Journal of Global Optimization manuscript No.
(will be inserted by the editor)

Efficient Multicriterial Optimization Based on Intensive Reuse of Search Information

Victor Gergel · **Evgeny Kozinov**

Received: date / Accepted: date

Abstract This paper proposes an efficient method for solving complex multicriterial optimization problems, for which the optimality criteria may be multiextremal and the calculations of the criteria values may be time-consuming. The approach involves reducing multicriterial problems to global optimization ones through minimax convolution of partial criteria, reducing dimensionality by using Peano curves and implementing efficient information-statistical methods for global optimization. To efficiently find the set of Pareto-optimal solutions, it is proposed to reuse all the search information obtained in the course of optimization. The results of computational experiments indicate that the proposed approach greatly reduces the computational complexity of solving multicriterial optimization problems.

Keywords Decision making · multicriterial optimization · scalarization · dimensionality reduction · global optimization algorithm · search information · computational complexity

1 Introduction

Multicriterial optimization (MCO) is a field of intensive research and applications – see, for example, monographs [5–8, 26, 30, 49] and reviews of theoretical and practical results in this field [9, 11, 23, 28, 29, 48].

It should be noted that problems of multicriterial optimization are among the most complex optimization problems. The statement of MCO problems covers many classes of optimization problems, including unconstrained optimization, nonlinear programming, global

This study was supported by the Russian Science Foundation, project No 16-11-10150.

V. Gergel
Institute of Information Technology, Mathematics and Mechanics
Lobachevsky State University of Nizhni Novgorod
Nizhni Novgorod, Russia
E-mail: gergel@unn.ru

E. Kozinov
Institute of Information Technology, Mathematics and Mechanics
Lobachevsky State University of Nizhni Novgorod
Nizhni Novgorod, Russia
E-mail: evgeny.kozinov@itmm.unn.ru

optimization, etc. In addition, the partial criteria of MCO problems are usually contradictory in nature. As a result, solving the MCO problems consists in finding some compromise (efficient) solutions, which cannot be improved with respect to all partial criteria simultaneously. In the course of optimization, one may need to find a number of different efficient solutions or even the entire set of non-dominated solutions (the Pareto domain).

A widely used method for finding the desired solution is to scalarize of the MCO criteria where the partial criteria are combined into a single generalized efficiency criterion – see, for instance, [8,9]. When using such an approach, the convolution coefficients can be considered indicators of the importance of partial criteria – the greater the coefficient value for some criterion, the greater the contribution of this partial criterion to the single scalar criterion.

The need to find several (or the entire set of) efficient solutions increases dramatically the computational complexity in solving MCO problems. Furthermore, it should be noted that the partial criteria can be multiextremal, and the calculations of the criteria values can be time-consuming. In such circumstances, even the finding of just one compromise solution requires a large amount of computations. Thus, finding several (or the whole set of) efficient solutions becomes a problem of huge computational complexity.

To overcome the computational complexity of MCO problems, this paper proposes an approach based on the following two main ideas. First, to solve MCO problems, efficient global optimization algorithms will be used, which are based on the information-statistical theory of global optimization [37,38]. Second, to accelerate computations, all the search information obtained during the optimization process is totally reused. In general, the reuse of search information will result in a substantial reduction of the amount of computations in searching for the subsequent efficient solutions.

Initially, the proposed approach was estimated experimentally in [47]. In the present work, the developed approach that involves the reuse of the search information has been generalized up to a general scheme for solving multicriterial optimization problems. Within the framework of this scheme, various global optimization algorithms can be used. The efficiency of the proposed scheme has been studied on the example of the information-statistical algorithms, for which the theoretical estimates have been obtained.

The rest of the paper is structured as follows. Section 2 states the multicriterial optimization problem and describes the approach for reducing MCO problems to one-dimensional global optimization ones. In Section 3, the proposed approach for solving multicriterial global optimization problems is considered. Section 4 presents the results of numerical experiments. The obtained results are discussed and the directions for further research are considered in the Conclusion.

2 Problem Statement

The *multicriterial optimization* (MCO) problem can be formulated as follows:

$$f(y) = (f_1(y), f_2(y), \ldots, f_s(y)) \to \min, y \in D, \tag{1}$$

where

- $y = (y_1, y_2, \ldots, y_N)$ is a vector of varied parameters,
- N is the MCO problem dimensionality,
- D is the search domain, which is a N-dimensional box

$$D = \{y \in R^N : a_i \le y_i \le b_i, 1 \le i \le N\}$$

with the given boundary vectors a and b. Without a loss of generality, it is supposed that the values of the partial criteria are not negative and their decrease corresponds to an increase in the solution efficiency.

Any efficient point can be considered as *partial solution* of the MCO problem. In general, solving a MCO problem may require finding the whole set of the Pareto-optimal points $PD \subseteq D$ (*the complete solution of the MCO problem*).

In this paper, the MCO problems are considered in relation to the most complicated decision making problems, in which the partial criteria $f_i(y)$, $1 \le i \le s$, can be multiextremal, and obtaining the criteria values at the search domain points $y \in D$ may require time-consuming computations. We also suppose that the partial criteria $f_i(y)$, $1 \le i \le s$, satisfy the Lipschitz condition

$$|f_i(y') - f_i(y'')| \le L_i \|y' - y''\|, y', y'' \in D, 1 \le i \le s. \tag{2}$$

Satisfying the Lipschitz condition allows us to compute numerical estimates of the possible partial criteria values based on their calculated values on a finite set of points within the search domain D.

The proposed approach to solving MCO problems is based on the following two key techniques.

1. Scalarization. One of the widely used approaches in addressing MCO problems is the scalarization technique, which employs various convolution schemes of the partial criteria $f_i(y)$, $1 \le i \le s$, to obtain a scalar generalized criterion $F(\lambda, y)$ – see, for instance, [8,9]. This approach makes it possible to reduce the solution of problem (2) to the solution of a family of nonlinear optimization subproblems

$$\min F(\lambda, y), y \in D \tag{3}$$

where $\lambda = (\lambda_1, \lambda_2, \ldots, \lambda_s)$ is the vector of coefficients used for formulating the scalar criterion $F(\lambda, y)$. It is assumed that in addition to (3) the scalar criterion $F(\lambda, y)$ also satisfies the Lipschitz condition

$$|F(\lambda, y') - F(\lambda, y'')| \le L \|y' - y''\|, y', y'' \in D, 1 \le i \le s. \tag{4}$$

In this paper, it is proposed to use the minimax convolution scheme in the following form:

$$F(\lambda, y) = \max_{1 \le i \le s} (\lambda_i f_i(y)), \tag{5}$$

with the given coefficients λ_i, $1 \le i \le s$, that should belong to the set

$$\Lambda \subset R^s : \sum_{i=1}^{s} \lambda_i = 1, \lambda_i \ge 0, 1 \le i \le s.$$

These coefficients λ_i, $1 \le i \le s$, can be considered indicators of partial criteria importance – the greater the value of some criterion coefficient λ_i, the greater the contribution of this partial criterion f_i in the scalar criterion $F(\lambda, y)$.

An important property of the minimax convolution scheme is the necessity and sufficiency of this approach for solving MCO problems: the result of minimizing $F(\lambda, y)$ from (6) yields an efficient solution[1] of a MCO problem. And, vice versa, every efficient solution of a MCO problem can be obtained through minimizing $F(\lambda, y)$ for the corresponding convolution coefficients λ_i, $1 \leq i \leq s$, – see, for instance, [19,29].

It should be pointed out that in the process of solving a MCO problem, the concept of the importance of partial criteria may change, which may require solving several subproblems (6) using different coefficients $\lambda \in \Lambda$. As a result, the possibility of finding a number (or the whole set) of efficient solutions with adequate computational costs becomes the key issue in solving MCO problems.

2. Dimensionality reduction. To decrease computational complexity of solving multidimensional optimization problems, the dimensionality reduction can be applied by using *Peano curves or mappings* $y(x)$, that map unambiguously the interval $[0,1]$ on a N-dimensional hypercube D [35,37,38].

Following such a reduction scheme, the initial multidimensional global optimization subproblem (4)-(6) is reduced to the one-dimensional subproblem:

$$F(\lambda, y(x^*)) = \min_{x \in [0,1]} F(\lambda, y(x)). \tag{6}$$

It should be mentioned that the reduced function $F(\lambda, y(x))$ satisfies the uniform Hölder condition, i.e.

$$|F(\lambda, y(x')) - F(\lambda, y(x''))| \leq H|x' - x''|^{1/N}, x', x'' \in [0,1], \tag{7}$$

where the constant H is determined by the relation $H = 2L\sqrt{N+3}$, where L is the Lipschitz constant from (5), and N is the dimensionality of the optimization problem (2).

The computational scheme for computing the values of the function $F(\lambda, y(x))$ obtained through the dimensionality reduction is as follows:

- The optimization algorithm solves the reduced one-dimensional function $F(\lambda, y(x))$,
- Once the next trial point $x \in [0,1]$ is selected, a multidimensional image $y \in D$ using the mapping $y(x)$, is calculated,
- At the point $y \in D$ the value of the function $F(\lambda, y)$ is calculated,
- The calculated value $z = F(\lambda, y)$ will be used further as the value of the reduced one-dimensional function $F(\lambda, y(x))$ at the point $x \in [0,1]$.

The algorithms for constructing the numerical approximations of Peano curves are given in [38].

3 Methods for Solving Multicriterial Global Optimization Problems

As it was mentioned above the scalarization approach makes it possible to use efficient global optimization methods for solving MCO problems. Finding numerical estimates for the globally optimal points is usually provided by a coverage of the search domain D – see,

[1] More precisely, minimizing $F(\lambda, y)$ can result in obtaining a weakly-efficient solution (a set of weakly-efficient points includes the Pareto domain). This can be corrected by adding an additional correction value in (6) – see, for example, [19,29].

for instance, [12,22,24,25,27,31,35,37–39,41,44,51,52]. As a result, the computational complexity of solving the global optimization problems is very high even for a fairly small number of varied parameters (the dimensionality of the problem). A dramatic reduction of the required computations can be achieved if the computing grids generated for covering the search domain are non-uniform, with the denser optimization points only in the vicinity of the desired globally optimal solutions. Such non-uniform coverage can be constructed by performing adaptive computations. However, this is possible if and only if all the search information (the previous optimization points and the values of the optimized function at these points) obtained through the calculations is taken into account for selecting each iteration point.

3.1 Reusing Search Information within Multicriterial Global Optimization

As follows from the above discussion, to improve the efficiency of the global optimization algorithms, the search information (the trial points and the corresponding values of the objective function) should be accumulated and used intensively within optimization procedures. Search information in conjunction with the Lipschitz condition allows us to calculate the estimates of the possible values of the optimized function. Besides, search information makes possible the adaptive selection of the points for the subsequent global search iterations. And contrariwise, a reduction in the amount of the search information usually leads to a significant increase in the number of iterations required to meet the stopping condition.

The search information obtained during the calculations may be represented as the set

$$A_k = \{(x_i, z_i, v_i)^T : 1 \le i \le k\} \tag{8}$$

where

- x_i, $1 \le i \le k$, are the points of the executed global optimization iterations; the points should be located in the increasing order;
- z_i, $1 \le i \le k$, are the scalar criterion values for the optimization subproblem (7) at the points x_i, $1 \le i \le k$, i.e.:

$$z_i = F(\lambda, x_i), 1 \le i \le k;$$

- v_i, $1 \le i \le k$, are the values of the vector criterion $f(y(x))$ of the initial multicriterial optimization problem (2) calculated at the points x_i, $1 \le i \le k$, i.e.:

$$v_i = (v_i^1, v_i^2, \ldots, v_i^s), v_i^j = f_j(y(x_i)), 1 \le j \le s, 1 \le i \le k.$$

It should be pointed out that the search information A_k contains the calculated partial criteria values of the initial MCO problem. The availability of such information allows the current values z_i, $1 \le i \le k$, to be recalculated to those for the next optimization problem (7) with any new value of the convolution coefficients λ'_j, $1 \le j \le s$, i.e.

$$z'_i = \max(\lambda'_j v_i^j, 1 \le j \le s), 1 \le i \le k, \tag{9}$$

without time-consuming calculations of the partial criteria values $f_i(y(x))$, $1 \le j \le s$. The number of computational operations required for recalculating the search information can be

evaluated as $M = 2sk$, that can be considered negligible compared with the time-consuming computations of the partial criteria values.

As a result of recalculating the values z_i, $1 \le i \le k$, the search information A_k from (9) will be readily available to use for solving the subsequent optimization subproblems (7). Thus, the global search can be continued instead of being restarted. Available search information will help reducing the number of global search iterations required to satisfy the stopping condition. It is expected that the amount of search information could reach such a large scale that solving each subsequent subproblem would require a minimum number of additional calculations (see the results of the numerical experiments in Section 4).

In particular, the availability of the search information allows computing the numerical estimates of the Hölder constant H from (8)

$$m = \begin{cases} rM, & M > 0 \\ 1, & M = 0 \end{cases}, M = \max_{1<i\le k} \frac{|z_i - z_{i-1}|}{\rho_i} \tag{10}$$

as the relative differences of the values of the objective function $z_i = F(\lambda, y(x_i))$, $1 \le i \le k$ on the set of points x_i,$1 \le i \le k$. Hereafter $\rho_i = \sqrt[N]{x_i - x_{i-1}}$, $1 < i \le k$. The constant r, $r > 1$ is the *reliability parameter* of the estimate of the constant H.

3.2 General Computational Scheme with Reusing Search Information for Solving MCO Problems

As a result of the above discussion the general computational scheme which provides reusing the search information for solving MCO problems can be proposed as follows.

Prior to beginning the computations, it is necessary to form the set Δ_q consisting of the values of the vectors $\lambda \in \Lambda$ from (6)

$$\Delta_q = \{\lambda_i \in \Lambda : 1 \le i \le q\}, \tag{11}$$

for each of which the problems (7) should be solved in order to find the efficient solutions of the MCO problem. The set Δ_q can be modified in the course of computations if necessary.

The computational scheme for solving the set of problems (7) for the corresponding vectors $\lambda \in \Delta_q$ is an iteration procedure, where at every iteration the following operations are performed.

1. Select the next vector $\lambda \in \Delta_q$ from the set Δ_q thus defining a new problem (7); the chosen vector λ should be excluded from the set Δ_q.
2. Solve the formulated problem (7) using any global optimization algorithm; the algorithm should utilize the search information A_k from (9) in the planning of the executed global search iterations; the additional search information obtained in the course of computations should be included into the set A_k.
3. Analyze the obtained results of solving the current problem (7); modify (reduce or extend) the content of the set Δ_q. if necessary.
4. Terminate the computations if the set Δ_q is empty, otherwise go to Step 1, and the computations should be continued further.

Each global optimization problem (7) is solved by the optimization method with the required accuracy in accordance to the MAGS convergence properties independently on the reused search information. This statement is confirmed also by the numerical results (see Section 4, Tables 3 - 5), where the number of problems solved with the required accuracy

are almost the same, and the indicators of completeness and uniformity of the approximation of the Pareto domain at approximately equal allowed number of iterations are higher when reusing the search information (see Table 5). The benefit of the reuse of the search information is a possible decrease in the number of executed iterations required to provide the required accuracy.

3.3 Multidimensional Algorithm for Multicriterial Global Optimization

According to the proposed approach, MCO problems can be solved by using numerous efficient one-dimensional optimization methods. However, it should be noted that one-dimensional optimization algorithms require certain generalizations when applied to multidimensional problems. Moreover, the applied algorithms should use the search information in accordance with the computational scheme proposed in Section 3.2.

The main results of using Peano curves for the dimensionality reduction are obtained within the framework of the information-statistical optimization theory. This theory has provided the basis for developing a large number of efficient multiextremal optimization methods – see, for instance, [1–3, 15–18, 21, 32, 33], etc.

The proposed approach employs the *Multidimensional Algorithm of Global Search* (MAGS) [37, 38] for solving one-dimensional multiextremal optimization subproblems (7). This method has a strong theoretical basis and has been proved experimentally to be highly efficient as compared to other global search algorithms (see also the results of computational experiments in Section 4).

The general computational scheme of MAGS is given below in summary form.

Let k, $k > 1$ be the number of executed optimization iterations, the values of the objective function $F(\lambda, y(x))$ (hereinafter, this procedure is called the *trial*) have been computed for all previous iterations, and the obtained search information is represented in the form of A_k from (9). The trial points of the next $(k+1)^{th}$ iteration are determined by the following rules.

Rule 1. Compute the *characteristic* $R(i)$ for each interval (x_{i-1}, x_i), $1 < i \le k$ from A_k

$$R(i) = \rho_i + \frac{(z_i - z_{i-1})^2}{m^2 \rho_i} - 2\frac{(z_i + z_{i-1})}{m}, 1 < i \le k, \tag{12}$$

where m, z_i, $1 \le i \le k$ and ρ_i, $1 < i \le k$ are defined in (11).

Rule 2. Determine the interval (x_{t-1}, x_t) with the highest characteristic $R(t)$, i. e.

$$R(t) = \max_{1<i\le k} R(i). \tag{13}$$

Rule 3. Compute the trial point x^{k+1} of the next iteration within the interval t from (14):

$$x^{k+1} = \frac{x_t + x_{t-1}}{2} - sign(z_t - z_{t-1})\frac{1}{2r}\left[\frac{r|z_t - z_{t-1}|}{m}\right]^N$$

The *stopping condition* is defined by the inequality

$$\rho_t \le \varepsilon \tag{14}$$

which should be checked for the interval t from (14). The value $\varepsilon > 0$ is the predefined *accuracy* of solving the global optimization problem. If the stopping condition is not fulfilled, the iteration index k is incremented by 1, and the execution of the algorithm should be continued.

The lowest computed value of the objective function $F(\lambda, y(x))$ is taken as the current estimate of the global minimum value, i. e.:

$$z_k^* = \min_{1 \le i \le k} z_i. \tag{15}$$

It can be noted that the characteristics $R(i)$, $1 < t \le k$ from (13) can be considered as some measures of importance of the intervals with respect to the location of the global minimum point in these intervals.

The MAGS algorithm applied for solving the MCO problems and combined with the computational scheme given in 3.2 will be further referred to as the *Multidimensional Algorithm of Multicriterial Global Search* (MAMGS).

3.4 Theoretical Analysis of the Multicriterial Global Optimization Algorithm

For the sake of simplicity, a theoretical analysis is implemented for the Algorithm of Global Search (AGS) which is the one-dimensional version of MAGS. The proven properties of AGS are also true for MAGS and MAMGS. In this case, a reduced subproblem of multiextremal optimization (6) can be stated as follows:

$$\phi(x^*) = \min \phi(x), x \in [0,1] \tag{16}$$

where $\phi(x) = F(\lambda, x)$ from (7).

First and foremost, the AGS algorithm is stable. The following theorem holds true.

Theorem 1 [37,38]. Suppose that along with minimizing the multiextremal function $\phi(x)$ the AGS is also used for optimizing the function $\psi(x)$, which differs from $\phi(x)$ by some limited variation, i.e.

$$|\psi(x) - \phi(x)| \le \delta, x \in [0,1] \tag{17}$$

and suppose that in the process of minimization $\psi(x)$ the stopping condition (15) is satisfied with the accuracy $\varepsilon > 0$. Then

$$|z_k^* - \phi(x^*)| \le \frac{\alpha L \varepsilon}{2} + \delta, \tag{18}$$

if for the value m from (11) and some $\alpha > 1$ the inequality

$$m \ge L \left(1 + \sqrt{\frac{\alpha + 4\beta(1+\beta)}{\alpha - 1}} \right), \tag{19}$$

is satisfied where

$$\beta = \frac{2r\delta}{L\varepsilon(r-1)} \tag{20}$$

and z_k^* from (16), $\phi(x^*)$ from (17), L from (5).

The statement of Theorem 1 can be used to analyze the results of solving the series of optimization subproblems (17). Suppose that in the course of solving the MCO problem, we proceeded from solving the subproblem (17) with the convolution coefficients $\lambda' \in \Lambda$ from (6) to solving the subproblem (17) with the coefficients $\lambda'' \in \Lambda$. Let us estimate the

maximum variation of the coefficient values λ_i, $1 \le i \le s$ obtained as a result of such a transition

$$\Delta\lambda_{max} = \max_{1 \le i \le s} |\lambda_i' - \lambda_i''|. \tag{21}$$

Under these conditions, the maximum variation of the objective function values after stating the new subproblem (17) is bounded by the value

$$\Delta\phi_{max} = \max\{\Delta\lambda_{max} f_i^{max}, 1 \le i \le s\}, \tag{22}$$

where

$$f_i^{max} = \max\{f_i(y), y \in D\}, 1 \le i \le s.$$

Using the obtained estimates, Theorem 1 can be reformulated in relation to the MCO problems.

Corollary 1. When switching from solving the optimization subproblem (17) with the convolution coefficients $\lambda' \in \Lambda$ from (6) to solving the next subproblem (17) with the coefficients $\lambda'' \in \Lambda$, the value of the function $F(\lambda'', x)$ at the point of the global minimum differs from the estimated minimum value z_k^* from (16) obtained by minimizing the function $F(\lambda', x)$, by no more than the bounded value

$$|z_k^* - F(\lambda'', x^*)| \le \frac{\alpha L \varepsilon}{2} + \delta, \alpha > 1, \tag{23}$$

where $\delta = \Delta\phi_{max}$ from (22)-(23) and the conditions (20)-(21) are satisfied.

This statement means that if the deviation $\Delta\phi_{max}$ from (23) of the estimates of the minimum values of $F(\lambda'', x)$ is acceptable, then solving the subproblem (17) with the new convolution coefficients $\lambda'' \in \Lambda$ does not require any additional global search iterations – the estimates of the minimum value of the function $F(\lambda'', x)$ can be obtained in accordance with (24) using the values z_i, $1 \le i \le k$, located within the search information A_k from (9).

Further analysis is related to the distribution of iteration points within the search domain. The density of trials P_{ab} in the subinterval $[a, b] \subset [0, 1]$ can be defined as the ratio of the number N_{ab} of executed trial points x_i, $1 \le i \le k$, belonging to the interval $[a, b]$, to its length, i.e.

$$P_{ab} = \frac{N_{ab}}{b - a}. \tag{24}$$

The following theorem holds true.

Theorem 2 [37,38]. Suppose that the objective function $\phi(x)$, $x \in [0, 1]$, from (17) is the Lipschitzian function with the constant L and

$$\phi(x) \ge \phi(x^*) + \Delta, x \in [a, b] \subset [0, 1], \tag{25}$$

where x^* is the global minimum point of the function $\phi(x)$ and

$$b - a > 2\Delta / L.$$

Then

$$P_{ab} < 3m / 2\Delta,$$

if the value m satisfies the inequality $m \geq 2L$, m from (11), P_{ab} from (25).

Theorem 2 can be used to estimate the density of the executed trials for solving the subproblems of the family (17).

Corollary 2. Suppose that in solving the next subproblem (17), the conditions of Theorem 2 are satisfied, all the previous trial points x_i, $1 \leq i \leq k$, from the search information A_k are taken into account and the density p_{ab} of a given interval $[a,b]$ has been obtained. Then the number of additional global search iterations n'_{ab} within the interval [a,b] executed for solving the next subproblem (17) cannot exceed the value

$$n'_{ab} \leq (3m/2\Delta - p_{ab})(b-a).$$

This statement means that, in solving the next subproblems of the family (17), when a sufficiently high trial density is obtained, the number of additional iterations within the intervals located far enough from the global minimum point will be constantly decreasing (up to the total absence of these additional iterations).

As a result of the theoretical analysis, two different schemes for forming various convolution coefficients $\lambda \in \Lambda$ can be proposed, at the use of which the conditions of Consequence 1 of Theorem 1 will be satisfied. In the first scheme, initially a sparse grid can be built on the set Λ, so that when setting new values of λ later, the close values could be always found in the grid built in earlier. In the second scheme, new values of λ could be chosen as a result of small variations of the values used earlier (this method is used in the practical solving the MCO problems often, when the decision maker wants to refine the estimates of the efficient solutions obtained earlier). It is worth noting that the reuse of the search information could be useful even at essential differences in the chosen values of the coefficients λ (see Section 4).

4 Results of computational experiments

As it was mentioned above, the proposed approach was developed on the basis of the minimax scalarization technique (6) and by reducing MCO problems to a series of global optimization problems (7). As a result, the numerical evaluation of the approach efficiency should focus on a comparison of the MAGS algorithm with other multiextremal optimization methods. Such a comparisons have been made in a sufficiently large number of works – see, for instance, [1–3, 16, 18, 20, 25, 31–33, 35]. These studies show, in particular, that MAGS is one of the most efficient multiextremal optimization methods.

The computational experiments have been performed using the Lobachevsky supercomputer at the State University of Nizhny Novgorod (operating system – CentOS 6.4, supercomputer management system – SLURM). Each computational node of the supercomputer had 2 Intel Sandy Bridge E5-2660 2.2 GHz, 64 Gb RAM processors. Each CPU had 8-cores (that is, 16 CPU cores were available at the computing node). To generate the executable program code, the Intel C++ Compiler 14.0.2 was used.

The experiments were executed as follows. At the beginning, a comparison of the proposed approach with a number of other multicriterial optimization algorithms was performed. Then the MAMGS efficiency is evaluated by solving a large set of bi-criteria two-dimensional optimization problems. Finally, the experiments were executed for more complex optimization problems with 5 varied parameters and 5 partial criteria.

In all numerical experiments considered below, in order to construct the Pareto domain approximation with the use of MAMGS, the set of subproblems (7) with different convolution coefficients λ from (6) distributed uniformly in Λ was solved.

Besides, every time after solving this set of subproblems, the additional non-dominated points have been selected from the obtained search information – this approach allows expanding the number of the efficient points making up the Pareto domain approximation.

In the first series of experiments the proposed approach is compared with the following five multicriterial optimization methods:

- The Monte-Carlo (MC) method, where the trial points are selected within the search domain D randomly and uniformly,
- The genetic algorithm SEMO from the PISA library [4, 10],
- The non-uniform coverage (NUC) method [10],
- The bi-objective Lipschitz optimization (BLO) method [45],
- The MAMGS method proposed herein.

To solve MCO problems, the compared methods should compute some numerical approximation $PDA \subseteq PD$ of the Pareto domain $PD \subseteq D$.

For the first three algorithms, the numerical results were used from [10]. The results of the BLO method were presented in [45].

To evaluate the quality of the approximation, the completeness and uniformity of the Pareto domain coverage were compared by using the following two indicators [10,46]:

- The *hypervolume index* (HV) defined as the total size of subdomains of the objective criterion values, dominated by the approximation points of the Pareto domain. This indicator's value can be obtained as the total volume of crossing boxes, the vertices of which are the approximation points of the Pareto domain and the reference point $\hat{f}$(see Figure 1). As a reference point in the experiments conducted, we used the vector value:

$$\hat{f} = (\hat{f}_1, \ldots, \hat{f}_s) : \hat{f}_i = \max_{1 \le i \le s} \hat{f}_i(y), y \in D.$$

  This indicator characterizes the Pareto domain approximation completeness (the higher the value, the more complete the Pareto domain coverage).
- The *distribution uniformity index* (DU) of the points from the Pareto domain approximation, defined by the expressions:

$$DU = \sum_{i=1}^{p} (d_i - d)^2, d = \frac{\sum_{i=1}^{p} d_i}{p},$$
$$d_i = \min \rho(y_i, y), y \in PDA, y_i \neq y, 1 \le i \le p,$$

  where p represents the number of points in the Pareto domain approximation PDA. This indicator characterizes the Pareto domain approximation uniformity (the lower the value, the more uniform the Pareto domain coverage).

The first benchmark problem is stated as follows [10]:

$$f_1(y) = (y_1 - 1)y_2^2 + 1, f_2(y) = y_2, 0 \le y_1, y_2 \le 1. \quad (26)$$

For MAMGS, the following parameters were used: the reliability parameter $r = 2$, the accuracy $\varepsilon = 0.06$, and 100 subproblems (7) have been solved. The reference point was (1,1).

The results of the experiments are presented in Table 1. The column "Iterations" shows the total number of the optimization iterations executed by the method (this is the same as the total number of the iteration points at which the objective function values are computed).

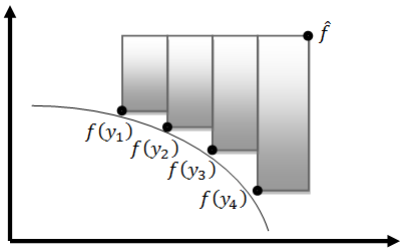

Fig. 1 An example of the hypervolume index (HV) estimate

Table 1 Numerical results of the methods compared for the benchmark problem (27)

Method	Iterations	PDA points	HV	DU
MC	500	67	0.300	1.277
SEMO	500	104	0.312	1.116
NUC	515	29	0.306	0.210
BLO	498	68	0.308	0.175
MAMGS	390	90	0.317	0.094

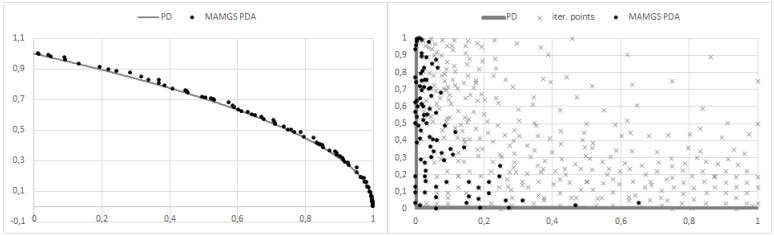

Fig. 2 The Pareto domain approximation (a) and iteration points (b) executed by MAMGS

The column "PDA points" demonstrates the number of points included in the numerical approximation of the Pareto domain. The efficiency indicators of the calculated Pareto domain approximations are given in the columns "HV" and "DU".

To illustrate the obtained results, Figure 1 shows the Pareto domain approximation obtained by using the MAMGS method.

Further, the second benchmark problem has been solved [10]:

$$f_1(y) = y_1, f_2(y) = \min(|y_1 - 1|, 1.5 - y_1) + y_2 + 1, 0 \le y_1, y_2 \le 2. \quad (27)$$

For MAMGS, the following parameters were used: the reliability parameter $r = 2$, the accuracy $\varepsilon = 0.06$, and 100 subproblems (7) have been solved. The reference point was (2,3). The results of the experiments are presented in Table 2 (the values of the DU indicator are not available in [10,45]).

As shown by the experimental results, MAMGS has an evident advantage over a number of other multicriterial optimization methods, even in solving relatively simple MCO problems.

It should be noted that the conditions for this experiment did not correspond to the initial assumptions about the properties of the multicriterial optimization problems: the partial criteria are neither multiextremal nor time-consuming.

Table 2 Numerical results of the methods compared for the benchmark problem (28)

Method	Iterations	PDA points	HV
MC	500	22	3.38
SEMO	500	221	3.27
NUC	490	36	3.42
BLO	435	92	3.61
MAMGS	380	92	3.59

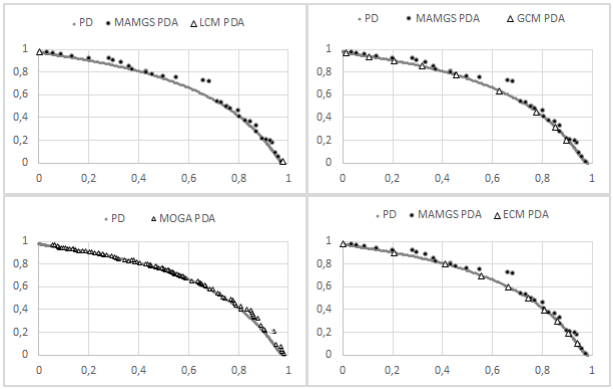

Fig. 3 The Pareto domain approximations calculated by the compared methods for the benchmark problem (29)

In the second series of experiments, MAMGS was compared with four other MCO methods considered in [50], namely:

- The linear combination method (LCM),
- The Multi-Objective Genetic Algorithm (MOGA),
- The global criterion method (GCM),
- The ε-constraint method (ECM).

The benchmark problems and the numerical results of these methods are taken form [50].

The first benchmark problem is stated as follows:

$$\begin{cases} f_1(y) = 1 - \exp\left(-\sum_{i=1}^{3}(y_i - 1/\sqrt{3})^2\right), \\ f_2(y) = 1 - \exp\left(-\sum_{i=1}^{3}(y_i + 1/\sqrt{3})^2\right), \end{cases} \quad -4 \le y_1, y_2 \le 4. \tag{28}$$

For MAMGS, the following parameters were used: the reliability parameter $r = 1.1$, the accuracy $\varepsilon = 0.04$, and 50 subproblems (7) have been solved.

Because the numerical results in [50] are given in the graphic form, the results of the experiment are given in the same way – see Figure 3. In addition, the number of executed iterations can be indicated: $LCM = 258$, $MOGA = 2500$, $GCM = 286$, $ECM = 257$. For MAMGS the number of executed iterations is 283.

The second benchmark problem is stated as follows:

$$f_1(y) = y_1, f_2(y) = (1 + 10y_2)\left[1 - (y_1/(1 + 10y_2))^2 - y_1/(1 + 10y_2)sin(8\pi y_1)\right], \tag{29}$$

where $0 \le y_1, y_2 \le 1$.

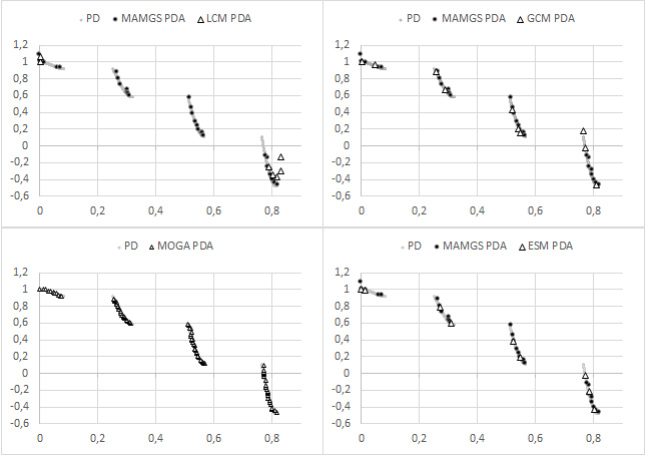

Fig. 4 The Pareto domain approximations calculated by the compared methods for the benchmark problem (30)

For MAMGS, the following parameters were used: the reliability parameter $r = 2$, the accuracy $\varepsilon = 0.01$, and 50 subproblems (7) have been solved.

The numerical results are given in the graphic form – see Figure 4. In addition the number of executed iterations can be indicated: $LCM = 875$, $MOGA = 3000$, $GCM = 427$, $ECM = 575$. For MAMGS the number of executed iterations is 277.

In the third series of experiments, the bi-criterial two-dimensional optimization problems have been solved, i.e. $N = 2$, $s = 2$. To obtain the MCO benchmark problems, the multiextremal functions were generated using the GKLS-generator [14]. This generator generates the multiextremal optimization problems with a priori known properties: the number of local minima, the size of their attraction domains, the point of global minimum, and the value of the functions therein, etc.

For MAMGS the following parameters were used: the reliability parameter $r = 4.5$, the accuracy $\varepsilon = 0.01$. A total of 100 MCO problems were solved, for each of them 50 global optimization subproblems (7) were solved with different convolution coefficients λ, distributed in Λ from (7) uniformly.

The experimental results are shown in Table 3. The columns ”10”,...,”50” show the average number of optimization iterations executed by the method for solving the indicated set of the subproblems (7).

These results show that if the method uses the search information then it can reduce the executed optimization iterations in solving multicriterial optimization problems up to 12.5 times. Moreover, as it has been noted previously, solving each subsequent scalar subproblem (7) to find another partial solution of the MCO problem requires a permanently decreasing number of optimization iterations. For more explicit demonstration of this key property of the proposed approach, Table 4 shows the results of experiments separated by the sequence of solved subproblems (7); each group includes 10 scalar subproblems (6).

The experimental results presented in Table 4 are even more convincing in showing a significant reduction in computations as the volume of obtained search information increases. Thus, the amount of computations was reduced 139.4 times for the group, which included from 41 to 50 subproblems.

Table 3 Experimental results for solving two-dimensional bi-criterial problems

Computational results (average per subproblem)	Number of subprolems being solved				
	10	20	30	40	50
Computations without using search information	647.7	817.8	840.6	882.0	835.2
Number of subproblems solved with the required accuracy	99.4%	99.1%	98.6%	98.7%	98.8%
Computations using search information	94.1	87.3	89.9	82.7	67.1
Number of subproblems solved with the required accuracy	99.4%	98.9%	98.6%	98.7%	98.9%
Reduction in the number of optimization iterations	**6.9**	**9.4**	**9.4**	**10.7**	**12.5**

Table 4 Average number of executed iterations per a subproblem within successive groups including 10 subproblems

Number of problems included in a group	1-10	11-20	21-30	31-40	41-50
Computations without using search information	647.7	987.8	886.2	1,006.1	648.2
Computations using search information	94.1	80.6	94.9	61.1	4.6
Reduction in the number of optimization iterations	**6.9**	**12.3**	**9.3**	**16.5**	**139.4**

For one of the solved problems more complete information is presented in Table 5. Row 2 contains the characteristics of the Pareto domain approximation computed numerically. In rows 3-4 the results of solving the MCO problem without the reuse of the search information are given for various values of required accuracy of computations. In the last row (row 5), the results of computations with the reuse of the search information are shown. One can see that the best results with respect to the completeness and the uniformity of approximation are presented in row 3 (the computations without the reuse of the search information). However, the number of performed iterations (the computations of the criteria values) exceeds the indicator given in row 5 more than 10 times. At approximately equal allowed number of iterations, the efficiency of computations with the reuse of the search information is higher (see rows 4 and 5).

In addition, an experiment on the solving two subproblems with the maximum differing values of the convolution coefficients $\lambda_1 = (1,0)$ and $\lambda_2 = (0,1)$ has been performed (i. e. the minimizations of the first criterion and of the second one have been performed successively). Total for finding the minimum values of the criteria without the reuse of the search information 855 iterations (the computations of the criteria values) have been executed, whereas with the reuse of the search information 530 iterations were necessary only (for the optimization of the second criterion at $\lambda_2 = (0,1)$ 395 iterations have been executed in the former case, and 50 iterations only in the latter one).

In the fourth experiment, the two-dimensional MCO problem with 5 criteria, i.e. for $N = 2$, $s = 5$, was solved. The problems were generated using the multiextremal functions generated by the GKLS-generator again. The following values were used: the reliability parameter $r = 5$, the search accuracy $\varepsilon = 0.01$. As in previous experiments, MCO problems (6) were solved using 50 different convolution coefficient values. These experiments showed that the average number of optimization iterations for solving a single problem (7) without

Table 5 The efficiency indicators of the solving of one of subproblems from Experiment 3

	Indicators		Average number of iterations per subproblems	Number of points in PDA
	DU	HV		
Pareto domain approx.	0.039	20.852		5087
Computations without using search information ($\varepsilon = 0.01$)	0.076	20.847	1022.74	803
Computations without using search information ($\varepsilon = 0.05$)	0.192	20.728	166.96	123
Computations using search information ($\varepsilon = 0.01$)	0.169	20.820	92.52	200

using the accumulated search information was 1,574.7. In the case when the search information was taken into account, the average number of optimization iterations executed by MAMGS was 89.2 (more than a 17-fold reduction).

In the final experiment, the bi-criteria five-dimensional MCO problem, i.e. for $N = 5$, $s = 2$, was solved. The problems were generated using the multiextremal functions generated by the GKLS-generator [14]. The following values were used: the reliability parameter $r = 4$, the accuracy $\varepsilon = 0.03$. As in the previous experiments, MCO problems (7) were solved using 50 different convolution coefficient values. These experiments showed that the average number of optimization iterations for solving a single problem (7) without using the accumulated search information was 300,383. In the case the search information was taken into account, the average number of optimization iterations executed by MAMGS was 49,751 (more than 6-fold reduction).

5 Conclusion

The paper proposes an efficient method for solving complex multicriterial optimization problems, for which the optimality criteria can be multiextremal and the calculation of the criteria values can be time-consuming. The proposed approach involves the transformation of multicriterial problems to global optimization problems through the minimax convolution of the partial criteria, the reduction of dimensionality using the Peano curves, and the use of the efficient information-statistical global optimization methods.

The key aspect of this approach is related to overcoming the huge computational complexity of the MCO problems. A significant increase in the efficiency and a considerable reduction of the amount of computations are provided by the intensive use of all search information obtained during the process of solving multicriterial optimization problems. As a part of this approach, the methods of recalculating all available search information to the values for the subsequent global optimization problem have been proposed. The updated search information is then used in the global optimization methods for the adaptive selecting of the global search iterations. According to the results of the computational experiments, the proposed approach reduces repeatedly the computational complexity of solving the multicriterial optimization problems.

To conclude, it can be noted that the proposed approach is promising and requires further investigation. First of all, it is necessary to continue conducting computational experiments to address the multicriterial optimization problems with a greater number of partial criteria and for greater dimensionality in the optimization problems being solved. The potential for parallel computing should also be evaluated in view of the high computational complexity for solving global optimization problems.

Acknowledgements

This work has been supported by the Russian Science Foundation, project No 16-11-10150 "Novel efficient methods and software tools for time-consuming decision making problems using superior-performance supercomputers."

References

1. Barkalov, K., Gergel, V.: Parallel global optimization on GPU. Journal of Global Optimization, 1–18 (2015). DOI 10.1007/s10898-016-0411-y.
2. Barkalov, K., Gergel, V., Lebedev, I.: Use of Xeon Phi Coprocessor for Solving Global Optimization Problems. LNCS, **9251**, 307–318. Springer, Heidelberg (2015)
3. Barkalov, K., Gergel, V., Lebedev, I.: Solving Global Optimization Problems on GPU Cluster. Simos T.E. (Ed.) ICNAAM 2015, AIP Conference Proceedings, **1738**, art. no. 400006 (2016)
4. Bleuler, S., Laumanns, M., Thiele, L., Zitzler, E.: PISA-A Platform and Programming Language Independent Interface for Search Algorithms. Evolutionary Multi-Criterion Optimization, LNCS, **2632**, 494–508 (2003)
5. Branke, J., Deb, K., Miettinen, K., Slowinski, R. (eds.): Multi-Objective Optimization-Interactive and Evolutionary Approaches. Springer, Berlin (2008)
6. Collette, Y., Siarry, P.: Multiobjective Optimization: Principles and Case Studies (Decision Engineering). Springer-Verlag, Berlin, Heidelberg (2011)
7. Deb, K.: Multi-Objective Optimization using Evolutionary Algorithms. Wiley, Chichester (2001)
8. Ehrgott, M.: Multicriteria Optimization. Springer-Verlag, Berlin, Heidelberg (2005, 2nd ed., 2010)
9. Eichfelder, G.: Scalarizations for adaptively solving multi-objective optimization problems. Comput. Optim. Appl., **44**, 249–273 (2009)
10. Evtushenko, Yu.G., Posypkin, M.A.: Method of non-uniform coverages to solve the multicriteria optimization problems with guaranteed accuracy. Automation and Remote Control, **75**(6), 1025–1040 (2014)
11. Figueira, J., Greco, S., Ehrgott, M. (eds.): Multiple criteria decision analysis: State of the art surveys. Springer, New York (2005)
12. Floudas, C.A., Pardalos, M.P.: Recent Advances in Global Optimization. Princeton University Press (2016)
13. Gablonsky, J.M., Kelley, C.T.: A Locally-Biased Form of the DIRECT Algorithm. Journal of Global Optimization, **21**(1), 27–37 (2001)
14. Gaviano, M., Lera, D., Kvasov, D.E., Sergeyev, Y.D.: Software for generation of classes of test functions with known local and global minima for global optimization. ACM Trans. Math. Software, **29**, 469–480 (2003)
15. Gergel, V.: A Unified Approach to Use of Coprocessors of Various Types for Solving Global Optimization Problems. 2nd International Conference on Mathematics and Computers in Sciences and in Industry, 13–18 (2015). DOI: 10.1109/MCSI.2015.18
16. Gergel, V., Grishagin, V., Gergel, A.: Adaptive nested optimization scheme for multidimensional global search. J. Glob. Optim., **66** (1), 35–51 (2016)
17. Gergel, V., Lebedev, I.: Heterogeneous Parallel Computations for Solving Global Optimization Problems. Procedia Computer Science, **66**, 53–62 (2015). DOI 10.1007/s10898-016-0411-y.
18. Gergel, V., Sidorov, S.: A Two-Level Parallel Global Search Algorithm for Solution of Computationally Intensive Multiextremal Optimization Problems. LNCS, **9251**, 505–515. Springer-Verlag, Berlin, Heidelberg (2015)
19. Germeyer, Yu.B.: Introduction in Theory of Operation Research. Moscow, Nauka (1971) (In Russian)
20. Grishagin, V.A.: Operating characteristics of some global search algorithms. Problems of Stochastic Search, **7**, 198–206 (1978) (In Russian)
21. Grishagin, V.A., Sergeyev, Y.D., Strongin, R.G.: Parallel characteristic algorithms for solving problems of global optimization. J. Global Optim., **10**, 185–206 (1997)
22. Hill, J.D.: A Search Technique for Multimodal Surfaces. IEEE Transactions on Systems and Cybernetics, **5**(1), 2–8 (1969)
23. Hillermeier, C., Jahn, J.: Multiobjective optimization: survey of methods and industrial applications. Surv. Math. Ind., **11**, 1–42 (2005)
24. Horst, R., Tuy, H.: Global Optimization: Deterministic Approaches. Springer-Verlag, Berlin (1990)
25. Kvasov, D.E., Pizzuti, C., Sergeyev, Y.D.: Local tuning and partition strategies for diagonal GO methods. Numerische Mathematik, **94**(1), 93-106 (2003)

26. Köksalan, M., Wallenius, J. and Zionts, S.: Multiple criteria decision making: From early history to the 21st century. World Scientific, Singapore (2011)
27. Locatelli, M., Schoen, F.: Global Optimization: Theory, Algorithms, and Applications. SIAM (2013)
28. Mardani, A., Jusoh, A., Nor, K., Khalifah, Z., Zakwan, N., Valipour, A.: Multiple criteria decision-making techniques and their applications – a review of the literature from 2000 to 2014. Economic Research-Ekonomska Istraživanja, **28**(1), 516–571 (2015). DOI: 10.1080/1331677X.2015.1075139
29. Marler, R. T., Arora, J. S.: Survey of multi-objective optimization methods for engineering. Struct. Multidisciplinary Optimization, **26**, 369–395 (2004)
30. Miettinen, K.: Nonlinear Multiobjective Optimization. Springer (1999)
31. Sergeyev, Y.D., Famularo, D., Pugliese, P.: Index Branch-and-Bound Algorithm for Lipschitz univariate global optimization with multiextremal constraints. Journal of Global Optimization, **21**(3), 317–341 (2001)
32. Sergeyev, Y.D., Grishagin, V.A.: Sequential and parallel algorithms for global optimization. Optimization Methods and Software, **3**, 111–124 (1994)
33. Sergeyev, Y.D., Grishagin, V.A.: Parallel Asynchronous Global Search and the Nested Optimization Scheme. J. Comput. Anal. Appl., **3**(2), 123–145 (2001)
34. Sergeyev, Y.D., Kvasov, D.E.: Global search based on efficient diagonal partitions and a set of Lipschitz constants. SIAM J. on Optimization, **16**(3), 910–937 (2006)
35. Sergeyev, Y.D., Strongin, R.G., Lera, D.: Introduction to global optimization exploiting space-filling curves. Springer-Verlag, New York (2013)
36. Siwale, I.: Practical multi-objective programming. Technical Report RD-14-2013. Apex Research Limited (2014)
37. Strongin, R.G.: Numerical methods in multiextremal problems: information-statistical algorithms. Nauka, Moscow (1978) (In Russian)
38. Strongin, R.G., Sergeyev, Y.D.: Global optimization with non-convex constraints. Sequential and parallel algorithms. Kluwer Academic Publishers, Dordrecht (2000, 2nd ed. 2013, 3rd ed. 2014)
39. Pintér, J.D.: Global optimization in action (continuous and Lipschitz optimization: algorithms, implementations and applications). Kluwer Academic Publishers, Dortrecht (1996)
40. Tan, K.C., Khor, E.F., Lee, T.H.: Multi-objective Evolutionary Algorithms and Applications. Springer-Verlag, London (2005)
41. Törn, A., Žilinskas, A.: Global Optimization. LNCS, **350**. Springer-Verlag, Berlin (1989)
42. Yang, X.-S.: Nature-inspired metaheuristic algorithms. Luniver Press, Frome (2008)
43. Zavadskas, E. K., Turskis, Z., Kildienė, S.: State of art surveys of overviews on MCDM/MADM methods. Technological and Economic Development of Economy, **20**, 165–179 (2014)
44. Zhigljavsky, A.A.: Theory of Global Random Search. Kluwer Academic Publishers, Dordrecht (1991)
45. Žilinskas, A., Žilinskas, J.: Adaptation of a one-step worst-case optimal univariate algorithm of bi-objective Lipschitz optimization to multidimensional problems. Commun Nonlinear Sci Numer Simulat, **21**, 89–98 (2015)
46. Zitzler, E., Knowles, J., Thiele, L.: Quality Assessment of Pareto Set Approximations. LNCS in Multiobjective Optimization-Interactive and Evolutionary Approaches. LNCS, **5252**, 373–404 (2008)
47. Strongin, R.G., Gergel, V.P. Markin, D.L.: Multicriterion multiextreme optimization with nonlinear constraints. Lecture Notes in Economics and Mathematical Systems, **351**, 120–127 (1988)
48. Chinchuluun, A., Pardalos, P.M.: A survey of recent developments in multiobjective optimization. Ann Oper Res, **154**(1), 29–50 (2007). DOI:10.1007/s10479-007-0186-0
49. Pardalos, P.M., Žilinskas, A., Žilinskas, J.: Non-Convex Multi-Objective Optimization. Springer (2017)
50. Chiandussi, G., Codegone, M., Ferrero, S., Varesio, F.E.: Comparison of multi-objective optimization methodologies for engineering applications. Computers and Mathematics with Applications, **63**(5), 912–942 (2012)
51. Zhigljavsky, A., Žilinskas, A.: Stochastic Global Optimization. Springer (2008)
52. Paulavičius, R., Žilinskas, J.: Simplicial Global Optimization. Springer (2014)