UDC 519.7 DOI: 10.18698/1812-3368-2019-2-17-31

NEW ADAPTIVE MULTI-MEMETIC GLOBAL OPTIMIZATION ALGORITHM

A.P. Karpenko apkarpenko@bmstu.ru
M.K. Sakharov msakharov@bmstu.ru

Bauman Moscow State Technical University, Moscow, Russian Federation

Abstract

This paper deals with the Simple *MEC* (*SMEC*) algorithm which belongs to a class of *MEC* algorithms. The algorithm was selected for investigation due to the following reasons: nowadays this algorithm and its modifications are successfully used for solving various optimization problems; the algorithm is highly suitable for parallel computations, especially for loosely coupled systems; the algorithm is not sufficiently studied — there are relatively few modifications of *SMEC* (while, for instance, tens of various modifications are known for particle swarm optimization). Authors proposed an adaptive multi-memetic modification of *SMEC* algorithm, which includes a stage of landscape analysis for composing a set of basic adaptation strategies; software implementation of the algorithm is also presented. Performance investigation was carried out with a use of multi-dimensional benchmark functions of different classes. It was demonstrated that the concept of multi-population along with the incorporated landscape analysis procedure allows making a rough static adaptation of the algorithm to the objective function at the very beginning of evolution process at the cost of small computational expenses. Utilization of memes, in turn, helps the algorithm to correct possible errors of static adaptation during the evolution due to a closer investigation of search sub-domains

Keywords

Multi-memetic algorithm, landscape analysis, mind evolutionary computation, global optimization

Received 25.12.2017

This work was supported by the Russian Foundation for Basic Research (RFBR project no. 16-07-00287)

Introduction. When solving real-world global optimization problems, one often faces a high-dimensional objective function with a non-trivial landscape, which is computationally expensive. In order to cope with such problems, many population-based algorithms were proposed [1, 2]. One of the main advantages of this class of algorithms, apart from their simplicity of implementation and diversity, is a high probability of localizing so called sub-optimal solutions, in other words, solutions that are close to the global optimum. In real-world optimization problems, such solutions are often sufficient.

In the meantime, it was demonstrated [2–4] that often a single method is not enough to obtain a high-quality solution. It is required to hybridize a method with other optimization techniques. One of the promising approaches in this field is so called memetic algorithms (*MA*). These methods are population meta-heuristic optimization algorithms based on neo-Darwinian evolution and a concept of meme proposed by R. Dawkins in 1976 [5]. In the context of *MA*, a meme can be considered as any local optimization method applied to a current solution during the evolution process. Memetic algorithms represent a combination of population-based global optimization technique and local search procedures.

Recent results of the works [4, 6, 7] demonstrate that if a memetic algorithm receives no prior knowledge on the problem hand, it can produce a solution not only equal to the one obtained using an ordinary population algorithm but even worse. In addition, there are relatively few theoretical papers that would suggest any particular *MA* configuration for black-box optimization problems. As a result, many scientists tend to utilize adaptive algorithms, which are capable of selecting the most suitable local optimization techniques for a particular search sub-domain during the evolution process.

Nowadays, many scientists actively work on the alternative approach, namely, an exploratory landscape analysis (*LA*) of an objective function [8, 9]. Instead of a dynamic adaptation of an algorithm during the optimization process, it is proposed to extract some information on the objective function's landscape and topology at the cost of additional evaluations (1…10 % of total computational budget). Landscape analysis methods identify either search sub-domains with rugged or smooth topology or sub-domains where values of the objective function are almost identical. In the works [10–12] several universal *LA* methods were proposed, including Cell Mapping and Information Content.

A class of Mind Evolutionary Computation (*MEC*) algorithms is considered [13–15] in this work. These algorithms belong to a family of methods inspired by a human society and simulate some aspects of a human behavior. An individual s is considered as an intelligent agent which operates in a group S made of analogous individuals. During the evolution process each individual is affected by other individuals within a group. This simulates the following logic. In order to achieve a high position within its group, an individual has to learn from the most successful individuals in this group. Groups themselves should follow the same principle to stay alive in the intergroup competition.

In this work, the Simple *MEC* algorithm is considered. It belongs to a class of *MEC* algorithms and was selected for investigation due to the following reasons: nowadays this algorithm and its modifications are successfully used for solving various optimization problems [6, 15]; the algorithm is highly suitable for parallel computations, especially for loosely coupled systems [7]; the algorithm is not sufficiently studied — there are relatively few modifications of *SMEC* (while, for instance, tens of various modifications are known for particle swarm optimization [2]).

We propose an adaptive multi-memetic modification of SMEC algorithm, which includes a stage of landscape analysis for composing a set of basic adaptation strategies; software implementation of the algorithm is also presented. Performance investigation was carried out with a use of multi-dimensional benchmark functions of different classes.

Problem statement and *SMEC* algorithm. A global deterministic unconstrained minimization problem is considered in this work

$$\min_{X \in R^n} \Phi(X) = \Phi\left(X^*\right) = \Phi^*, \tag{1}$$

where $\Phi\left(X\right)$ is the scalar objective function; $\Phi\left(X^*\right) = \Phi^*$ is its required minimal value; $X = \left(x_1, x_2, \ldots, x_n\right)$ is n-dimensional vector of variables; R^n is n-dimensional arithmetical space. A domain D_0 is defined as follows:

$$D_0 = \left\{X \mid x^{\min} \le x_i \le x^{\max},\ i \in \left[1:n\right]\right\} \tag{2}$$

and used for generating the initial population of solutions.

A population in the *SMEC* algorithm consists of leading groups $S^b = \left(S_1^b, S_2^b, \ldots, S_{|S^b|}^b\right)$ and lagging groups $S^w = \left(S_1^w, S_2^w, \ldots, S_{|S^w|}^w\right)$; the number of individuals within each group is set to be the same and equals $|S|$. The *SMEC* algorithm is based on the following procedures: initialization, similartaxis and dissimilation.

The initialization stage creates groups S^b, S^w and put them in the search domain. We illustrate the initialization stage by an example of the group S_i.

1. Generate a random vector $X_{i,1}$ whose components are distributed uniformly within the corresponding search subdomain. Identify this vector with the individual $s_{i,1}$ of the group S_i.

2. Determine the initial coordinates of the rest of the individuals in the group using the formula

$$X_{i,j} = X_{i,1} + N_n(\sigma),\ j \in [2:|S|], \tag{3}$$

where $N_n(\sigma)$ is n-dimensional vector of independent random real numbers, distributed normally with math expectation and standard deviation equaling 0 and σ respectively.

The similar-taxis stage implements a local search inside every group S^b, S^w and can be described as follows.

1. Determine the current best individual s_{i,j_b}, $j_b \in [1:|S|]$, of the group S_i.

2. Determine new coordinates of the rest individuals $s'_{i,j}$, $j \in [1:|S|]$, $j \neq j_b$ in this group using formula (3).

3. Calculate the objective function's values for all individuals in the group $\Phi'_{i,j} = \Phi(X'_{i,j})$, $j \in [1:|S|]$. Here vector $X'_{i,j}$ corresponds to the individual $s'_{i,j}$.

4. Determine a new winner of the group s'_{i,k_b}, $k_b \in [1:|S|]$, as an individual with the lowest value of the objective function Φ.

The dissimilation stage implements a global search between all groups and uses the following steps.

1. Determine the best individuals of all groups S^b, S^w.

2. Compare their scores and rank them. If a score of any leading group S_i^b is less than a score of any lagging group S_j^w, then the latter becomes a leading group and the leading group becomes a lagging one. If a score of any lagging group S_j^w is lower than scores of all leading groups for ω consecutive iterations, then it's removed from the population.

3. Replace each removed group with a new one using the initialization procedure.

Similar-taxis and dissimilation stages are repeated iteratively while the best obtained value of the objective function $\Phi(X)$ changes. When the best obtained value stops changing, the winner of the best group is selected as a solution to the optimization problem (1).

Modified Multi-Population *SMEC*. An extension performance investigation of the *SMEC* algorithm was carried out by the authors in the work [16] in order to determine the influence of the free parameters' values on the efficiency of the algorithm. The following parameters were considered: σ, the standard deviation, utilized when generating new individuals in groups; ω, the removing frequency of lagging groups; η, the ratio between numbers of leading and lagging groups in the population.

Results of that work demonstrated a strong dependency of the algorithm's efficiency on the values of those parameters and also revealed that optimal values of those parameters differ for various objective functions Φ.

These conclusions allowed formulating a multi-population modification of *SMEC* algorithm. Instead of a single population, a set of sub-populations $K = \left(K_1, K_2, \ldots, K_{|K|}\right)$ is considered.

Evolution inside each sub-population is governed by the individual values of the free parameters described above. Those values can be set based on the distinct features of the objective function's topology if they are known a priori or determined with a use of some heuristics [16].

In the modified algorithm, the required optimal value Φ^* of the objective function is determined as the minimum of the values obtained by every sub-population independently

$$\Phi^* = \min_l \Phi_l^*, \ l \in \left[1 : |K|\right]. \tag{4}$$

The modified initialization stage of sub-population K_l is executed using an individual value σ_l. As a result, each sub-population has its own level of search intensification and diversification and provides a balance for the whole multi-population. Parameter σ_l is also used at the similar-taxis stage within every sub-population K_l.

The ratio between numbers of leading and lagging groups in every sub-population is determined by the value η_l. An experimental result shows [16], that a small number of lagging groups affects adversely the diversification properties and can be useful only when the computational process approaches stagnation.

The dissimilation stage in the modified algorithm is executed with the use of an individual value ω_l. Based on the experiments [16] it was determined that the higher the removing frequency is (the lower value of ω_l), the lower are diversification properties of the algorithm because lagging groups don't have enough time to explore their search sub-domains.

Landscape analysis procedure. In this work, we present a new method for landscape analysis of the objective function for a case when no initial information is available with any limitation on a problem's dimension. This method allows putting a function into one of six groups, each group corresponds to a static adaptation strategy which is utilized on the further stages. Each proposed group corresponds to a certain type objective function's topology. This classification was proposed based on the experimental studies [16] as the new algorithm is designed for particularly for parallel loosely coupled systems. It allows considering different parallelization procedures in terms of load balancing which will be studied in the future works. The proposed *LA* procedure can be described as follows.

1. Generate N quasi-random *n*-dimensional vectors with domain D_0. Here N is a total number of all groups in a multi-population (a free parameter of the algorithm). In this work LP_τ sequence was used to generate quasi-random numbers since it provides a high-quality coverage of a domain [17].

2. For every $X_r, r \in [1:N]$, calculate the corresponding values of the objective function Φ_r and sort those vectors in ascending order of values Φ_r, $r \in [1:N]$.

3. Equally divide a set of vectors $(X_1, X_2, \ldots, X_N)$ into $|K|$ groups in accordance with a given number of sub-populations $|K|$ (one freer parameter).

4. For every group K_l, $l \in [1:|K|]$, calculate a value of its diameter d_l — a maximum Euclidian distance between any two individuals within this group [18].

5. Build a linear approximation for the dependency of diameter *d* on group number *l* using the least squares method [18].

6. Calculate an estimation of the size of domain D_0 using the formula

$$d_D = \sqrt{n\left(x^{\max} - x^{\min}\right)^2}. \tag{5}$$

Put the objective function Φ into one of the six categories (Table 1) based on the calculated parameters.

Table 1

Classification of objective functions based on the *LA* results

$d_j(l)$ increases	$d_j(l)$ neither increases nor decreases	$d_j(l)$ decreases
	$d_D / d_1 > 2$	
Nested sub-domains with the dense first domain (category I)	Non-intersected domains of the same size (category III)	Distributed domains with potential minima (category V)
	$d_D / d_1 \le 2$	
Nested sub-domains with the sparse first domain (category II)	Intersected domains of the same size (category IV)	Highly distributed domains with potential minima (category VI)

There are three possible cases for the approximated dependency $d(l)$: d can be an increasing function of l; d can decrease as l grows; $d(l)$ can be neither decreasing nor increasing. Within the scope of this work it is assumed that the latter scenario takes place when a slope angle of the approximated line is less than ±5°.

The ratio between d_D and d_1 helps to estimate the density of the first group K_1 with respect to the original domain D_0. In other words, we can understand whether vectors X with the least values of the function Φ are sparsely or densely distributed. We consider two possible cases:

$$\frac{d_D}{d_1} > 2, \ \frac{d_D}{d_1} \le 2.$$

Here the value 2 was obtained from the empirical studies [18].

Each of the six categories represents a certain topology of the objective function Φ and subsequently the rules for determining numerical values of the *SMEC* algorithm's free parameters, specified in the previous sections.

For objective functions that belong to the categories I and II, there is a high probability that the required global minimum is located within a domain defined by group K_1. In this case, values of the parameters σ, η and ω are selected to provide a high level of intensification properties for the first groups and increase diversification properties for groups with high index numbers.

For objective functions that belong to the categories III and IV, values of the parameters σ, η and ω are selected randomly from given intervals.

Finally, objective functions that belong to the categories V and VI can be characterized with a large search sub-domain that can include the desired minimum. In this case, the values of the parameters are selected to increase the diversification properties of the first groups.

Figure 1 displays a few examples of the proposed landscape analysis procedure for various two-dimensional benchmark functions, including the traditional functions of Rastrigin, Griewank and Styblinski — Tang [19] as well as the composition functions from CEC 2014 [20].

Multi-memetic modification. Memetic algorithms appeared to be successful for solving optimization problems in various fields. However, just like for any other heuristic algorithm, the adjustment of the free parameters is required for their efficient performance. For instance, it is crucial to choose the most suitable meme for a problem in hand. It was demonstrated in Ref. [4] that this choice affects *MA's* efficiency greatly.

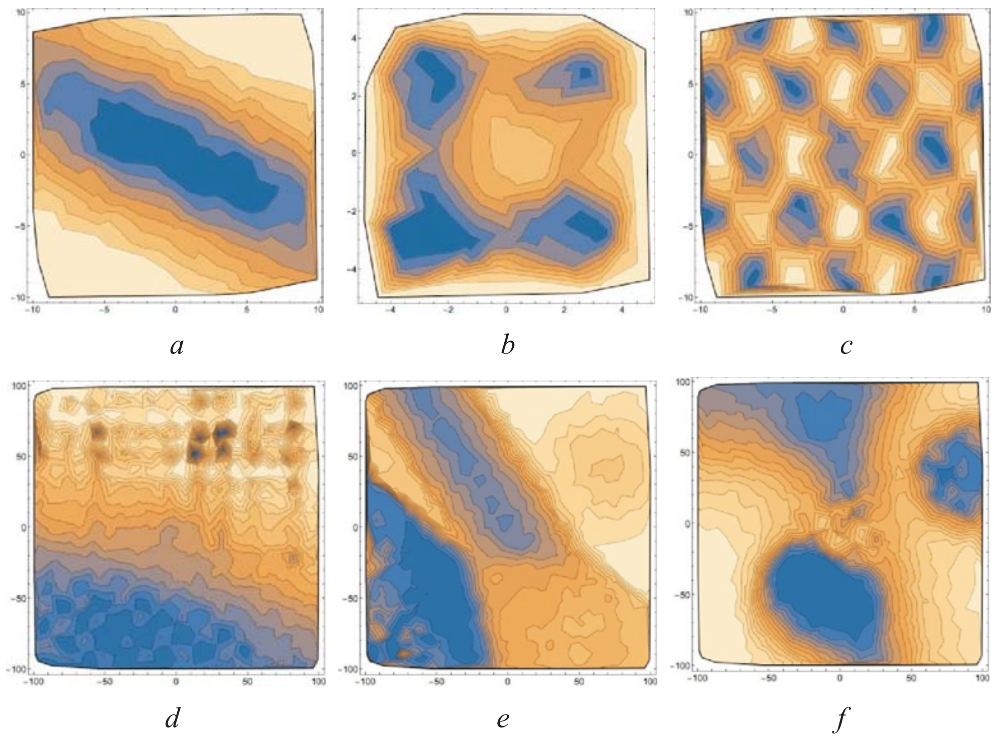


Fig. 1. Results of the landscape analysis procedure for a few benchmark functions: *a* Zakharov function (category I); *b* Griewank function (category II); *c* Styblinski — Tang function (category IV); *d* composition function 2 from CEC'14 (category II); *e* composition function 4 from CEC'14 (category I); *f* composition function 5 from CEC'14 (category IV)

Nowadays, there are many papers, which propose different schemes for hybridization of meta-heuristic methods with local search procedures [4, 21, 22]. Often these algorithms utilize complicated heuristic local search procedures designed specifically for certain problems. Despite high efficiency their applications are limited because in many real-world problems scientists do not have any prior information on what meme they should use for a particular problem. Multi-memetic algorithms were proposed to overcome these difficulties [23].

A distinct feature of this algorithm is a use of several memes during the evolution. In this class of algorithms, a decision on which meme to use for one or another individual in a population is usually made dynamically. Such a class of *MA* provides a competition between different specialized local search methods. As a result, the algorithm preserves high efficiency despite lacking any initial information about a problem under investigation.

When designing a multi-memetic algorithm, one should carefully select a practical strategy for applying one meme or another from a set of available memes $M = (m_j, j \in [1:|M|])$. The choice can be made based on the characteristics of memes or/and search sub-domains [24, 25].

In this work three ($|M| = 3$) local search methods were utilized, namely, Nelder — Mead method [26], Solis — Wets method [27], and Monte-Carlo method [1]. Only zero-order methods were used to deal with problems where the objective function's derivative is not available explicitly and its approximation is computationally expensive. While the Nelder — Mead method is purely local, Solis — Wets and Monte-Carlo methods can solve both local and global optimization tasks depending on their parameters. As a hyper-heuristic for selecting a meme the following rule was utilized.

1. Within sub-population K_l in every group select the best individual s'_{l,i,k_b}.

2. Launch all available memes from their current positions. An iteration number for each meme is limited with P.

3. Select the most efficient meme for every group based on the obtained values of $\Phi(X)$.

4. Use the best meme m_b to refine all individuals' positions in a group at the similar-taxis stage.

To save computational budget, memes are utilized with an individual frequency for each group $\omega_l / 2$.

The algorithm proposed in this paper with a use of multi-population concept and multi-memetic approach was named Multi-Memetic Modified *MEC* (*M3MEC*). The concept of multi-population along with the incorporated landscape analysis procedure allows making a rough static adaptation of the algorithm to the objective function at the very beginning of evolution process at the cost of small computational expenses. Utilization of memes, in turn, helps the algorithm to correct possible errors of static adaptation during the evolution due to a closer investigation of search sub-domains.

Performance investigation. The *M3MEC* algorithm was implemented by authors in *Wolfram Mathematica*. Software implementation has a modular structure, which helps to modify algorithms easily and extend them with additional assisting methods.

A study was carried out in this work to compare the efficiency of the proposed algorithms with a landscape analysis procedure and *SMEC* with optimal values of the free parameters from the work [16]. All numeric experiments were carried out using the multi-start method with 50 launches. The best obtained value of an objective function Φ^* as well as its average value $\overline{\Phi}$ based on the results of all launches were utilized as the performance indices for comparison two algorithms and their software implementations along with the average iteration number $\overline{\lambda}$.

Benchmark functions. Multi-dimensional benchmark optimization functions are considered in this paper [19]. An original domain for generating the initial population equals

$$D_0 = \{ X \mid -10 \le x_i \le 10,\ i \in [1:n] \}. \quad (6)$$

During the experiments the following values of free parameters were used for the *SMEC* algorithm: standard deviation $\sigma = 0.1$; total number of groups $\gamma = 100$; number of individuals in each group $|S| = 50$; ratio between numbers of leading and lagging groups number of groups $\eta = 1$; removing frequency for lagging groups $\omega = 20$. In order to provide approximately the same level of computational expenses per iteration the following settings were used for *M3MEC*: number of sub-populations $|K| = 5$; number of groups in every sub-population $\gamma_l = 16$; number of individuals in each group within the sub-population $|S_l| = 30$.

The number of stagnation iterations $\lambda_{stop} = 50$ was used as a termination criterion for the algorithms. Tolerance used for identifying stagnation was equal to $\varepsilon = 10^{-5}$.

Experimental results. Obtained results (Table 2) demonstrate superiority of the proposed algorithm over Simple Mind Evolutionary Computation algorithm.

Table 2

Numerical experiment results

Function	*SMEC*		*M3MEC*	
	$n = 8$	$n = 16$	$n = 8$	$n = 16$
Ackley function	$\overline{\Phi_1} = 3.9E + 0$ $\Phi_1^* = 3.3E - 1$	$\overline{\Phi_1} = 4.3E + 0$ $\Phi_1^* = 8.7E - 1$	$\overline{\Phi_1} = 1.01E - 3$ $\Phi_1^* = 2.34E - 6$	$\overline{\Phi_1} = 8.31E - 2$ $\Phi_1^* = 2.44E - 5$
Dixon function	$\overline{\Phi_2} = 6.7E - 1$ $\Phi_2^* = 3.0E - 1$	$\overline{\Phi_2} = 2.3E + 0$ $\Phi_2^* = 1.9E + 0$	$\overline{\Phi_2} = 0.47E + 0$ $\Phi_2^* = 2.09E - 6$	$\overline{\Phi_2} = 1.12E - 1$ $\Phi_2^* = 3.16E - 4$
Griewank function	$\overline{\Phi_3} = 3.7E - 2$ $\Phi_3^* = 2.9E - 2$	$\overline{\Phi_3} = 4.2E - 2$ $\Phi_3^* = 2.1E - 2$	$\overline{\Phi_3} = 6.95E - 2$ $\Phi_3^* = 1.97E - 2$	$\overline{\Phi_3} = 6.47E - 2$ $\Phi_3^* = 1.85E - 5$
Levy function	$\overline{\Phi_4} = 2.6E + 0$ $\Phi_4^* = 0.9E + 0$	$\overline{\Phi_4} = 1.1E + 1$ $\Phi_4^* = 3.9E + 0$	$\overline{\Phi_4} = 6.28E - 5$ $\Phi_4^* = 1.47E - 7$	$\overline{\Phi_4} = 4.19E - 1$ $\Phi_4^* = 2.92E - 6$
Powell function	$\overline{\Phi_5} = 1.2E - 1$ $\Phi_5^* = 1.1E - 1$	$\overline{\Phi_5} = 2.8E + 0$ $\Phi_5^* = 1.5E + 0$	$\overline{\Phi_5} = 9.76E - 4$ $\Phi_5^* = 3.92E - 5$	$\overline{\Phi_5} = 1.03E - 1$ $\Phi_5^* = 4.49E - 4$

End of Table 2

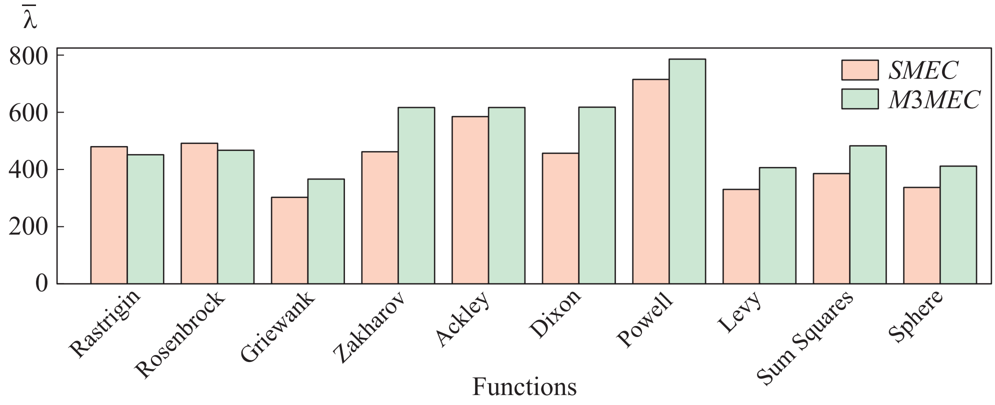
Function	*SMEC*		*M3MEC*	
	$n = 8$	$n = 16$	$n = 8$	$n = 16$
Rastrigin function	$\overline{\Phi_6} = 6.3\text{E} + 1$ $\Phi_6^* = 4.0\text{E} + 1$	$\overline{\Phi_6} = 2.4\text{E} + 2$ $\Phi_6^* = 1.4\text{E} + 2$	$\overline{\Phi_6} = 1.29\text{E}{+}0$ $\Phi_6^* = 1.33\text{E} - 2$	$\overline{\Phi_6} = 5.66\text{E} + 1$ $\Phi_6^* = 9.97\text{E} + 0$
Rosenbrock function	$\overline{\Phi_7} = 6.5\text{E} + 0$ $\Phi_7^* = 2.4\text{E} + 0$	$\overline{\Phi_7} = 3.2\text{E} + 1$ $\Phi_7^* = 2.5\text{E} + 1$	$\overline{\Phi_7} = 5.17\text{E} - 1$ $\Phi_7^* = 1.04\text{E}{-}\ 3.$	$\overline{\Phi_7} = 2.21\text{E} + 1$ $\Phi_7^* = 2.64\text{E} - 1$
Sphere function	$\overline{\Phi_8} = 2.4\text{E} - 2$ $\Phi_8^* = 2.1\text{E} - 2$	$\overline{\Phi_8} = 1.9\text{E} - 1$ $\Phi_8^* = 1.8\text{E} - 1$	$\overline{\Phi_8} = 1.71\text{E} - 5$ $\Phi_8^* = 2.94\text{E} - 7$	$\overline{\Phi_8} = 6.97\text{E} - 3$ $\Phi_8^* = 2.18\text{E}{-}6$
Sum of Squares function	$\overline{\Phi_9} = 8.0\text{E} - 2$ $\Phi_9^* = 7.3\text{E} - 2$	$\overline{\Phi_9} = 1.6\text{E} + 0$ $\Phi_9^* = 9.4\text{E} - 1$	$\overline{\Phi_9} = 3.99\text{E} - 5$ $\Phi_9^* = 2.34\text{E} - 7$	$\overline{\Phi_9} = 1.08\text{E} - 2$ $\Phi_9^* = 1.51\text{E} - 6$
Zakharov function	$\overline{\Phi_{10}} = 4.6\text{E} - 2$ $\Phi_{10}^* = 3.9\text{E} - 2$	$\overline{\Phi_{10}} = 3.9\text{E} - 1$ $\Phi_{10}^* = 3.1\text{E} - 1$	$\overline{\Phi_{10}} = 4.68\text{E} - 4$ $\Phi_{10}^* = 9.03\text{E} - 7$	$\overline{\Phi_{10}} = 6.91\text{E} - 1$ $\Phi_{10}^* = 2.72\text{E} - 4$

For the majority of the benchmark functions the results obtained with the use of *M3MEC* are better than ones obtained using *SMEC* by several orders of magnitude both for the average values $\overline{\Phi}$ and least found values Φ^*. While the high accuracy of Φ^* is caused by memes [25], decrease in the average values $\overline{\Phi}$ is conditioned upon *LA* procedure.

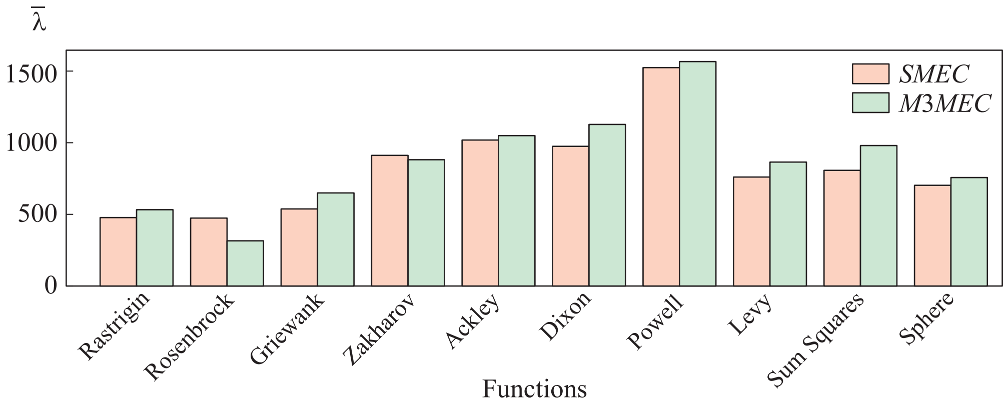
On the other hand, *M3MEC* algorithm requires more iterations than the *SMEC* algorithm (Fig. 2). However, in terms of the objective function's evaluations this advantage of *SMEC* is not obvious. It was discovered that sub-populations that were made of individuals with high values of $\Phi(X)$ often requires more iterations to reach stagnation. This sometimes makes a decent number of last iterations unnecessary and can be overcome with a use of more advanced termination criteria.

Conclusion. This paper presents a new two-stage adaptive multi-memetic algorithm with the incorporated landscape analysis procedure. The algorithm is capable of adapting to various objective functions using both static and dynamic adaptation. Static adaptation was implemented with a use of landscape analysis, while dynamic adaptation was made possible by utilizing several memes.

A comparative study of the proposed method with a traditional *SMEC* algorithm was carried out. Obtained results demonstrate the superiority of proposed technique over Simple Mind Evolutionary Computation algorithm with tuned values of the free parameters. *M3MEC* provides a high-quality solution for multi-dimensional optimization problems but requires more



a



b

Fig. 2. Iteration number for different benchmark functions:
a dimension $n = 8$; *b* dimension $n = 16$

evaluations of an objective function. This drawback, however, can be overcome with a use more advanced termination criteria.

Further research will be devoted to the investigation of different strategies for selecting memes and parallelization schemes for *M3MEC*.

REFERENCES

[1] Weise T. Global optimization algorithms — theory and application. University of Kassel, 2008.

[2] Karpenko A.P. Sovremennye algoritmy poiskovoy optimizatsii. Algoritmy, vdokhnovlennye prirodoy [Modern algorithms of search optimization. Nature-inspired algorithms]. Moscow, BMSTU Publ., 2014.

[3] Krasnogor N. Studies on the theory and design space of memetic algorithms. PhD Thesis. University of the West of England, 2002.

[4] Neri F., Cotta C., Moscato P. Handbook of memetic algorithms. Springer, 2011.

[5] Dawkins R. The Selfish Gene. Oxford Univ. Press, 1976.

[6] Sakharov M.K., Karpenko A.P. Multi-memetic mind evolutionary computation algorithm for loosely coupled systems of desktop computers. *Nauka i obrazovanie: nauchnoe izdanie MGTU im. N.E. Baumana* [Science and Education: Scientific Publication], 2015, no. 10 (in Russ.). DOI: 10.7463/1015.0814435

[7] Sakharov M., Karpenko A. New parallel multi-memetic MEC-based algorithm for loosely coupled systems. *Proc. VII Int. Conf. Optimization Methods and Application "Optimization and applications" OPTIMA–2016*, 2016, pp. 124–126.

[8] Mersmann O., Bischl B., Trautmann H., et al. Exploratory landscape analysis. *GECCO'11*, ACM, 2011, pp. 829–836. DOI: 10.1145/2001576.2001690

[9] Vassilev V.K., Fogarty T.C., Miller J.F. Smoothness, ruggedness and neutrality of fitness landscapes: from theory to application. In: Ghosh A., Tsutsui S. (eds). *Advances in Evolutionary Computing. Natural Computing Series*. Berlin, Heidelberg, Springer, 2003, pp. 3–44. DOI: https://doi.org/10.1007/978-3-642-18965-4_1

[10] Bischl B., Mersmann O., Trautmann H., et al. Algorithm selection based on exploratory landscape analysis and cost-sensitive learning. *GECCO'14*, ACM, 2012, pp. 313–320.

[11] Kerschke P., Preuss M., Hernándezet C., et al. Cell mapping techniques for exploratory landscape analysis. In: Tantar AA., et al. (eds). *EVOLVE — A Bridge between Probability, Set Oriented Numerics, and Evolutionary Computation V. Advances in Intelligent Systems and Computing*, vol. 288. Cham, Springer, 2014, pp. 115–131.

[12] Muñoz M.A., Kirley M., Halgamuge S.K. Exploratory landscape analysis of continuous space optimization problems using information content. *IEEE Trans. Evol. Comput.*, 2015, vol. 19, iss. 1, pp. 74–87. DOI: 10.1109/TEVC.2014.2302006

[13] Sun Ch., Sun Y., Wang W. A survey of MEC: 1998–2001. *IEEE SMC 2002 Int. Conf.*, 2002, p. 9. DOI: 10.1109/ICSMC.2002.1175629

[14] Jie J., Zeng J., Ren Y. Improved mind evolutionary computation for optimizations. *Proc. 5th World Congress Intell. Contr. Automat.*, 2004, pp. 2200–2204.
DOI: 10.1109/WCICA.2004.1341978

[15] Jie J., Zeng J., Han C. An extended mind evolutionary computation model for optimizations. *Appl. Math. Comput.*, 2007, vol. 185, iss. 2, pp. 1038–1049.
DOI: 10.1016/j.amc.2006.07.037

[16] Sakharov M., Karpenko A. Performance investigation of mind evolutionary computation algorithm and some of its modifications. In: Abraham A., Kovalev S., Tarassov V., Snášel V. (eds). *Proc. 1st Int. Sci. Conf. IITI'16. Proceedings of the First International Scientific Conference "Intelligent Information Technologies for Industry" (IITI'16). Advances in Intelligent Systems and Computing*, vol. 450. Cham, Springer, 2016, pp. 475–486. DOI: https://doi.org/10.1007/978-3-319-33609-1_43

[17] Sobol' I.M. On the distribution of points in a cube and the approximate evaluation of integrals. *USSR Comput. Maths. Phys.*, 1967, vol. 7, iss. 4, pp. 86–112.
DOI: 10.1016/0041-5553(67)90144-9

[18] Sakharov M., Karpenko A. A new way of decomposing search domain in a global optimization problem. In: Abraham A., Kovalev S., Tarassov V., Snasel V., Vasileva M., Sukhanov A. (eds). *Proceedings of the Second International Scientific Conference "Intelligent Information Technologies for Industry" (IITI'17). Advances in Intelligent Systems and Computing*, vol. 679. Cham, Springer, 2017, pp. 398–407.
DOI: https://doi.org/10.1007/978-3-319-68321-8_41

[19] Floudas A.A., Pardalos P.M., Adjiman C., et al. Handbook of test problems in local and global optimization. Boston, MA, Springer, 1999.

[20] Liang J.J., Qu B.Y., Suganthan P.N. Problem definitions and evaluation criteria for the CEC 2014 special session and competition on single objective real-parameter numerical optimization. Technical Report. Zhengzhou Univ., Nanyang Technological Univ., 2013.

[21] Nguyen Q.H., Ong Y.S., Krasnogor N. A study on the design issues of memetic algorithm. *IEEE Congress on Evolutionary Computation*, 2007, pp. 2390–2397.
DOI: 10.1109/CEC.2007.4424770

[22] Hart W., Krasnogor N., Smith J.E., Memetic evolutionary algorithms. In: Hart W.E., Smith J.E., Krasnogor N. (eds). *Recent Advances in Memetic Algorithms. Studies in Fuzziness and Soft Computing*, vol. 166. Berlin, Heidelberg, Springer, 2005, pp. 3–27.
DOI: https://doi.org/10.1007/3-540-32363-5_1

[23] Krasnogor N., Blackburne B., Burke E.K., et al. Multimeme algorithms for protein structure prediction. In: Guervós J.J.M., Adamidis P., Beyer H.G., Schwefel H.P., Fernández-Villacañas J.L. (eds). *Parallel Problem Solving from Nature — PPSN VII. PPSN 2002. Lecture Notes in Computer Science,* vol. 2439. Berlin, Heidelberg, Springer, pp. 769–778.
DOI: https://doi.org/10.1007/3-540-45712-7_74

[24] Ong Y.S., Lim M.H., Zhu N., et al. Classification of adaptive memetic algorithms: a comparative study. *IEEE Trans. Syst., Man, Cybern. {B}*, 2006, pp. 141–152.
DOI: 10.1109/TSMCB.2005.856143

[25] Karpenko A.P., Sakharov M.K. Multi-memes global optimization based on the algorithm of mind evolutionary computation. *Informatsionnye tekhnologii* [Information Technologies], 2014, no. 7, pp. 23–30 (in Russ.).

[26] Nelder J.A., Meade R. A simplex method for function minimization. *Comput. J.*, 1965, vol. 7, iss. 4, pp. 308–313. DOI: 10.1093/comjnl/7.4.308

[27] Solis F.J., Wets R. J.-B. Minimization by random search techniques. *Math. Oper. Res.*, 1981, vol. 6, no. 1, pp. 19–30. DOI: 10.1287/moor.6.1.19

Karpenko A.P. — Dr. Sc. (Phys.-Math.), Professor, Head of Department of Computer Added Design, Bauman Moscow State Technical University (2-ya Baumanskaya ul. 5, str. 1, Moscow, 105005 Russian Federation).

Sakharov M.K. — Assist. Professor, Department of Computer Added Design, Bauman Moscow State Technical University (2-ya Baumanskaya ul. 5, str. 1, Moscow, 105005 Russian Federation).

Please cite this article as:

Karpenko A.P., Sakharov M.K. New adaptive multi-memetic global optimization algorithm. *Herald of the Bauman Moscow State Technical University, Series Natural Sciences*, 2019, no. 2, pp. 17–31. DOI: 10.18698/1812-3368-2019-2-17-31