The problem of the genetic algorithm initial configuration selection

N A Timofeev[1], **P V Peresunko**[1], **S R Nekhonoshin**[1], **V V Kukartsev**[1,2], **V V Tynchenko**[1,2] **and A S Mikhalev**[1]

[1] Siberian Federal University, 79, Svobodny pr., Krasnoyarsk, 660041, Russian Federation
[2] Reshetnev Siberian State University of Science and Technology, 31, Krasnoyarsky Rabochy Av., Krasnoyarsk, 660037, Russian Federation

E-mail: peres94@yandex.ru; space.trucking@mail.ru; crashik11@yandex.ru; vlad_saa_2000@mail.ru; 051301@mail.ru

Abstract. The genetic algorithm is one of the most well-known and frequently used global optimization algorithms. The software implementation of the algorithm in many programming languages and the development of various modifications of the selection parameters only encourage authors to analyze the GA parameters and search for their optimal values. Many works are trying to answer the question: what values of input parameters should be used for a greater likelihood of successfully finding the result. However, they only consider a specific set of parameters. This article describes the entire set of GA output parameters and draws up a recommendation for choosing initial values. Favorable sets of parameter values were found. Based on these sets, you can customize them for specific tasks. This study provides an example of using these sets to solve the problem of determining the parameters of a welded beam. Results were obtained corresponding to the best values of similar studies. It is important to note that in solving this problem, an initial set of parameters was already formulated, which facilitated the search for a global minimum.

1. Introduction

The genetic algorithm (GA) was developed in 1975. For all its existence, it has shown its effectiveness in solving optimization problems [1]. The abundance of scientific papers related to the research of this algorithm, apart from two factors: large number of software implementations of the algorithm in different programming languages [2-3]; various modifications of the algorithm, including the emergence of new types of selection, crossover and mutations [4-5].

There are many different modifications of the GA. Recent modifications have not been fully investigated and did not have time to prove their viability with each other. The developers leave recommendations on the standard values of the parameters, but they cannot be optimal for all applications.

Due to the large number of adjustable GA parameters, there is the problem of selecting their optimal values, which are solved, for example, using other optimization methods [6]. This process is also called meta-optimization [7]. However, the meta-optimization process takes a lot of machine time, as it implies a multiple launch of the optimization algorithm.

This article contains the initial sets of parameter values. GA with these parameters effectively finds a minimum on test problems. The results were used as initial GA settings for solving a real problem. The object of the research is the function of the dependence of finding the extremum on the values of the GA parameters. This paper assumes that GA operators are also its parameters, since they affect the efficiency of its work. As efficiency, we take the probability that, with the given input parameters, the algorithm falls in the vicinity of the minimum / maximum.

In most of the reviewed research papers on relevant topics, such parameters as mutation probability, crossover probability, and population size and, in rare cases, selection parameters most often change.

Authors such as S. S. Polyakov [8] and V Kapoor [9] investigated the effects of GA parameters on the algorithm's operation using the example of a number of test functions and came to general conclusions regarding the importance of the "mutation probability" parameter. O Roeva [10], M Javidi [11] and A Rexhepi [12], in addition to studying the influence of GA parameters, used them to solve problems. Thus, they supported the theoretical information with practical results.

Moreover, at the same time, it can be noted that in most of the works described, there are no rules and developments for the selection of parameters [9–12], or the exact numerical characteristics of the GA parameters [8] are not given.

2. Materials and methods

For the study, the "GA" package was used in the article [13] with the R programming language, since this package contains a large number of GA modifications.

Values of GA parameters recorded in Table 1 participate in the formation of all possible combinations of algorithm parameter values.

Table 1. Values of GA input parameters.

	Two-dimensional functions	Five-dimensional functions	Ten-dimensional functions
Population size	50, 100, 300, 500	100, 300, 500	300, 500, 700
Probability of mutation	0.05, 0.1	0.05, 0.1	0.05, 0.1
Crossover probability	0.7, 0.8, 0.9	0.7, 0.8, 0.9	0.7, 0.8, 0.9
Elitism	0.01, 0.05	0.01, 0.05	0.01, 0.05
Crossover type	Mixed crossover (blxCrossover), a: 0.36, 0.5, 0.7.		
	Local arithmetic crossover (laCrossover).		
	Laplace crossover (laplaceCrossover), a,b: ((0, 0.1), (0, 0.5)).		
	Single point crossover (spCrossover).		
	Full arithmetic crossover (waCrossover).		
Type of selection	Linear ranking selection (lrSelection).		
	Nonlinear rank selection (nlrSelection), q: 0.1, 0.25, 0.75.		
	Selection with linear scaling (lsSelction).		
	Roulette selection (rwSelection).		
	Tournament selection (tourSelection), k: 0.05, 0.1.		
Mutation type	Uniform mutation (raMutation).		
	Uneven mutation (nraMutation).		
	Powerful mutation (powMutation), pow: 10, 50.		
	Mutation around the solution (rsMutation).		

Within the framework of this study, we will assume that the main time costs during the operation of the GA are observed when the optimized function is started. Then, in order to limit the simultaneous growth of *NP* and the number of iterations of *iterCount*, we assume that the optimized function can only be run *N* times. Then the number of iterations will be calculated by the formula: *iterCount* = *N* / *NP*. *N* depends on the dimension of the problem: 5000, 7000 and 15000 in dimensions 2, 5 and 10, respectively.

Also for each dimension, a different set of population size values was used. The population sizes of 300 and 500 are common for each dimension, while 50 is only for two-dimensional, and 700 is only for ten-dimensional. This feature is related to the fact that as the dimension increases, it is necessary to explore the search space in more detail.

Let's formulate a criterion for the effectiveness of GA.

Let some function $f(\bar{x})$ be given, which has a global extremum at a point x^*. There is M – number of GA launches with a given set of input parameters. Then the effectiveness of this set is the ratio of the number of hits in a square neighborhood of size $2*e$ of the desired point x_{opt} to the number M (or, in other words, the relative probability of falling into an extremum).

For the study was selected the following set of test functions [14]: Rastrigin, Rosenbrock and Schwefel. Each of the functions will be considered in three dimensions: 2-dimensional, 5-dimensional and 10-dimensional. A similar set of dimensions was used in other scientific works [15], therefore, for the task, he, according to the authors, is preferable to achieve the goal of the study.

To analyze the results, the following strategies were developed for selecting the input parameter values:

- Search for the best parameter sets.
- Search for the worst values.

Tools have been used to demonstrate the estimation of probability density data at various levels. Based on this, each parameter of a specific data set has its own median of the probability of successfully finding a minimum and the interquartile range. Screening occurs if the difference between the medians of two parameter values is too high (more than 30%) or the interquartile range of one parameter value is significantly higher than the other (more than 30%).

The following experiment was carried out: such parameter values were selected that for a given function of each dimension show high efficiency. The results of the experiment are shown in Table 2, where the sign of the dash in some cells means that there is no such value of the input parameter that shows high efficiency for all dimensions of this function.

Table 2. Favorable values of parameters.

Input parameter	Rastrigin function	Rosenbrock function	Schwefel function
Crossover type	laCrossover	-	blxCrossover
Additional parameter blxCrossover	-	-	0.36, 0.5, 0.7
Type of selection	lsSelection	-	-
Mutation type	nraMutation, raMutation	powMutation (10 and 50)	raMutation
Population size	-	-	300
Probability of mutation	0.1	0.05, 0.1	0.1
Crossover probability	0.8, 0.9	0.7, 0.8, 0.9	0.7, 0.8, 0.9
Elitism	0.05	0.05	0.01, 0.05

You can see that all functions have parameters that do not have an optimal value, effective for any dimension of a given function. Based on this, it cannot be argued that for one function, but of different dimensions, the same set of parameters will work with the same efficiency.

At the end of the first strategy stage, intermediate conclusions can be drawn: certain values of auxiliary parameters are equally effective for different selection methods, regardless of the function and its dimensionality. Certain values of such GA parameters as mutation probability, crossover probability, and elitism show themselves well for a given set of functions. Moreover, increasing the dimension does not affect the efficiency of these parameters, while there is a need to increase the size of the population.

After the results of the first strategy, the second was carried out. As a result, the following values of input parameters most often show low efficiency for the proposed set of functions:

- Mutation type: rsMutation.
- Type of selection: rwSelection.
- Population size: 50 и 700.

These values show efficiency only for a specific function and do not have universality.

The obtained effective values of the parameters will be used as the initial approximation of the GA parameters for the real optimization problem.

3. Experimental study

Determining the parameters of welded beam was defined as a realistic objective [16-23]. It is necessary to find the minimum cost of the beam design, determined by the vector of parameters *x = (x1, x2, x3, x4)^T*. In addition to the specified objective function, the following restrictions must be taken into account: shear stress *t*; bending *q*; longitudinal load *Pc*; beam deflection *qe*. Mathematically, we can formulate the problem as follows:

$$
\begin{gathered}
f(x) = 1{,}10471 \cdot x_1^2 \cdot x_2 + 0{,}04811 \cdot x_3 \cdot x_4 \cdot (14 + x_2),\\
g^1(x) = \tau(x) - 13600 \le 0,\ g^2(x) = \sigma(x) - 30000 \le 0,\ g^3(x) = x_1 - x_4 \le 0,\\
g^4(x) = 0{,}10471 \cdot x_1^2 + 0{,}04811 \cdot x_3 \cdot x_4 \cdot (14 + x_2) - 5 \le 0,\\
g^5(x) = 0{,}125 - x_1 \le 0,\ g^6(x) = \delta(x) - 0{,}25 \le 0,\\
g^7(x) = 6000 - P_c(x) \le 0,\ D = [0{,}1; 2{,}0] \times [0{,}1; 2{,}0] \times [0{,}1; 10{,}0] \times [0{,}1; 2{,}0]]\\
\tau(x) = \left((\tau')^2 + (2 \cdot \tau' \cdot \tau'') \cdot \frac{x_2}{2 \cdot R} + (\tau'') \right)^{1/2},\ \tau' = \frac{6000}{\sqrt{2} \cdot x_1 \cdot x_2},\ \tau'' = \frac{M \cdot R}{J},\ M = 6000\left(14 + \frac{x_2}{2}\right),\\
R = \left(\frac{x_2^2}{4} + \left(\frac{x_1 + x_3}{2} \right)^2 \right)^{\frac{1}{2}},\ J = 2 \cdot 2^{\frac{1}{2}} \cdot x_1 \cdot x_2 \cdot \left(\frac{x_2^2}{12} + \left(\frac{x_1 + x_3}{2} \right)^2 \right),\ \sigma(x) = \frac{504000}{x_3^2 \cdot x_4},\ \delta(x) = \frac{65{,}856}{30 \cdot x_3^3 \cdot x_4},\\
P_c(x) = \frac{4{,}013 \cdot 5 \cdot 10^6 \cdot \left(x_3^2 \cdot x_4^6 \right)^{\frac{1}{2}}}{x_3^2 \cdot x_4} \cdot \left(1 - \frac{x_3 \cdot \left(\frac{5}{8} \right)^{\frac{1}{2}}}{28} \right)
\end{gathered} \tag{1}
$$

As the initial parameters of the GA, we used the values of the parameters obtained as a result of research on test functions. Thus formed 3 sets of parameters that are used as the initial approximation of the values of the parameters of the GA. After that, the values of the types of selection, mutations and crossover are fixed, changing only the values of the other parameters, which are saved when the result is improved. For each set, 10 runs of the algorithm were made, after which the best-found value was saved.

As a result, the smallest minimum of the function was found using the following set of parameters: number of individuals is 150, probability of mutation is 0.1, probability of the crossover is 0.8, elitism is 0.05, type of mutation is nraMutation, the type of selection is tourSelection, type of crossover is blxCrossover. Lets's compare the results of the genetic algorithm with these parameters with the results of other works (Table 3).

Table 3. Results of solving the optimization problem in other studies.

	This article	Deb [16]	Siddall [17]	Ragsdell [18]	Coello [19]	Wang [20]	Coello [21]	Coello [22]	Yongquan [23]
x1	0.2057	0.2489	0.2444	0.2455	0.2060	0.2024	0.2088	0.2057	0.2573
x2	3.4705	6.1730	6.2189	6.1960	3.4713	3.5442	3.4205	3.4705	3.4705
x3	9.0366	8.1789	8.2915	8.2730	9.0202	9.0482	8.9975	9.0366	9.0366
x4	0.2057	-0.2533	0.2444	0.2455	0.2065	0.2057	0.2100	0.2057	0.2057
f(x)	1.7249	2.4331	2.3815	2.3859	1.7282	1.7280	1.7483	1.7249	1.7249

The result obtained corresponds to the best solution found in [22-26], which indicates the effectiveness of applying the recommended parameters.

4. Discussion

In most real-world optimization problems, their specificity is determined by the type of the objective function. For the research in this article, 2 classes of objective functions were selected that can be encountered in practice when describing real problems. The results were used as initial approximations for solving the problem of engineering optimization. Comparison with the results in other articles allows making a conclusion about the effectiveness of the approach used to the selection of the initial parameters of the algorithm. In the future, we plan to conduct research on the functions of other classes. The approach used in the article can also be used to select configurations of other global optimization algorithms.

In the process of selecting the parameters of the algorithm on test functions, it was found that as the dimension increases, the optimal value of the population number increases, at which the maximum probability of finding an extremum is reached. This observation needs additional research. However, not always the larger the population, the better is the minimum. First of all, with an increase in the number of the population, the time spent on the work of the HA also increases, and the increase in the number does not always pay off [24].

5. Conclusion

Summarizing the results of this study, favorable sets of parameter values were found. Based on these sets, you can customize them for specific tasks.

This study provides an example of using these sets to solve the problem of determining the parameters of a welded beam. Results were obtained corresponding to the best values of similar studies. It is important to note that in solving this problem, an initial set of parameters was already formulated, which facilitated the search for a global minimum.

References

[1] Tynchenko V S, Petrovsky E A and Tynchenko V V 2016 The parallel genetic algorithm for construction of technological objects neural network models *2nd International Conference on Industrial Engineering, Applications and Manufacturing, ICIEAM 2016* 7911573

[2] Panchenko T V 2007 *Genetic Algorithms* (Astrakhan: Astrakhan University Publishing House)

[3] Bessonov D V 2014 C ++ library of components of genetic algorithms *Young Scientist* **6** 73-7

[4] Zbigniew M 1996 *Genetic Algorithms + Data Structures = Evolution Programs* (Springer-Verlag: Berlin)

[5] Deep K and Thakur M 2007 A new crossover operator for real coded genetic algorithms *Applied Mathematics and Computation* **188** 895-912

[6] Groshev S V and Karpenko A P 2016 Meta-optimization of population algorithms for multipurpose optimization *Internet-magazine "SCIENCE"* **8(6)** 1-11

[7] Pedersen M E H 2010 *Tuning & simplifying heuristical optimization* (Doctoral dissertation, University of Southampton)

[8] Polyakov S S 2013 Investigation of the significance of the parameters of the genetic algorithm *IX All-Russian Conference "Youth and Science"* 183-6

[9] Kapoor V, Dey S and Khurana A P 2011 An empirical study of the role of control parameters of genetic algorithms in function optimization problems *International Journal of Computer Applications* **0975** 8887

[10] Roeva O, Fidanova S and Paprzycki M 2015 Population size influence on the genetic and ant algorithms performance in case of cultivation process modeling *Recent Advances in Computational Optimization* 107-20

[11] Javidi M M and Fard R H 2013 Research in random parameters of genetic algorithm and its application on TSP and optimization problems *Walailak Journal* **12(1)** 27-34

[12] Rexhepi A, Maxhuni A and Dika A 2013 Analysis of the impact of parameters values on the genetic algorithm for TSP *International Journal of Computer Science Issues* **3** 158

[13] Storn R and Price K 1997 Differential evolution–a simple and efficient heuristic for global optimization over continuous spaces *Journal of global optimization* **11(4)** 341-59

[14] Karaboga D 2005 An idea based on honey bee swarm for numerical optimization *Technical report-tr06, Erciyes university, engineering faculty, computer engineering department* **200**

[15] Chen S, Bolufe-Rohler A and Montgomery O 2015 Measuring the curse of dimensionality and its effects on particle swarm optimization and differential evolution *Applied Intelligence* **42(3)** 514-26

[16] Syswerda G 1989 Uniform Crossover in Genetic Algorithms *Proceedings of the Third International Conference on Genetic Algorithms* 2-9

[17] Siddall J N 1972 *Analytical Design-Making in Engineering Design* (Upper Saddle River, New Jersey: Prentice Hall)

[18] Ragsdell K M and Phillips D T 1976 Optimal design of a class of welded structures using geometric programming *ASME Journal of Engineering for Industries* **98(3)** 1021-5

[19] Coello C A C and Montes E M 2002 Constraint-handling in genetic algorithms through the use of dominance-based to urnament selection *Advanced Engineering Informatics* **16** 193-203

[20] He Q and Wang L 2007 An effective co-evolutionary particle swarm optimization for constrained engineering design problems *Engineering Applications of Artificial Intelligence* **20** 89-99

[21] Coello C A C 2000 Use of a self-adaptive penalty approach for engineering optimization problems *Computers in Industry* **41** 113-27

[22] Coello C A C and Becerra R L 2004 Efficient evolutionary optimization through the use of a cultural algorithm *Engineering Optimization* **36** 219-36

[23] Zhou Y Q and Liu L 2010 An effective chaotic cultural-based particle swarm optimization for constrained engineering design problems *Applied Mechanics and Materials* **20** 64-6

[24] Zhang Y, Ma Q, Sakamoto M and Furutani H 2010 Effects of population size on the performance of genetic algorithms and the role of crossover *Artificial Life and Robotics* **15(2)** 239-43

[25] Tynchenko V S, Tynchenko V V, Bukhtoyarov V V and Agafonov E D 2018 Decision support system for designing an effective configuration of a computing network for distributed complex problem solving *RPC 2018 - Proceedings of the 3rd Russian-Pacific Conference on Computer Technology and Applications* 8482126

[26] Tynchenko V S, Tynchenko V V, Bukhtoyarov V V, Tynchenko S V and Petrovskyi E A 2016 The multi-objective optimization of complex objects neural network models *Indian Journal of Science and Technology* **9(29)** 99467