

# SEARCH OPTIMIZATION OPPORTUNITIES OF MODIFIED SELF-ORGANIZING MIGRATING ALGORITHM IN MULTI-EXTREMAL TASKS ENVIRONMENT

*Ivan CHERNOGOROV\**, *Victor POLYAKH*, *Orkhan YARAKHMEDOV*

Department of Software Engineering for Computers and Computer-based Systems, Faculty of  
Computer Science and Facilities, Don State Technical University, Gagarin sq. 1, 344000  
Rostov-on-Don, Russia

\*[hintaivr@gmail.com](mailto:hintaivr@gmail.com)

(Received: 06-August-2017; accepted: 24-September-2017; published: 30-November-2017)

DOI: <http://dx.doi.org/10.25073/jaec.201712.60>

**Abstract.** *The paper studies the search optimization task of multi-extremal objects, which are more complicated than mono-extremal. Paper postulates that to find extreme suitable values on complex test function the heuristic algorithm is one way. Self-Organizing Migrating Algorithm and devised approach applied to this task are considered. Conducted research established common test environment to compare multi-extremal test functions. Specific characteristics for problem solving of detection and identification of global and local extreme are included. Additional clustering mechanism is described. Obtained measurements of Self-Organizing Migrating Algorithm on a range of multi-extremal test functions are illustrated.*

## Keywords

*Clustering, Multi-Extremal, Searching Optimization, SOMA, Test Function.*

## 1. Introduction

The most advanced problems in science, technology, economics, military affairs and other applied modern trends are connected with the

tasks of finding optimums in designs, technologies, models and environments, through the possibility of controlling the dynamic and static states, as well as, other requirements put forward in the specifications of the design objects. In other words, the developers have to solve the problems of Searching Optimization (SO) [1]. It is very typical that most of the current known SO methods are developed and effectively used to find only one extreme, which is often the global one.

However, many tasks in solving complex technological systems and transportation problems require optimization. Especially, the objects of discrete nature are characterized by MultiExtremal (ME) properties [2] and [3]. A significant distinctive property for solving such tasks requires specific methods to reach the solution. It is unlikely that these methods should be sought in the class of the SO deterministic methods, though such attempts are already well known. These methods are too sensitive to the sign variation of discontinuous functions within their continuum response factor spaces. For solving real optimization problems, it has been common to apply methods called "heuristic". These methods are the most perspective to obtain solutions for the ME problems [4] and [5]. The bright representative this type of methods is

Self-Organizing Migrating Algorithm (SOMA) [6], [7] and [8]. SOMA is an algorithm developed in 1999, whose operation is based, like the Scatter Search [9] or Particle Swarm Optimization (PSO) [10] on vector operations.

The original idea that led to its creation, is to mimic the behavior of a group of intelligent individuals who cooperate in solving common problems such as finding food sources, etc. Since working with similar populations, such as Genetic Algorithms (GA) [11] and evolutionary outcome after one it is identical with genetic algorithm and differential evolution, it can be understood like sort of evolutionary algorithm despite the fact that during its run it is not in terms of philosophy of creating a new algorithm descendants, as in other classical evolutionary algorithms.

To test the effectiveness of the developed modifications require Test Functions (TFs) with not only global extremum, but also a variety of global, local or sub-extreme values. The authors are chosen next TFs: Rastrigin [12], Rosenbrock [13], Himmelblau [14], Lambda [15], Schwefel [16], Giunta [17], Ursem [18], Shubert [19] and Plateau [20].

These TF have different property. For example, Rastrigin TF has only one global extremum. Himmelblau has 4 equals global extreme, etc.

## 2. Canonical SOMA MM

SOMA is not based on the philosophy of evolution, but on the behavior of a social group of individuals, e.g. a herd of animals looking for food. One can classify SOMA as an evolutionary algorithm, because the final result, after one migration loop, is equivalent to the result from one generation derived by the classic evolutionary algorithms - individuals hold new positions on the N dimensional hyper-plane. When the group of individuals is created, then the rule mentioned above governs the behavior of all individuals so that they demonstrate 'self-organization' behavior. Because no new individuals are created, and only existing ones are moving over the N-dimensional hyper-plane. This algorithm was described in more detail in the [6], [7] and [8]

works. Below is a brief description of the algorithm, for an understanding of its principles.

### 2.1. SOMA Parameters and Their Recommended Values

SOMA parameters divided into two types: "Controlling" - are used for the quality of results of optimization process and "Stopping" - are used to stop the search process when criteria are fulfilled:

- *PathLength*. This parameter defines how far an individual stops in front of/behind the Leader,
- *Step*. This parameter defines the fraction of step individual to the Leader,
- *PRT*. This parameter determines whether an individual will go directly to the Leader or to dimensional subspace, which is perpendicular to the original space. This is the mutation and crossover GA prototypes,
- *PopSize*. This is individuals number in population,
- *Migrations*. This is the number of maximum iteration, the implementation of which will stop the algorithm,
- *MinDiv*. This parameter defines the largest allowed difference between the best and the worst individual from actual population. If the difference is too small, then the optimizing process is will stop. Recommended value is 0.1, but parameter is very dependent on target SO problem [6].

The parameters and their recommended domains are shown in Tab. 1.

### 2.2. SOMA Strategies and Choice of Target Strategy for Experiments

Select a population Leader ("attraction" of population or individual in particular) is one of the

**Tab. 1:** Recommended domains of SOMA parameters.

Name	Recommended domain	Type
PathLength	[1.1; 3]	Controlling
Step	[0.11; PathLength]	Controlling
PRT	[0; 1]	Controlling
PopSize	[10; $+\infty$ )	Controlling
Migrations	[10; $+\infty$ )	Stopping
MinDiv	(0; $+\infty$ )	Stopping

key moments in SOMA. The canonical SOMA version has four basic strategies:

- *All to One.* All individuals move towards the Leader, except the Leader,
- *All to Random.* All individuals move towards a randomly selected individual during the migration loop. This strategy has sub-strategies, which are change the number of randomly selected individuals,
- *All to All.* All individuals move towards the all other individuals. Each individual try to find the best position between it and other individuals,
- *All to All Adaptive.* It is a modified All to All strategy. Individuals do not begin a new migration from the same old position, but from the last best position found during the last traveling to the previous individual.

As a result of previously conducted research [6] the most logical strategy is "All to All Adaptive", due to the high efficiency (in spite of the increasing computing time). Also, the authors attempt was create new strategies ("All to One plus Random", "All to Neighbor and All to One plus Neighbor"), but the results of computational experiments have shown that the basic strategies were more effective.

### 2.3. SOMA Principles

The canonical SOMA version [6] consists of the following steps:

1. *Parameter definition.* Before starting the algorithm, SOMA's parameters and the Cost Function (CF) needs to be defined,

2. *Population initialization.* A population of individuals is randomly generated,
3. *Migrating loop.* Each individual is evaluated by CF and the Leader is chosen for the current migration loop. Migration can be also viewed as a competitive-cooperative phase. During the competitive phase each individual tries to find the best position on its way and also the best from all individuals. Thus during migration, all individuals compete among themselves. When all individuals are in new positions, they release information as to their cost value. This can be regarded as a cooperative phase. All individuals cooperate so that the best individual (Leader) is chosen. Then all other individuals begin to jump, towards the Leader. Each individual is evaluated after each jump using the CF. The jumping continues, until a new position defined by the PathLength has been reached,
4. *Test for stopping condition.* If MinDiv or Migrations stopping criterion are fulfilled, go to step 5, otherwise return to step 3,
5. *Stop.* Recall the best solution(s) found during the search.

## 3. Modified SOMA for ME SO Tasks

ME SOMA modification based on the canonical SOMA, but it has additional discrete mechanism and clustering process.

First problem in ME SO is an allocation of all global/local/sub-extremum areas. In this work, authors propose the sampling of search space approach. That means that all searching space divided into  $N$  sub-spaces.

In every sub-space ME SOMA initialize independent population (population from one sub-space do not communicate with other populations). After fulfilled stopping criterion in each population SOMA work is stop (population in one sub-space do not stop the work of other populations). Sampling the searching space process means, that number of maximum

extremums were allocated equally  $N$ . However, it makes modification as "parallel" and reducing the computing time, because all SOMA populations searching the extrema at the same time.

After allocating all extreme in each space the result values are clustered, to get a true picture without "nearly-extremum" values. For this problem authors propose "A Quasi-Equivalence" clustering algorithm [21] and [22]. Clustering does not require resulting number of clusters. It can be described by the following equations:

The matrix of normal similarity measures:

$$\mu_{x_q}(x_i) = 1 - \frac{d(x_q, x_i)}{\max_{k \in [1, Q]} (d(x_q, x_k))}, \quad (1)$$

where:  $x$  is the plurality of elements;  $Q$  is a number of elements in plurality;  $q, i \in 1 \dots Q$ ,  $d(x, y)$  is a clustering criterion (like Euclidean distance between points, etc.).

The relative similarity measures:

$$\zeta_{x_q}(x_i, x_j) = 1 - |\mu_{x_q}(x_i) - \mu_{x_q}(x_j)|, \quad (2)$$

where  $j \in 1 \dots Q$ .

The matrix of similarity measures of elements plurality:

$$\begin{aligned} \zeta(a, b) &= T(\zeta_{x_1}(a, b), \dots, \zeta_{x_Q}(a, b)) = \\ &= \min_{i=1, Q} \zeta_{x_i}(a, b), \end{aligned} \quad (3)$$

where  $a, b \in x$  plurality.

The result matrix:

$$R_\zeta^q = R_\zeta^{q-1} \cdot R_\zeta, \quad (4)$$

where  $R_\zeta$  is relation between clustering points.

Values in result  $R$  matrix will show whether the pair of points belongs  $R$  relation, their called "quasi-equivalence levels" ( $a$ ). The choice of a particular level divides the plurality into equivalence classes, which correspond to separate clusters. Fig. 1 demonstrates flow-chart of "A Quasi-Equivalence" clustering algorithm.

ME SOMA modification requires to do "A Quasi-Equivalence" clustering by 2 different criterion: *first* - by Euclidean distance between

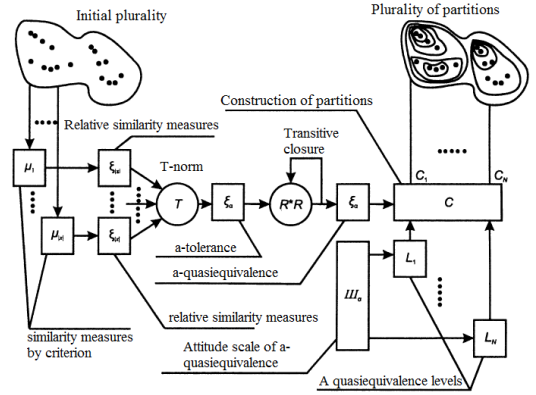


Fig. 1: "A Quasi-Equivalence" clustering flow-chart.

allocated extrema, to drop-out around the extreme values and *second* - by CF values, to group the extrema with the same values.

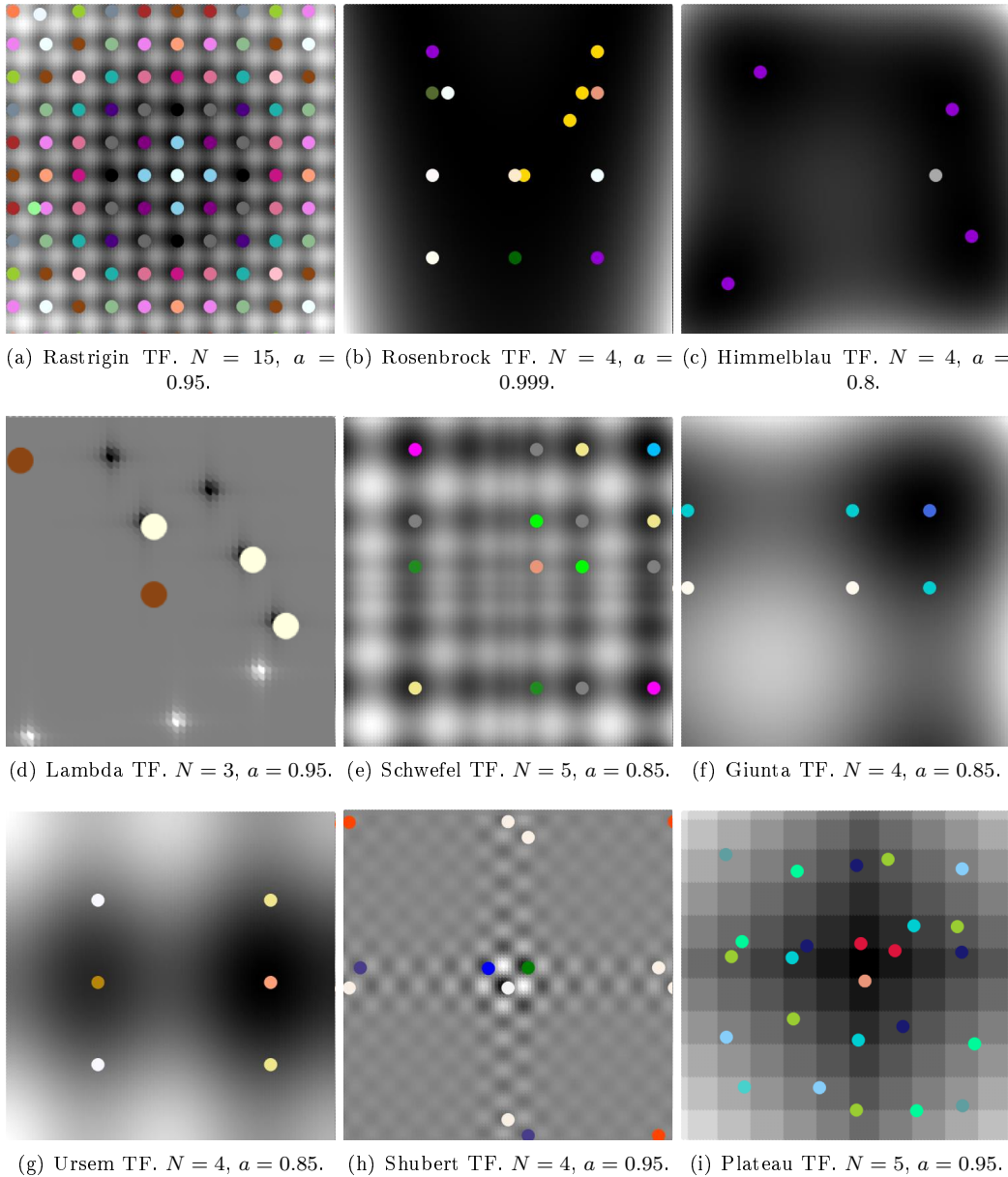
Using Euclidean distance clustering criterion allows to groups the SOMA populations around the same extrema. After the first clustering criterion developed mechanism delete all individuals, except the Leader (extremum). It allows to dropout the sub-local values.

## 4. Check ME SOMA Effectiveness

To test the effectiveness of developed ME SOMA modification chosen 9 famous TF. An appropriate software tool "ME SOMA" was also developed. For development C# programming language in Microsoft Visual Studio 2015 IDE was used.

Search purpose for each function are the minima. For all experiments identical parameter settings are used:  $N$  is dependent on TF,  $Migration = 10$ ,  $PopSize = 7$ ,  $Step = 0.11$ ,  $PathLength = 3$ ,  $PRT = 0.1$ ,  $MinDiv = 1e^{-15}$ . *Strategy* is "All to All Adaptive".  $a$  is dependent on TF

To demonstrate ME SOMA modification result the figures for each TF with allocated extreme area clusters are illustrated. As can be seen on Fig. 2(a) global and local extrema were localized. Although  $N >$  (global + local extreme



**Fig. 2:** ME SOMA modification results for different TF.

count) and it is not a problem, because clustering by Euclidean distance delete sub-local population. But one sub-local extremum localized, defined by light-green color. This can be correct by changing a parameter. When SOMA try find plurality of extrema in not ME TF it gets sub-local clusters as seen on Fig. 2(b). Figure 2(c) is good demonstration of deleting sub-local extrema by varying a parameter. A

little number of  $N$  leads to allocating not all global extrema as seen above on Fig. 2(d). A little number of  $N$  leads to allocating not all global and local extrema as seen above on Fig. 2(e). Positions of clusters on Fig. 2(i) not symmetrical, as in other functions due to the nature of this function.

The experiments described above have shown that to find the global and local extrema at ME TF recommended to use  $N > 5$ . Also, a parameter can significantly improve the drop-out process of sub-local populations. To optimize TF with plurality of global extrema recommended use  $a < 0.85$ . To optimize TF with plurality of global and local extrema recommended use  $a > 0.85$ .

## 5. Comparing ME SOMA Result

ME SOMA comparing with other analogues has two problems: first - is a little number of ME SO algorithm and second - is papers, which describing these algorithms usually do not contain any numerical experimental data or optimization problem is far from TF. Therefore, the comparison will be with the results presented in the [23], which described modifications of PSO, GA and Ant Colony Optimization (ACO). Table 2 illustrates standard values of Rastrigin TF extreme. Table 3 demonstrates the result, that were conducted on Rastrigin TF with  $(x, y) \in [-1.5, 1.5]$  dimension domains and how much PSO, GA, ACO and ME SOMA invocated CF to obtained these results.

ME SOMA parameters:  $N = 3$ ,  $Migrations = 10$ ,  $PopSize = 10$ ,  $Step = 0.11$ ,  $PathLength = 3$ ,  $PRT = 0.1$ ,  $MinDiv = 0.01$ ,  $a = 0.95$ ,  $Strategy = All\ to\ All\ Adaptive$ .

The time consuming criteria was omitted, because of the inexpediency of its comparison in these two-dimensional test problems. If we will carry out the research in tasks of large dimensions (NP-complete problems), this criterion is very important for comparing the effectiveness of methods.

As a result, the minimum and average errors (in percentage) of the obtained approximation (see Tab. 1), relative to the standard, were:

- for PSO:  $\sim 1.94e^{-6}$  % and  $\sim 0.4456$  %,
- for GA:  $\sim 7.1e^{-4}$  % and  $\sim 1.7976$  %,
- for ACO:  $\sim 1.5e^{-6}$  % and  $\sim 0.7829$  %,

Tab. 2: Rastrigin TF extreme standard.

No.	$x$	$y$	$F(x, y)$
1	-1	1	2
2	-1	0	1
3	-1	-1	2
4	0	1	1
5	0	0	0
6	0	-1	1
7	1	1	2
8	1	0	1
9	1	-1	2

Tab. 3: ME SOMA, PSO, GA and ACO CF comparing.

No.	PSO	GA	ACO	ME SOMA
	$F(x, y)$			
1	1.9901	2.01775	2.01637	1.98993
2	0.995	1.00137	0.98972	0.9954
3	1.9899	2.06707	2.0165	1.98991
4	0.995	1.0611	0.9889	0.9954
5	$1.94e^{-6}$	0.00071	$1.5e^{-6}$	$2.77e^{-18}$
6	0.995	1.01436	0.9893	0.9952
7	1.9899	2.00947	1.9898	1.98992
8	0.995	1.03398	0.9899	0.9953
9	1.9899	1.993	2.0135	1.98994
	$CF\ invocated\ number$			
	20200	22500	20000	33055

- for ME SOMA:  $\sim 2.77e^{-18}$  % and  $\sim 0.4317$  %.

ME SOMA has the best global result and a smaller obtained spread of the values. This fact indicates greater reliability of the algorithm.

But, for these experiments, ME SOMA need more CF invocated number. This is due to the fact that MM SOMA contains a deterministic part ("Step" parameter). However, it allows converge to all system agents and get a more accurate result. And CF invocated number of all current algorithms can be significantly changed at other tasks (e. g. in NP-complete tasks).

## 6. Related Work

In the design optimization process, we are often confronted with problems facing the ME conditions. Such situation requires decisions, which take into consideration several identical or close extremes, and the best choice in-between them has to be made. The classical theory of scheduling gives examples, where several identical op-

timums and identical sub-optimums, close to them exist [1], [3], [4] and [5]. The majority of discrete, integer and combinatory programming problems differs in such property [24], [22], [23], [24] and [25], in particular, when finding solution for graphs [26], [27], [28] and [29]. The finite number of admissible decisions requires considering the ME solutions for the discrete environment optimization. There are many additional conditions, which can help to choose the extreme, equivalent or close in size, and satisfy both, the numerical criteria estimates and the heuristic ideas. Therefore, the choice, of the most effective methods and algorithms, is an extremely important step to find such solution of the ME task.

## 7. Conclusions

The analysis of SOMA application for solving the ME tasks showed that modification is efficient, effective, and bring some essential features to the presented solutions. The specific approaches to solve the task for each of these particular cases is determined through the analysis of the algorithm features; the detection and identification of local extremes, clustering method and subsequent operations resulting from such analysis. Also, ME SOMA modification showed reasonable performance.

To conclude, studied SOMA is relevant and promising for future applications. The specific choice of the algorithm tool for solving ME tasks depends on the experience and personal researcher preferences, as well as on the special features of the domain specific research area.

## References

- [1] BOETTCHER, S. and A. G. PERCUS. Extremal optimization: methods derived from co-evolution. *Proceedings of the 1st Annual Conference on Genetic and Evolutionary Computation (GECCO)*. San Francisco: Morgan Kaufmann Publishers Inc., 1999, pp. 825–832.
- [2] SHREVES, R. Drupal search engine optimization, Packt Publishing LTD, Birmingham. 2012. Available at: <https://www.webpage.com>.
- [3] VINOGRADOV, I. M. *Mathematical encyclopedia. Soviet Encyclopedia. Volume 4*. 1st ed. Moscow: Soviet Encyclopedia, 1985.
- [4] STRONGIN, R. G. Algorithms for multi-extremal mathematical programming problems. *Journal of Global Optimization*. 1992, vol. 2, iss. 4, pp. 357–378.
- [5] NEYDORF, R. A., A. V. FILIPPOV and Z. H. YAGUBOV. Commute algorithm of biextreme solutions of the homogeneous distribution problem. *Herald of DSTU*. 2011, vol. 11, no. 5(56), pp. 655–666.
- [6] ONWUBOLU, G. C. and B. V. BABU. *New Optimization Techniques in Engineering*. 1st ed. New York: Springer, 2004.
- [7] DAVENDRA, D. and I. ZELINKA. *Self-organizing migrating algorithm*. 1st ed. New York: Springer, 2016.
- [8] NOLLE, L., I. ZELINKA, A. A. HOPGOOD and A. GOODYEAR. Comparison of an self-organizing migration algorithm with simulated annealing and differential evolution for automated waveform tuning. *Advances in Engineering Software*. 2005, vol. 36, iss. 10, pp. 645–653.
- [9] MARTIA, R., M. LAGUNAB and F. GLOVERB. Principles of scatter search. *European Journal of Operational Research*. 2006, vol. 169, iss. 2, pp. 359–372.
- [10] EBERHART, R. and J. KENNEDY. New optimizer using particle swarm theory. In: *Proceedings of the Sixth International Symposium on Micro Machine and Human Science (MHS)*. Nagoya: IEEE, 1995, pp. 39–43.
- [11] ZIARATI, K. and R. AKBARI. A multi-level evolutionary algorithm for optimizing numerical functions. *International Journal of Industrial Engineering Computations*. 2011, vol. 2, iss. 2, pp. 419–430.

- [12] RASTRIGIN, L. A. *Systems of Extremal Control*. 1st ed. Moscow: Nauka, 1974.
- [13] ROSENBROCK, H. H. An Automatic Method for Finding the Greatest or Least Value of a Function. *The Computer Journal*. 1960, vol. 3, iss. 3, pp. 175–184.
- [14] HIMMELBLAU, D. *Applied Nonlinear Programming*. 1st ed. New York: McGraw-Hill, 1972.
- [15] NEYDORF, R. A. Complicated test function to research multiextremal problems and solutions. In: *Proceedings of the VI International Seminar "System analysis, management and information processing"*. Rostov-on-Don: DSTU, 2015, pp. 6–14.
- [16] LAGUNA, M. and R. MARTI. Experimental Testing of Advanced Scatter Search Designs for Global Optimization of Multimodal Functions. *Journal of Global Optimization*. 2005, vol. 33, iss. 2, pp. 235–255.
- [17] JAMIL, M. and X.-S. YANG. A literature survey of benchmark functions for global optimization problems. *Journal of Mathematical Modelling and Numerical Optimization*. 2013, vol. 4, no. 2, pp. 150–194.
- [18] ALROOMI, A. R. The Farm of Unconstrained Benchmark Functions. *University of Bahrain, Electrical and Electronics Department* [online]. 2013. Available at: <http://www.al-roomi.org/cv/publications>.
- [19] MOLGA, M. and C. SMUTNICKI. Test functions for optimization needs. 2005. Available at: <http://www.robertmarks.org/Classes/ENGR5358/Papers/functions.pdf>.
- [20] Test functions for optimization. In: *wikipedia* [online]. 2017. Available at: [https://en.wikipedia.org/wiki/Test\\_functions\\_for\\_optimization](https://en.wikipedia.org/wiki/Test_functions_for_optimization).
- [21] BARSEGIAN, A. A., M. S. KUPRIJANOV, V. V. STEPANENKO. *Methods and analysis models of OLAP and Data Mining*. 1st ed. Saint Petersburg: BKhV-Peterburg, 2004.
- [22] Clustering. Algorithm A-quasiequivalence. In: *habrahabr* [online]. 2011. Available at: <https://habrahabr.ru/post/124978>.
- [23] NEYDORF, R., I. CHERNOGOROV, V. POLYAKH, O. YARAKHMEDOV, Y. GONCHAROVA and D. VUCINIC. Study of Search Optimization Opportunities of Heuristic Algorithms for Solving Multi-Extremal Problems. In: *Proceedings of The X International Conference on Advanced Engineering Computing and Applications in Sciences (ADVCOMP2016)*. Venice: IARIA, 2016, pp. 44–51.
- [24] ZHURAVLEV, Y. I. Discrete programming. *Encyclopedia of Mathematics*. In: *encyclopediaofmath* [online]. 2015. Available at: [https://www.encyclopediaofmath.org/index.php/Discrete\\_programming](https://www.encyclopediaofmath.org/index.php/Discrete_programming).
- [25] KNUTH, D. E. *The Art of Computer Programming, Volume 4A: Combinatorial Algorithms, Part 1*. 1st ed. New Jersey: Addison-Wesley Professional, 2011.
- [26] DIESTEL, R. *Graph Theory*. 5th ed. Berlin: Springer, 2016.
- [27] RUOHONEN, K. *Graph Theory*. 1st ed. Berlin: Springer, 2013. Available at: [http://math.tut.fi/~ruohonen/GT\\_English.pdf](http://math.tut.fi/~ruohonen/GT_English.pdf).
- [28] GRIFFIN, C. *Graph Theory: Penn State Math 485 Lecture Notes Version 1.4.2.1*. In: *personal* [online]. 2017. Available at: <http://www.personal.psu.edu/cxg286/Math485.pdf>.
- [29] Dooren, P. V. *Graph Theory and Applications*. In: *perso.uclouvain.be* [online]. 2009. Available at: <https://perso.uclouvain.be/paul.vandooren/DublinCourse.pdf>.

## About Authors

**Ivan CHERNOGOROV** is Ph.D. student at Don State Technical University. His research



is focused on multiextremal modification of method of swarming particles for different complex tasks. Currently, he is on the verge of preparing for the defense of his doctoral thesis.

**Victor POLYAKH** is Ph.D. student at Don State Technical University. His research is focused on multiextremal modification of evolutionary-genetic algorithm for different

complex tasks. Currently, he is on the verge of preparing for the defense of his doctoral thesis.

**Orkhan YARAKHMEDOV** is Ph.D. student at Don State Technical University. His research is focused on multiextremal modification of ant colony optimization algorithm for different complex tasks. Currently, he is on the verge of preparing for the defense of his doctoral thesis.

"This is an Open Access article distributed under the terms of the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited (CC BY 4.0)."