Numerische Mathematik manuscript No.
(will be inserted by the editor)

Local tuning and partition strategies for diagonal GO methods

Dmitri E. Kvasov[1], **Clara Pizzuti**[2], **Yaroslav D. Sergeyev**[1,2]

[1] University of Nizhni Novgorod, Gagarin Av., 23, Nizhni Novgorod – Russia
[2] Istituto per la Sistemistica e l' Informatica – C.N.R., c/o D.E.I.S. – Università della Calabria, 87036 Rende (CS) – Italy, e-mail: {pizzuti,yaro}@si.deis.unical.it

Summary In this paper, global optimization (GO) Lipschitz problems are considered where the multi-dimensional multiextremal objective function is determined over a hyperinterval. An efficient one-dimensional GO method using local tuning on the behaviour of the objective function is generalized to the multi-dimensional case by the diagonal approach using two partition strategies. Global convergence conditions are established for the obtained diagonal geometric methods. Results of a wide numerical comparison show a strong acceleration reached by the new methods working with estimates of the local Lipschitz constants over different subregions of the search domain in comparison with the traditional approach.

Key words Global optimization – diagonal approach – local tuning – partition strategies.

1 Introduction

In [13,14,16] diagonal global optimization algorithms have been introduced for solving multi-dimensional Lipschitz global optimization (GO) problems with box constraints. In its general form such a problem can be stated as minimization of a multiextremal function satisfying the Lipschitz condition with a constant $0 < L < \infty$ over a hyperinterval, i.e., finding the value f^* and points x^* such that

$$f^* = f(x^*) = \min_{x \in D} f(x), \tag{1}$$

Correspondence to: Yaroslav D. Sergeyev where

$$|f(x') - f(x'')| \leq L \parallel x' - x'' \parallel, \qquad x', x'' \in D \subset \mathbb{R}^n, \tag{2}$$

$$D = [a, b\,] = \{x \in \mathbb{R}^n \,:\, a \leq x \leq b\}, \qquad a \leq b, \;\; a, b \in \mathbb{R}^n. \tag{3}$$

Such problems very often can be faced in real-life applications (for example, in data classification, nonlinear approximation, globally optimized calibration of complex system models etc.). A number of such problems solved by the diagonal methods can be found in [16].

The diagonal approach is a simple and powerful tool for extending one-dimensional global optimization methods to the multi-dimensional case. The main idea is to describe the behaviour of the objective function $f(x)$ over a hyperinterval (we shall also use the term *cell* or simply *interval*) $D_i = [a_i, b_i]$ by information obtained from evaluating $f(x)$ at the vertices a_i, b_i being the ends of the main diagonal defining the interval D_i. During every $(l+1)$-th iteration to each subinterval $D_i \subset D$ generated in the course of the previous l iterations a *characteristic* $R_i = R(a_i, b_i, f(a_i), f(b_i))$ is associated in such a way that R_i tends to be higher if D_i contains lower values of $f(x)$. Then, among all subintervals created so far within D, an interval D_t with the maximal characteristic is chosen for further subdivision. It is subdivided in p subcells and $f(x)$ is evaluated at the vertices a_j, b_j of all the intervals $D_j, 1 \leq j \leq p$. The process is repeated until satisfaction of a stopping rule.

The diagonal method proposed in [13,14,16] and extending the univariate algorithm from [12] uses a global estimate of the Lipschitz constant L in its work. GO algorithms using in their work the global Lipschitz constant L (or its estimates) do not take into account local information about behaviour of the objective function over every small subregion of D. In fact, it is supposed in such algorithms (see [7]) that $f(x)$ has the same constant L over every subdomain of D without paying any attention to situations where $f(x)$ has a very low local Lipschitz constant over the subdomain under consideration. It has been shown for a number of global optimization algorithms (see [18,19,23]) that using local information for estimating local Lipschitz constants can accelerate the global search significantly. Importance of such information in the diagonal approach context has been highlighted in [16]. Of course, the local data must be in an appropriate way balanced with the global information about the objective function otherwise the global solution can be lost [22].

In this paper a new diagonal algorithm generalizing an efficient deterministic one-dimensional GO method using local tuning on the behaviour of the objective function (see [19]) is extended to the multi-dimensional case by the diagonal approach using two partition strategies widely used in literature [3,9,13,14,16]:

- Bisection, where $p = 2$ and the interval D_t is subdivided in two subintervals by a hyperplane orthogonal to the longest edge of D_t;
- Partition 2^n, where $p = 2^n$ and D_t is partitioned into 2^n new subintervals generated by the intersection of the boundary of D_t and the hyperplanes that contain a point x^{l+1} belonging to the main diagonal of D_t and are parallel to the boundary hypersurfaces of D_t.

The new method uses a local information about the objective function over the *whole* search region D *during* the global search in contrast with techniques which do it only in a *neighborhood* of local minima *after* stopping their global procedures (see e.g. [7]). Global convergence conditions are established for the new method. Results of a wide numerical comparison show a strong acceleration reached by the new method working with estimates of the local Lipschitz constants over different subregions of the search domain in comparison with the traditional approach using global estimates of L.

2 The new algorithm with local tuning

In this section the New Diagonal Algorithm with Local tuning (NDAL) is described.

The method starts by setting the number of iterations, l, and the number of generated intervals, $m = m(l)$, equal to 1. The first two *trials* (evaluations of the objective function) are executed at the points $x_0 = a$, $x_1 = b$ from (3). The results of trials are indicated as $z_0 = f(x_0)$, $z_1 = f(x_1)$, and the initial number $k = k(l)$ of trial points generated by the algorithm is taken equal to 2. The initial estimate of the global optimum is taken as $z_1^* = \min\{z_0, z_1\}$. The estimate λ_1 of the local Lipschitz constant over the initial interval $D_1 = D = [a, b]$ (in this case, of course, the local estimate coincides with the global one) is calculated as follows

$$\lambda_1 = \frac{\mid f(a) - f(b) \mid}{\parallel a - b \parallel}.$$

Suppose now that $l \geq 1$ iterations of the method have already been executed. The iteration $l + 1$ consists of the following steps.

Step 1. For each interval $D_i = [a_i, b_i]$, $1 \leq i \leq m(l)$, calculate its characteristic

$$R_i = 0.5(K_i \parallel a_i - b_i \parallel - f(a_i) - f(b_i)) \qquad (4)$$

where

$$K_i = K_i(l) = (r + \frac{C}{l}) \max\{\lambda_i, \gamma_i, \xi\}, \qquad (5)$$

the values $r > 1$, $\xi > 0$, and $C > 0$ are parameters of the method, λ_i is the estimate of the local Lipschitz constant over the interval D_i calculated at the moment of creation of D_i, and

$$\gamma_i = \mu \frac{\| a_i - b_i \|}{d^{\max}}. \tag{6}$$

The values μ and d^{max} are evaluated as follows

$$\mu = \max_{1 \le i \le m(l)} \lambda_i, \tag{7}$$

$$d^{\max} = \max_{1 \le i \le m(l)} \| a_i - b_i \| . \tag{8}$$

Step 2. Among all the intervals D_i choose an interval D_t such that

$$R_t = \max_{1 \le i \le m(l)} R_i. \tag{9}$$

Step 3. If

$$\| a_t - b_t \| > \varepsilon \| a - b \|,$$

where a and b are from (3) and t is from (9), then go to Step 4, otherwise take the value

$$z_l^* = \min_{1 \le i \le k(l)} f(x_i)$$

(where x_i, $1 \le i \le k(l)$, are the trial points generated by the algorithm in the course of the previous l iterations) as an estimate of the global optimum of the problem (1) – (3) and **Stop**.

Step 4. Choose the new point x^{l+1} belonging to the main diagonal (the diagonal joining the vertices a_t and b_t) of the subinterval D_t, where t is from (9), as follows (see [13,14,16]):

$$x^{l+1} = \frac{a_t + b_t}{2} - \frac{f(b_t) - f(a_t)}{2\hat{K}} \times \frac{b_t - a_t}{\| a_t - b_t \|}. \tag{10}$$

Here

$$\hat{K} = \hat{K}(l) = (4 + \frac{C}{l}) \max\{\mu, \xi\}, \tag{11}$$

where ξ is from (5) and μ is from (7).

Step 5. Subdivide the interval D_t into p new subintervals by Bisection strategy or by Partition 2^n.

Step 6. Denote by x_i, $i = 1, \ldots, s$, the vertices of the new p subintervals generated during Step 5 where $f(x)$ must be evaluated.

–In the case of Bisection strategy it is necessary to evaluate $f(x)$ at two vertices, $s = 2$ (the points a_t and b_t come from the subdivided interval D_t and $f(x)$ has already been evaluated at its vertices during the previous iterations).

–In the case of Partition 2^n, the number $s = 2 \times 2^n - 3$ because the new 2^n subintervals are identified by their two vertices, x^{l+1} is common to two intervals, and $f(a_t)$ and $f(b_t)$ of the subdivided interval D_t have already been evaluated.

Step 7. For all the new intervals D_i, $1 \leq i \leq p$, get an estimate of the local Lipschitz constant as

$$\lambda_i = \max\{\frac{\mid f(a_t) - f(b_t) \mid}{\parallel a_t - b_t \parallel}, \max_{1\leq j \leq p} \frac{\mid f(a_j) - f(b_j) \mid}{\parallel a_j - b_j \parallel}\}. \qquad (12)$$

Set $l := l + 1$, $m := m + p - 1$, $k := k + s$, and go to Step 1.

Let us give a few comments on the introduced method. The key idea of the algorithm is estimating local Lipschitz constants by balancing local and global data. In contrast with the traditional approach (see [13,14]) where the global estimate $\hat{K}$ of the Lipschitz constant L from (2) is used in the form (11), the local estimate K_i from (5) is the result of the balance between the local and the global information represented by the values λ_i and γ_i, respectively. When the subinterval D_i has a small main diagonal (in comparison with the current maximal diagonal $d^{\max}$ over all subintervals in D) then (see (6)–(8)), γ_i is small too and the local information represented by λ_i has a decisive influence (see (5)) on K_i. When the interval D_i is very wide (its diagonal $\parallel a_i - b_i \parallel$ is close to $d^{\max}$), the local information is not reliable and the global information (see (6)) represented by γ_i is used.

The values r, C, and ξ influence K_i as global parameters. By increasing r and C we augment reliability of the method over the whole region D. The parameter $\xi > 0$ is a small number allowing the NDAL to work also when $f(x_i) = const$ for all trial points x_i. The importance of the parameter ξ for the correct work of the method can be seen from (4) – (5) and (10) – (11). If $\gamma_i < \xi$ and $\lambda_i < \xi$ it follows

$$K_i(l) = \hat{K}(l) = (r + \frac{C}{l})\xi.$$

Of course, this case is degenerate for the method.

The introduced algorithm belongs to the class of diagonally extended geometric algorithms and also to more general classes of *adaptive partition* and *divide the best* algorithms (see [15,16] and [20], respectively). Let us study the convergence properties of the infinite ($\varepsilon = 0$ in the stopping rule) sequence $\{y^k\}$ of trial points generated by the NDAL during minimization of the function $f(x)$ from (1)–(3). Hereinafter we shall designate by Y' the set of limit points of the sequence $\{y^k\}$.

Theorem 1 *Let y' be a limit point of the sequence $\{y^k\}$ then, for all trial points $y^k \in \{y^k\}$, it follows $f(y^k) \geq f(y')$. If there exists another limit point $y'' \in Y'$ then $f(y') = f(y'')$.*

Proof. This result can be obtained as a particular case of the general convergence study from [20] and its proof is so omitted. □

The next theorem presents sufficient global convergence conditions for the NDAL.

Theorem 2 *Let there exist an iteration number l^* such that for a cell D_j, $j = j(l)$, containing a global minimizer x^* of $f(x)$ during the l-th iteration of the NDAL the following inequality takes place*

$$K_j(l) \geq 2H_j, \qquad l > l^*, \tag{13}$$

where

$$H_j = \max\{\frac{f(a_j) - f(x^*)}{\| x^* - a_j \|}, \frac{f(b_j) - f(x^*)}{\| b_j - x^* \|}\}. \tag{14}$$

Then, x^* is a limit point of the trial sequence $\{y^k\}$ generated by the NDAL.

Proof. We start the proof by showing that the estimates $K_i(l)$ of the local Lipschitz constants L_i from (5) are bounded values. In fact, since the global Lipschitz constant $L < \infty$ and the constants $r > 1$, $C > 0$, and $\xi > 0$, it follows

$$0 < r\xi < K_i(l) \leq (r + C)\max\{L, \xi\} < \infty, \qquad l \geq 1. \tag{15}$$

Suppose, that there exists a limit point $y' \neq x^*$ of the trial sequence $\{y^k\}$. Taking into consideration (4), (10), (11), and (15) we can conclude for an interval D_i, $i = i(l)$, containing y' during the l-th iteration of the NDAL, that

$$\lim_{l \to \infty} R_i(l) = -f(y'). \tag{16}$$

Consider now the cell D_j, $j = j(l)$, such that the global minimizer $x^* \in D_j$ and suppose that x^* is not a limit point of $\{y^k\}$. This signifies that there exists an iteration number q such that for all $l \geq q$

$$x^{l+1} \notin D_j, \quad j = j(l).$$

Estimate now the characteristic $R_j(l)$, $l \geq q$, of the interval D_j. It follows from (14) and the fact of $x^* \in D_j$ that

$$f(a_j) - f(x^*) \leq H_j \| a_j - x^* \| \leq H_j \| a_j - b_j \|,$$

$$f(b_j) - f(x^*) \leq H_j \| b_j - x^* \| \leq H_j \| a_j - b_j \| .$$

Then, by summarizing these inequalities we obtain

$$f(a_j) + f(b_j) \leq 2f(x^*) + 2H_j \parallel a_j - b_j \parallel .$$

From this inequality and (13), (14) we can deduce for all iteration numbers $l > l^*$ that

$$\begin{gathered} R_j(l) = 0.5(K_j \parallel a_j - b_j \parallel - f(a_j) - f(b_j)) \geq \\ 0.5(K_j \parallel a_j - b_j \parallel - 2f(x^*) - 2H_j \parallel a_j - b_j \parallel) = \\ 0.5 \parallel a_j - b_j \parallel (K_j - 2H_j) - f(x^*) \geq -f(x^*). \end{gathered} \tag{17}$$

Since x^* is a global minimizer, it follows from (16) and (17) that an iteration number $q^* > \max\{l^*, q\}$ will exist such that

$$R_j(q^*) \geq R_i(q^*).$$

But this means that during the q^*-th iteration, trials will be executed at the cell D_j. Thus, our assumption that x^* is not a limit point of $\{y^k\}$ is not true and theorem has been proved. □

Let us denote the set of global minimizers of the problem (1)–(3) as X^*. Then the following corollary ensures the inclusion $Y' \subseteq X^*$.

Corollary 1 *Given the conditions of Theorem 2, all limit points of the sequence $\{y^k\}$ are global minimizers of $f(x)$, $Y' \subseteq X^*$.*

Proof. The corollary follows immediately from Theorems 1 and 2. □

The sets Y' and X^* coincide if conditions established by Corollary 2 are fulfilled.

Corollary 2 *If condition (13) is fulfilled for all points $x^* \in X^*$, then the set of limit points of $\{y^k\}$ coincides with the set of global minimizers of the objective function $f(x)$, i.e. $Y' = X^*$.*

Proof. Again, the corollary is a straightforward consequence of Theorems 1 and 2. □

3 Numerical comparison

The goal of this section is dual: first, to show advantages of the local tuning in comparison to the traditional approach using global estimates of the Lipschitz constant; second, to establish which of two partitioning strategies, Bisection or Partition 2^n, works better.

Table 1. Test problems

N^o	Formula	Domain	Source
1	$0.25x_1^4 - 0.5x_1^2 + 0.1x_1 + 0.5x_2^2$	$[-10,10]^2$	[8]
2	$(4-2.1x_1^2+x_1^4/3)x_1^2 + x_1x_2 + (-4+4x_2^2)x_2^2$	$[-2.5,2.5]\times$ $[-1.5,1.5]$	[24]
3	$2x_1^2 - 1.05x_1^4 + x_1^6/6 + x_1x_2 + x_2^2$	$[-5,5]^2$	[2]
4	$(x_2 - 5.1x_1^2/(4\pi^2) + 5x_1/\pi - 6)^2 + 10(1-1/(8\pi))\cos x_1 + 10$	$[-5,10]\times$ $[0,15]$	[1]
5	$(1-2x_2+0.05\sin(4\pi x_2) - x_1)^2 + (x_2 - 0.5\sin(2\pi x_1))^2$	$[-10,10]^2$	[2]
6	$1+(x_1+x_2+1)^2(19-14x_1+3x_1^2-14x_2+6x_1x_2+3x_2^2)]\times$ $[30+(2x_1-3x_2)^2(18-32x_1+12x_1^2+48x_2-36x_1x_2+27x_2^2)]$	$[-2,2]^2$	[4]
7	$\sum_{i=1}^5 i\cos((i+1)x_1+i)\sum_{j=1}^5 j\cos((j+1)x_2+j)$	$[-10,10]^2$	[8]
8	$\sum_{i=1}^5 i\cos((i+1)x_1+i)\sum_{j=1}^5 j\cos((j+1)x_2+j)+$ $(x_1+1.42513)^2+(x_2+0.80032)^2$	$[-10,10]^2$	[8]
9	$100(x_2-x_1^2)^2+(x_1-1)^2$	$[-2,8]^2$	[2]
10	$(x_1^2+x_2-11)^2+(x_1+x_2^2-7)^2$	$[-6,6]^2$	[6]
11	$-4x_1x_2\sin(4\pi x_2)$	$[\,0,1]^2$	[10]
12	$-\sin(2x_1+1)-2\sin(3x_2+2)$	$[\,0,1]^2$	[10]
13	$(x_1-2)^2+(x_2-1)^2-0.04/(0.25x_1^2+x_2^2-1)+5(x_1-2x_2+1)^2$	$[\,1,2]^2$	[17]
14	$-\lvert\sin(x_1)\sin(2x_2)\rvert+0.01(x_1x_2+(x_1-\pi)^2+3(x_2-\pi)^2)$	$[\,0,2\pi]^2$	[21]
15	$(\pi/n)\{10\sin^2(\pi y_1)+\sum_{i=1}^{n-1}[(y_i-1)^2(1+10\sin^2(\pi y_{i+1}))]+$ $(y_n-1)^2\},\ where\ y_i = 1+(1/4)(x_i-1), \quad i=1,\dots,n$	$[-10,10]^n$	[8]
16	$0.1\{\sin^2(3\pi x_1)+\sum_{i=1}^{n-1}[(x_i-1)^2(1+\sin^2(3\pi x_{i+1})]\}+$ $0.1(x_n-1)^2[1+\sin^2(2\pi x_n)\ ]$	$[-10,10]^n$	[8]
17	$-\sum_{i=1}^4 c_i\exp(-\sum_{j=1}^3 \alpha_{ij}(x_j-p_{ij})^2)$	$[\,0,1]^3$	[5]
18	$100[x_3-0.25(x_1+x_2)^2]^2+(1-x_1)^2+(1-x_2)^2$	$[\,0,1]^3$	[17]
19	$(x_1^2-2x_2^2+x_3^2)\sin(x_1)\sin(x_2)\sin(x_3)$	$[-1,1]^3$	[11]
20	$\sum_{i=1}^3[(x_1-x_i^2)^2+(x_i-1)^2]$	$[-10,10]^3$	[25]

Thus, four methods are compared:

– the traditional method with Partition 2^n and the global estimate;
– the traditional method with Bisection and the global estimate;
– the new algorithm using local tuning and Partition 2^n;
– the new algorithm using local tuning and Bisection.

The list of problems used in the experiments is shown in Table 1, where the following quantities are specified:

N^o : problem number;
$Formula$: formula of the test function;
$Domain$: feasible region of the test function;
$Source$: bibliographic reference.

Problems 1–14 are two-dimensional, problems 17–20 are three-dimensional, and problems 15–16 are of arbitrary dimension $n > 1$ ($n = 2$ and $n = 3$ have been used).

Table 2. Results of numerical experiments with two-dimensional functions for $r = 1.1$

Problem	Global Estimate		Local Tuning	
Number	Partition 2^n	Bisection	Partition 2^n	Bisection
1	12412	8950	4742	3508
2	8037	2670	2947	1354
3	19427	20392	14832	14244
4	4687	2762	1332	998
5	4187	2818	807	602
6	20522	17732	14572	10924
7	6837	4766	5532	3936
8	4057	3922	2822	3372
9	16187	16446	10307	7328
10	6267	4384	1797	1286
11	312	256	272	146
12	292	200	167	96
13	1827	2002	282	238
14	1127	96^*	592	186
15	4857	2736	2237	1336
16	1627	532	492	118
Average	7041.36	5666.50	3983.25	3104.50

To show the influence of the parameter r on the search characteristics, the experiments for the two-dimensional case have been realized for two different values of the parameter r in all the methods: $r = 1.1$ and $r = 1.3$. The value $C = 10$ was taken in all the two-dimensional experiments. We have executed these experiments with the accuracy $\varepsilon = 0.01$ in the stopping rule.

The numbers of function evaluations executed by the methods before satisfaction of the stopping rule for the two-dimensional case are reported in Tables 2 and 3. Global optima have been located in all the experiments. For Problem 14 and the method with the global estimate of the Lipschitz constant and Bisection strategy the value $r = 1.1$ was too small: the method has not located the global minimizer in this case. The sufficient value of the reliability parameter r for finding the global minimizer for Problem 14 is $r = 1.3$.

In Table 4 the experimental results for three-dimensional test functions are shown. The following parameters have been chosen in all the experiments: $r = 1.2$, $C = 100$. The search accuracy $\varepsilon = 0.02$ has been used.

Performance of all the methods during solving Problem 10 is illustrated in Figs. 1 – 4. Trials points are shown by the black dots.

The new algorithm was faster than the method using the global estimate for both strategies in all the cases. The smaller values of the accuracy ε ensure higher values of acceleration. For example, Table 5 shows

Table 3. Results of numerical experiments with two-dimensional functions for $r = 1.3$

Problem	Global Estimate		Local Tuning	
Number	Partition 2^n	Bisection	Partition 2^n	Bisection
1	13987	9874	7012	5620
2	9862	4774	3357	2072
3	20057	21608	16802	16754
4	5812	3728	2332	1190
5	4817	3180	1402	650
6	21922	22424	17812	12622
7	7267	7374	6422	5128
8	5467	4504	3717	3938
9	16752	17378	10852	8250
10	8852	6820	3432	1858
11	417	324	362	174
12	347	232	177	114
13	2102	2306	307	284
14	1297	800	747	360
15	7167	3880	3137	1740
16	1852	778	612	162
Average	7998.56	6874.00	4905.13	3807.25

Table 4. Results of numerical experiments with three-dimensional functions for $r = 1.2$

Problem	Global Estimate		Local Tuning	
Number	Partition 2^n	Bisection	Partition 2^n	Bisection
15	173513	43780	98412	12060
16	26938	3732	12625	1032
17	6879	1810	4825	1020
18	83475	27760	15862	3470
19	8556	2040	7568	1358
20	122436	74254	59646	21756
Average	70299.50	25562.67	33156.33	6782.67

that the NDAL works better when accuracy increases and the improvement is stronger for higher values of the parameter r.

It can be seen from the numerical experiments that the new method with local tuning significantly outperforms the traditional approach. In its turn, Bisection works better then Partition 2^n strategy. The best combination is the new algorithm with local tuning working with Bisection strategy.

Higher values of the parameter r increase the reliability of the methods and lead to a fast growth of the iterations number. This happens because by increasing r we uniformly augment the estimates of the Lipschitz constants (both global and local ones). The obtained improvement increases for higher values of the parameter r.

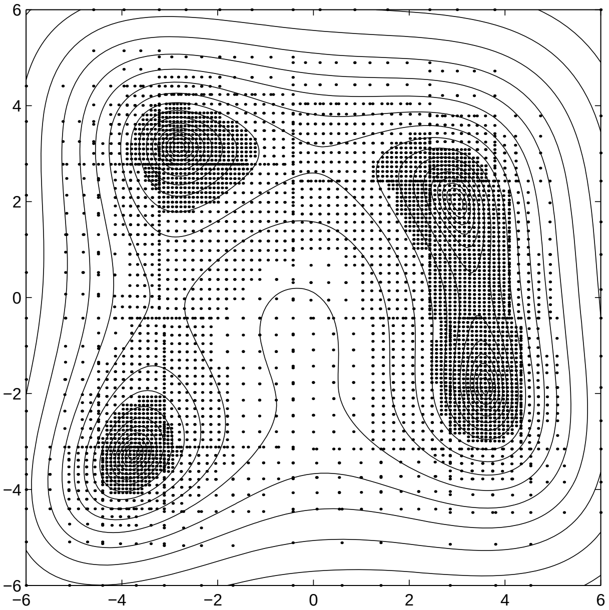

Fig. 1. Level curves of Problem 10 with the trial points generated by strategy Partition 2^n and method with global estimate of Lipschitz constant with $r = 1.1$, the number of trials $= 6267$

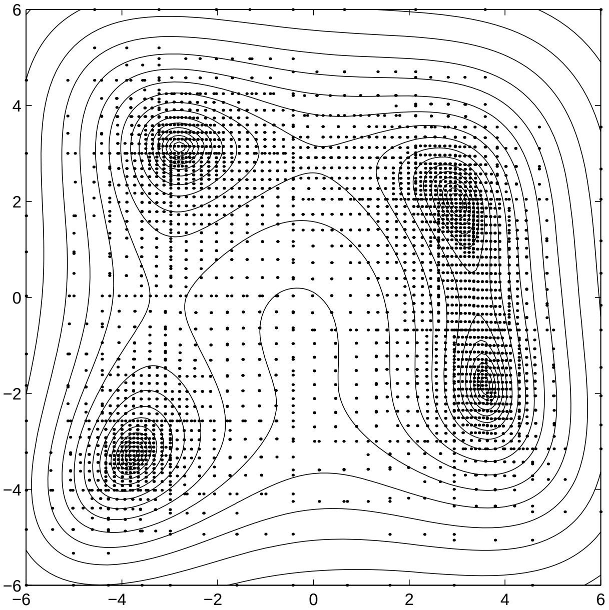

Fig. 2. Level curves of Problem 10 with the trial points generated by strategy Bisection and method with global estimate of Lipschitz constant with $r = 1.1$, the number of trials $= 4384$

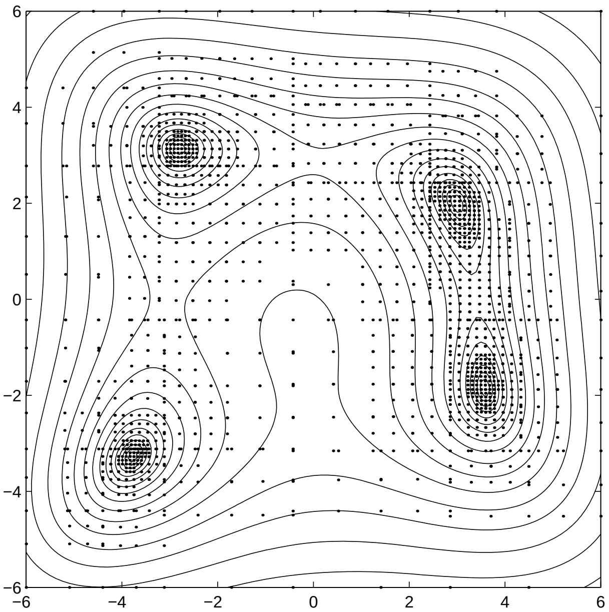

Fig. 3. Level curves of Problem 10 with the trial points generated by strategy Partition 2^n and method with local tuning with $r = 1.1$, the number of trials $= 1797$

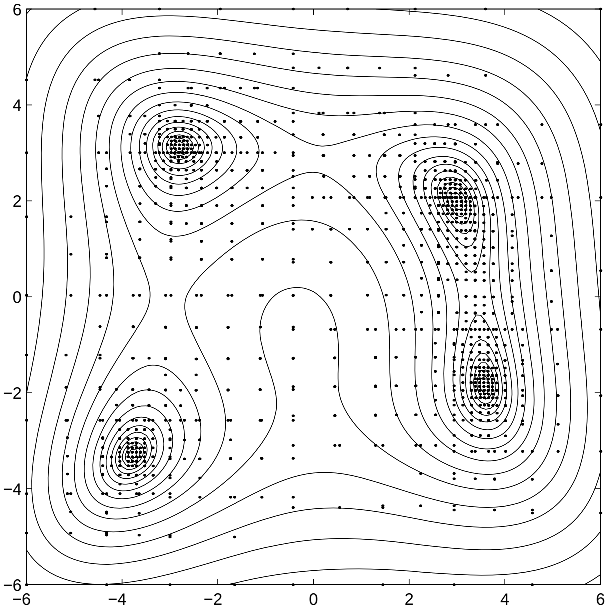

Fig. 4. Level curves of Problem 10 with the trial points generated by strategy Bisection and method with local tuning with $r = 1.1$, the number of trials $= 1286$

Table 5. Number of trials for Problem 7 in dependence on the parameter r and accuracy ε

r	ε	Global Estimate		Local Tuning	
		Partition 2^n	Bisection	Partition 2^n	Bisection
1.1	0.0100	6837	4766	5532	3936
	0.0010	10742	11664	7012	4662
	0.0001	35697	32218	7367	4694
1.3	0.0100	7267	7374	6422	5128
	0.0010	23712	17322	8962	8270
	0.0001	54397	42584	11862	8582

If in the search region there exists a neighborhood of the global solution having local Lipschitz constants smaller than the global one (this is true, for example, for differentiable functions having the global solution in an interior point of the search domain), then smaller values of the accuracy ε ensure higher values of acceleration.

References

1. Branin, F.H., Widely convergent method for finding multiple solutions of simultaneous nonlinear equations, *IBM J. Res. Dev.* **16(5)**, (Sept. 1972) 504–522.
2. Dixon, L.C.W., and Szegö, G.P., (eds.), *Towards Global Optimization* (North-Holland Publishing, Amsterdam, 1975).
3. Gergel, V.P., A global optimization algorithm for multivariate functions with Lipschitzian first derivatives, *J. Glob. Optim.* **10(3)**, (1997) 257-281.
4. Goldstein, A.A., and Price, J.F., On descent from local minima, *Math. Comput.* **25(115)**, (1971) 569–574.
5. Hartman, J.K., Some experiments in global optimization, *Nav. Res. Logist.* **20**, (1973) 569–576.
6. Himmelblau, D., *Applied Nonlinear Programming* (McGraw-Hill, New-York, 1972).
7. Horst, R., and Pardalos, P.M., (eds.), *Handbook of Global Optimization* (Kluwer Academic Publishers, Dordrecht, 1995).
8. Lucidi, S., and Piccioni, M., Random tunneling by means of acceptance-rejection sampling for global optimization, *J. Optimization Theory Appl.* **62(2)**, (1989) 255–277.
9. Meewella, C.C., and Mayne, D.Q., Efficient domain partitioning algorithms for global optimization of rational and Lipschitz continuous functions, *J. Optimization Theory Appl.* **61 (2)**, (1989) 247–270.
10. Mladineo, R.H., An algorithm for finding the global maximum of a multimodal, multivariate function, *Math. Program.* **34**, (1986) 188–200.
11. Mladineo, R.H., Stochastic minimization of Lipschitz functions, in *Recent Advances in Global Optimization*, C.A. Floudas and P.M. Pardalos, (eds.), (Princeton University Press, 1992), 369–383.
12. Pijavskii, S.A., An algorithm for finding the absolute extremum of a function, *Comput. Math. Math. Phys.* **12**, (1972) 57–67.
13. Pintér, J., A unified approach to globally convergent one-dimensional optimization algorithms, Research Report 83-5, IAMI CNR, Milano (1983).

14. Pintér, J., Extended univariate algorithms for n-dimensional global optimization, *Computing* **36**, (1986) 91-103.
15. Pintér, J., Convergence qualification of adaptive partition algorithms in global optimization, *Math. Program.* **56**, (1992) 343–360.
16. Pintér, J., *Global Optimization in Action* (Kluwer Academic Publishers, Dordrecht, 1996).
17. Schittkowski, K., *More Test Examples for Nonlinear Programming Codes*, in Lecture Notes in Economics and Mathematical Systems 282 (Springer-Verlag, Berlin, 1987).
18. Sergeyev, Ya.D., An information global optimization algorithm with local tuning, *SIAM J. Optim.* **5(4)**, (1995) 858–870.
19. Sergeyev, Ya.D., A one-dimensional deterministic global minimization algorithm, *Comput. Math. Math. Phys.* **35(5)**, (1995) 553–562.
20. Sergeyev, Ya.D., On convergence of “Divide the Best” global optimization algorithms, *Optimization* **44(3)**, (1999) 303–325.
21. Sergeyev, Ya.D., Pugliese, P., and Famularo, D., Index information algorithm with local tuning for solving multidimensional global optimization problems with multiextremal constraints, (2001) *Submitted*.
22. Stephens, C.P., and Baritompa, W.P., Global optimization requires global information, *J. Optimization Theory Appl.* **96(3)**, (1998) 575–588.
23. Strongin, R.G., and Sergeyev, Ya.D., *Global Optimization with Non-Convex Constraints: Sequential and Parallel Algorithms* (Kluwer Academic Publishers, Dordrecht, 2000).
24. Törn, A., and Žilinskas, A., *Global Optimization*, in Lecture Notes in Computer Science 350 (Springer-Verlag, Berlin, 1989).
25. Walster, G., Hansen, E., and Sengupta, S., Test results for global optimization algorithm, in *Numerical Optimization 1984 SIAM*, P.T.Boggs, R.H.Byrd, and R.B.Schnabel, (eds.) (SIAM, Philadelphia, 1985), 272–287.