Parallel Computing for Time-Consuming Multicriterial Optimization Problems

Victor Gergel, Evgeny Kozinov

Lobachevski National Research University
of Nizhnii Novgorod
gergel@unn.ru, evgeny.kozinov@itmm.unn.ru

Abstract. In the present paper, an efficient method for parallel solving the time-consuming multicriterial optimization problems, where the optimality criteria can be multiextremal, and the computation of the criteria values can require a large amount of computations, is proposed. The proposed scheme of parallel computations allows obtaining several efficient solutions of a multicriterial problem. During performing the computations, the maximal use of the whole search information is provided. The results of the numerical experiments have demonstrated such an approach to allow reducing the computational costs of solving the multicriterial optimization problems essentially – several tens and hundred times.

Keywords: Decision making, multicriterial optimization, parallel computing, dimensionality reduction, criteria convolution, algorithm of global search, computation complexity.

1 Introduction

The statements of the multicriterial optimization (MCO) problems belong to the most general models of the decision making problems. A general state of the art in the field of multicriterial optimization is presented in the monographs [2-4,19], the reviews of the scientific and practical results are given in [1,5-8,20].

At the same time, the MCO problems are the most complicated ones. As a rule, the solving of the MCO problems is reduced to finding some compromised solutions, which obtaining the best values with respect to particular partial criteria is coordinated in.

The necessity to find several efficient solutions increases the computational complexity of solving the MCO problems essentially. In such conditions, finding even a single compromised solution requires a large amount of computations whereas the finding of several efficient solutions (or of the whole Pareto set) becomes a problem of high computation complexity. Addressing this problem becomes possible at the maximum utilization of huge computational capabilities of high-performance systems. And, besides, full utilization

adfa, p. 1, 2011.

of the whole search information obtained in the course of computations is necessary for efficient computations.

Further structure of the paper is as follows. In Section 2, the multicriterial optimization problem statement is given. In Section 3, a scheme of parallel computations for the simultaneous solving of a set of multicriterial global optimization problems is proposed. In Section 4, a parallel algorithm of multicriterial global search for high performance computing systems is presented. Section 5 presents the results of numerical experiments. In Conclusion, the obtained results are discussed and main directions of further investigations are outlined.

2 Multicriterial Optimization Problem Statement

The problem of multicriterial (or multi-objective) optimization (MCO) can be defined as follows:

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

where $y = (y_1, y_2, \dots, y_N)$ is a vector of varied parameters, N is the dimensionality of the problem being solved, and D is the search domain being an N-dimensional hyperparallelepiped

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

at given boundary vectors a and b. Without a loss in generality, the partial criteria values in problem (1) are supposed to be non-negative, and the decrease of these ones corresponds to the increase of the efficiency of the considered solutions $y \in D$. It is also supposed that the partial criteria $f_i(y)$, $1 \le i \le s$ satisfy the Lipschitz condition

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

where L_i, $1 \le i \le s$ are the Lipschitz constants for the corresponding partial criteria $f_i(y)$, $1 \le i \le s$, and $\|.\|$ denotes the norm in R^N space.

As a solution of the MCO problem, any efficient solution (*a partial solution*) is considered. In the general case, in solving a MCO problem, it is necessary to find the whole set of Pareto-optimal solutions $PD(f, D)$ (*a full solution of a MCO problem*).

In the present work, the MCO problems are considered in the context of the most complicated problems of decision making, where the partial criteria $f_i(y)$, $1 \le i \le s$ can be multiextremal, and the obtaining of the criteria values at the points of the search domain $y \in D$ can require a considerable amount of computations. In the paper, it is proposed to use the minimax convolution of

partial criteria, according to which the solving of the problem (1) is reduced to the solving of a family of the global optimization problems:

$$\min\{F(\lambda,y)=\max\left(\lambda_i f_i(y),1\le i\le s\right)+\rho\sum_{i=1}^{s}(\lambda_i f_i(y))\},$$
$$\lambda\in\Lambda\subset R^s:\ \sum_{i=1}^{s}\lambda_i=1,\ \lambda_i\ge 0,\ 1\le i\le s, \tag{3}$$

where $\rho>0$ is a small positive number (the last term in the expression for $F(\lambda,y)$ allows obtaining the Pareto optimal solutions at the appropriate value of coefficient ρ – see, for example, [22, 23]). It is worth noting that the integrated scalar criterion $F(\lambda,y)$ also satisfies the Lipschitz condition

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

3 Parallel Computations in the Multicriterial Global Optimization Problems

The scalarization of the vector criterion allows reducing the solving of the MCO problem (1) to solving a series of the multiextremal problems (3). And, therefore, the problem of development of the methods for solving the MCO problems is resolved by the possibility of a wide use of the efficient parallel global search algorithms.

The proposed approach of parallel computations for solving time consuming global search problems is based on the following main statements:

- The parallelism of the performed computations is provided by means of simultaneous computing the values of the partial criteria $f_i(y)$, $1\le i\le s$ at several different points of the search domain D. Such an approach provides the parallelization of the most computation costly part of the global search processes and is a general one – it can be applied for many global search methods for various global optimization problems.
- The parallel computations of the partial criteria values $f_i(y)$, $1\le i\le s$ is provided by means of the simultaneous solving of several global optimization problems (3) for various values of the coefficients λ_i, $1\le i\le s$. For solving the problems of the family (3), a set of computational nodes of the high performance systems with distributed memory can be applied.
- In the course of parallel computations, the results of computations are interchanged between all employed processors because of the information compatibility of the global optimization problems of the family (3).

Below, these statements will be considered in more details.

3.1 Structure of the Global Search Information

The numerical solving of the optimization problems consists usually in a sequential computing the values of the partial criteria $f^i = f(y^i)$ at the points $y^i,\ 1 \le i \le k$ of the search domain D. The search information obtained as a result of computations can be represented in the form of the *Set of the Search Information* (SSI):

$$\Omega_k = \left\{\left(y^i, f^i = f(y^i)\right)^T : 1 \le i \le k\right\}. \tag{5}$$

The availability of SSI allows reducing the results of the previous computations to the values of any next optimization problem (3) being solved without any time-consuming computations of the partial criteria values $f_i(y)$, $1 \le i \le s$ from (1) for new values of the convolution coefficients λ'_j, $1 \le j \le s$ i. e.

$$z'_i = max\left(\lambda'_j f_i^j, 1 \le j \le s\right), 1 \le i \le k. \tag{6}$$

And, this way, all the search information in full amount can be employed for continuing the computations. In general, the reuse of the search information will require less and less computations for solving every next optimization problem down to the execution of several iterations only to find the next efficient solution.

3.2 General Scheme of Parallel Computations

As it has been already mentioned above, when solving a multicriterial optimization problem (1), in order to find several different efficient solutions, solving a series of scalar problems (3) with various values of the coefficients of the minimax convolution of the partial criteria may be required

$$\vec{\Phi}(y) = \{\varphi_1(y), \dots, \varphi_q(y)\},\ \ \varphi_i(y) = F(\lambda_i, y),\ 1 \le l \le q. \tag{7}$$

The problems of the family $\vec{\Phi}(y)$ can be solved sequentially, various global optimization methods can be used for solving the problems. On the other hand, these problems can be solved simultaneously with the use of several processors as well. At that, it is important to note that the obtained family of the one-dimensional problems $\vec{\Phi}(y)$ is an information-linked one – the values of the optimized functions computed for any problem $\varphi_l(y)$, $1 \le l \le q$ can be reduced to the values of all the rest problems of the family without the time-consuming recalculations of the partial criteria values $f_i(y)$, $1 \le i \le s$ according to (6).

The information compatibility of the problems from the family (7) allows proposing the following method of parallel computations. The solving of each particular problem can be performed on a separate processor of the computing system; the exchange of the obtained search information between the processors should be performed in the course of computations.

As a result of the use of such a parallelization approach, a unified approach to the parallel computations for the multiprocessor computing systems with the distributed memory can be developed. The general scheme of the parallel computations consists in the following.

1. The family of the one-dimensional reduced information-linked problems $\vec{\Phi}(y)$ from (7) is distributed among the processors of the computing system. A single problem from the family $\vec{\Phi}(y)$ as well as several ones can be allocated to each particular processor.
2. For solving the selected problems from the family (7), the algorithm of global search updated by the following rules of the information interaction is applied on each processor:
   (a) Prior to the beginning of a new global optimization iteration, for any problem $\varphi_l(y)$, $1 \le l \le q$ at any point $y' \in D$, this point y should be transferred to all employed processors in order to exclude the repeated computation of the particular criteria values $f_i(y)$, $1 \le i \le s$ at this point. In order to organize the data transfer, a queue for receiving the transferred points and the partial criteria values at these points can be organized at each computational node.
   (b) Upon completing the iteration for any problem $\varphi_l(y)$, $1 \le l \le q$ at any point $y' \in D$, the point y' together with the particular criteria values $f_i(y)$, $1 \le i \le s$ computed at this point should be transferred to all employed processors.
   (c) Prior to beginning the next global search iteration, the algorithm should check the queue of the received messages; if there are any data in this queue, the received information should be included into the search information Ω_k from (5).

The scheme of the parallel computing considered above provides the completeness of the whole search information Ω_k from (5) for all problems from the family $\vec{\Phi}(y)$.

The possibility of the asynchronous data transfer is a principal feature of such a scheme of the parallel computations. Besides, any single control node is absent in this scheme, and the number of computational nodes can vary in the course of global search.

4 Parallel Methods for Solving Multicriterial Global Optimization Problems

The multiextremal optimization is a research direction being developed extensively – the general state of the art is presented, for example, in [9-11,13,14,17,18]. The information-statistical theory of global optimization is one of the promising approaches – see, for example, [10,11,27,31]. The high performance computing systems are used widely for solving the time-consuming global search problems [11,16,24-26,29,31].

4.1 Parallel Algorithm of Multicriterial Global Optimization

The approach is based on the following two statements:

- In order to reduce the complexity of the computational analysis of a large multidimensional search information Ω_k from (5), the reduction of the dimensionalities of the MCO problems is applied.
- For solving the problems from the family (3), the efficient global search algorithms developed within the framework of the information-statistical theory of the multiextremal optimization [10,11] are used.

These statements are presented in more details below.

Dimensionality reduction. Within the framework of the proposed approach, the Peano *space-filling curves* or *evolvents* $y(x)$ mapping the interval [0,1] onto the *N*-dimensional hypercube *D* unambiguously (see, for example, [10-12]) were used for the dimensionality reduction. As a result of such reduction, the initial multidimensional global optimization problem (3) is reduced to a one-dimensional problem:

$$\varphi(x^*) = \min\{\varphi(x): x \in [0,1]\}, \tag{8}$$

where $\varphi(x) = F(\lambda, y(x))$.

The dimensionality reduction scheme reduces the multidimensional problem (3) with the Lipschitzian minimized function to a one-dimensional problem (8), where the corresponding functions satisfy the uniform Hölder condition i. e.

$$|F(\lambda, y(x') - F(\lambda, y(x'')| \le H|x' - x''|^{1/N}, \; x', x'' \in [0,1], \tag{9}$$

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

As a result of the dimensionality reduction, the search information Ω_k from (5) can be transformed into the *Matrix of the Search State* (MSS)

$$A_k = \{(x_i, z_i, l_i)^T : 1 \le i \le k\}, \tag{10}$$

where x_i, $1 \le i \le k$ are the reduced points of the executed global search iterations[1], z_i, $1 \le i \le k$ are the values of the scalar criterion of current optimization problem (8) being solved, l_i, $1 \le i \le k$ are the indices of the global search iterations, which the points x_i, $1 \le i \le k$ were computed within.

The matrix of the search state can be used by the optimization algorithms in order to improve the efficiency of global search – selecting the points for the scheduled iterations can be performed taking into account the results of all computations performed before. Besides, the availability of the MSS allows computing the numerical estimates of the Hölder constant H from (9)

$$m = \begin{cases} rM, & M > 0, \\ 1, & M = 0, \end{cases} \qquad M = \max_{1<i\le k} \frac{|z_i - z_{i-1}|}{\rho_i}, \tag{11}$$

as the relative differences of the values of the minimized function $z_i = \varphi(x_i)$, $1 \le i \le k$ 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 *parameter of reliability* of the estimate of the constant H.

Parallel algorithm. Within the proposed approach, for solving the reduced one-dimensional multiextremal optimization subproblems (8), it is proposed to use well-known Multidimensional Algorithm of Global Search (MAGS) developed within the framework of the information-statistical theory of the multiextremal optimization [10,11]. This method has a good theoretical substantiation and has demonstrated a high efficiency as compared to other global search algorithms (see also the results of numerical experiments in Section 5).

For the sake of completeness, let us consider briefly the general computational scheme of MAGS, which consists in the following.

Let k, $k > 1$ global search iterations have been executed, the computing of the minimized function values $\varphi(x)$ (hereafter this procedure is called the *trial*) has been performed within every iteration, and the obtained search information is represented in the form of A_k from (10). The choice of the trial points for the next $(k+1)^{th}$ iteration is determined by the following rules.

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

[1] The lower indices denote the increasing order of the coordinate values of the points x_i, $1 \le i \le k$.

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

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

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

Rule 3. Compute the trial point of the next global search iteration x^{k+1} within the interval t, $1 < t \le k$ with the highest characteristic $R(t)$ from (13):

$$x^{k+1} = \frac{x_t + x_{t-1}}{2} - \frac{z_t - z_{t-1}}{2m}.$$

Termination condition, which the calculations are stopped according to, is defined by the inequality

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

which should be checked for the interval t from (13), which the scheduled trial is performed in. The quantity $\varepsilon > 0$ is the predefined *accuracy* of the problem solution. If the termination condition is not fulfilled, the iteration index k is incremented by 1, and the execution of the algorithm is continued.

As the current estimate of the optimization problem solution at every global search iteration, the lowest computed value of the minimized function is accepted i. e.:

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

Additional information on the MAGS algorithm is given in [11]. Here, it should be noted that the characteristics $R(i)$, $1 < t \le k$ being computed from (12) can be interpreted as some measures of importance of the intervals with respect to the location of the global minimum point in these ones.

Within the framework of the proposed approach, the MAGS algorithm is applied to solving every problem from the family $\vec{\Phi}(y)$ in combination with the general scheme of the parallel computations presented in Sec. 3. The method obtained as a result of such extension is called hereafter Parallel Multicriterial Global Algorithm (PMGA) for high-performance computing systems with distributed memory.

4.2 Multilevel Parallel Algorithm of Multicriterial Global Optimization

The general scheme of parallel computations considered in Subsection 4.1 can be extended for the simultaneous computing of several minimized function

values for every optimization problem from the family $\overrightarrow{\Phi}(y)$ being solved on a separate multiprocessor multicore node with shared memory. For this purpose, a parallel generalization of the MAGS method can be applied – see, for example, [11,16,22]. This generalization consists in the following.

Let p is the number of employed parallel computational units (processors or cores) of a system with shared memory. The rules of the parallel algorithm correspond to the computational scheme of the MAGS method (see Subsection 4.1) except the rules of computation of the next global search iteration points. Below, for the sake of brevity, the modified rules for the parallel algorithm only are given.

Rule 2 (updated). Arrange the characteristics of the intervals obtained in (12) in the decreasing order

$$R(t_1) \geq R(t_2) \geq \ldots \geq R(t_{k-1}) \geq R(t_k) \tag{16}$$

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

Rule 3 (updated). Perform new trials (the computations of the minimized function values $\varphi(x)\big)$ at the points x^{k+j},$1 \leq j \leq p$ located in the intervals with the highest characteristics from (16)

$$x^{k+j} = \frac{x_{t_j}+x_{t_j-1}}{2} - sign\left(z_{t_j} - z_{t_j-1}\right)\frac{1}{2r}\left[\frac{\left|z_{t_j}-z_{t_j-1}\right|}{m}\right]^N, 1 \leq t_j \leq p.$$

The termination condition (14) in the parallel algorithm, which the trials are terminated according to, should be checked for all intervals from (16), which the scheduled trials are performed in, i. e.:

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

The PMGA algorithm updated by the scheme of parallel computations for the computational nodes with shared memory will be named hereafter Multilevel Parallel Multicriterial Global Algorithm (MPMGA).

5 Results of Numerical Experiments

The numerical experiments have been carried out on the «Lobachevskii» supercomputer at State University of Nizhnii Novgorod (operating system – CentOS 6.4, supercomputer management system – SLURM). Each supercomputer node had 2 Intel Sandy Bridge E5-2660 2.2 GHz processors, 64 Gb RAM. Each processor had 8 cores (i. e. total 16 CPU cores were available at each node). To generate the executable program code, Intel C++ 14.0.2 compiler was used.

Prior to conducting the numerical experiments, let us consider the results of comparison of the proposed approach with a number of other multicriterial optimization algorithms presented in [28]. For the comparison, the bi-criterial test problem proposed in [29] was used:

$$f_1(y) = (y_1 - 1)y_2^2 + 1,\ f_2(y) = y_2,\ 0 \le y_1, y_2 \le 1. \qquad (17)$$

As a solution of a MCO problem, the construction of a numerical approximation of the Pareto domain (PDA) was considered. To evaluate the quality of approximation, the completeness and the uniformity of coverage of the Pareto domain were compared using the following two indicators [28, 29]:

- The *hypervolume index* (HV) defined as the volume of the subdomain of the values of the vector criterion *f*(*y*) dominated by the points of approximation of the Pareto domain. This indicator characterizes the completeness of approximation of the Pareto domain (the higher the value, the more complete the coverage of the Pareto domain).
- The *distribution uniformity index* (DU) of the points from the Pareto domain approximation. This indicator characterizes the uniformity of coverage of the Pareto domain (the less the value, the more uniform the coverage of the Pareto domain).

Within the framework of the considered experiment, five multicriterial optimization algorithms were compared: the Monte-Carlo (MC) method, the genetic algorithm SEMO from the PISA library [20,32], the Non-uniform coverage (NUC) method [20], the bi-objective Lipschitz optimization (BLO) method proposed in [32], and the serial version of the MPMGA algorithm proposed in the present paper. Total 50 problems (3) have been solved by MPMGA with various values of the convolution coefficients λ distributed in Λ from (3) uniformly. The results of experiments from [28] are presented in the complete form in Table 1.

Table 1. Results of numerical experiments from [28] for the test problem (17)

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+	370	100	0.316	0.101

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

The numerical experiments have been carried out on solving the bi-criterial two-dimensional MCO problems i. e. for $N = 2$, $s = 2$. As the problem criteria, the multiextremal functions were used defined by the relations:

$$\phi(y_1, y_2) = -(AB + AC)^{\frac{1}{2}}$$

$$AB = \left(\sum_{i=1}^{7} \sum_{j=1}^{7} \left[A_{ij} a_{ij}(y_1, y_2) + B_{ij} b_{ij}(y_1, y_2) \right] \right)^2$$

$$CD = \left(\sum_{i=1}^{7} \sum_{j=1}^{7} \left[C_{ij} a_{ij}(y_1, y_2) - D_{ij} b_{ij}(y_1, y_2) \right] \right)^2$$

where

$$a_{ij}(y_1, y_2) = \sin(\pi i y_1) \sin(\pi j y_2),\ b_{ij}(y_1, y_2) = \cos(\pi i y_1) \cos(\pi j y_2)$$

are defined in the range $0 \le y_1, y_2 \le 1$, and the parameters $-1 \le A_{ij}, B_{ij}, C_{ij}, D_{ij} \le 1$ are the independent random numbers distributed uniformly.

Table 2. Results of a series of experiments on solving the two-dimensional bi-criterial MCO problems

P	Q	P*Q	1-25	26-50	1-50	S_1	S_2
Computations without the reuse of the search information							
1	1	1	8 571,6	8 590,2	17 165,9	-	**1**
Computations with the reuse of the search information							
1	1	1	1 199,5	573,9	1 773,4	**1**	**9,7**
1	25	25	52,1	27,6	79,7	**22,2**	**215,4**
25	1	25	135,1	54,8	189,8	**9,3**	**90,4**
5	5	25	66,8	37,1	103,9	**17,1**	**165,2**
25	25	625	8,6	8,1	16,7	**106,3**	**1 029,1**

The solving of 100 multicriterial problems has been performed in the course of experiments. In each problem, the search of the Pareto-optimal solutions has been performed for 50 convolution coefficients λ distributed in Λ from (3) uniformly. The obtained results were averaged over the number of solved MCO problems.

The results of numerical experiments are presented in Table 2. The first two columns in Table 2 denote the numbers of the processors (P) and of the parallel threads on each processor (Q) employed. The third column (P*Q) contain the total number of threads employed. In the fourth, fifth, and sixth columns, the numbers of iterations necessary to find the solution in given groups of problems from the family (3) for the corresponding numbers of different coefficients λ from (3) are given. The last two columns contain the information on the speedup of the parallel computations obtained with the use of the search information (S_1) and without the one (S_2).

The obtained results of experiments demonstrate that even simple reuse of the search information allows reducing the total amount of computations 9.7 times without the use of additional computational resources. When using additional 25 processors, one can obtain the speedup from 9.3 up to 22.2 times. If 25 processors were employed in the computations and 25 parallel threads were used for each processor, the speedup with the reuse of the search information reaches 106.3 times. The overall speedup in this case relative to the initial algorithm without the reuse of the search information was more than 1029 times.

6. Conclusion

In the present article, an efficient method of solving the time-consuming multicriterial optimization problems, where the optimality criteria can be multiextremal and computing the criteria values can require a large amount of computations has been proposed. The key aspect of the developed approach consists in the overcoming of the high computational complexity of the global search of the set of the efficient solutions in solving the multicriterial optimization problems. A considerable improvement of the efficiency and an essential reduction of the amount of computations have been provided by means of the maximal possible utilization of the whole search information obtained in the course of computations. Within the framework of the developed approach, the methods for reducing the whole available search information to the values of current scalar nonlinear programming problem being solved have been proposed. The search information is used by the optimization methods for the adaptive planning of the executed global search iterations.

The results of the numerical experiments have demonstrated such an approach to allow reducing the computation costs of solving the multicriterial optimization problems considerably – tens and hundreds times.

In conclusion, one can note that the developed approach is a promising one and requires further investigations. First of all, it is necessary to continue carrying out the numerical experiments on solving the multicriterial optimization

problems with more partial criteria of efficiency and for a greater dimensionality of the optimization problems being solved.

Acknowledgements

This work has been supported by 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. 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
2. Miettinen K.: Nonlinear Multiobjective Optimization. Springer (1999)
3. Ehrgott, M.: Multicriteria Optimization. Springer (2nd ed., 2010)
4. Collette, Y., Siarry, P.: Multiobjective Optimization: Principles and Case Studies (Decision Engineering). Springer (2011)
5. Marler, R. T., Arora, J. S.: Survey of multi-objective optimization methods for engineering. Struct. Multidisciplinary Optimization **26**, 369–395 (2004)
6. Figueira,J., Greco, S., Ehrgott, M. (eds.): Multiple criteria decision analysis: State of the art surveys. New York (NY): Springer (2005)
7. Eichfelder, G.: Scalarizations for adaptively solving multi-objective optimization problems. Comput. Optim. Appl. **44**, 249–273 (2009)
8. Siwale, I.: Practical multi-objective programming. Technical Report RD-14-2013. Apex Research Limited (2014)
9. Pintér, J.D.: Global optimization inaction (continuous and Lipschitz optimization: algorithms, implementations and applications). Kluwer Academic Publishers, Dordrecht (1996)
10. Strongin, R.G.: Numerical methods in multiextremal problems: information-statistical algorithms. Nauka, Moscow (1978) (in Russian)
11. Strongin, R., Sergeyev, Ya.: Global optimization with non-convex constraints. Sequential and parallel algorithms. Kluwer Academic Publishers, Dordrecht (2000)
12. Sergeyev Y.D., Strongin R.G., Lera D.: Introduction to global optimization exploiting space-filling curves. Springer (2013)
13. Floudas, C.A., Pardalos, M.P.: Recent Advances in Global Optimization. Princeton University Press (2016)
14. Locatelli, M., Schoen, F.: Global Optimization: Theory, Algorithms, and Applications. SIAM. (2013)
15. Sergeyev Y.D.: An Information Global Optimization Algorithm with Local Tuning. SIAM Journal on Optimization, **5**(4), 858–870 (1995)
16. Sergeyev, Y.D., Grishagin, V.A.: Parallel Asynchronous Global Search and the Nested Optimization Scheme. J. Comput. Anal. Appl., **3**(2), 123–145 (2001)
17. Törn, A., Žilinskas, A.: Global Optimization. Lecture Notes in Computer Science 350. Berlin: Springer-Verlag (1989)

18. Zhigljavsky, A.A.: Theory of Global Random Search. Dordrecht: Kluwer Academic Publishers (1991)
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. Forrester,A.I.J., Keane,A.J.: Recent advances in surrogate-based optimization. Prog.Aerosp.Sci. **45**(1), 50–79 (2009)
22. Krasnoshekov, P.S., Morozov, V.V., Fedorov, V.V.: Decompozition in Design Problems. Engineering Cybernetics, **2**, 7–17 (1979) (in Russian)
23. Wierzbicki, A.: The Use of Reference Objectives in Multiobjective Optimization. G. Fandel and T. Gal (Eds.), Multiple Objective Decision Making, Theory and Application. Springer-Verlag, New York. **177**, 468–486 (1980)
24. Gergel, V., Sidorov, S.: A Two-Level Parallel Global Search Algorithm for Solution of Computationally Intensive Multiextremal Optimization Problems. LNCS. Springer-Verlag, Berlin Heidelberg. **9251**, 505–515. (2015)
25. Barkalov, K., Gergel, V., Lebedev, I.: Use of Xeon Phi Coprocessor for Solving Global Optimization Problems. LNCS, Springer, Heidelberg. **9251**, 307–318 (2015)
26. Gergel, V.: An 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
27. 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)
28. Gergel, V., Kozinov, E.: Efficient Multicriterial Optimization Based on Intensive Reuse of Search Information (to be published).
29. Evtushenko, Yu.G., Posypkin, M.A.: A deterministic algorithm for global multi-objective optimization. Optimization Methods and Software, **29**(5), 1005–1019 (2014)
30. 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
31. Strongin R., Gergel V., Grishagin V., Barkalov K.: Parallel computations for global optimization problems. Moscow State University (2013) (in Russian)
32. Zilinskas, A., Zilinskas, 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)