Ya. D. Sergeyev, Multidimensional global optimization using the first derivatives, 

Use of the all-Russian mathematical portal Math-Net.Ru implies that you have read and agreed to these terms of use
http://www.mathnet.ru/eng/agreement

Download details:
IP: 54.70.40.11
February 3, 2020, 11:02:08
MULTIDIMENSIONAL GLOBAL OPTIMIZATION USING THE FIRST DERIVATIVES

© 1999 г. Ya. D. Sergeyev

ISI CNR c/o DEIS, Univ. Calabria, Arcavacata di Rende, 87030 Cosenza, Italy and Software Department, Nizhni Novgorod State Univ., pr. Gagarina, 23; Nizhni Novgorod, 603600 Russia

Received May 21, 1997; in final form, September 7, 1998

We propose a new multidimensional algorithm for solving global optimization problems with the objective function having Lipschitzian first derivatives and determined over a multidimensional interval. The method does not belong to the class of multistart algorithms. It is based on the following three new proposals and is an illustration how it is possible to generalize to the multidimensional case the one-dimensional algorithms belonging to the classes of adaptive partition and characteristic global optimization methods. The first proposal is to estimate the local Lipschitz constants for derivatives in different subintervals of the search region during the course of optimization to provide a local tuning on the behavior of the objective function. The second one is a new partitioning scheme providing an efficient keeping of the search information. The last proposal is a way to calculate characteristics of multidimensional intervals to provide convergence to the global minimizers.

1. INTRODUCTION

In this paper the global optimization problem of minimizing a multidimensional multiextremal function $f(x)$ having a finite number of local extrema over a hyperinterval $D \subset \mathbb{R}^N$ is considered

$$ x^* = \text{argmin}\{f(x): x \in D\}, \quad D = \{x: a_i \leq x_i \leq b_i, 1 \leq i \leq N\}, $$

(1.1)

Many methods have been proposed for solving this problem under different assumptions about $f(x)$ (see, for example, [1-26]).

In this paper we assume that the objective function is such that the following condition holds:

$$ |f'(x_1) - f'(x_2)| \leq L|x_1 - x_2| \quad \forall x_1, x_2 \in D, $$

(1.2)

where $f'(x_1), f'(x_2)$ are the directional derivatives calculated at the points $x_1, x_2$ in the direction $x_2 - x_1, ||\cdot||$ is Euclidean norm and $0 < L < \infty$ is called the Lipschitz constant.

For solving the problem (1.1) under assumption (1.2) together with well known multistart algorithms (see e.g., [7, 8]) some other methods (which can be described in a general framework of “divide the best” global optimization algorithms (see [15]) have been proposed. This class includes as subclasses adaptive partition algorithms (see [27]) and characteristic algorithms (see [28, 29]).

Thus, [30] (see also modifications proposed in [31]) present an algorithm for (1.1) with $N \geq 1$ where the constant $L$ from (1.2) is a priori known. Usually in real optimization problems it is very difficult to know the precise value of $L$, that is why $L$ is substituted by an a priori fixed estimate. In [32] another approach estimating $L$ in the course of optimization (note, that he considers a one-dimensional problem (1.1)) is proposed. There exists the third way when you do not use global constant at all but estimate the local Lipschitz constants for subintervals of the search region. In [16-18, 22] this approach has been introduced for solving several differently stated global optimization problems with the Lipschitzian objective function.

On the one hand the Breiman-Cutler scheme [30] can not be applied to extend to the multidimensional case methods which estimate $L$ in the course of the search because the domain partitioning constructed by this one uses a unique constant $L$ during the whole process of optimization. Since changing $L$ you must recalculate the domain partitioning (and this operation is very expensive numerically) the multidimensional methods using this scheme and adaptively estimating $L$ (or local Lipschitz constants $L_i$) will be very slow.

On the other hand it has been shown (see [33-35]) that simple partitioning schemes applying for generalizing one-dimensional characteristic algorithms to the multidimensional case consume too much memory already in the case when the methods extended do not use derivatives.

In this paper a new scheme permitting extending to the multidimensional case “divide the best” one-dimensional algorithms using derivatives is proposed. The paper is structured as follows. In the Sect. 2 we describe how characteristics for one-dimensional intervals (see [27-29]) can be generalized to the case...
when the region to assign a characteristic is a hyperinterval and the first derivatives can be used. The Sect. 3 presents a new scheme for splitting the hyperinterval $D$ from (1,1) into subintervals. The Sect. 4 describes how the local Lipschitz constants $L_i$ for subintervals are estimated. The Sect. 5 introduces the multidimensional global optimization algorithm. Sufficient conditions of global convergence are established for the method in Sect. 6. Sect. 7 concludes the paper.

2. CHARACTERISTICS FOR HYPERINTERVALS

In [33] a scheme extending to the multidimensional case one-dimensional adaptive partition global optimization algorithms not using derivatives is proposed. In this paper we introduce a scheme permitting to extend the notion of characteristic to the multidimensional case for “divide the best” algorithms including the class of adaptive partition methods and allowing also to construct characteristics using the first derivatives.

We associate to a hyperinterval $D(j)$ a characteristic $R(j)$ such that

$$R(j) = R(a_j, b_j, f(a_j), f(b_j), f'(a_j), f'(b_j)),$$  

(2.1)

where $a_j$ and $b_j$ are points determining the main diagonal of the hyperinterval $D(j)$ (and, therefore, the interval $D(j)$ also). The values $f'(a_j)$ and $f'(b_j)$ are determined as follows. We will denote as $f'(a_j)$ the derivative of $f(x)$ at the point $a_j$ in the direction $b_j - a_j$, i.e.,

$$f'(a_j) = \frac{\partial f}{\partial x_1}(a_j) \frac{b_{j1} - a_{j1}}{d_j} + \ldots + \frac{\partial f}{\partial x_N}(a_j) \frac{b_{jN} - a_{jN}}{d_j},$$  

(2.2)

where

$$d_j = ||b_j - a_j||$$  

(2.3)

is the length of the main diagonal of the interval $D(j)$. Analogously we calculate $f'(b_j)$ as the directional derivative at the point $b_j$

$$f'(b_j) = \frac{\partial f}{\partial x_1}(b_j) \frac{b_{j1} - a_{j1}}{d_j} + \ldots + \frac{\partial f}{\partial x_N}(b_j) \frac{b_{jN} - a_{jN}}{d_j}.$$  

(2.4)

Consider now the formula for the characteristic $R(i)$ calculated for every one-dimensional interval $[x_{i-1}, x_i]$, $1 \leq i \leq k$, in [17]

$$R(i) = \min \left\{ z_{i-1}, z_i + z_{i-1}(\hat{x}_i - x_{i-1}) - 0.5\mu_i(x_{i} - x_{i-1})^2 \right\},$$

where $z_i = f(x_i), z'_i = f'(x_i), 1 \leq i \leq k,$

$$\hat{x}_i = -z_i + z_{i-1} + z'_ix_i - z'_{i-1}x_{i-1} + 0.5\mu_i(x_i^2 - x_{i-1}^2)$$

and $\mu_i$ is an estimate of the local Lipschitz constants $L_i$ of the interval $[x_{i-1}, x_i]$. Generalizing these expressions and using (2.2)–(2.4) we associate to a hyperinterval $D(j)$ the following characteristic:

$$R(j) = \min \{ f(a_j), f(b_j), f'(a_j)\hat{x}_j - 0.5\mu_j\hat{x}_j^2 \},$$  

(2.5)

where a way to calculate $\mu_j$ will be presented below (see (4.10)) and $\hat{x}_j, 0 < \hat{x}_j < d_j$, is a point laying at the main diagonal and calculated as

$$\hat{x}_j = 0.5d_j + \frac{f(b_j) + f(a_j) + 0.5d_j[f'(b_j) + f'(a_j)]}{\mu_j d_j + f'(b_j) - f'(a_j)}.$$  

(2.6)

Fig. 1 illustrates formulae (2.5), (2.6) for the case $N = 2$. The characteristic $R(j) = \varphi(\hat{x}_j)$, where $\varphi(x), x = a_j + \alpha(b_j - a_j), 0 \leq \alpha \leq 1$, is a support function for $f(x)$ over the main diagonal $d_j$ if $\mu_j \geq L_j$.

Remark 1. Note, that for $\hat{x}_j$ from (2.6) the following estimate

$$\max \{ \hat{x}_j, d_j - \hat{x}_j \} \leq 0.5(r + 1)r^{-1}d_j$$  

(2.7)

can be proved on the basis of lemma 3.1 from [17], where $r > 1$ is the reliability parameter (see (4.10)).
Remark 2. The characteristics of the one-dimensional methods from [30–32, 36] can be generalized to the multidimensional case by using the same approach (see (2.1)).

3. PARTITIONING SCHEME

In this section we propose a new scheme of sequential splitting $D$ into $N$-dimensional subintervals. Numbers of hyperintervals for subdivision will be generated one after another by the multidimensional global optimization algorithm to be described in Sect. 5. To calculate the characteristics for every subinterval we need (see (2.5), (2.6)) the values

$$a_j, b_j, f(a_j), f(b_j), \frac{\partial f}{\partial x_j}(a_j), \ldots, \frac{\partial f}{\partial x_N}(a_j), \frac{\partial f}{\partial x_j}(b_j), \ldots, \frac{\partial f}{\partial x_N}(b_j),$$

i.e. $2(N + 1)$-component real vector. It has been shown (see [32, 34, 35]) that simple partitioning schemes consume too much memory already in the case when they are applied for extending to the multidimensional case methods do not using derivatives, i.e. it is necessary to keep only the $2(N + 1)$-component real vector $a_j, b_j, f(a_j), f(b_j)$ for every interval $D(j)$. That is why we propose a new partitioning scheme (which can be used also for solving the problems discussed in [32, 34, 35]) which permits to store trial points in a data base separately from the intervals they belong to. Here in after we use the term trial to identify the operation of evaluating the values

$$f(x), \frac{\partial f}{\partial x_1}(x), \ldots, \frac{\partial f}{\partial x_N}(x)$$

at a point $x \in D$. To achieve the access to the data a special indexation for hyperintervals is introduced. Thus we can avoid the multiple keeping of information corresponding to trial points which are vertices of different hyperintervals.

The procedure of splitting consists in the following. A subinterval $D(j) \subset D$ chosen for partitioning is split into two equal subintervals by the hyperplane orthogonal to the edge parallel to the $i$-th coordinate axis and passing through the point

$$u = (a_{j1}, a_{j2}, \ldots, a_{ji-1}, a_{ji} + 0.5(b_{ji} - a_{ji}), a_{ji+1}, \ldots, a_{jN}),$$

where $i$ is defined as follows

$$i = \min \{ t : |b_t - a_t| = \max \{ |b_n - a_n| : 1 \leq n \leq N \} \}.$$

We indicate every subinterval $D(j)$ by the vector $m(j) = (m_1, \ldots, m_N)$, where $m_i, 1 \leq i \leq N$, is a number composed by $n'_i + 1$ characters from the alphabet $\{0, 1\}$. The number $n_i$ demonstrates how many times split-
ting orthogonal to the coordinate axis $x$, has been done to obtain the subinterval $D(j) = D(m_1, ..., m_N)$. As it will be shown further, characters composing $m_i$ permit to determine for the interval $D(j)$ the vertices $a_j, b_j$ the objective function $f(x)$ has been calculated at. Without loss of generality here in after we consider the domain

$$D = \{x: 0 \leq x_n \leq 1, 1 \leq n \leq N\}.$$ 

At the first iteration we evaluate

$$f(a_1), f(b_1), \frac{\partial f}{\partial x_1}(a_1), ..., \frac{\partial f}{\partial x_N}(a_1), \frac{\partial f}{\partial x_1}(b_1), ..., \frac{\partial f}{\partial x_N}(b_1),$$

at the vertices $a_1 = (0, ..., 0)$ and $b_1 = (1, ..., 1)$. The hypercube $D$ is indicated as $D(1) = D(0, 0)$. Due to the scheme of division the hypercube $D$ is split by the hyperplane orthogonal to the coordinate axis $x_1$ and passing through the point $(0.5, 0, ..., 0)$ (see (3.1)). This operation generates two new subintervals

$$D(1) = D(00, 0, ..., 0), \quad D(2) = D(01, 0, ..., 0).$$

Their vertices (which determine the corresponding main diagonals) are:

$$D(00, 0, ..., 0) \rightarrow a_1 = (0, 0, ..., 0), \quad b_1 = (0.5, 1, ..., 1);$$

$$D(01, 0, ..., 0) \rightarrow a_2 = (0.5, 0, ..., 0), \quad b_2 = (1, 1, ..., 1).$$

Let us consider now a general case. After $k$ iterations we wish to split by the hyperplane orthogonal to a coordinate axis $x_i$ a hyperinterval $D(i) = D(m_1, ..., m_i, ..., m_N)$. To execute the partition we need coordinates of the vertices $a_i, b_i$, where $f(x)$ has been evaluated. We do not keep coordinates of the points $a_i, b_i$ but reestablish these ones by using the characters $h_i, ..., h_N$: composing the numbers $m_i, 1 \leq i \leq N$.

In fact, coordinates calculated $a_i, 1 \leq i \leq N$, of the point $a_i$ are calculated as

$$a_{i_j} = \sum_{j=0}^{n_i} 2^{-j} h_j, \quad 1 \leq i \leq N. \quad (3.2)$$

Coordinates $b_i, 1 \leq i \leq N$, of the point $b_i$ are calculated analogously:

$$b_{i_j} = 2^{-n_i}(h_n + 1) + \sum_{j=0}^{n_i-1} 2^{-j} h_j, \quad 1 \leq i \leq N. \quad (3.3)$$

Having calculated the points $a_i$ and $b_i$, we create the following two subintervals substituting the old interval

$$D(i) = D(m_1, ..., m_i, ..., m_N)$$

by the new one:

$$D(i) = D(m_1, ..., m_{i-1}, h_0 h_1 ... h_{n_i-1} h_{n_i} 0, m_{i+1}, ..., m_N). \quad (3.4)$$

$$D(k + 1) = D(m_1, ..., m_{i-1}, h_0 h_1 ... h_{n_i-1} h_{n_i} 1, m_{i+1}, ..., m_N). \quad (3.5)$$

Thus, to obtain the access to the information

$$f(a_j), f(b_j), \frac{\partial f}{\partial x_1}(a_j), ..., \frac{\partial f}{\partial x_N}(a_j), \frac{\partial f}{\partial x_1}(b_j), ..., \frac{\partial f}{\partial x_N}(b_j)$$

every interval $D(j)$ has to conserve only its numbers $m_i, 1 \leq i \leq N$.

Fig. 2 contains an example illustrating partitioning of $D$. Starting we have $D(1) = D(00, 0)$. After executing the second iteration we obtain

$$D(1) = D(00, 00), \quad D(2) = D(00, 01).$$

If in the course of the third iteration the interval $D(2)$ has been split we have

$$D(1) = D(00, 00), \quad D(2) = D(00, 01), \quad D(3) = D(01, 01).$$

Partitioning of the interval $D(1)$ during the forth iteration leads to the following situation

$$D(1) = D(00, 00), \quad D(2) = D(00, 01),$$

$$D(3) = D(01, 01), \quad D(4) = D(01, 00).$$
It is seen from the figure that $b_1 = a_1$. It is easily to demonstrate that in general every internal trial point in $D$ can belong to a pair of intervals. This property is very useful because permits (by using the introduced intervals indexation) to reduce calculations and the quantity of the information to store significantly.

In fact, suppose that after splitting we have obtained an interval $D(j)$ with the vertices $a_j$, $b_j$ to calculate the information (3.6). First of all we go to verify the existence of these data in the data base. It is possible if the values (3.6) have been already calculated in the course of previous iterations. Four cases are possible:

- there are no the points $a_j$, $b_j$ in our data base. Therefore, we have to calculate the information (3.6) and to keep it in the data base;
- the data have been calculated at the point $a_j$, i.e. we have
  \[ f(a_j), \frac{\partial f}{\partial x_1}(a_j), \ldots, \frac{\partial f}{\partial x_N}(a_j) \]  
  and to obtain (3.6) we have to calculate only the values
  \[ f(b_j), \frac{\partial f}{\partial x_1}(b_j), \ldots, \frac{\partial f}{\partial x_N}(b_j); \]  
  (3.8)

- the data have been calculated at the point $b_j$. This means that we already have (3.8) and we need only (3.7) to have (3.6);
- the data have been calculated at both the points $a_j$ and $b_j$ and we have the complete information (3.6) in our data base. Thus, we have nothing to calculate.

4. ESTIMATES OF THE LOCAL LIPSCHITZ CONSTANTS FOR DERIVATIVES

To calculate for an interval $D(i)$, $1 \leq i \leq k$, the characteristic $R(j)$ from (2.5) we need the value $\mu_j$ estimating the local Lipschitz constant $L$. In this section we propose a scheme for calculating $\mu_j$.

Let us consider the following auxiliary problem. Suppose that we have two points $u, v \in D$ and it is necessary to estimate the value $L(u, v)$ from

\[ |f'(u) - f'(v)| \leq L(u, v) \|u - v\|, \]  
(4.1)

where $f'(u), f'(v)$ are the directional derivative calculated at the points $u, v$ in the direction $v - u$, i.e.:

\[ f'(u) = \frac{\partial f}{\partial x_1}(u) \frac{v_1 - u_1}{\|u - v\|} + \ldots + \frac{\partial f}{\partial x_N}(u) \frac{v_N - u_N}{\|u - v\|}, \]

\[ f'(v) = \frac{\partial f}{\partial x_1}(v) \frac{v_1 - u_1}{\|u - v\|} + \ldots + \frac{\partial f}{\partial x_N}(v) \frac{v_N - u_N}{\|u - v\|}. \]

Naturally, it is supposed that the values

\[ f(u), f(v), \frac{\partial f}{\partial x_1}(u), \ldots, \frac{\partial f}{\partial x_N}(u), \frac{\partial f}{\partial x_1}(v), \ldots, \frac{\partial f}{\partial x_N}(v) \]

are known. Note, that due to (1.2) the inequality (4.1) is fulfilled and $L(u, v) \leq L$, where $L$ is from (1.2). We will deduce an estimate $\hat{\mu}(u, v)$ for the value $L(u, v)$ from the following inequalities which should take place:

\[ \phi^+(u, y) \geq \phi^-(v, y), \quad y \in [0, \|u - v\|], \]  
(4.2)

\[ \phi^+(v, y) \geq \phi^-(u, y), \quad y \in [0, \|u - v\|]. \]  
(4.3)
where
\[ \phi^+(u, y) = f(u) + f'(u)y + 0.5\mu(u, v)y^2 \]
is the upper bound based on the point \( u \) for the objective function \( f(x) \), \( x = u + \alpha(v - u), \alpha \in [0, 1] \), and
\[ \phi^-(v, y) = f(v) + f'(v)(y - \|u - v\|) - 0.5\mu(u, v)(y - \|u - v\|)^2 \]
is the lower bound based on the point \( v \), and analogously
\[ \phi^+(v, y) = f(v) + f'(v)(y - \|u - v\|) + 0.5\mu(u, v)(y - \|u - v\|)^2 \]
is the upper bound based on the point \( v \) and
\[ \phi^-(u, y) = f(u) + f'(u)y - 0.5\mu(u, v)y^2 \]
is the lower bound based on the point \( u \).

From (4.2), (4.3) we obtain that \( \mu(u, v) \) should be taken in a way providing fulfillment of the following inequality:
\[ \mu(u, v) \geq \tau(u, v, y) \quad \forall y \in [0, \|u - v\|], \]
where
\[ \tau(u, v, y) = 2\left[\frac{f(v) - f(u) - f'(v)(\|u - v\| - y) - f'(u)y}{(\|u - v\| - y)^2 + y^2}\right]. \]
It can be shown that the function
\[ V(u, v) = \max \{\tau(u, v, y) : y \in [0, \|u - v\|]\}, \]
can be represented in the form
\[ V(u, v) = \begin{cases} T(u, v), & f'(u) = f'(v), \\ \overline{T}(u, v), & \text{otherwise}, \end{cases} \tag{4.4} \]
where
\[ T(u, v) = \tau(u, v, 0.5\|u - v\|), \]
\[ \overline{T}(u, v) = \max \{\tau(u, v, 0), \tau(u, v, \|u - v\|), \tau(d^+(u, v)), \tau(d^-(u, v))\}. \]
The value \( \tau(d^+(u, v)) \) in the last formula is taken in consideration only if
\[ 0 \leq d^+(u, v) \leq \|u - v\|. \]
Analogously \( \tau(d^-(u, v)) \) is included in max if
\[ 0 \leq d^-(u, v) \leq \|u - v\|. \]
The values \( d^+(u, v), d^-(u, v) \) are determined in the following way:
\[ d^+(u, v) = \frac{f(u) - f(v) + f'(v)\|u - v\| - d(u, v)}{f'(v) - f'(u)}, \]
\[ d^-(u, v) = \frac{f(u) - f(v) + f'(v)\|u - v\| + d(u, v)}{f'(v) - f'(u)}, \]
\[ d(u, v) = 0.5([2(f(u) - f(v)) + f'(v) + f'(u)]\|u - v\|)^2 + (f'(v) - f'(u))^2\|u - v\|^2)^{1/2}. \]
Now we are ready to return to our problem and to estimate local Lipschitz constants for hyperintervals. Suppose, that in the course of the \((k + 1)\)-th iteration an interval \( D(t) \) has been split on two new intervals \( D(t) \) and \( D(k + 1) \) from (3.4), (3.5). The vertices \( a_t, b_t, a_{k+1}, b_{k+1} \) of these intervals (see Fig. 3) are calculated according to (3.2)–(3.5).

To estimate the values \( L_n, L_{k+1} \) we calculate the values \( \lambda_n, \lambda_{k+1} \) and \( \gamma_n, \gamma_{k+1} \) being local and global estimates respectively. We determine the values \( \lambda_n, \lambda_{k+1} \) as follows:
\[ \lambda_n = \max \{V(a_n, b_n), V(a_n, a_{k+1}), V(b_n, b_{k+1}), V(b_n, a_{k+1})\}, \tag{4.5} \]
\[ \lambda_{k+1} = \max \{V(a_n, b_{k+1}), V(b_n, a_{k+1}), V(b_n, b_{k+1}), V(a_{k+1}, b_{k+1})\}. \tag{4.6} \]
Thus, we calculate the values \( \lambda_n, \lambda_{k+1} \) as the maximum of the estimates \( V(u, v) \), where different combinations of the vertices are used as the points \( u \) and \( v \). The pairs of vertices, for which we calculate \( V(u, v) \) are
connected in Fig. 3 by the bold lines. The value $\gamma_i$ for the interval $D(i)$ is computed as follows:

$$\gamma_i = M^{k+1} \frac{d_i}{X^{k+1}}$$

(4.7)

where

$$M^{k+1} = \max \{ \lambda_i : 1 \leq i \leq k + 1 \},$$

(4.8)

$$X^{k+1} = \max \{ d_i : 1 \leq i \leq k + 1 \},$$

(4.9)

and $d_i, 1 \leq i \leq k + 1$, are calculated due to (2.3). The value $\gamma_{k+1}$ for the interval $D(k+1)$ is computed analogously.

We will estimate $L_i$ by the value

$$\mu_i = r \max \{ \lambda_i, \gamma_i, \xi \},$$

(4.10)

where $r > 1$ is a reliability parameter and $\xi > 0$ is a small number. The value $\mu_{k+1}$ is computed by a complete analogy.

The value $\mu_j$ estimates the local Lipschitz constant $L_j$ over the hyperinterval $D(j)$. The values $\lambda_i$ and $\gamma_i$ reflect the influence on $\mu_j$ respectively of the local and global information obtained in the course of the previous iterations. When the main diagonal $d_j$ of the interval $D(j)$ is small then (see (4.7)–(4.9)) $\gamma_j$ is small also and we use (see (4.10)) the local information presented by $\lambda_j$. When $d_j$ is very large then the local information is not reliable and the global information represented by $\gamma_j$ is used.

5. THE MULTIDIMENSIONAL ALGORITHM

Now we are ready to describe the decision rules of the multidimensional global optimization algorithm.

At the first iteration we set the counter of iterations $k = 1$, evaluate

$$f(a_1), f(b_1), \frac{df}{dx_1}(a_1), ..., \frac{df}{dx_N}(a_1), \frac{df}{dx_1}(b_1), ..., \frac{df}{dx_N}(b_1),$$

at the vertices $a_i = (0, ..., 0)$ and $b_i = (1, ..., 1)$ and save this information in the data base. The hypercube $D$ is indicated as $D(1) = D(0, ..., 0)$. Suppose now that $k > 1$ iterations of the algorithm have already been done and a hyperinterval $D(j^k)$ where the value $X^k$ from (4.9) is achieved has been determined, i.e.

$$j^k = \text{argmax}\{d_i : 1 \leq i \leq k\}.$$

(5.1)

Moreover, the value $M^k$ from (4.8) has been computed. At the end of the $k$-th iteration an interval $D(i)$ has been chosen to be divided in the course of the $(k+1)$-th iteration of the algorithm. This iteration consists in the following:

**Step 1.** Create two new subintervals $D(i)$ and $D(k+1)$ due to (3.2)–(3.5) and calculate for them (or read from the data base) the information (3.7) at the point $a_{i+1}$ and (3.8) at the point $b_i$. If a new information (which can be (3.7) or (3.8) or (3.6)) has been obtained then, save it in the data base.

**Step 2.** Calculate and save the values $\lambda_i$ and $\lambda_{k+1}$ from (4.10) by using (4.4)–(4.6).

**Step 3.** If $i = j^k$ then recalculate $X^{k+1}$ and corresponding number $J^{k+1}$ following (4.9), (5.1). Otherwise set

$$X^{k+1} = X^k, \quad J^{k+1} = J^k.$$

**Step 4.** In this step we determine the value $M^{k+1}$. If

$$M^k + \delta > \max \{ \lambda_i, \lambda_{k+1} \} > M^k,$$

where $\delta > 0$, then set

$$M^{k+1} = M^k,$$

$$m' = \max \{ \lambda_i, \lambda_{k+1} \}.$$

If

$$M^k + \delta < \max \{ \lambda_i, \lambda_{k+1} \},$$
set \[ M^{k+1} = \max \{ \lambda, \lambda_{k+1} \}. \]

Otherwise set
\[ M^{k+1} = M^k. \]

**Step 5.** If \( X^{k+1} = X^k \) and \( M^{k+1} = M^k \) then compute and save the characteristics \( R(t) \) and \( R(k+1) \) using (2.5), (2.6) and taking \( \mu \) and \( \mu_{k+1} \) following (4.7)–(4.10). Otherwise it is necessary to recalculate all characteristics \( R(j), 1 \leq j \leq k+1 \).

**Step 6.** Find among the hyperintervals \( D(j), 1 \leq j \leq k+1 \), an interval \( D(t) \) with the minimal characteristic
\[ R(t) = \min \{ R(j) : 1 \leq j \leq k+1 \}. \]

**Step 7** (stopping rule). If the condition
\[ d_t \leq \varepsilon, \]

is fulfilled and \( m' \leq M^{k+1} \) then stop. Here \( t \) is from (5.2), \( d_t \) is the main diagonal of the interval \( D(t) \), \( \varepsilon > 0 \) is a preset accuracy. If (5.3) holds but \( m' > M^{k+1} \) then, set \( M^{k+1} = m' \) and go to Step 5. Otherwise set the counter of iterations \( k = k+1 \) and go to Step 1.

**Remark 3.** We have introduced the values \( m' \) and \( \delta \) in the scheme because the operation of recalculating characteristics is quite expensive. Thus, we recalculate all characteristics only if \( M^k \) has been increased more than a number \( \delta > 0 \). Another reason for recalculating characteristics is the situation when the \( (k+1) \)-th iteration has been executed at the interval with the maximal diagonal \( X^k \). As after splitting this diagonal is no more it is necessary to find a new maximal diagonal \( X^{k+1} \).

**6. CONVERGENCE CONDITIONS**

In this section we present sufficient conditions of convergence to global minimizers for the multidimensional algorithm. We shall denote by \( X^* \) the set of global minimizers of the objective function and by \( \{ x^k \} \) the sequence of trial points generated by the method during solving the problem (1.1), (1.2).

The set of limit points of the sequence \( \{ x^k \} \) is indicated by \( X^* \).

**Theorem.** For every function \( f(x) \) satisfying (1.1), (1.2) there exist numbers \( r^*, \xi^* \) such that for all \( r \geq r^* \), \( \xi \geq \xi^* \) where \( r \) and \( \xi \) are from (4.10), we have \( X^* = X^* \).

**Proof.** Suppose, that there exists a limit point \( x' \) of \( \{ x^k \} \) such that
\[ f(x') > f(x^*), \quad x^* \in X^*, \quad D(j(k)) \text{ be a hyperinterval containing } x' \text{ in the course of the } k \text{-th iteration}. \]

As \( x' \) is a limit point then due to decision rules of the method for the characteristic \( R(j(k)) \) of \( D(j(k)) \) we have
\[ \lim_{k \to \infty} R(j(k)) = f(x'). \]

Let us now estimate the characteristic \( R(i), i = i(k) \), of the hyperinterval \( D(i(k)) \) containing a point \( x^* \in X^* \) in the course of the \( k \)-th iteration. By using the Taylor series we obtain
\[ R(i) = \min \{ f(a_i) + f'(a_i) \delta_i - 0.5\mu \delta_i^2, \quad R(i) = f(a_i) + f'(a_i) \delta_i - 0.5\mu \delta_i^2 \leq f(a_i) + f'(a_i) \delta_i - 0.5\mu \delta_i^2, \]

where \( L_i \) is the local Lipschitz constant for \( D(i) \) and
\[ \alpha_i = a_i + \delta_i (b_i - a_i) / d_i. \]

Now we estimate separately three items from (6.2). By using again Taylor series and (2.7) we have for the first one
\[ f(\alpha_i) \leq f(x^*) + 0.5 L_i \| x^* - \alpha_i \|^2 \leq f(x^*) + 0.5 L_i \| 0.5 (r+1) r^{-1} d_j \|^2. \]

The second item is estimated due to (2.7) as follows
\[ 0.5 L_i \delta_i^2 \leq 0.5 L_i \| 0.5 (r+1) r^{-1} d_j \|^2. \]

Finally, by taking in consideration (2.7) and (4.10) we can write for the third item
\[ -0.5\mu \delta_i^2 \leq -0.5\mu \| 0.5 (r+1) r^{-1} d_j \|^2 \leq -0.5\mu \| 0.5 (r+1) r^{-1} d_j \|^2. \]
MULTIDIMENSIONAL GLOBAL OPTIMIZATION USING THE FIRST DERIVATIVES

Summarizing we obtain from (6.2)–(6.5) that

\[ R(i) \leq f(x^*) + L_i [(r + 1)r^{-1} d_j]^2 - 0.5 r \xi [0.5(r - 1)r^{-1} d_j]^2. \]  

(6.6)

It follows from (6.6) and the fact that \( L_i \leq L \) that

\[ R(i) \leq f(x^*) \]  

(6.7)

if

\[ (r - 1)^2 r \xi /(r + 1)^2 \geq 2L_i. \]  

(6.8)

Since \( L_i \) is a constant, then for every objective function there exist \( r^*, \xi^* \) such that for all \( r \geq r^*, \xi > \xi^* \) the estimate (6.7) takes place. Thus, we obtain from (6.1), (6.6) that, in accordance with step 6 of the algorithm, \( x^* \) can not be a limit point of \( \{x^k\} \). Since (6.7) holds for all \( x^* \in X^* \) theorem has been completely proved.

7. CONCLUSION

In this paper a new scheme for generalizing to the multidimensional case one-dimensional methods belonging to the "divide the best" algorithms family (including as subclasses adaptive partition algorithms and characteristic algorithms) has been introduced. The scheme provides:

(i) extending to the multidimensional case methods using derivatives;
(ii) an efficient keeping the search information in the course of optimizing;
(iii) possibility to construct algorithms using (for the objective function and/or its derivatives) given Lipschitz constant, its global or local estimates making in the course of the search.

A new multidimensional algorithm for solving global optimization problems with the objective function given over a hyperinterval and having Lipschitzian first derivatives has been proposed on the basis of the scheme introduced.

ACKNOWLEDGMENT

The author thanks the anonymous referees for their useful suggestions.

REFERENCES