An Approach for Parallel Solving the Multicriterial Optimization Problems with Non-convex Constraints

✉Victor Gergel and Evgeny Kozinov

Lobachevsky State University of Nizhni Novgorod, Nizhni Novgorod, Russia
gergel@unn.ru, evgeny.kozinov@itmm.unn.ru

Abstract. In the present paper, an efficient method is proposed for parallel solving of the multicriterial optimization problems with non-convex constraints, where the optimality criteria could be the multiextremal ones and computing the values of the criteria and constraints could require a large amount of computations. The developed approach is based on the reduction of the multicriterial problems to the nonlinear programming ones by means of the minimax convolution of the partial criteria, on the dimensionality reduction with the use of Peano space-filling curves, and on the application of efficient information-statistical global optimization methods with a novel index scheme of the constraints handling instead of the penalty functions applied usually. When performing the parallel computations, the maximum utilization of the whole search information obtained in the course of the search process is provided. The results of the computational experiments have demonstrated such an approach to allow reducing the computational costs of solving the multicriterial optimization problems essentially – tens and hundred times.

Keywords: Decision Making · Multicriterial Optimization · Global Optimization with Non-Convex Constraints · High Performance Computations · Dimensionality Reduction · Criteria Convolution · Global Search Algorithms · Computational Complexity.

1 Introduction

The multicriterial optimization (MCO) problems are among the most general problem statements for the decision-making problems – the statement of MCO problems covers many classes of optimization problems, including unconstrained optimization, nonlinear programming, global optimization, etc. The opportunity to specify several criteria is very useful in formulating the complex decision-making problems, and is used in the applications widely. The practical importance has caused a high research activity in the field of the MCO problems. As a result of intensive research, a plenty of efficient methods for solving the MCO problems have been proposed, and many practically important problems have been solved - see, for example, the monographs [1-3,19] and reviews of scientific and practical results [4,5,7,20,32,33].

Among key features of the multicriterial optimization problems is a potential contradiction between the partial efficiency criteria. This makes impossible to achieve the optimum (the best) values with respect to all partial criteria simultaneously. Consequently, the finding of some compromised (effective, non-dominated) decisions, when the achieved values of particular partial criteria are consistent with each other is understood as a solution of a MCO problem usually. It is important to note that the viewpoint on an expedient compromise can be changed in the course of computations that could require finding several different compromised decisions.

Among the developed approaches for solving the MCO problems, one can outline the methods of lexicographic optimization, when some arrangement of the criteria according to the importance of these ones is made, and the optimization of the partial criteria is performed successively according to the decreasing of their importance – see, for example, [3]. Another approach is represented by the iterative methods [4,17], when the researcher (the decision-maker) takes an active part in the process of selecting the decisions. One more direction developed extensively consists in the development of the evolutionary algorithms based on the simulation of some natural phenomena and the application of these ones to solving the MCO problems [17,18,22,23]. The scalarization, when some methods for the convolution of the partial criteria into a single criterion are applied, is an approach used widely – see, for example, [2,6].

The present work is devoted to the solving of the MCO problems, which are used for formulating the decision-making problems in the computer-aided design of the complex technical objects and systems. In these applications, the partial criteria can have a multiextremal form, and the domain of feasible decisions can be defined by non-convex constraints. The presence of constraints can result in a partial computability, when the computations of some criteria and constraints are impossible if even one constraint is not satisfied. Also, it was supposed that the computations of the values of criteria and constraints could require a large amount of computations. In these conditions, the finding of even one compromised decision requires a considerable amount of computations whereas the finding of several effective decisions (or of the complete set of these ones) becomes a problem of a huge computational complexity.

The properties of the considered class of the MCO problems listed above determine the key feature of these ones – a high computational complexity. One of the promising directions of the search for the methods of solving such problems consists in the use of the model-based approach, when after a small number of computations of the values of the computation-costly criteria and constraints, the fast-computed approximation functions are constructed [25,26]. Such an approach is efficient enough, however, the construction of good approximations is difficult at the essentially multiextremal behavior of the optimized criteria and constraints.

The approach to solving the computational-costly class of the MCO problems proposed in the present paper is based on the following key statements. First of all, the scalarization of the vector criterion is used that allows reducing the solving of a MCO problem to the solving of a series of global optimization problems [2,6]. Next, an efficient global search algorithm developed in the framework of the information-statistical theory of the multiextremal optimization [9,10] is applied for solving the constrained global optimization (CGO) problems with the non-convex constraints.

The parallelization methods developed for this algorithm provide high indicators of efficiency of the parallel computations allowing full utilization of the great computational potential of modern supercomputer systems. Finally, the whole search information obtained in the course of solving a MCO problem is utilized in full amount when performing all necessary computations. In general, the developed approach allows reducing the amount of computations performed for the searching of the next efficient decisions essentially – down to the execution of several iterations only.

Further structure of the paper is as follows. In Section 2, the statement of a multicriterial optimization problem with non-convex constraints is given. In Section 3, the basics of the developed approach are presented. In Section 4, the global search algorithm for solving the reduced scalar nonlinear programming problems is described. In Section 5, the issues of the parallel computations with the reuse of the search information obtained in the course of computations are discussed. Section 6 presents the results of numerical experiments. In Conclusion, the obtained results are discussed and main directions of further investigations are outlined.

2 Problem Statement

A problem of multicriterial optimization with non-convex constraints can be stated in the following form:

$$\begin{aligned} f(y) &= \big(f_1(y), f_2(y), \dots, f_s(y)\big) \to min,\ y \in Q, \\ Q &= \{\, y \in D : g_i(y) \le 0, 1 \le i \le m \,\}, \\ D &= \{\, y \in R^N : a_i \le y_i \le b_i, 1 \le i \le N \,\} \end{aligned} \tag{1}$$

where

- $y = (y_1, y_2, \dots, y_N)$ is the vector of varied parameters,
- N is the dimensionality of the multicriterial optimization problem being solved,
- $f(y) = (f_1(y), f_2(y), \dots, f_s(y))$ is the vector criterion of efficiency,
- $g(y) = (g_1(y), g_2(y), \dots, g_s(y))$ is the vector function of the constraints,
- Q is the domain of feasible solutions, D is the search domain and a, $b \in R^N$ are given constant vectors.

In further consideration, the following notations will be used also:

$$g_{m+1}(y) = f_1(y), g_{m+2}(y) = f_2(y), \dots, g_{m+s}(y) = f_s(y), M = s + m.$$

Without any loss in generality, the partial criteria values in the problem (1) are supposed to be non-negative, and the decrease of these ones corresponds to increasing efficiency of the considered decisions $y \in D$.

Usually, the partial criteria of the MCO problem (1) contradict to each other, and there is no decision $y \in D$, which would provide the optimal (minimal) values for all criteria simultaneously. In such cases, the decisions $y^* \in D$, where the values of particular partial criteria cannot be improved without worsening the efficiency values with respect to other criteria, are considered as the solutions of the MCO problem.

Such unimprovable decisions are called the effective or Pareto-optimal ones. Any effective decision can be considered as a *partial solution*, and the set of all unimprovable decisions represent a *complete solution* of the MCO problem.

As it has been already mentioned above, in the present paper, the problem (1) will be considered in application to the most complex decision-making problems where the partial criteria $f_i(y)$, $1 \le i \le s$ could be multiextremal, the constraints could be non-convex, and obtaining the values of the criteria and constraints at the points of the search domain $y \in D$ could require a large amount of computations. Let us suppose also the partial criteria $f_i(y)$, $1 \le i \le s$ and the constraints $g_i(y)$, $1 \le i \le m$ to satisfy the Lipschitz condition

$$|g_i(y') - g_i(y'')| \le L_i \|y' - y''\|,\ y', y'' \in D,\ 1 \le i \le M, \tag{2}$$

where L_i are the Lipschitz constants for the functions $g_i(y)$, $1 \le i \le M$ and $\|*\|$ denotes the Euclidean norm in R^N.

3 The Basics of the Approach

3.1 The reduction of the MCO problems to the global optimization problems with the non-convex constraints

The approach applied in the present work is based on the scalarization of the vector criterion by means of the minimax convolution scheme that allows reducing the solving of the problem (1) to solving a nonlinear programming problem

$$\begin{aligned} &\min [F(\lambda, y) = \max(\lambda_i f_i(y), 1 \le i \le s)], y \in Q, \\ &\lambda \in \Lambda \subset R^s : \textstyle\sum_{i=1}^{s} \lambda_i = 1,\ \lambda_i \ge 0,\ \ 1 \le i \le s. \end{aligned} \tag{3}$$

The necessity and sufficiency of this approach for solving the MCO problem is a key property of the minimax convolution scheme: the result of the minimization of $F(\lambda, y)$ leads to the obtaining of an effective decision[1] for the MCO problem and, vise versa, any effective decision of the MCO problem can be obtained as a result of the minimization of $F(\lambda, y)$ at the corresponding values of the convolution coefficients λ_i, $1 \le i \le s$ – see, for example, [4].

The coefficients λ_i, $1 \le i \le s$ in (3) can be understood as the indicators of importance of the partial criteria – the larger the value of the coefficient λ_i of a particular partial criterion, the more the contribution of this partial criterion in the scalar criterion $F(\lambda, y)$. As a result, a method of solving the MCO problems can be formulated in a step-by-step manner. At every step, the decision maker chooses the desired values of the coefficients λ_i, $1 \le i \le s$. Then, the solving of the formed problem (3) is performed. Afterwards, the decision maker analyzes the obtained effective decisions and corrects the chosen coefficients λ_i, $1 \le i \le s$ if necessary. Such a multistep method corresponds to the practice of the choice of the compromised decision in the

[1] More exactly, the minimization of $F(\lambda, y)$ can lead to the obtaining of the weakly – effective decisions (the set of the weakly effective decisions includes the Pareto domain).

complex decision-making problems to much extent. And the possibility to determine several effective decisions (or the whole set of these ones) at reasonable computational costs becomes a key problem in solving the complex multicriterial optimization problems.

It is worth noting that the scalar criterion $F(\lambda, y)$ satisfies the Lipschitz condition also:

$$\left|F(\lambda, y') - F(\lambda, y'')\right| \le L\left\|y' - y''\right\|,\ y', y'' \in D. \qquad (4)$$

3.2 The dimensionality reduction for the multidimensional global optimization problems

The use of the global search algorithms developed within the framework of the information-statistical theory of global optimization [8-11] for solving the multiextremal optimization problems (3) is one more key statement of the approach developed in the present work. This theory has served as the basis for the development of a large number of optimization algorithms, which have been substantiated mathematically and have demonstrated a high efficiency, and have allowed solving many complex optimization problems in various fields of application [11,28-31,34].

The reduction of the dimensionality of the problems being solved with the use of Peano space-filling curves or evolvents $y(x)$ mapping the interval $[0,1]$ onto an N-dimensional hypercube D unambiguously is a distinctive feature of the information-statistical global optimization algorithms – see, for example, [9-11]. As a result of such reduction, the initial multidimensional global optimization problem (3) is reduced to a one-dimensional problem:

$$F(\lambda, y(x^*)) = min\,\{\, F(\lambda, y(x)) \colon\ g_i(y(x)) \le 0,\, 1 \le i \le m, x \in [0,1]\}. \qquad (5)$$

It is important to note that the one-dimensional functions obtained as a result of the reduction satisfy the uniform Hölder condition (see [9,10]) i. e.

$$\begin{aligned} &|F(\lambda, y(x')) \ - F(\lambda, y(x''))| \le H|x' - x''|^{\frac{1}{N}}, x', x'' \in [0,1], \\ &|g_i(y') - g_i(y'')| \le H_i|x' - x''|, x', x'' \in [0,1], 1 \le i \le m \end{aligned} \qquad (6)$$

where the Hölder constant H (H_i) is defined by the relation $H = 4L\sqrt{N}$ $(H_i = 4L_i\sqrt{N})$, $1 \le i \le m$, $L(L_i)$ is the Lipschitz constant from (2) and (4) and N is the dimensionality of the optimization problem (1).

4 An Efficient Method for Solving the Global Optimization Problems with the Non-convex Constraints

The basics of the approach presented in Section 3 allow reducing the solving of the MCO problem (1) to the solving of a series of the reduced multiextremal problems with the constraints (5). And, thus, the global search algorithms can be applied for solving the MCO problems [8,12-16].

It is worth noting that the presence of the non-convex constraints complicates solving the global optimization problems considerably – the obtained solutions should belong to the feasible domain Q. The situation becomes even more complicated in the case of the partial computability, when the computing of some criteria and constraints is impossible if there is even one unsatisfied constraint. Often, for solving the constrained optimization problems, more simple cases are selected – for example, the problems with the linear or quadratic constraints are considered. Various methods of approximation of the complex constraints using the constraints of simpler forms (linear, convex, etc.) are applied as well. However, the most often applied method is the penalty function method. The approach used in the present work is based on a novel method of the constraint handling. This approach was developed within the framework of the information-statistical theory of global search [10]. The idea of the approach consists in the construction of a scalar unconstrained objective function, the solving of which leads to the solving of the initial problem (5) – more detailed description of the approach is given below.

Within the framework of this approach, the algorithm of constrained global search (ACGS) for the multiextremal optimization problems with the non-convex constraints[2] makes the basis of the developed optimization methods. The general computational scheme of the algorithm can be represented in the following form [9,10].

Let us introduce a simpler notation for the one-dimensional problems (5) as

$$min\{\, \varphi(x) \colon g_i\big(y(x)\big) \leq\ 0, 1 \leq i \leq m, x \in [0,1]\big\},$$
$$\varphi(x)\ = g_{m+1}(x)\ =\ F\big(\lambda, y(x)\big). \tag{7}$$

The problem (7) can be considered in the partial computability form, when each function g_j, $1 \leq j \leq m+1$ is defined and computable in the corresponding subdomain $\Delta_j \subset [0,1]$ only, where

$$\Delta_1 = [0,1], \Delta_{j+1} = \{\, x \in \Delta_j \colon g_j\big(y(x)\big) \leq 0\, \}, 1 \leq j \leq m. \tag{8}$$

Taking into account the condition (8), the initial problem (7) can be represented as follows

$$\varphi(x^*) = min\big\{g_{m+1}\big(y(x)\big) \,:\, x \in \Delta_{m+1}\big\}. \tag{9}$$

This form of the problem (7) allows defining an *index* $\nu = \nu(x)$ for the points x from the search domain $[0,1]$ where $\nu - 1$ is the number of constraints, which are satisfied at this point. The index ν is defined by the conditions

$$g_\nu\big(y(x)\big) > 0, g_j\big(y(x)\big) \leq 0,\ \ 1 \leq j \leq \nu - 1, 1 \leq \nu = \nu\,(x) \leq m + 1. \tag{10}$$

where the last inequality is insufficient if $\nu = m + 1$. Computing the index ν can be provided by the sequential computation of the values $g_j\big(y(x)\big)$, $1 \leq j \leq \nu = \nu\,(x)$, i. e. the next value $g_{j+1}(x)$ is computed in the case, when $g_j(x) \leq 0$ only. The pro-

[2] This algorithm is known also as the index method - see [10].

cess of computations is terminated either as a result of the fulfillment of the inequality $g_j(x) > 0$ or as a result of the achievement of the value $\nu\,(x) = m + 1$ (this procedure is called hereafter a *trial*).

The main idea of such index scheme consists in the reduction of the constrained problem (7) to an unconstrained problem

$$\Phi(x^*) = min\{\Phi(x): x \in [0,1]\}, \tag{11}$$

where

$$\Phi(x) = \begin{cases} g_\nu(y(x))/H_\nu, & \nu < m+1, \\ (g_{m+1}(y(x)) - g^*_{m+1})/H_{m+1}, & \nu = m+1. \end{cases}$$

It is worth noting that the values of the Lipschitz constants $L_\nu, 1 \le \nu \le m+1$ and the value g^*_{m+1} are unknown. However, when performing the computations, one can use the adaptive estimates of these values obtained in the course of solving the optimization problem (see the description of the algorithm below) instead.

The general computational scheme of the ACGS method consists in the following.

The first trial is performed at an arbitrary point $x_1 \in (0, 1)$. The choice of the point $x_{k+1}, k \ge 1$ of any next trial is determined by the following rules.

Rule 1. Renumber the points of preceding trials $x_1, \dots, x_k$ by the lower indices in the order of increasing of the coordinate values i. e.

$$0 = x_0 < x_1 < \cdots < x_i < \cdots < x_k < x_{k+1} = 1, \tag{12}$$

and juxtapose these ones to the values $z_i = g_\nu(x_i), \nu = \nu\,(x_i), 1 \le i \le k$ from (10) computed at these points. The points $x_0 = 0$ and $x_{k+1} = 1$ are introduced additionally for convenience of further notations (the values z_0 and z_{k+1} are undefined).

Rule 2. Subdivide the indices $i, 1 \le i \le k$ of the points from (12) with respect to the number of constraints of the problem fulfilled at these points by constructing the sets

$$I_\nu = \{i: 1 \le i \le k, \nu = \nu\,(x_i)\},\ \ 1 \le \nu \le m+1 \tag{13}$$

containing the indices of all points $x_i, 1 \le i \le k$ having the indices equal to the same value ν. The boundary points $x_0 = 0$ and $x_{k+1} = 1$ are interpreted as the ones having the zero indices, and are juxtaposed to an auxiliary set $I_0 = \{0, k+1\}$.

Determine the maximum value of the index

$$M = max\{\nu = \nu\,(x_i), 1 \le i \le k\}. \tag{14}$$

Rule 3. Compute the current estimates

$$\mu_\nu = \max\left\{|z_i - z_j|/\sqrt[N]{(x_i - x_j)}, i, j \in I_\nu, i > j\right\} \tag{15}$$

for the Hölder constants H_ν of the functions $g_\nu, 1 \le \nu \le m+1$ from (6). If the set I_ν contains less than two elements or if μ_ν from (15) appears to equal zero, then accept $\mu_\nu = 1$.

Rule 4. Compute the estimates $z_\nu^*, 1 \le \nu \le M$ for all nonempty sets I_ν, $1 \le \nu \le m + 1$ from (13),

$$z_\nu^* = \begin{cases} 0, & \nu < M, \\ \min\{g_\nu(x_i) : i \in I_\nu\}, & \nu = M. \end{cases} \tag{16}$$

Rule 5. Compute the characteristics $R(i)$ for each interval (x_{i-1}, x_i), $1 \le i \le k + 1$, where

$$R(i) = \begin{cases} \rho_i + \dfrac{(z_i - z_{i-1})^2}{r_\nu^2 \mu_\nu^2 \rho_i} - 2\dfrac{(z_i + z_{i-1} - 2z_\nu^*)}{r_\nu \mu_\nu}, & \nu = \nu(x_{i-1}) = \nu(x_i), \\ 2\rho_i - 4\dfrac{(z_i - z_\nu^*)}{r_\nu \mu_\nu}, & \nu = \nu(x_i - 1) < \nu(x_i), \\ 2\rho_i - 4\dfrac{(z_{i-1} - z_\nu^*)}{r_\nu \mu_\nu}, & \nu = \nu(x_{i-1}) > \nu(x_i), \end{cases} \tag{17}$$

$$\rho_j = \sqrt[N]{(x_i - x_{i-1})}, 1 \le j \le k + 1 .$$

where $z_\nu^*, 1 \le \nu \le M$ from (16), M from (14).

The values $r_\nu > 1,\ 1 \le \nu\ \le m + 1$ are the parameters of the algorithm. The appropriate values r_ν allows using the products $r_\nu \mu_\nu$ as the estimates of the Hölder constants H_ν, $1 \le \nu \le m + 1$.

Rule 6. Determine the interval (x_{t-1}, x_t) with the maximum characteristic:

$$R(t) = max\{R(i) \colon 1 \le i \le k + 1\}. \tag{18}$$

Rule 7. Execute the next trial at the point of the interval $x^{k+1} \in (x_{t-1}, x_t)$ determined according to the expression

$$x^{k+1} = \begin{cases} \dfrac{x_t + x_{t-1}}{2}, & \upsilon(\mathrm{x}_{\text{t-1}}) \ne \upsilon(\mathrm{x}_{\text{t-1}}), \\ \dfrac{x_t + x_{t-1}}{2} + sign(z_t - z_{t-1})\dfrac{1}{2r_\nu}\left[\dfrac{|z_t - z_{t-1}|}{\mu_\nu}\right]^N, & \upsilon = \upsilon(\mathrm{x}_{\text{t-1}}) = \upsilon(\mathrm{x}_{\text{t-1}}). \end{cases} \tag{19}$$

The iterations of the algorithm are terminated if the stopping condition is satisfied

$$\rho_t \le \varepsilon, \tag{20}$$

where t is from (18), and $\varepsilon > 0$ is the predefined accuracy.

Various modifications of this algorithm and the corresponding theory of convergence are presented in [9,10].

5 Parallel Computations for the Time-consuming Multicriterial Constrained Optimization Problems

The proposed approach for parallel computations when solving the computation-costly multicriterial optimization problems is based on the simultaneous computing of the values of partial criteria and constraints of the initial problem (1) at several different points of the search domain D. Such an approach provides the parallelization of the most time-consuming part of the global search, and is a general one – it can be applied for many global search methods for various global optimization problems. Besides, an essential speedup of the computations can be provided by means of full utilization of the whole search information obtained in the course of optimization.

5.1 The reuse of the search information for accelerating the computations

The numerical solving of the optimization problems consists in the sequential computation of the values of the partial criteria $f^i = f(y^i)$ and constraints $g^i = g(y^i)$ at the points $y^i,\ 1 \le i \le k$ of the search domain D. The search information obtained can be represented in the form of the *search information set* (SIS):

$$\Omega_k = \{(y^i, f^i, g^i)^T : 1 \le i \le k\}. \tag{21}$$

The availability of SIS allows reducing the results of previous computations to the values of any next optimization problem (11) being solved without any time-consuming computations of the values of partial criteria and constraints of the initial problem (1) at any new values of the convolution coefficients $\lambda \in \Lambda$.

And, thus, all search information can be utilized for continuing the computations in full amount. In general, the reuse of the search information will require less and less amount of computations for solving every next optimization problem downto performing several iterations only to find the next effective decision (see Section 6 for the results of the numerical experiments).

As a result of the dimensionality reduction, the search information Ω_k from (21) can be transformed into the matrix of search state (MSS)

$$A_k = \{(x_i, z_i, \text{v}_i, l_i)^T : 1 \le i \le k\}, \tag{22}$$

where x_i, $1 \le i \le k$ are the reduced trial points of the executed global search iterations, z_i, $1 \le i \le k$ are the values of scalar criterion of current reduced optimization problem (11) being solved, ν_i, $1 \le i \le k$ are the indices of the scalar criterion values, and l_i, $1 \le i \le k$ are the indices of the global search iterations, where the points x_i, $1 \le i \le k$ have been computed.

The ACGS algorithm improved by the possibility to use the search information A_k from (22) will be called hereafter the Algorithm of Multicriterial Constrained Global Search (AMCGS).

5.2 Parallel algorithm of the multicriterial global search

The choice of the points in the search domain D for the simultaneous execution of several trials (computing the values of the criteria and constraints of initial MCO problem (1)) can be provided by means of the following parallel generalization of the ACGS method – see, for example, [10,34].

Let p be the number of employed parallel computing nodes (processors or cores) of the computational system with shared memory. The rules of the parallel algorithm correspond to the computational scheme of the ACGS method except the steps of computing the points of the next global search iteration. The modified rules for the parallel algorithm can be presented as follows.

Rule 6 (updated). Arrange the characteristics of the intervals (x_{i-1}, x_i), $1 \leq i \leq k+1$ obtained in (17) in the decreasing order

$$R(t_1) \geq R(t_2) \geq \ldots \geq R(t_{k-1}) \geq R(t_k) \tag{23}$$

and select p intervals with the indices t_j, $1 \leq j \leq p$ having the maximum values of the characteristics.

Rule 7 (updated). Perform new trials at the points x^{k+j},$1 \leq j \leq p$ placed into the intervals with the maximum characteristics from (23) according the expression (19).

The stopping condition (20) of the parallel algorithm, which terminates the trials, should be checked for all intervals, where the scheduled trials are performed, i. e.

$$\rho_{\mathrm{t_j}} \leq \varepsilon,\ 1 \leq t_j \leq p\,.$$

The ACGS algorithm updated by the opportunity of the parallel computations for the computing nodes with shared memory will be named hereafter the Parallel Algorithm of Multicriterial Constrained Global Search (PAMCGS).

6 Results of Numerical Experiments

The numerical experiments have been carried out using the Lobachevsky supercomputer at State University of Nizhni Novgorod (the operating system – CentOS 6.4, the supercomputer management system – SLURM). Each supercomputer node had 2 Intel Sandy Bridge E5-2660 2.2 GHz, 64 Gb RAM processors. The central processor units (CPUs) had 8 cores (i.e., total 16 CPU cores were available per a node).

First, let us consider the results of the comparison of the developed PAMCGS algorithm with several other multicriterial optimization algorithms. A bi-criterial test problem proposed in [21]:

$$f_1(y) = (y_1 - 1)y_2^2 + 1, f_2(y) = y_2, 0 \leq y_1, y_2 \leq 1. \tag{24}$$

was used for this experiment. The construction of a numerical approximation of Pareto domain was understood as the solution of the problem (24). To evaluate the quality of approximation, the completeness and uniformity of coverage of the Pareto domain were evaluated with the use of the following two indices [21,24]:

- The hypervolume index (HV). This index features the completeness of approximation of the Pareto domain (a larger value corresponds to a denser coverage of the Pareto domain).
- The distribution uniformity index (DU). This index features the uniformity of coverage of the Pareto domain (a lower value corresponds to more uniform coverage of the Pareto domain).

Within the framework of this experiment, five multicriterial optimization algorithms were compared: the Monte-Carlo (MC) method, the genetic algorithm SEMO from the PISA library [20,24], the Non-Uniform Coverage (NUC) method [21], the Bi-objective Lipschitz Optimization (BLO) method proposed in [24], and the serial variant of the PAMCGS algorithm proposed in the present paper. The results of solving the problem (24) for all methods listed above were obtained in [24].

For the AGCS method, 50 problems (3) have been solved at various values of the convolution coefficients λ distributed in Λ uniformly. The results of performed experiments are presented in Table 1.

Table 1. Comparison of the efficiency of the multicriterial optimization algorithms

Method	MC	SEMO	NUC	BLO	ACGS
Iterations	500	500	515	498	370
Number of points in the Pareto domain approximation	67	104	29	68	100
HV index	0.300	0.312	0.306	0.308	0.316
DU index	1.277	1.116	0.210	0.175	0.101

The results of the performed experiments have demonstrated that the ACGS algorithm have a considerable advantage as compared to the considered multicriteria optimization methods even when solving the relatively simple MCO problems.

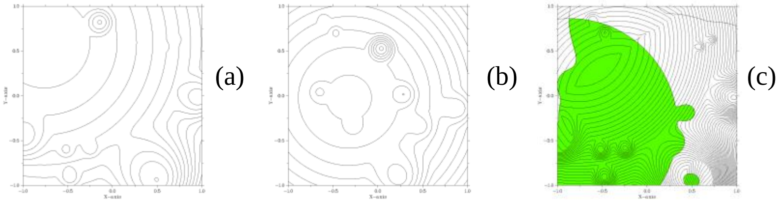

Fig. 1. Contour plots of two criteria obtained with the use of the GKLS generator (a, b); the problem to be solved obtained by the convolution of the criteria $\lambda = \{0.5, 0.5\}$ (c). The feasible domain is highlighted by green

In the second series of the numerical experiments, the solving of the bi-criterial two-dimensional MCO problems with two constraints i. e. N=2, s=2, m=2 has been performed. The multiextremal functions obtained with the use of the GKLS generator [27] were used as the problem criteria. In the course of experiments, the solving of 100 multicriterial problems of this class has been performed. In every problem, the search of the Pareto-optimal decisions for 50 convolution coefficients λ from (3) distributed in Λ uniformly has been performed. The obtained results were averaged over the number of solved MCO problems. In Figure 1 an example of two criteria as well as the result of convolution of the criteria and the feasible domain are presented.

The numerical experiments have been performed with stopping upon achievement the method accuracy. For the checking, the points of the solution found by the method have been compared to the points of the Pareto domain approximation computed taking into account the selected convolution coefficients λ. The accuracy of method $\varepsilon = 0.02$ and the reliability parameter $r = 5.6$ were used when solving the series of problems. The results of the numerical experiments are presented in Table 2.

Table 2. The results of the series of experiments on solving the two-dimensional bi-criterial constrained MCO problems

Number of computing cores	Search information				S1	S2
	not used		Used			
	Number of iterations	Problems solved	Number of iterations	Problems solved		
1	26191.8	88%	1420.5	93%	18.4	18.4
2	12146.1	85%	653.3	90%	18.6	40.1
5	5019.3	85%	285.7	91%	17.6	91.7
10	2141.5	85%	152.5	93%	14.0	171.8
25	1022.4	88%	88.6	94%	11.5	295.6

In the first column of Table 2, the number of computing cores employed for solving the problems from the considered series of experiments is given. In the second and fourth columns, the averaged number of iterations executed by the PAMCGS algorithm for the solving of the optimization problem are presented. The third and fifth columns contain the percentage of the solved problems at given parameters of the method. The last two columns contain the information on the obtained speedup. The column (S1) shows the effect of the reuse of the accumulated search information. The column (S2) contains the information on the total speedup achieved as compared to the initial algorithm without the use of the search information.

Table 3. The results of the series of experiments on solving the four-dimensional three-criterial constrained MCO problems

Number of computing cores	Search information				S1	S2
	not used		Used			
	Number of iterations	Problems solved	Number of iterations	Problems solved		
1	49 988 246.5	91%	6 153 261.0	90%	8.1	8.1
2	20 369 550.2	90%	2 400 575.1	89%	8.5	20.8
5	8 228 684.5	90%	709 672.4	92%	11.6	70.4
10	5 582 125.8	92%	702 522.4	91%	7.9	71.2
25	1 704 359.8	91%	204 342.8	90%	8.3	244.6

The obtained results of experiments (Table 2) demonstrate that the reuse of the search information to allow reducing the total amount of computations by the factor of 18.4 without employing any additional computational resources. When using 25 computer cores, the maximun speedup reaches 295.6 times.

In the third series of the numerical experiments, the solving of the three-criterial four-dimensional MCO problems with five constraints (i. e. *N*=4, *s*=3, *m*=5) has been performed. The criteria and constraints of the MCO problems to be solved were generated with the use of the GKLS generator [27] as in the previous experiment. When solving the problem series, the accuracy of the method $\varepsilon = 0.01$ and the reliability parameter $r = 5.6$ were used. The results of the numerical are presented in Table 3. As it can be noted, for example, the speedup achieved when using 25 computing cores was 244.6 times.

7 Conclusion

In the present paper, an efficient parallel method is proposed for solving complex multicriterial optimization problems with non-convex constraints, where the criteria of optimality could be the multiextremal ones, and the computing of the criteria values could require a large amount of computations. The proposed approach is based on the reduction of the multicriterial problems to the nonlinear programming ones by means of the minimax convolution of the partial criteria, on the dimensionality reduction with the use Peano space-filling curves, and of application of the efficient information-statistical global optimization methods with a novel index scheme of the constraints handling instead of the penalty functions applied usually.

The key aspect of the developed approach is the overcoming of a large computational complexity of the global search of the set of effective decisions when solving the multicriterial optimization problems. A considerable increase in the efficiency and an essential reduction of the amount of computations was provided by means of the maximum possible use of the search information obtained in the course of computations. To do so, it was necessary to provide the possibilities of storing large amounts of the search information, of its efficient processing, and of using the search data in the course of solving the multicriterial optimization problems. Within the framework of the developed approach, the methods for converting all available search information to the values of current scalar problem of nonlinear programming being solved have been proposed. The search information transformed to current optimization problem was used by the applied optimization methods for the adaptive planning of the global search iterations performed. The availability of the search information allows also executing the parallel computation efficiently providing the choice of the most promising points of the search domain when searching the effective decisions for the MCO problems.

The results of the numerical experiments have demonstrated the developed approach to allow reducing the computational costs of solving the multicriterial optimization problems with the non-convex constraints essentially – tens and hundreds times.

As a conclusion, one can note that the developed approach is a promising one and needs continuing the investigations further. First of all, it is necessary to continue carrying out the numerical experiments on solving the multicriterial optimization problems with larger number of partial criteria and constraints for lager dimensionalities of the optimization problems to be solved. Also, a possibility of parallel computations for the high-performance systems with distributed memory should be explored.

Acknowledgements

This research was supported by the Russian Science Foundation, project No 16-11-10150 “Novel efficient methods and software tools for time-consuming decision making problems using supercomputers of superior performance.”

References

1. Miettinen K.: Nonlinear Multiobjective Optimization. Springer (1999)
2. Ehrgott, M.: Multicriteria Optimization. Springer (2nd ed., 2010)
3. Collette, Y., Siarry, P.: Multiobjective Optimization: Principles and Case Studies (Decision Engineering). Springer (2011)
4. Marler, R. T., Arora, J. S.: Survey of multi-objective optimization methods for engineering. Struct. Multidisciplinary Optimization 26, 369-395 (2004)
5. Figueira,J., Greco, S., Ehrgott, M., eds: Multiple Criteria Decision Analysis: State of the Art Surveys. New York (NY): Springer (2005)
6. Eichfelder, G.: Scalarizations for adaptively solving multi-objective optimization problems. Comput. Optim. Appl. 44, 249–273 (2009)
7. 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)
8. Pintér, J.D.: Global Optimization in Action (continuous and Lipschitz optimization: algorithms, implementations and applications). Kluwer Academic Publishers, Dortrecht (1996)
9. Strongin, R.G.: Numerical Methods in Multiextremal Problems: Information-Statistical Algorithms. Nauka, Moscow (1978) (in Russian)
10. Strongin, R., Sergeyev, Ya.: Global Optimization with Non-convex Constraints. Sequential and Parallel Algorithms. Kluwer Academic Publishers, Dordrecht (2nd ed. 2013, 3rd ed. 2014)
11. Sergeyev Y.D., Strongin R.G., Lera D.: Introduction to Global Optimization Exploiting Space-filling Curves. Springer (2013)
12. Floudas, C.A., Pardalos, M.P.: Recent Advances in Global Optimization. Princeton University Press (2016)
13. Horst, R., Tuy, H.: Global Optimization: Deterministic Approaches. Berlin: Springer-Verlag (1990)
14. Locatelli, M., Schoen, F.: Global Optimization: Theory, Algorithms, and Applications. SIAM (2013)
15. Törn, A., Žilinskas, A.: Global Optimization. Lecture Notes in Computer Science 350. Berlin: Springer-Verlag (1989)
16. Zhigljavsky, A.A.: Theory of Global Random Search. Dordrecht: Kluwer Academic Publishers (1991)

17. Branke, J., Deb, K., Miettinen, K., Slowinski, R., eds.: Multi-Objective Optimization—Interactive and Evolutionary Approaches. Springer, Berlin (2008)
18. Yang, X.-S.: Nature-inspired metaheuristic algorithms. Luniver Press, Frome (2008)
19. Marler, R. T., Arora, J. S.: Multi-Objective Optimization: Concepts and Methods for Engineering. VDM Verlag (2009)
20. Hillermeier, C., Jahn, J.: Multiobjective optimization: survey of methods and industrial applications. Surv. Math. Ind. 11, 1–42 (2005)
21. Evtushenko, Yu.G., Posypkin, M.A.: A deterministic algorithm for global multi-objective optimization. Optimization Methods & Software, 29 (5), 1005–1019 (2014)
22. Deb, K.: Multi-Objective Optimization using Evolutionary Algorithms. Wiley, Chichester (2001)
23. Tan, K.C., Khor, E.F., Lee, T.H.: Multi-Objective Evolutionary Algorithms and Applications. Springer-Verlag, London (2005)
24. 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)
25. Jones, D.R.: A taxonomy of global optimization methods based on response surfaces. J. Glob. Optim. 21, 345–383 (2001)
26. Voutchkov, I., Keane, A.: Multi-objective optimization using surrogates. Comput. Intell. Optim. Adapt. Learn. Optim. 7, 155–175 (2010)
27. Gaviano, M., Kvasov, D.E, Lera, D., and Sergeyev, Ya.D.: Software for generation of classes of test functions with known local and global minima for global optimization. ACM Transactions on Mathematical Software 29(4), 469-480 (2003)
28. Barkalov, K., Gergel, V., Lebedev, I.: Use of Xeon Phi Coprocessor for Solving Global Optimization Problems. LNCS, Springer, Heidelberg. 9251, 307–318 (2015)
29. Gergel, V.P., Grishagin, V.A., Gergel, A.V.: Adaptive nested optimization scheme for multidimensional global search. Journal of Global Optimization, 66(1), 1–17 (2015)
30. Barkalov, K., Gergel, V.: Parallel Global Optimization on GPU. J. Glob. Optim. 66(1), 3-20 (2016)
31. Gergel, V., Lebedev, I.: Heterogeneous Parallel Computations for Solving Global Optimization Problems. In: Procedia Computer Science, 66, pp. 53–62 (2015)
32. Gergel, V.P., Kuzmin, M.I., Solovyov, N.A., Grishagin, V.A.: Recognition of surface defects of cold-rolling sheets based on method of localities. International Review of Automatic Control 8 (1), 51-55 (2015)
33. Kvasov D.E., Sergeyev Y.D.: Deterministic approaches for solving practical black-box global optimization problems. Advances in Engineering Software, vol. 80, pp.58-66 (2015)
34. Sergeyev Ya.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)