Multidimensional Constrained Global Optimization in Domains with Computable Boundaries

Vladimir Grishagin and Ruslan Israfilov

Lobachevsky State University of Nizhni Novgorod, Nizhni Novgorod, Russia
vagris@unn.ru,ruslan@israfilov.com

Abstract. Multidimensional constrained global optimization problem with objective function under Lipschitz condition and constraints generating a feasible domain with computable boundaries is considered. For solving this problem the dimensionality reduction approach on the base of the nested optimization scheme is used. This scheme reduces initial multidimensional problem to a family of one-dimensional subproblems and allows applying univariate methods for the execution of multidimensional optimization. Sequential and parallel modifications of well-known information-statistical methods of Lipschitz optimization are proposed for solving the univariate subproblems arising inside the nested scheme in the case of domains with computable boundaries. A comparison with classical penalty function method being traditional means of taking into account the constraints is carried out. The results of experiments demonstrate a significant advantage of the methods proposed over the penalty function method.

Keywords: global optimum · multidimensional problems · nested recursive optimization · computable boundaries

1 Introduction

The global optimization problem [1–3] is considered in the following form:

$$f(y) \to \min, \quad y \in Q \subseteq \mathbb{R}^N, \tag{1}$$

where the feasible domain

$$D = \{y \in \mathbb{R}^N : \ a_i \le y_i \le b_i, 1 \le i \le N\}, \tag{2}$$

$$Q = \{y \in D : g_i(y) \le 0, 1 \le i \le m\}, \tag{3}$$

is defined by constant (2) and functional (3) constraints. The objective function is supposed to satisfy in the domain Q the Lipschitz condition

$$|F(y') - F(y'')| \le L\|y' - y''\|, \quad y', y'' \in Q \tag{4}$$

with the Lipschitz constant $L > 0$, where the denotation $\|\cdot\|$ signifies the Euclidean norm in $\mathbb{R}^N$.

The traditional way of solving the constrained problems of mathematical programming consists in solving the unconstrained problem

$$F(y) \to \min, \quad y \in D, \tag{5}$$

where

$$F(y) = f(y) + CG(y). \tag{6}$$

An auxiliary function $G(y)$ (penalty function) satisfies the condition

$$G(y) = 0,\ y \in Q, \quad G(y) > 0,\ y \notin Q, \tag{7}$$

and the constant $C > 0$ (see [1, 2, 4]).

If the constant C is sufficiently large and functions $f(y)$ and $G(y)$ are, for instance, continuous, solutions of the problems (1) and (5) coincide. For some function classes there is a finite penalty constant providing the coincidence of solutions (Eremin-Zangwill exact penalty functions [4, 5]).

As an example of penalty function one can take the function

$$G(y) = \max\{0; g_1(y), \ldots, g_m(y)\}. \tag{8}$$

If all the functions $g_j(y)$, $1 \le j \le m$ are continuous in D, the function (8) is continuous as well.

On the one hand, the problem (5) is simpler than the initial problem (1) because of simplifying the feasible domain D. On the other hand, a choice of the penalty constant is rather difficult. If it is insufficiently large the global minimizer of the problem (5) can fall out of feasible domain. If the penalty constant is too large, it worsens the properties of the function $F(y)$ in comparison with the initial function $f(y)$ (the function $F(y)$ can have a ravine surface, the Lipschitz constant of $F(y)$ can increase considerably, etc.).

Another factor which influences the complexity of the optimization significantly is the dimension N of the problem. To overcome this complexity, the approaches connected with reducing the multidimensional problem to one or several univariate subproblems are often applied. We consider one of approaches to the dimensionality reduction based on the nested optimization scheme which replaces solving the multidimensional problem (1) by solving a family of univariate subproblems connected recursively. In the framework of this approach in combination with different univariate global search methods [6–12] many sequential and parallel multidimensional algorithms have been proposed [7, 8, 13–16] and applied to practical problems (see, for example, papers [17–19]). The other interesting approaches to parallelizing global search algorithms can be found in publications [21–24].

The scheme of nested optimization consists in the following [3, 7, 20].

Let us introduce the notations

$$u_i = (y_1, \ldots, y_i), \quad \nu_i = (y_{i+1}, \ldots, y_N), \tag{9}$$

allowing to write down the vector y as a pair $y = (u_i, \nu_i)$ for $1 \le i \le N-1$. Assume that $y = \nu_0$ if $i = 0$ and $y = u_N$ for $i = N$.

Let us define also two series of sets. The first series contains sections of the domain Q:

$$S_1 = Q, \quad S_{i+1}(u_i) = \{\nu_i \in \mathbb{R}^{N-i} \colon (u_i, \nu_i) \in Q\}, \quad 1 \le i \le N-1. \tag{10}$$

The second collection of sets consists of projections

$$P_{i+1}(u_i) = \{y_{i+1} \in \mathbb{R} \colon \exists (y_{i+1}, \nu_{i+1}) \in S_{i+1}(u_i)\}, \quad 0 \le i \le N-1, \tag{11}$$

of the sections $S_{i+1}(u_i)$ onto the axis y_{i+1}.

Then, according to [3] and [8] the basic relation of the nested optimization scheme

$$\min_{y \in Q} f(y) = \min_{y_1 \in P_1} \min_{y_2 \in P_2(u_1)} \dots \min_{y_N \in P_N(u_{N-1})} f(y) \tag{12}$$

takes place.

Now let us introduce one more family of functions generated by the objective function $f(y)$:

$$f^N(y) \equiv f(y), \tag{13}$$

$$f^i(u_i) = \min\{f^{i+1}(u_i, y_{i+1}) \colon y_{i+1} \in P_{i+1}(u_i)\}, \quad 1 \le i \le N-1, \tag{14}$$

defining in the projections

$$Q_i = \{u_i \in \mathbb{R}^i \colon \exists (u_i, \nu_i) \in Q, \}, \quad 1 \le i \le N, \tag{15}$$

of the domain Q onto the coordinate axes $y_1, \dots, y_i$.

As it follows from (11), in order to solve the problem (1) – (3) it is sufficient to solve a one-dimensional problem

$$f^1(y_1) \to \min, \quad y_1 \in P_1 \subseteq \mathbb{R}^1. \tag{16}$$

According to (14), each estimation of the function $f^1(y_1)$ at some fixed point $y_1 \in P_1$ consists in solving a one-dimensional problem

$$f^2(y_1, y_2) \to \min, \quad y_2 \in P_2(y_1) \subseteq \mathbb{R}^1. \tag{17}$$

This problem is a one-dimensional minimization with respect to y_2 since y_1 is fixed, etc., up to solving the univariate problem with fixed vector u_{N-1}.

$$f^N(u_{N-1}, y_N) = f(u_{N-1}, y_N) \to \min, \quad y_N \in P_N(u_{N-1}), \tag{18}$$

On the whole, the solving of the problem (1) – (3) is reduced to solving a family of "nested" one-dimensional subproblems

$$f^i(u_{i-1}, y_i) \to \min, \quad y_i \in P_i(u_{i-1}), \tag{19}$$

where the fixed vector $u_{i-1} \in Q_{i-1}$.

The approach of nested optimization can be applied to the problem (5) as well. In this case the domains of one-dimensional search in (19) are simple, namely, they are closed intervals $P_i = [a_i, b_i]$, $1 \le i \le N$.

However, in general case subject to continuity of the penalty function $G(y)$ the projection $P_i(u_{i-1})$ is a system of non-intersecting closed intervals

$$P_i(u_{i-1}) = \bigcup_{s=1}^{q_i} [a_i^s, b_i^s]. \tag{20}$$

where the number of the intervals q_i and their bounds a_i^s, b_i^s, $1 \le s \le q_i$, depend on the vector u_{i-1}, i.e.,

$$q_i = q_i(u_{i-1}), \quad a_i^s = a_i^s(u_{i-1}), \quad b_i^s = b_i^s(u_{i-1}). \tag{21}$$

If the domain Q is such that it is possible to obtain the explicit (analytical) expressions for the values q_i, a_i^s, b_i^s as functions of $u_{i-1} \in Q_{i-1}$ for all $1 \le i \le N$, then the feasible set Q is called *the domain with the computable boundaries.*

2 Sequential and parallel algorithms of global search over a system of closed intervals

Let us introduce a unified form for one-dimensional problems (19) arising inside the nested scheme

$$\varphi(x) \to \min, \quad x \in X = \bigcup_{s=1}^{q} [\alpha_s, \beta_s], \tag{22}$$

where $q \ge 1$, $\alpha_s \le \beta_s$, $1 \le s \le q$, $\beta_s < \alpha_{s+1}$, $1 \le s \le q-1$.

One of the most efficient methods of univariate multiextremal optimization is the information-statistical algorithm of global search (AGS) proposed by Strongin [7] for the case (22) with $q = 1$. We will consider a sequential modification of AGS for the more general situation (22) when $q > 1$, i.e., the domain X of one-dimensional search consists of several intervals and call this modification as AGS-G. This method belongs to the class of characteristical algorithms [21] and can be described like these methods in the following manner.

Let the term "trial" denote an estimation of the objective function value at a point of the domain X, k be a trial number, x^k be the coordinate and $z^k = \varphi(x^k)$ be the result of k-th trial.

The initial stage of AGS-G consists in carrying out $2q$ initial trials at points $x^1 = \alpha_1$, $x^2 = \beta_1$, $x^3 = \alpha_2$, $x^4 = \beta_2, \ldots, x^{2q-1} = \alpha_q$, $x^{2q} = \beta_q$, i.e., $x^{2j-1} = \alpha_j$, $x^{2j} = \beta_j$, $1 \le j \le q$, with corresponding trial results $z^j = \varphi(x^j)$, $1 \le j \le 2q$.

The choice of a point x^{k+1}, $k > 2q$, for implementation of a new $(k+1)$-th trial consists in the following steps.

Step 1 Renumber by subscripts the points $x^1, \ldots, x^k$ of previous trials in increasing order, i.e.,

$$\alpha_1 = x_1 < x_1 < \cdots < x_k = \beta_q, \tag{23}$$

and juxtapose to them the values $z_j = \varphi(x_j)$, $1 \le j \le k$, obtained earlier, i.e., $z_j = \varphi(x^l)$, if $x_j = x^l$.

Step 2 Assign to each interval (x_{j-1}, x_j), $2 \le j \le k$, generated by the points from (25) a feasibility indicator τ_j in the following way: if $x_{j-1} = \beta_j$ and $x_j = \alpha_{i+1}$, or $x_{j-1} = x_j$, then $\tau_j = 0$, otherwise, $\tau_j = 1$.

Step 3 For feasible intervals (x_{j-1}, x_j) with indicator $\tau_j = 1$ calculate the divided differences

$$\lambda_j = \frac{|z_j - z_{j-1}|}{x_j - x_{j-1}}, \tag{24}$$

and the value

$$\Lambda_k = \max\{\lambda_j \colon 2 \le j \le k, \tau_j = 1\}. \tag{25}$$

Step 4 Determine an adaptive estimation

$$M_k = \begin{cases} r\Lambda_k, & \Lambda_k > 0, \\ 1, & \Lambda_k = 0, \end{cases} \tag{26}$$

where $r > 1$ – a parameter of the method.

Step 5 Juxtapose a numerical value (*characteristic*)

$$R(j) = M_k(x_j - x_{j-1}) + \frac{(z_j - z_{j-1})^2}{M_k(x_j - x_{j-1})} - 2(z_j + z_{j-1}), \tag{27}$$

to each feasible interval (x_{j-1}, x_j), $2 \le j \le k$, $\tau_j = 1$.

Step 6 Among feasible intervals select the interval (x_{t-1}, x_t) which the maximal characteristic $R(t)$ corresponds to.

Step 7 Choose in the interval (x_{t-1}, x_t) the point

$$x^{k+1} = \frac{x_t + x_{t-1}}{2} - \frac{z_t - z_{t-1}}{2M_k}, \tag{28}$$

as the coordinate of $(k+1)$-th trial and calculate the value $z^{k+1} = \varphi(x^{k+1})$.

Step 8 Increase the iteration number k by 1 ($k = k + 1$) and go to **Step 1**.

The computational scheme described above generates an infinite sequence of trials. It can be truncated by introducing a termination criterion. For characteristical algorithms as such criterion the inequality

$$x_t - x_{t-1} \le \varepsilon, \tag{29}$$

is used, as a rule, where $\varepsilon > 0$ is a predefined search accuracy, i.e., the search is considered to have been completed when the length of the interval with maximal characteristic is less than accuracy ε.

As AGS-G is a simple modification of AGS it is easily to show that their convergence conditions (see [3]) are the same. In particular, convergence to global minima is provided by fulfillment of the inequality $M_k > 2L$, where L is the Lipschitz constant of the function (22).

In order to describe a parallel version of AGS-G let us assume that at our disposal there are $p > 1$ independent processors, and modify the steps 6 – 8 of the algorithmic scheme presented above. Initial stage of executing trials in end points of intervals (22) can be realized either sequentially or in parallel – no matter.

Step 6 Arrange all characteristics (27) in the decreasing order

$$R(t_1) \geq R(t_2) \geq \ldots \tag{30}$$

and take intervals $(x_{t_{j-1}}, x_{t_j})$, $1 \leq j \leq p$, which the first p characteristics in the series (30) correspond to.

Step 7 Within intervals with maximal characteristics determined at the previous step calculate points

$$x^{k+j} = \frac{x_{t_j} + x_{t_j-1}}{2} - \frac{z_{t_j} - z_{t_j-1}}{2M_k}, \quad 1 \leq j \leq p, \tag{31}$$

as the coordinates of new p trials and calculate in parallel values $z^{k+j} = \varphi(x^{k+j})$, $1 \leq j \leq p$, (each value on a separate processor).

Step 8 Increase the iteration number k by p ($k = k + p$) and go to **Step 1**.

The described method (PAGS-G) belongs to the class of parallel characteristical algorithms the theoretical substantiation of which has been done in [25].

3 Numerical experiments

In order to compare the efficiency of the penalty function method and the method of explicit boundaries in the framework of nested optimization scheme a class of known two-dimensional multiextremal test functions has been considered and on the base of this functions taken as objective ones the problems of constrained optimization have been constructed with constraints forming a feasible domain with computable boundaries, which provide the complicated structure (non-convex and disconnected) of this domain.

For the experiment test functions two-dimensional [10, 25] as objective ones in the problem (1) have been choosen. The domain (2) is the square $0 \leq y_1, y_2 \leq 1$ and 5 non-linear constraints

$$\begin{aligned}
g_1(y_1, y_2) &= 0.5\exp(-y_1) - y_2 - 0.25 \leq 0, \\
g_2(y_1, y_2) &= -4(y_1 - 0.9)^2 + y_2 - 0.8 \leq 0, \\
g_3(y_1, y_2) &= -4(y_2 - 0.6)^2 + y_1 - 0.7 \leq 0, \\
g_4(y_1, y_2) &= -10|y_2 - 0.5y_1 - 0.1| + |\sin(7\pi y_1)| \leq 0, \\
g_5(y_1, y_2) &= -(y_1 - 0.2)^2 - (y_2 - 0.8)^2 + 0.1 \leq 0
\end{aligned}$$

form the feasible domain (3).

Hundred test problems of this type have been minimized by means of the nested optimization scheme with the algorithm AGS-G applied inside for univariate optimization (19) (method MAGS-G). Moreover, the same problems have been solved by the penalty function method (MAGS-P) with different penalty constants C. For illustration of methods behavior Figure 1 shows level curves of a test function, the feasible domain and trial distribution in the region (2) for MAGS-G and MAGS-P (trials are marked with crosses). Figure 1 demonstrates that all the trials of MAGS-G are placed in the feasible domain only while MAGS-P carried out a significant part of trials of the the feasible region.

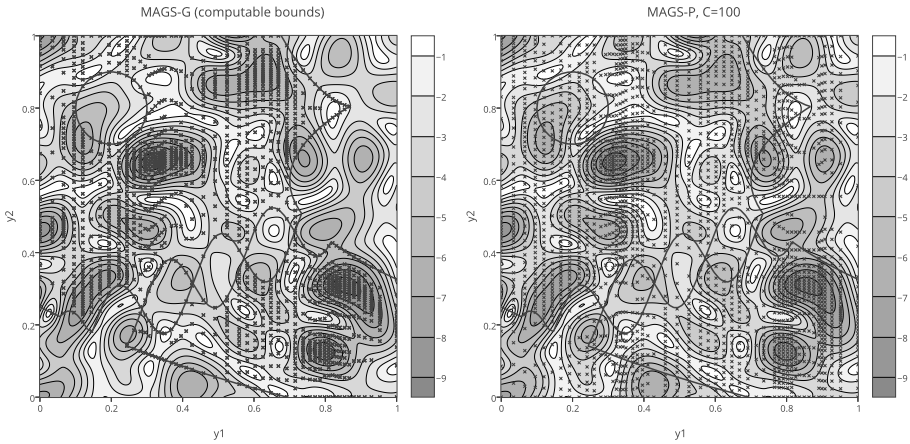

Fig. 1. Trial distributions of MAGS-G and MAGS-P

For conclusive comparison of the optimization algorithms considered above the method of operating characteristics [3, 16] was used. In the framework of this method several test problems are taken and each of them is solved by an optimization algorithm with a given set of its parameters. After the experiment one can determine a number P of problems solved successfully and average number K of trials spent by the algorithm. Repeating such experiment for different sets of parameters we obtain a set of pairs (K, P) called *operating characteristic* of the algorithm. It is convenient to represent operating characteristics of the algorithms compared on the plane with axes K and P. Such representation enables to compare efficiencies of the algorithms in a visual manner. If for the same K the operating characteristic of a method is located above the characteristic of another one, the first method is better as it has solved successfully more problems than its rival spent the same computational resources.

Figure 2 contains the operating characteristics of MAGS-G and MAGS-P with penalty constants $C = 1$, $C = 10$, $C = 100$, obtained after minimization of 100 two-dimensional random functions [10, 25] with $r = 3$ and different values of accuracy ε from (29).

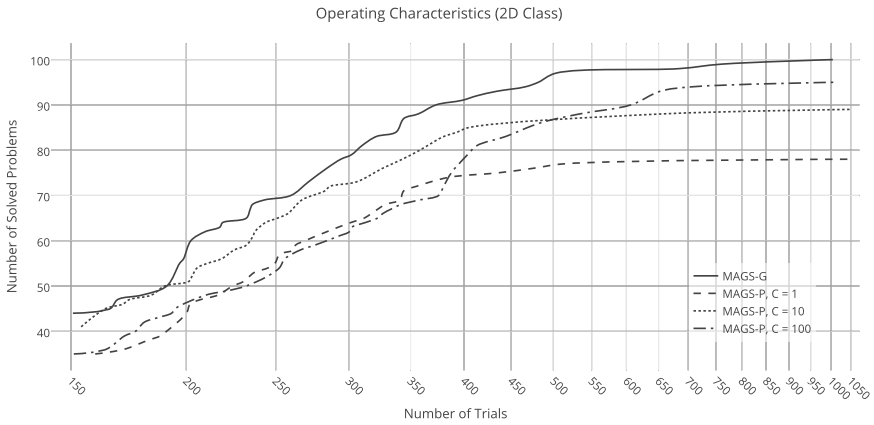

Fig. 2. Operating characteristics of MAGS-G and MAGS-P

The experiment confirms known situation when for small penalty constant MAGS-P are not able to find the global minimum of certain constrained problems. On the other hand, if the penalty constant is overestimated MAGS-P spends too many trials for searching the global minimum. Moreover, the MAGS-P efficiency worsens because of placing trials out of the feasible domain while MAGS-G executes trials at feasible points only.

For evaluation of efficiency of parallelizing the nested optimization scheme let us take again the test class (33) with the same 5 constraints and apply to optimization the nested scheme where for solving problem (16) the sequential method AGS-G is used, but the parallel algorithm PAGS-G with different numbers of processors performs solving the univariate problems for the coordinate y_2. This parallel multidimensional method will be called PMAGS-G. Let $K(1)$ be the average number of trials executed by MAGS-G and $K(p)$ be the average number of trials spent by PMAGS-G with $p > 0$ processors for optimizing 100 test functions. As a criterion of efficiency *the speed-up in trials* is taken. This criterion is defined [21] as $s(p) = K(1)/(pK(p))$ and characterizes the speed-up under assumption that the time spent for realization of algorithmic scheme and data transmission is much less than the time of objective function evaluations. The graph of speed-up in trials is presented in Figure 3.

Experiments were performed using two-socket machine with Intel® Xeon™ E5-2680 v3 processors (Haswell, 2.5GHz, 12 Cores; OS: Red Hat Enterprise Linux Server 6.6; compiler: GCC 4.8.2). Library Intel® Threading Building Blocks 4.3 Update 3 was used for implementing parallel version of the method.

4 Conclusion

A class of multidimensional multiextremal problems with special type of constraints has been considered. Constraints of this type can generate non-convex

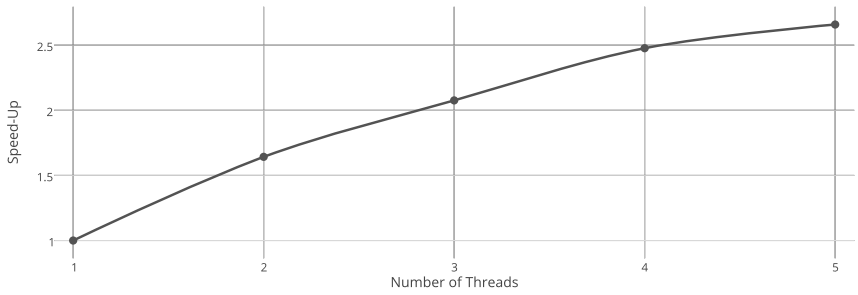

Fig. 3. Speed-up of PMAGS-G Non-linear character of the speed-up is accounted for by the loss of information while planning the new trials in the decision rule (30) – (31) of the parallel methods (detailed explanation of this effect can be found in [25])

and even disconnected feasible domains. For solving the problems under consideration sequential and parallel global optimization methods on the base of nested optimization scheme have been adapted. Decision rules of these methods provide estimating the objective function within the feasible domain only and do not require any parameters for taking into account the constraints as opposed to classical penalty function method. Computational experiments demonstrate advantages of the algorithms proposed in comparison with the penalty function method for constraints of the considered type.

As a further way of the development of researches it would be interesting to investigate different parallelization schemes inside the nested optimization approach in combination with other methods of univariate optimization.

Acknowledgments This research was supported by the Russian Science Foundation, project No. 15-11-30022 "Global optimization, supercomputing computations, and applications".

References

1. Horst, R., Pardalos, P., Thoai, N.: Introduction to Global Optimization. Kluwer Academic Publishers, Dordrecht (1995)
2. Pintr, J.: Global Optimization in Action (Continuous and Lipschitz Optimization: Algorithms, Implementations and Applications). Kluwer Academic Publishers, Dordrecht (1996)
3. Strongin, R., Sergeyev, Ya.: Global Optimization with Non-convex Constraints: Sequential and Parallel Algorithms. Kluwer Academic Publishers, Dordrecht (2000)
4. Eremin, I., Astafyev, N.: Introduction to the theory of linear and convex programming. Nauka, Moscow (1976)
5. Di Pillo, G., Grippo, L.: Exact penalty functions in constrained optimization. SIAM Journal on Control and Optimization 27(6), 1333–1360 (1989)
6. Piyavskij, S.: An Algorithm for Finding the Absolute Extremum of a Function. USSR Comput. Math. Math. Phys. 12(4), 57–67 (1972)

7. Strongin, R.: Numerical Methods in Multiextremal Problems (Information - Statistical Algorithms). Nauka, Moscow (1978)
8. Grishagin, V., Strongin, R.: Optimization of multiextremal functions subject to monotonically unimodal constraints. Engineering Cybernetics 5, 117–122 (1984)
9. Sergeyev, Ya., Grishagin, V.: A parallel algorithm for finding the global minimum of univariate functions. Journal of Optimization Theory and Applications 80(3), 513–536 (1994)
10. Sergeyev, Ya., Grishagin, V.: Sequential and parallel global optimization algorithms. Optimization Methods and Software 3, 111–124 (1994)
11. Sergeyev, Ya.: An information global optimization algorithm with local tuning. SIAM Journal on Optimization 5(4), 858–870 (1995)
12. Gergel, V., Strongin, R.: Parallel computing for globally optimal decision making. Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics) 2763, 76–88 (2003)
13. Shi, L., lafsson, S.: Nested Partitions Method for Global Optimization. Operations Research 48(3), 390–407 (2000)
14. Sergeyev, Ya., Grishagin, V.:, Parallel Asynchronous Global Search and the Nested Optimization Scheme. Journal of Computational Anaysis and Applications 3(2), 123-145 (2001)
15. Barkalov, K., Gergel, V:. Multilevel scheme of dimensionality reduction for parallel global search algorithms. In: Proceedings of the 1-st International Conference on Engineering and Applied Sciences Optimization, Kos Island, 2111–2124 (2014)
16. Gergel, V, Grishagin, V., Israfilov, R.: Local Tuning in Nested Scheme of Global Optimization. Procedia Computer Science 51, 865–874 (2015)
17. Gergel, V.: A software system for multiextremal optimization. European Journal of Operational Research 65(3), 305–313, (1993)
18. Dam, E., Husslage, B., Hertog, D.: One-dimensional Nested Maximin Designs. Journal of Global Optimization 46(2), 287–306 (2010)
19. Gergel, V., Kuzmin, M., Solovyov, N., Grishagin, V.: Recognition of Surface Defects of Cold-Rolling Sheets Based on Method of Localities. International Review of Automatic Control (IREACO). Theory and Applications 8(1), 52–55 (2015)
20. Carr, C., Howe, C.: Quantitative Decision Procedures in Management and Economic: Deterministic Theory and Applications. McGrawHill, New York, (1964)
21. He, J., Verstak, A., Watson, L., Sosonkina, M.: Design and implementation of a massively parallel version of DIRECT. Comput. Optim. Appl. 40(2), 217–245 (2008)
22. Evtushenko, Yu., Malkova, V., Stanevichyus, A.: Parallel global optimization of functions of several variables. Computational Mathematics and Mathematical Physics 49(2), 246–260 (2009)
23. Ferreiro, A.M., Garcia, J.A., Lopez-Salas, J.G., Vazquez, C. An efficient implementation of parallel simulated annealing algorithm in GPUs. Journal of Global Optimization 57(3), 863–890 (2013)
24. Barkalov, K., Polovinkin, A., Meyerov, I., Sidorov, S., Zolotykh, N. SVM Regression Parameters Optimization Using Parallel Global Search Algorithm. Lecture Notes in Computer Science, 7979, 154–166 (2013)
25. Grishagin, V., Sergeyev, Ya., Strongin, R.: Parallel Characteristical Algorithms for Solving Problems of Global Optimization. Journal of Global Optimization 10(2), 185–206 (1997)