IMPROVED LIPSCHITZ BOUNDS WITH THE FIRST NORM FOR FUNCTION VALUES OVER MULTIDIMENSIONAL SIMPLEX

R. PAULAVIČIUS and J. ŽILINSKAS

Institute of Mathematics and Informatics
Akademijos 4, LT-08663, Vilnius, Lithuania
E-mail: r.paulavicius@vpu.lt; julius.zilinskas@ktl.mii.lt

Received October 1, 2007; revised July 24, 2008; published online December 1, 2008

Abstract. A branch and bound algorithm for global optimization is proposed, where the maximum of an upper bounding function based on Lipschitz condition and the first norm over a simplex is used as the upper bound of function. In this case the graph of bounding function is intersection of n-dimensional pyramids and its maximum point is found solving a system of linear equations. The efficiency of the proposed global optimization algorithm is evaluated experimentally.

Key words: global optimization, branch and bound algorithms, Lipschitz optimization, the first norm.

1 Introduction

Our aim is to find at least one globally optimal solution to the problem

$$f^* = \max_{x \in D} f(x),$$

where an objective function $f(x), f: \mathbb{R}^n \to \mathbb{R}$, is a real-valued Lipschitz function, $D \subseteq \mathbb{R}^n$ is a feasible region, $n$ is the number of variables. A function $f: D \to \mathbb{R}, D \subseteq \mathbb{R}^n$, is said to be Lipschitz if it satisfies the condition

$$|f(x) - f(y)| \leq L \|x - y\|, \quad \forall x, y \in D,$$

where $L > 0$ is a constant called Lipschitz constant, $D$ is a compact set and $\|\cdot\|$ denotes the norm. The Euclidean norm is used most often, but other norms could also be considered. In [15] we showed that for Lipschitz function $f(x)$

$$|f(x) - f(y)| \leq L_p \|x - y\|_q,$$

where $L_p \|x - y\|_q$.
where \( L_p = \sup \{ || \nabla f(x) ||_p : x \in D \} \) is Lipschitz constant, the gradient of the function \( f(x) \) is denoted by \( \nabla f(x) = (\frac{\partial f}{\partial x_1}, \ldots, \frac{\partial f}{\partial x_n}) \), and \( 1/p + 1/q = 1, 1 \leq p, q \leq \infty \).

The most often studied case of problem (1.1) is the univariate one \( (n = 1) \), for which many algorithms have been proposed, compared and theoretically investigated. In the presented paper, we are interested in the multivariate case \( (n \geq 2) \).

In Lipschitz optimization the upper bound of the optimal value \( f^* \) is evaluated exploiting Lipschitz condition. It follows from (1.2) that, for all \( x, y \in D \)

\[
f(x) \leq f(y) + L \| x - y \|.
\]

If \( y \in D \) is fixed, then the concave function \( F(x) = f(y) + L \| x - y \| \) estimates \( f(x) \) on \( D \).

Almost all methods for solving the multivariate problem fall into two main classes. The first class contains the direct extensions of Piyavskii’s method [19] to the multivariate case and various modifications with different norms or approximations [7, 11, 12, 13, 14, 21]. Note that when the Euclidean norm is used in the multivariate case, the upper bounding functions are envelopes of circular cones with parallel symmetry axes. A problem of finding maximum of such a bounding function becomes a difficult global optimization problem involving systems of quadratic and linear equations. Most of these algorithms can be improved by interpreting them as branch and bound methods [5, 7, 8].

The second class contains many simplicial and rectangular branch and bound techniques [4, 6, 17, 18, 20]. They differ in the ways how branching is performed and bounds are computed. Simplicial partitions are preferable when the values of an objective function at the vertices of partitions are used to compute bounds [22, 27]. Another advantage of simplicial partitions is that they may be used to vertex triangulate feasible regions of non-rectangular shape defined by linear inequality constraints [27], what allows reduction of search space of problems with symmetric objective functions [26].

In general, bounds belong to the following two simple families \( \mu_1(P) \) and \( \mu_2(P) \). Let

\[
delta(P) = \max \{ \| x - y \| : x, y \in P \}
\]

denotes the diameter of \( P \). For example, if \( P = \{ x \in \mathbb{R}^n : a \leq x \leq b \} \) is \( n \)-rectangle, then \( \delta(P) = \| b - a \| \) and if \( P \) is an \( n \)-simplex then diameter \( \delta(P) \) is the length of its longest edge. Then simpler upper bound can be derived:

\[
\mu_1(P) = \min_{y \in T} f(y) + L \delta(P), \tag{1.3}
\]

here \( T \subset P \) is finite sample of points in \( P \) where the function values of \( f \) have been evaluated. When \( P \) is a rectangle or a simplex the set \( T \) often coincides with the vertex set \( V(P) \). The more tight but computationally more expensive than (1.3) is the estimate

\[
\mu_2(P) = \min_{y \in T} \left\{ f(y) + L \max_{z \in V(P)} \| y - z \| \right\}. \tag{1.4}
\]
Performance of optimization algorithms depends on tightness of bounds [23]. For $D \subseteq P$, the sharpest upper bound given the knowledge of the function values $f(y), y \in T$, and of the Lipschitz constant $L$, is provided by

$$\max_{x \in D} \min_{y \in T} (f(y) + L \|x - y\|).$$

(1.5)

Although it may be more natural to formulate this problem as min-max problem, we keep formulation used in [5, 13, 14]. Anyway (1.5) is a difficult optimization problem when the space has dimension $n \geq 2$.

The Euclidean norm is used most often, but other norms could be also considered. In [15, 16] we investigated how different norms and corresponding Lipschitz constants influence speed of algorithms for global optimization when (1.4) is used on vertices of simplex:

$$UB(I) = \min_{v \in V(I)} \left\{ f(v) + L_{\|x - v\|_\infty} \right\},$$

(1.6)

where $v$ is a vertex of the simplex $I$ and values of the function at all vertices of this simplex $V(I)$ are used. Experiments have shown that better results may be achieved when non Euclidean norms are used. Therefore for better upper bound we proposed the combination of two extreme (infinite and first) and the Euclidean norms:

$$UB_{1,2,\infty}(I) = \min_{v \in V(I)} \left( f(v) + \min \{ L_1 \max_{x \in I} \|x - v\|_\infty, L_2 \max_{x \in I} \|x - v\|_2, L_\infty \max_{x \in I} \|x - v\|_1 \} \right).$$

(1.7)

where the first norm is defined as $\|x\|_1 = \sum_{i=1}^n |x_i|$.

In this paper we investigate improved bounds with the first norm. For finding upper bound we exploit (1.5) formula:

$$F(I) = \max_{x \in I} \left( \min_{v \in V(I)} (f(v) + L_{\|x - v\|_1}) \right).$$

(1.8)

In this case the graph of $F$ is the intersection of $n$-dimensional pyramids and the maximum point is found solving system of linear equations. Branch and bound algorithm with simplicial partitions and face to face vertex triangulation [27] is used to find the global maximum.

2 Improved Upper Bound

Let us formulate two propositions, which are used for solving global optimization problem (1.1) with (1.8) for evaluation of improved upper bound.

**Proposition 1.** If two $n$-pyramids $F_{v_1}(x) = f(v_1) + L_{\|x - v_1\|_1}$ and $F_{v_2}(x) = f(v_2) + L_{\|x - v_2\|_1}$ are defined and $f(v_1) \geq f(v_2)$ then the intersection of pyramids is contained in manifold of dimensionality $n - 1$ defined by

$$\sum_{i=1}^n d(v_{1i}, v_{2i}) - \frac{f(v_2) - f(v_1)}{L_\infty} = 0,$$

(2.1)
where
\[ d(v_{1i}, v_{2i}) = \begin{cases} 
2x_i - v_{1i} - v_{2i}, & \text{when } v_{1i} < v_{2i}, \\
-2x_i + v_{1i} + v_{2i}, & \text{when } v_{1i} > v_{2i}, \\
0, & \text{when } v_{1i} = v_{2i}
\end{cases} \]

and all points \( x \) in the intersection are closer to the vertex \( v_1 \) than to \( v_2 \), i.e.
\[ ||x - v_1||_1 \leq ||x - v_2||_1. \] (2.2)

**Proof.** From equality \( F_{v_1}(x) = F_{v_2}(x) \) we get
\[ \sum_{i=1}^{n} (|x_i - v_{1i}| - |x_i - v_{2i}|) = \frac{f(v_2) - f(v_1)}{L_\infty}. \] (2.3)

For each of difference \(|x_i - v_{1i}| - |x_i - v_{2i}|\), \( i = 1, \ldots, n \) case from the following three possibilities is true:
\[ \begin{cases} 
2x_i - v_{1i} - v_{2i}, & \text{when } v_{1i} < v_{2i}, \\
-2x_i + v_{1i} + v_{2i}, & \text{when } v_{1i} > v_{2i}, \\
0, & \text{when } v_{1i} = v_{2i}.
\end{cases} \]

Therefore from (2.3) we get (2.1).

Because \( f(v_1) > f(v_2) \), from equality \( F_{v_1}(x) = F_{v_2}(x) \) we get
\[ ||x - v_1||_1 - ||x - v_2||_1 = \frac{f(v_2) - f(v_1)}{L_\infty} \leq 0, \]

therefore (2.2) is true:
\[ ||x - v_1||_1 \leq ||x - v_2||_1. \]

**Proposition 1.** The maximum point of \( F \) can be found solving a system of \( n \) linear equations.

**Proof.** Let us define numeration of vertices \( v_i \) so that \( f(v_1) \geq f(v_2) \geq \ldots \geq f(v_{n+1}) \).

Intersection of pyramids \( F_{v_1}(x) = f(v_1) + L_\infty ||x - v_1||_1 \) and \( F_{v_i}(x) = f(v_i) + L_\infty ||x - v_i||_1, \) \( i = 2, \ldots, n + 1, \) is \((n-1)\)-manifold defined by (2.1) (see Proposition 1). Taking into account (2.2), it is possible to consider only part of the manifold, which is defined by linear equation and constraints. Therefore it is possible to form a system of \( n \) linear equations defining intersections \( F_{v_i}(x) = F_{v_1}(x) \). If the solution of this system satisfies the constraints (see Example 1) then the upper bounding function is maximal at the solution point. If the solution of this system does not satisfy the constraints, then the maximum of the upper bounding function is the minimum of the function value at the intersections (see Example 2).

### 3 Algorithm

Branch and bound is a technique for implementation of covering global optimization methods [27] as well as combinatorial optimization algorithms. An iteration of a classical branch and bound algorithm processes a node in the
search tree representing a not yet explored subspace of the solution space. The iteration has three main components: selection of the node to process, branching of the search tree and bound calculation. Bound may be estimated using interval arithmetic \cite{24, 25, 28, 29} as well as Lipschitz condition. Subspaces which cannot contain a global minimum are discarded from further search pruning the branches of the search tree. Although covering, selection, branching and bounding rules differ in different branch and bound algorithms, the structure of the algorithm remains the same, what enables implementation of branch and bound algorithms using prepared templates \cite{1, 2, 3}.

In the proposed algorithm simplicial partitions, face to face vertex triangulation for initial covering, subdivision through the middle of the longest edge of simplex, the breadth first selection strategy and improved bounds (1.8) with first norm are used. The proposed algorithm is shown in Algorithm 1.

\begin{algorithm}
\caption{Branch and bound algorithm with simplicial partitions.}
\begin{algorithmic}
\State An \( n \)-dimensional hyper-rectangle \( D \) is face-to-face vertex triangulated into \( n \)-dimensional simplices \( I = \{ I_k | D \subseteq \cup I_k, k = 1, \ldots, n! \} \).
\State LB(D) = \(-\infty\).
\While \( I \) is not empty: \( I \neq \emptyset \)
\State Choose and exclude \( I_k \in I \) from the set of non-solved simplices \( I \).
\State LB(D) = max\{LB(D), max_v \in V(I_k) f(v)\}
\State UB(I_k) = max_v \in I_k (\min_v \in V(I_k) [f(v) + L_\infty ||x - v||_1])
\If {UB(I_k) - LB(D) > \varepsilon}
\State Branch \( I_k \) into 2 simplices: \( I_{k_1}, I_{k_2} \).
\State \( I = \{ I, I_{k_1}, I_{k_2} \} \)
\EndIf
\EndWhile
\end{algorithmic}
\end{algorithm}

\section{Numerical Example}

\textbf{Example 1.} Suppose that the test objective function is \( f(x_1, x_2) = \sin(2x_1 + 1) + 2 \sin(3x_2 + 2) \) and feasible region \([0, 1] \times [0, 1]\) is covered by two right-angled equilateral simplices \( I_1(v_1, v_2, v_3) \) and \( I_2(v_1, v_3, v_4) \) (see Fig. 1a).

Let us consider the first simplex \( I_1(v_1, v_2, v_3) \). Because \( f(v_1) = f(0, 0) = 2.6601 > f(v_2) = f(1, 0) = 1.9597 > f(v_3) = f(1, 1) = -1.7767 \), the intersection point is closer to vertex \( v_1 \) and it is enough to find intersection of pyramids \( F_{v_1} = F_{v_2} \) and \( F_{v_1} = F_{v_3} \).

Intersection of pyramids \( F_{v_1} = F_{v_2} \) gives the following equation
\[
|x_1 - 0| - |x_1 - 1| + |x_2 - 0| - |x_2 - 0| = \frac{f(1, 0) - f(0, 0)}{6}.
\]
However as \( 0 \leq x_1 \leq 1 \), this equation can be simplified and solved:
\[
x_1 + x_1 - 1 = \frac{f(1, 0) - f(0, 0)}{6} \quad \Rightarrow \quad x_1 = 0.44164.
\]

\textit{Improved Lipschitz Bounds over Multidimensional Simplex} 557

Analogously intersection of pyramids $F_{v_1} = F_{v_3}$ may be found:

$$|x_1 - 0| - |x_1 - 1| + |x_2 - 0| - |x_2 - 1| = \frac{f(1,1) - f(0,0)}{6},$$

or (taking into account that $f(1,1) - f(0,0) = 0.63027$)

$$\begin{cases} x_1 + x_2 = 0.63027, & \text{when } 0 \leq x_1 \leq 1, \ 0 \leq x_2 \leq 1, \\ x_2 = 0.63027, & \text{when } x_1 \leq 0, \\ x_1 = 0.63027, & \text{when } x_2 \leq 0. \end{cases} \tag{4.2}$$

From (4.1), (4.2) and (2.2) it follows that

$$\begin{cases} x_1 = 0.44164 \\ x_1 + x_2 = 0.63027 \\ 0 \leq x_1 \leq 0.5 \\ 0 \leq x_2 \leq 1.0 \end{cases} \Rightarrow \text{intersection point } p_1 = (0.44164, 0.18863)$$

and improved upper bound is given by $F(I_1) = F_{v_1}(p_1)$ (see, Fig. 1):

$$F_{v_1}(p_1) = F_{v_1}(p_1) = f(v_1) + L_\infty \| p_1 - v_1 \|_1 = f(0,0) + 6 \cdot 2 = 14.66,$$

$UB_{v_1}(x) = f(0,0) + 6 \max_{x \in I} \|x - v_1\|_1 = f(0,0) + 6 \cdot 2 = 14.66$,

$UB_{v_2}(x) = f(1,0) + 6 \max_{x \in I} \|x - v_2\|_1 = f(1,0) + 6 \cdot 1 = 7.9597$,

$UB_{v_3}(x) = f(1,1) + 6 \max_{x \in I} \|x - v_3\|_1 = f(1,1) + 6 \cdot 2 = 10.223$,

$UB(I_1) = \min \{ UB_{v_1}, UB_{v_2}, UB_{v_3} \} = UB_{v_2} = 7.9597. \tag{4.4}$

From (4.3) and (4.4) it follows that $F(I_1) < UB(I_1)$. 

**Figure 1.** a) Projection of intersection lines; b) Visualization of upper bounding functions.
Example 2. Subdivision of simplices $I_1$ and $I_2$ produces four simplices $I_3, \ldots, I_6$, see Fig. 2. Let us consider the simplex $I_3(v_1, v_2, v_5)$. Because $f(v_1) = 2.6601 > f(v_2) = 1.9597 > f(v_5) = 0.20773$, the intersection point is closer to vertex $v_1$ and it is enough to find intersection of pyramids $F_{v_1} = F_{v_2}$ and $F_{v_1} = F_{v_5}$. Intersection of pyramids $F_{v_1} = F_{v_2}$ is found in Example 1:

$$x_1 = 0.44164.$$  \hfill (4.5)

Intersection of pyramids $F_{v_1} = F_{v_5}$ gives the equation:

$$|x_1 - 0| - |x_1 - 0.5| + |x_2 - 0| - |x_2 - 0.5| = \frac{f(0.5, 0.5) - f(0, 0)}{6},$$

or (taking into account that $(f(0.5, 0.5) - f(0, 0)/12 + 0.5 = 0.29564)$

$$\begin{cases} x_1 + x_2 = 0.29564, & \text{when } 0 \leq x_1 \leq \frac{1}{2}, \ 0 \leq x_2 \leq \frac{1}{2}, \\ x_2 = 0.29564, & \text{when } x_1 \leq 0, \\ x_1 = 0.29564, & \text{when } x_2 \leq 0. \end{cases} \hfill (4.6)$$

From (4.5), (4.6) and (2.2) we get a system

$$\begin{cases} x_1 = 0.44164 \\ x_1 + x_2 = 0.29564 \\ 0 \leq x_1 \leq 0.5 \\ 0 \leq x_2 \leq 0.5 \end{cases}$$

which defines an empty set of solutions (see Fig. 2). Therefore improved upper bound is achieved in intersection line belonging to the upper bounding function (see Fig. 2), i.e.

$$F(I_3) = F_{v_1}(p_1),$$

where \( p_1 \) is any point belonging to the line \( x_1 + x_2 = 0.29564 \) and \( 0 \leq x_1 \leq 0.5, \ 0 \leq x_2 \leq 0.5 \). Let \( p_1 = (0.29564, 0) \in x_1 + x_2 = 0.29564 \) and
\[
F_{v_1} (p_1) = f (v_1) + L_\infty \| p_1 - v_1 \|_1
= f (0, 0) + 6 ([0.29564 - 0] + [0 - 0]) = 4.4339. \tag{4.7}
\]
Let us verify that this upper bound is better than (2.1):
\[
U B_{v_1} (x) = f (0, 0) + 6 \max_{x \in I} \| x - v_1 \|_1 = f (0, 0) + 6 \cdot 1 = 8.66,
U B_{v_2} (x) = f (1, 0) + 6 \max_{x \in I} \| x - v_2 \|_1 = f (1, 0) + 6 \cdot 1 = 7.9597,
U B_{v_5} (x) = f (0.5, 0.5) + 6 \max_{x \in I} \| x - v_5 \|_1 = f (0.5, 0.5) + 6 \cdot 1 = 6.20773,
U B (I_3) = \min \{ U B_{v_1}, U B_{v_2}, U B_{v_5} \} = U B_{v_2} = 6.20773. \tag{4.8}
\]
From (4.7) and (4.8) it follows that
\[
F (I_3) < U B (I_3).
\]

5 Results of Experiments

Various test functions for global optimization from [5, 9, 10] have been used in our experiments. Lipschitz constants have been estimated using Theorem 1 from [15]. Test functions with \( n = 2 \) and \( n = 3 \) are numbered according to [5, 10]. For \( n \geq 4 \) functions names from [9] are used. The speed of global optimization is measured using the number of function evaluations criterion. The results are presented in Table 1.

<table>
<thead>
<tr>
<th>Function</th>
<th>( n = 2 )</th>
<th>( n = 3 )</th>
<th>( n \geq 4 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>( F (I_3) )</td>
<td>4.4339</td>
<td>8.66</td>
<td>15% smaller</td>
</tr>
<tr>
<td>( U B (I_3) )</td>
<td>8.66</td>
<td>7.9597</td>
<td>15% smaller</td>
</tr>
<tr>
<td>( F (I_3) )</td>
<td>6.20773</td>
<td>6.20773</td>
<td>15% smaller</td>
</tr>
</tbody>
</table>

Results of combination of bounds with different norms [15, 16] are also given in Table 1. Combination of bounds gives better results when the first norm is not preferable. However for some functions improved bound with first norm produces even better results than the combination. The results suggest to include improved bound with first norm into combination instead of simpler bound with first norm. It is also promising to develop improved bounds for other norms.

6 Conclusions

In this paper an improved Lipschitz bound with first norm has been proposed and applied in multidimensional Lipschitz global optimization. Test functions of different dimensionality \( n = 2, 3, 4, 5, 6 \) have been used for experimental investigation of branch and bound algorithm.
Table 1. Numbers of function evaluations.

<table>
<thead>
<tr>
<th>Test function</th>
<th>Dimension (n)</th>
<th>Precision (ε)</th>
<th>F(1.8)</th>
<th>UB(1.6)</th>
<th>UB (1.7)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. [5]</td>
<td>2</td>
<td>0.355</td>
<td>1085</td>
<td>1174</td>
<td>556</td>
</tr>
<tr>
<td>2. [5]</td>
<td>2</td>
<td>0.0446</td>
<td>181</td>
<td>242</td>
<td>158</td>
</tr>
<tr>
<td>3. [5]</td>
<td>2</td>
<td>11.9</td>
<td>5229</td>
<td>5673</td>
<td>4168</td>
</tr>
<tr>
<td>4. [5]</td>
<td>2</td>
<td>0.0141</td>
<td>30</td>
<td>36</td>
<td>9</td>
</tr>
<tr>
<td>5. [5]</td>
<td>2</td>
<td>0.1</td>
<td>59</td>
<td>62</td>
<td>62</td>
</tr>
<tr>
<td>6. [5]</td>
<td>2</td>
<td>44.9</td>
<td>1552</td>
<td>1620</td>
<td>1038</td>
</tr>
<tr>
<td>7. [5]</td>
<td>2</td>
<td>542.0</td>
<td>15180</td>
<td>17946</td>
<td>10685</td>
</tr>
<tr>
<td>10. [5]</td>
<td>2</td>
<td>0.691</td>
<td>1209</td>
<td>1321</td>
<td>1285</td>
</tr>
<tr>
<td>11. [5]</td>
<td>2</td>
<td>0.335</td>
<td>3290</td>
<td>3557</td>
<td>3085</td>
</tr>
<tr>
<td>12. [5]</td>
<td>2</td>
<td>0.804</td>
<td>13555</td>
<td>14929</td>
<td>14929</td>
</tr>
<tr>
<td>20. [5]</td>
<td>3</td>
<td>10.6</td>
<td>21186</td>
<td>23626</td>
<td>23626</td>
</tr>
<tr>
<td>21. [5]</td>
<td>3</td>
<td>0.369</td>
<td>11190</td>
<td>13677</td>
<td>1142</td>
</tr>
<tr>
<td>23. [5]</td>
<td>3</td>
<td>41.65</td>
<td>153461</td>
<td>170081</td>
<td>9009</td>
</tr>
<tr>
<td>25. [5]</td>
<td>3</td>
<td>0.0506</td>
<td>9652</td>
<td>11365</td>
<td>5117</td>
</tr>
<tr>
<td>Rosenbrock [10]</td>
<td>3</td>
<td>50000.0</td>
<td>29960</td>
<td>33517</td>
<td>25378</td>
</tr>
<tr>
<td>Rosenbrock [10]</td>
<td>4</td>
<td>2 \cdot L_2</td>
<td>3053</td>
<td>4172</td>
<td>2235</td>
</tr>
<tr>
<td>Shekel 5 [9]</td>
<td>4</td>
<td>2 \cdot L_2</td>
<td>11798</td>
<td>13485</td>
<td>13485</td>
</tr>
<tr>
<td>Shekel 7 [9]</td>
<td>4</td>
<td>2 \cdot L_2</td>
<td>11917</td>
<td>13485</td>
<td>13485</td>
</tr>
<tr>
<td>Shekel 10 [9]</td>
<td>4</td>
<td>2 \cdot L_2</td>
<td>11942</td>
<td>13485</td>
<td>13485</td>
</tr>
<tr>
<td>Levy No. 9 [9]</td>
<td>4</td>
<td>2 \cdot L_2</td>
<td>12829</td>
<td>16778</td>
<td>14087</td>
</tr>
<tr>
<td>Levy No. 15 [9]</td>
<td>4</td>
<td>2 \cdot L_2</td>
<td>1760604</td>
<td>&gt; 2000000</td>
<td>86396</td>
</tr>
<tr>
<td>Schwefel 1.2 [9]</td>
<td>4</td>
<td>2 \cdot L_2</td>
<td>28223</td>
<td>34828</td>
<td>16774</td>
</tr>
<tr>
<td>Powell [9]</td>
<td>4</td>
<td>2 \cdot L_2</td>
<td>2647</td>
<td>3686</td>
<td>1129</td>
</tr>
<tr>
<td>Rosenbrock [10]</td>
<td>5</td>
<td>2 \cdot L_2</td>
<td>124440</td>
<td>146346</td>
<td>141446</td>
</tr>
<tr>
<td>Levy No. 10 [9]</td>
<td>5</td>
<td>2 \cdot L_2</td>
<td>211002</td>
<td>&gt; 300492</td>
<td>262623</td>
</tr>
<tr>
<td>Levy No. 16 [9]</td>
<td>5</td>
<td>2 \cdot L_2</td>
<td>&gt; 1000000</td>
<td>&gt; 1000000</td>
<td>39051</td>
</tr>
<tr>
<td>Schwefel 3.7 [9]</td>
<td>5</td>
<td>2 \cdot L_2</td>
<td>33</td>
<td>33</td>
<td>33</td>
</tr>
<tr>
<td>Rosenbrock [10]</td>
<td>6</td>
<td>4 \cdot L_2</td>
<td>&gt; 500000</td>
<td>&gt; 500000</td>
<td>41327</td>
</tr>
<tr>
<td>Levy No. 10 [9]</td>
<td>6</td>
<td>4 \cdot L_2</td>
<td>56870</td>
<td>62946</td>
<td>62892</td>
</tr>
</tbody>
</table>

The improved upper bound gives better results for Lipschitz optimization than one usually used. Depending on dimensionality of problems, the number of function evaluations is from 4% to 30% smaller than with simpler bound.

Acknowledgements

The research is partially supported by Lithuanian State Science and Studies Foundation within the project B-03/2007 “Global optimization of complex systems using high performance computing and grid technologies”. We thank anonymous reviewer for constructive remarks enabling us to improve the algorithm and the paper.

References


Improved Lipschitz Bounds over Multidimensional Simplex


