GLOBAL MINIMIZATION ALGORITHMS FOR HÖLDER FUNCTIONS

D. LERA[1] and YA.D. SERGEYEV[2]

[1] *Department of Mathematics, University of Cagliari, Via Ospedale 72 09124, Cagliari, ITALY. email: lera@unica.it*

[2] *ISI-CNR, c/o DEIS, Università degli Studi della Calabria, Via Pietro Bucci 41C-42C 87036 Rende (CS), ITALY and Software Department, University of Nizhni Novgorod Gagarin Av. 23, Nizhni Novgorod, RUSSIAN FEDERATION. email: yaro@si.deis.unical.it*

Abstract.

This paper deals with the one-dimensional global optimization problem where the objective function satisfies Hölder condition over a closed interval. A direct extension of the popular Piyavskii method proposed for Lipschitz functions to Hölder optimization requires an a priori estimate of the Hölder constant and solution to an equation of degree N at each iteration. In this paper a new scheme is introduced. Three algorithms are proposed for solving one-dimensional Hölder global optimization problems. All of them work without solving equations of degree N. The case (very often arising in applications) when Hölder constant is not given a priori is considered. It is shown that local information about the objective function used inside the global procedure can accelerate the search significantly. Numerical experiments show quite promising performance of the new algorithms.

AMS subject classification: 65K05, 90C30, 65B99.

Key words: Global optimization, Hölder functions, local information, acceleration.

1 Introduction

We consider the one-dimensional global optimization problem

$$\min\{f(x) : \ x \in [a,b]\}, \qquad a,b \in \mathbf{R} \tag{1.1}$$

where the function $f(x)$ satisfies the Hölder condition

$$|f(x) - f(y)| \leq h|x-y|^{1/N}, \qquad \forall x,y \in [a,b] \tag{1.2}$$

with a constant $0 < h < \infty$. It is supposed that the value N is known and the objective function (1.1) can be represented by a "black box" procedure. This problem arises in many applications, for instance, the simple plant location problem under a uniform delivered price policy [5], infinite horizon optimization problems [6], etc.

Two cases can be examined:

i) a constant $H \geq h$ is given a priori;

ii) nothing is known about the Hölder constant h.

In the case i), Gourdin et al. [4] have proposed an extension of the Piyavskii method [7], [8] for Lipschitz optimization to Hölder optimization in order to solve a problem of globally maximization analogous to (1.1), (1.2). They consider an iterative construction of an upper-bounding function corresponding to the lower envelope of parabolic functions. At each iteration they must determine the maxima of the piecewise concave function through line search techniques, that is by solving an equation of degree N. The drawback of this approach is that for N "large" the computation of the local maxima of the upper-bounding function can be tricky. We propose a technique that at each iteration does not require the solution of nonlinear equations of degree N, so that this method turns out to be very easy to apply even with N large, and it requires a smaller computing time. Moreover we can consider the extension to the case ii) where the constant H is not available a priori. In particular we use two different procedures: one that estimates the global constant during the search, and another that adaptively estimates the local Hölder constants in different subintervals of the search region during the course of the optimization. Note that the algorithm [4] can not be extended in such a way because if an adaptive estimate of h has been changed during the k-th iteration, then it is necessary to solve $k-1$ equations of degree N in the course of this iteration.

This paper has the following structure: in Section 2 we describe a general scheme of the algorithm for solving (1.1), (1.2) and three specific algorithms belonging to the scheme. In Section 3 we illustrate the convergence conditions to the global minima for the three methods, while in Section 4 we give some numerical results. Section 5 concludes the paper.

2 The algorithms

Let us describe the general algorithm firstly in a compact form and then, by specifying STEP 2, we will give three different algorithms. By the term *trial* we will denote the evaluation of the function at any point, which we will be refereed to as a *trial point*.

General Scheme - GS -.

STEP 0. The first two trials are performed at the points $x^1 = a$ and $x^2 = b$. The point x^{k+1}, $k \geq 2$, of the current (k+1)-th iteration is chosen as follows.

STEP 1. Renumber the trial points x^1, x^2, ..., x^k of the previous iterations by subscripts so that

$$a = x_1 < \ldots < x_k = b.$$

STEP 2. Compute in a certain way the value $\bar{m}_i$ being an estimate of the Hölder constant of $f(x)$ over the interval $[x_{i-1}, x_i]$, $i = 2, ...k$, such that

$$0 < \bar{m}_i = rm_i < \infty \tag{2.1}$$

and $r > 1$ is a reliability parameter of the method. The way to calculate the value m_i will be specified in each concrete algorithm.

STEP 3. For each interval (x_{i-1}, x_i), $i = 2, ..., k$, compute

$$y_i = \frac{1}{2}(x_i + x_{i-1}) - \frac{z_i - z_{i-1}}{2rm_i(x_i - x_{i-1})^{\frac{1-N}{N}}}, \tag{2.2}$$

where $z_j = f(x_j)$, $j = 1, ..., k$.

STEP 4. Calculate, for the interval (x_{i-1}, x_i), $i = 2, ..., k$, the characteristic

$$M_i = \min\{f(x_{i-1}) - rm_i(y_i - x_{i-1})^{1/N},\ f(x_i) - rm_i(x_i - y_i)^{1/N}\}. \tag{2.3}$$

STEP 5. Find the interval (x_{t-1}, x_t) with the minimal characteristic

$$t = argmin\{M_i\ :\ 2 \le i \le k\}. \tag{2.4}$$

STEP 6. If

$$|x_t - x_{t-1}| > \varepsilon$$

where $\varepsilon > 0$ is a given search accuracy, then execute the next trial at the point

$$x^{k+1} = y_t \tag{2.5}$$

and go to STEP 1. Otherwise, calculate an estimate of the minimum as

$$f^* = \min\{z_i\ :\ 1 \le i \le k\}$$

and STOP.

We introduce now some observations with regard to the scheme above. During the course of the (k+1)-th iteration we construct an auxiliary piecewise function

$$L^k(x) = \bigcup_{i=2}^{k} l_i(x) \tag{2.6}$$

with

$$l_i(x) = \max\{f(x_{i-1}) - \bar{m}_i(x - x_{i-1})^{1/N},\ f(x_i) - \bar{m}_i(x_i - x)^{1/N}\}, \tag{2.7}$$

$x \in [x_{i-1}, x_i]$.

In [4] Gourdin et al. propose to choose as the new trial point

$$\tilde{x}^{k+1} = argmin\{l_i(x)\ :\ x \in [x_{i-1}, x_i],\ i = 2, ..., k\}.$$

Since the method [4] uses the a priori given Hölder constant h, then due to (1.2), the function $L^k(x)$ in (2.6) has the following property:

$$L^k(x) \le f(x), \quad \forall\ x \in [a, b],$$

i.e., $L^k(x)$ is a low-bounding function for $f(x)$ for every interval $[x_{i-1}, x_i],\ i = 2, ..., k$, (see Fig. 2.1) and

$$L^k(\tilde{x}^{k+1}) \leq L^k(x) \leq f(x),\ x \in [x_{i-1}, x_i],\ i = 2, ..., k.$$

This method clearly requires to calculate the solution of the following system

$$\begin{cases} A_i = f(x_{i-1}) - \bar{m}_i(\tilde{y}_i - x_{i-1})^{1/N} \\ A_i = f(x_i) - \bar{m}_i(x_i - \tilde{y}_i)^{1/N} \end{cases} \tag{2.8}$$

in order to find the "peak" value A_i and the corresponding point $\tilde{y}_i$ (see Fig. 2.1) for every interval $[x_{i-1}, x_i],\ i = 2, ..., k$ and to choose among them the point $\tilde{x}^{k+1}$.

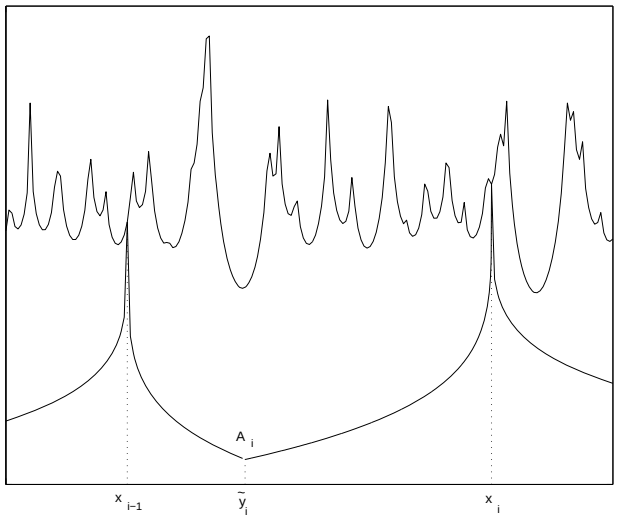

Figure 2.1: the support function

In the scheme introduced in this paper, for each interval $[x_{i-1}, x_i]$, $i = 2, ..., k$, we approximate the point $\tilde{y}_i$ by the point y_i from (2.2) found as intersection of the lines $r_{left}(x)$ and $r_{right}(x)$ (see Fig. 2.2):

$$r_{left}(x) = -\bar{m}_i(x_i - x_{i-1})^{\frac{1-N}{N}}x + \bar{m}_i(x_i - x_{i-1})^{\frac{1-N}{N}}x_{i-1} + f(x_{i-1}),$$

$$r_{right}(x) = \bar{m}_i(x_i - x_{i-1})^{\frac{1-N}{N}}x - \bar{m}_i(x_i - x_{i-1})^{\frac{1-N}{N}}x_i + f(x_i).$$

Note that the analytical expression (2.2) of y_i is very easy to calculate even with N large.

The characteristic M_i in STEP 3, related to the interval $[x_{i-1}, x_i]$, represents the minimum value of the auxiliary functions

$$l_i^-(x) = f(x_{i-1}) - \bar{m}_i(x - x_{i-1})^{1/N},\quad l_i^+(x) = f(x_i) - \bar{m}_i(x_i - x)^{1/N} \tag{2.9}$$

evaluated at the point y_i (see Fig. 2.2).

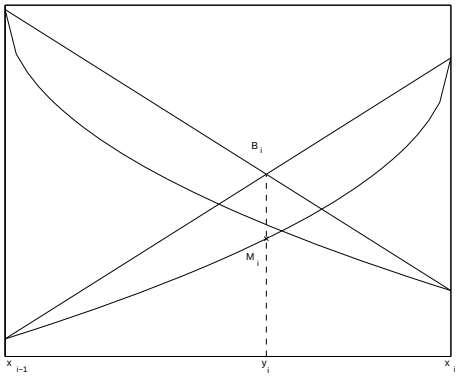

Figure 2.2: the point y_i

Now we perform three different choices of the value m_i at STEP 2 in order to get three different algorithms. The first choice is

ALGORITHM 2.1.

STEP 2.

$$m_i = h, \qquad i = 2, ..., k. \tag{2.10}$$

Here the exact value of the a priori given Hölder constant h is used. Actually it is quite difficult to know the Hölder constant "a priori", and a typical way (see [11]) to avoid this obstacle is to look for an approximation of h during the course of the search.

For the second algorithm we consider a "global" estimate of the Hölder constant, for each iteration k. More precisely we have

ALGORITHM 2.2.

STEP 2. Set

$$m_i = \max\{\xi, H^k\} \tag{2.11}$$

where $\xi > 0$ is a small number (the second parameter of the method) that takes into account our hypothesis that $f(x)$ is not constant over the interval $[a, b]$ and the value H^k is calculated as follows

$$H^k = \max\{H_i : i = 2, ..., k\} \tag{2.12}$$

with

$$H_i = \frac{|z_i - z_{i-1}|}{|x_i - x_{i-1}|^{1/N}}, \qquad i = 2, ..., k. \tag{2.13}$$

Note that if during the course of the k-th iteration $H^k = H^{k-1}$, then the auxiliary function $L^{k+1}(x)$ will differ from $L^k(x)$ only in the two subintervals obtained after splitting the interval $[x_{t-1}, x_t]$ by the point x^{k+1}. Otherwise, if $H^k > H^{k-1}$, we have to calculate completely the function $L^{k+1}(x)$.

In the previous algorithms we have used either the global a priori given exact constant h or its adaptive estimate for the construction of the auxiliary function $L^k(x)$ in the whole interval $[a, b]$. This procedure has one main drawback: the constant h does not give a lot of information about the actual slope of the function $f(x)$, in each particular interval $[x_{i-1}, x_i] \subset [a, b]$. If the local Hölder constant h_i related to the interval $[x_{i-1}, x_i]$ is significantly less than the global constant h, then the methods using only h work slowly over such intervals (see [10], [12]).

Now we propose an approach that does not use h for the whole interval $[a, b]$, but adaptively estimates the values of the local Hölder constants h_i related to the intervals $[x_{i-1}, x_i]$. The auxiliary function $L^k(x)$ is then constructed by using these local values for each interval $[x_{i-1}, x_i]$, $i = 2, ..., k$. It has been proved for a number of global optimization algorithms that using local information can significantly accelerate the search [10], [12].

Let us introduce the third algorithm:

ALGORITHM 2.3.

STEP 2. Set

$$m_i = \max\{\lambda_i, \gamma_i, \xi\} \tag{2.14}$$

with

$$\lambda_i = \max\{H_{i-1}, H_i, H_{i+1}\}, \quad i = 2, ..., k-1, \tag{2.15}$$

where H_j is from (2.13), and when $i = 1$ and $i = k$ we consider only H_1, H_2, and H_{k-1}, H_k respectively. The value

$$\gamma_i = H^k \frac{|x_i - x_{i-1}|}{X^{max}}, \tag{2.16}$$

where H^k is from (2.12) and

$$X^{max} = \max\{|x_i - x_{i-1}|^{1/N}, \ 1 = 2, ..., k\}.$$

The parameter $\xi > 0$ has the same sense as in Algorithm 2.2.

Note that in (2.14) we consider two different components, λ_i and γ_i, that take into account respectively local and global information obtained during the previous iterations. When the interval $[x_{i-1}, x_i]$ is large, the local information is not reliable and the global part γ_i increases. When $[x_{i-1}, x_i]$ is small, then the local information becomes relevant, γ_i is small, and the local component λ_i is used.

Let us also remark that by using the local estimates m_i we construct the auxiliary function $L^k(x) = \bigcup_{i=2}^{k} l_i(x)$ where $l_i(x)$ are closer to $f(x)$ than those which have been constructed by using only the global Hölder constant h, or its estimates. If in the course of the k-th iteration X^{max} was not equal to $(x_t - x_{t-1})$ and

$$\max\left\{\frac{|z_t - f(x^{k+1})|}{|x_t - x^{k+1}|^{1/N}}, \frac{|f(x^{k+1}) - z_{t-1}|}{|x^{k+1} - x_{t-1}|^{1/N}}\right\} \le H^k,$$

then the function $L^{k+1}(x)$ will differ from $L^k(x)$ (see (2.15)) only in the intervals

$$[x_{i-1}, x_i], \qquad i = t(k) - 1, \quad t(k), \quad t(k) + 1, \quad t(k) + 2,$$

otherwise $L^{k+1}(x)$ is completely reconstructed.

3 Convergence properties

In this section we investigate convergence properties of the three algorithms introduced above by studying an infinite trial sequence $\{x^k\}$ generated by an algorithm belonging to the general scheme GS for solving the problem (1.1), (1.2). First of all, we obtain a simple result regarding the properties of the characteristic M_i.

LEMMA 3.1. *If $\bar{m}_i > h_i$, where h_i is the local Hölder constant related to the interval $[x_{i-1}, x_i]$, then*

$$M_i < f(x), \qquad x \in [x_{i-1}, x_i]. \tag{3.1}$$

PROOF. If $\bar{m}_i > h_i$, then, due to (1.2), (2.7), and (2.9), the function

$$l_i(x) = \max\{l_i^-(x), l_i^+(x)\}$$

is a low-bounding function for $f(x)$ over the interval $[x_{i-1}, x_i]$. Moreover, since $r > 1$, it follows

$$l_i(x) < f(x), \qquad x \in (x_{i-1}, x_i]).$$

The function $l_i^-(x)$ is strictly decreasing on $[x_{i-1}, x_i]$ and $l_i^+(x)$ is strictly increasing on this interval. Thus, it follows

$$\min\{l_i^-(x), l_i^+(x)\} \le \min\{l_i(x) : x \in [x_{i-1}, x_i]\}, \qquad \forall x \in [x_{i-1}, x_i].$$

Particularly, this is true for $x = y_i$, where y_i is from (2.2). To conclude the proof it is sufficiently to recall that, due to (2.2), (2.3), and (2.7),

$$M_i = \min\{l_i^-(y_i), l_i^+(y_i)\}$$

and $y_i \in (x_{i-1}, x_i)$. □

Let us now return to the trial sequence $\{x^k\}$ generated by GS for solving the problem (1.1), (1.2). We need the following definition.

DEFINITION 3.1. *The convergence to a point $x' \in (a,b)$ is said bilateral if there exist two subsequences of $\{x^k\}$ converging to x' one from the left, the other from the right.*

THEOREM 3.2. *Let x' be any limit point of $\{x^k\}$ such that $x' \neq a$, $x' \neq b$. Then the convergence to x' is bilateral.*

PROOF. Consider the interval $[x_{t-1}, x_t]$ determined by (2.4) at the $(k+1)$-th iteration. By (2.2) and (2.5) we have that the new trial point x^{k+1} divides the interval $[x_{t-1}, x_t]$ into the subintervals $[x_{t-1}, x^{k+1}]$ and $[x^{k+1}, x_t]$. Considering (2.2), (2.10), (2.11), and (2.14) we can write

$$\begin{aligned}
\max(x^{k+1} - x_{t-1}, x_t - x^{k+1}) &\leq 0.5\left[x_t - x_{t-1} + \frac{|z_t - z_{t-1}|}{(x_t - x_{t-1})^{1/N}}\frac{(x_t - x_{t-1})}{rm_t}\right] \\
&\leq 0.5\left[x_t - x_{t-1} + \frac{1}{r}(x_t - x_{t-1})\right] \\
&\leq 0.5(1 + \frac{1}{r})(x_t - x_{t-1}). \qquad (3.2)
\end{aligned}$$

Consider now an interval $[x_{s-1}, x_s]$, $s = s(k)$, such that $x' \in [x_{s-1}, x_s]$; then, because x' is a limit point of $\{x^k\}$ and using (2.2), (2.4), (2.5), and (3.2), we obtain

$$\lim_{k\to\infty}(x_{s(k)} - x_{s(k)-1}) = 0. \qquad (3.3)$$

If $x' \notin \{x^k\}$ the subsequences $\{x_{s(k)-1}\}$ and $\{x_{s(k)}\}$ are the ones we are looking for, and the theorem has been proved. Suppose now that $x' \in \{x^k\}$ and that the convergence to x' is not bilateral, i.e. no sequence converging to x' from the left exists. In this case there exist integers $q, n > 0$, such that $x' = x^q$ and for any iteration number $k > \max(q, n)$ no trials will fall into the interval $[x^n, x^q] = [x_{j(k)-1}, x_{j(k)}]$. For the value M_j of this interval we have:

$$M_j = \min\{z_{j-1} - \bar{m_j}(y_j - x_{j-1})^{\frac{1}{N}}, f(x') - \bar{m_j}(x_j - y_j)^{\frac{1}{N}}\} \qquad (3.4)$$

that is

$$M_j \leq f(x') - \underbrace{\bar{m_j}(x_j - y_j)^{\frac{1}{N}}}_{>0} < f(x'). \qquad (3.5)$$

On the other hand, it follows from (2.1) and (3.3) that

$$\lim_{k\to\infty} M_{s(k)} = f(x') \qquad (3.6)$$

thus, for a sufficiently large iteration number k the inequality

$$M_{j(k)} < M_{s(k)} \qquad (3.7)$$

is satisfied. This means that, by (2.4) and (2.5), a trial will fall into the interval $[x^n, x^q]$ which contradicts our assumption that there is no subsequence converging to x' from the left. In the same way we can consider the case when there is

no subsequence converging to x' from the right. Hence the convergence to x' is bilateral. □

COROLLARY 3.3. *For all trial points x^k, it follows $f(x^k) \geq f(x')$, $k \geq 1$.*
PROOF. Suppose that exists a point x^q such that

$$z^q = f(x^q) < f(x'). \tag{3.8}$$

Consider the value M_j of the interval $[x_{j-1}, x_j]$ where $x_j = x^q$. We have:

$$M_j = \min\{z_{j-1} - \bar{m}_j(y_j - x_{j-1})^{\frac{1}{N}}, z_j - \bar{m}_j(x_j - y_j)^{\frac{1}{N}}\},$$

$$M_j < \min\{z_{j-1}, z_j\} < f(x').$$

Again, from (3.6) and (3.8) the inequality (3.7) holds. By (2.4) and (2.5) this fact contradicts the assumption that x' is a limit point of $\{x^k\}$. Thus $f(x^q) \geq f(x')$ and the corollary has been proved. □

COROLLARY 3.4. *If another limit point $x" \neq x'$ exists, then $f(x") = f(x')$.*
PROOF. Follows directly from Corollary 3.3. □

COROLLARY 3.5. *If the function $f(x)$ has a finite number of local minima in $[a, b]$, then the point x' is locally optimal.*
PROOF. If x' is not a local minimizer then, taking into account the bilateral convergence of $\{x^k\}$ to x' and the fact that $f(x)$ has a finite number of local minima in $[a, b]$, a point w such that $f(w) < f(x')$ will be found. But this is impossible by Corollary 3.3. □

Now let us give sufficient conditions for the convergence to the global minima.
THEOREM 3.6. *Let x^* be a global minimizer of $f(x)$. If there exists an iteration number k^* such that for all $k > k^*$ the inequality*

$$\bar{m}_{j(k)} > h_{j(k)} \tag{3.9}$$

holds, where $h_{j(k)}$ is the Hölder constant for the interval $[x_{j(k)-1}, x_{j(k)}]$, i.e.,

$$|f(x) - f(y)| \leq h_{j(k)}|x - y|^{1/N}, \quad \forall x, y \in [x_{j(k)-1}, x_{j(k)}] \tag{3.10}$$

and the interval $[x_{j(k)-1}, x_{j(k)}]$ is such that $x^* \in [x_{j(k)-1}, x_{j(k)}]$, then x^* is a limit point of $\{x^k\}$.
PROOF. Suppose that x^* is not a limit point of the sequence $\{x^k\}$ and a point $x' \neq x^*$ is a limit point of $\{x^k\}$. Then there exists an iteration number n such that for all $k \geq n$

$$x^{k+1} \notin [x_{j-1}, x_j], \qquad j = j(k).$$

Lemma 3.1 and (3.9) imply

$$M_j < f(x^*). \tag{3.11}$$

However, since x^* is a global minimizer, the inequality

$$f(x^*) \leq f(x') \tag{3.12}$$

holds. Thus, considering (3.11), (3.6), and (3.12) together with the decision rules of the algorithm, we conclude that a trial will fall into the interval $[x_{j-1}, x_j]$. This fact contradicts our assumption and proves that x^* is a limit point of the sequence $\{x^k\}$. □

COROLLARY 3.7. *If the conditions of Theorem 3.6 are satisfied, then all limit points of $\{x^k\}$ are global minimizers of $f(x)$.*

PROOF. The result follows immediately from Corollary 3.4. □

THEOREM 3.8. *For every function $f(x)$ satisfying (1.2) with $h < \infty$ there exists r^* such that for all $r > r^*$ the algorithms 2.2 and 2.3 determine all global minimizers of the function $f(x)$ over the search interval $[a, b]$.*

PROOF. Since $h < \infty$ and any value of r can be chosen in the methods 2.2 and 2.3, it follows that there exists r^* such that condition (3.9) will be satisfied for all global minimizers for $r > r^*$. This fact, due to Theorem 3.6, proves the theorem. □

4 Numerical experiments

In this section we report some numerical results obtained by comparing the algorithms 2.1, 2.2, and 2.3 with the method from [4]. Here in after we denote the method from [4] by GJE and the algorithms 2.1, 2.2, and 2.3 from Section 2 by $A1$, $A2$, and $A3$, respectively. Three series of experiments have been done in order to compare the methods.

In the first and second series of experiments a set of eight functions described in [4] have been used (see Table 1). Since the GJE algorithm requires, at each iteration, the solution to the system (2.8) in order to find the "peak" point $(\tilde{y}_i, A_i)$ (see Fig. 2.1) we distinguish two cases. In the first series we use the integers $N = 2, 3$, and 4 because it is possible to use explicit expressions for the coordinates of the intersection point $(\tilde{y}_i, A_i)$ (see [4]). The second series of experiments considers the case of fractional N.

Table 2 contains the number of trials executed by the algorithms with accuracy $\varepsilon = 10^{-4}(b-a)$ (this accuracy is used in all series of experiments). The constants $H \geq h$ from [4] have been used in GJE and $A1$. Parameters of $A2$ and $A3$ were $\xi = 10^{-8}$ and $r = 1.1$. In this case, all the global minimizers have been found by all the methods. A strong acceleration has been obtained by the algorithm $A3$ in comparison with the other methods.

In Table 3 we present numerical results for the problems from Table 1 with fractional values of N. In this case, in GJE the system (2.8) should be solved by using a line search technique (see [4]) during each iteration. The following methods have been used for this goal:

i) the routine FSOLVE from the Optimization Toolbox of MATLAB 5.3;

ii) the routine NEWT of Numerical Recipes [9] that combines the Newton's method for solving nonlinear equations with a globally convergent strategy that will guarantee progress towards the solution at each iteration even if the initial guess is not sufficiently close to the root.

no.	*function*	*interval*	*optimal points*
1	$x^6 - 15x^4 + 27x^2 + 250$	$[-4, 4]$	3.0
2	$(x^2 - 5x + 6)/(x^2 + 1)$	$[-5, 5]$	2.414213
3	$\begin{cases} (x-2)^2 & \text{if } x \leq 3 \\ 2ln(x-2)+1 & \text{otherwise} \end{cases}$	$[0, 6]$	2.0
4	$\begin{cases} -\sqrt{2x - x^2} & \text{if } x \leq 2 \\ -\sqrt{-x^2 + 8x - 12} & \text{otherwise} \end{cases}$	$[0, 6]$	4.0
5	$(3x - 1.4)\sin 18x$	$[0, 1]$	0.966085
6	$2(x-3)^2 + e^{x^2/2}$	$[-3, 3]$	1.590717
7	$-\sum_{k=1}^{5} k\sin[(k+1)x + k]$	$[-10, 10]$	-6.774576 -0.49139 5.791785
8	$-\sum_{k=1}^{5} k\cos[(k+1)x + k]$	$[-10, 10]$	-7.083506 -0.8003 5.48286

Table 1

These methods have been chosen because they can be easily found by a final user. Unfortunately, our experience with both algorithms has shown that solving the system (2.8) can be a problem itself. Particularly, we note that, when N increases, the two curves l_i^- and l_i^+ from (2.9) tend to flatten (see Figs. 4.1, 4.2) and if the intersection point $(\tilde{y}_i, A_i)$ is close to the boundaries of the subinterval $[x_{i-1}, x_i]$, then the system (2.8) can be difficult to solve. In some cases the methods looking for the roots of the system do not converge to the solution. For example, Fig. 4.2 presents the case when the point (denoted by "x") which approximates the root is obtained out of the search interval $[x_{i-1}, x_i]$). Thus, the system (2.8) is not solved and, as a consequence, the algorithm GJE does not find the global minima of the objective function. These cases are shown in Table 3 by "–".

function	*N*	*GJE*	*A1*	*A2*	*A3*
	2	5569	5477	1477	216
1	3	11325	11075	2368	302
	4	12673	15841	2615	397
	2	4517	5605	2270	265
2	3	5890	7908	3801	343
	4	7027	8945	3486	474
	2	1683	1515	1249	112
3	3	2931	2521	1574	167
	4	3867	3162	1697	196
	2	4077	4371	1568	156
4	3	4640	7605	2023	254
	4	6286	9453	2451	309
	2	1160	1091	279	44
5	3	3169	2823	367	68
	4	4777	4188	424	81
	2	2879	2532	1761	201
6	3	5191	4200	3186	318
	4	5370	5093	4165	421
	2	4273	4478	580	198
7	3	8682	11942	710	234
	4	10304	15996	756	286
	2	3336	3565	380	180
8	3	8489	9516	312	202
	4	8724	15538	550	220
	2	3436	3579	1195	171
Average	3	6289	7198	1792	236
	4	7378	9777	2018	298

Table 2

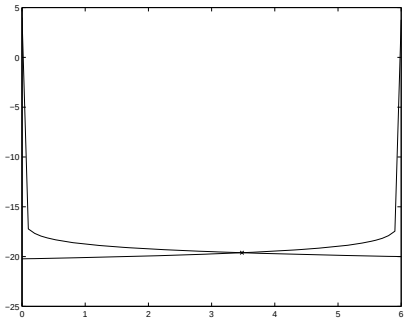

Figure 4.1: The two curves l_i^- and l_i^+

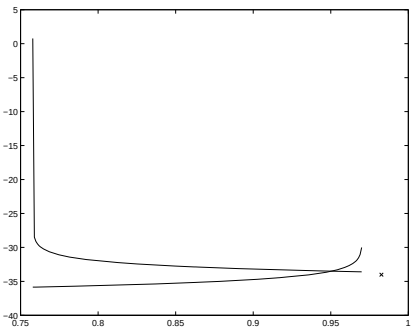

Figure 4.2: No convergence

The numerical experiments described in Table 3 have been executed with the

following parameters. The constants $H \geq h$ from [4] have been used in GJE and $A1$. Parameter $\xi = 10^{-8}$ has been used in $A2$ and $A3$. The reliability parameter $r = 1.5$ for $A2$ and $r = 1.1$ for $A3$. All global minimizers have been found by the algorithms GJE, $A1$, and $A3$. Note that the parameter r influences the reliability of the methods $A2$ and $A3$. For example, the algorithm $A2$ has found only one global minimizer in the experiments marked by "*". The value $r = 3.5$ allows us to find all global minimizers.

The third series of experiments (see Table 4) has been done with the following function (see Fig. 4.3):

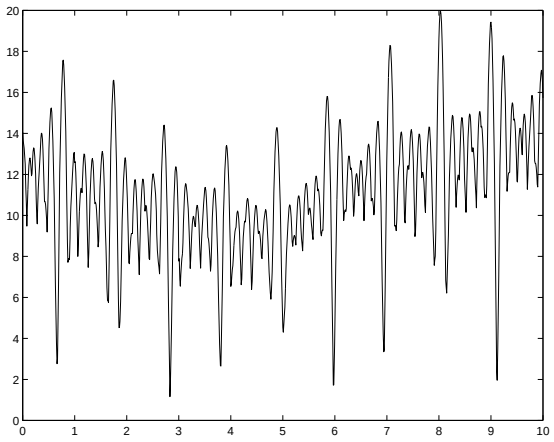

Figure 4.3: The function $F_N(x)$, with $N = 5$

$$F_N(x) = \sum_{k=1}^{5} k|\sin((3k+1)x+k)||x-k|^{1/N}, \quad x \in [0,10]. \tag{4.1}$$

It is a hölderian function, with exponent $1/N$, $N > 1$, that is

$$|F_N(x) - F_N(y)| \leq h|x-y|^{1/N}, \qquad \forall x, y \in [0,10].$$

It can be shown (see [3]) that the constant

$$H_N = 15 + \sum_{k=1}^{5} k 2^{1-1/N}(3k+1)^{1/N}(10-k)^{1/N} \geq h.$$

In this series of experiments a new algorithm $A0$ has been used to show efficiency of the choice of the point y_i from (2.2). The method $A0$ works as $A1$ but in STEP 3 and STEP 4

$$y_i = 0.5(x_i + x_{i-1}) \tag{4.2}$$

and

$$M_i = \min\{f(x_{i-1}) - rm_i(y_i - x_{i-1})^{1/N},\ f(x_i) - rm_i(x_i - y_i)^{1/N}\}$$

where y_i is from (4.2).

function	*N*	*GJE*	*A1*	*A2*	*A3*
	4/3	1913	1923	1053	131
1	53/2	10483	15899	2972	581
	100/3	10195	15757	2108	601
	4/3	2341	2329	1484	144
2	53/2	6883	9243	4215	934
	100/3	7201	8671	4090	958
	4/3	705	680	649	58
3	53/2	−−	5921	2207	388
	100/3	−−	5399	2023	407
	4/3	1213	1768	1025	84
4	53/2	6763	10057	3073	592
	100/3	6609	9458	2828	609
	4/3	397	381	278	45
5	53/2	5895	8056	725	131
	100/3	4127	6783	667	137
	4/3	1160	1108	1664	109
6	53/2	−−	5050	4491	783
	100/3	−−	4699	4196	801
	4/3	730	722	473	145
7	53/2	9833	16169	103*	441
	100/3	10078	15982	154*	447
	4/3	557	549	378	151
8	53/2	9617	16083	94*	272
	100/3	9094	15617	153*	373
	4/3	1127	1182	875	108
Average	53/2	−−	10809	2235	515
	100/3	−−	10295	2027	541

Table 3

The parameters of the methods have been chosen as follows: H_N, for any given N, has been used in the GJE, $A0$, and $A1$ algorithms as an overestimate for h. The parameter $\xi = 10^{-8}$ has been used in $A2$ and $A3$. Due to Theorem 3.8, every hölderian function optimized by $A1$ or $A2$ has a crucial value r^* of the parameter r. Thus, different values of r have been chosen for different values of N. In the method $A3$, r was equal to 1.3 if $N = 5, 10, 20$ and equal to 1.7 in the cases $N = 40, 60, 80, 100$. In the method $A2$ we have chosen $r = 1.3$ for the case $N = 5$, $r = 1.7$ for $N = 10$, $r = 2.8$ for $N = 20$ and $r = 9.3$ if $N = 40, 60, 80$; the result for $N = 100$ has been obtained with $r = 15$.

The algorithms $A2$ and $A3$ have found good estimates of the global solution in all the cases. The methods $A0$ and $A1$ have made the same for $N = 5, 10, 20$. It can be seen that $A1$ outperforms $A0$. For $N = 40, 60, 80, 100$, these methods stop after a few iterations in neighborhoods of local minimizers because the used

accuracy ε was not small enough in order to find the global solution. Augmenting accuracy allows to locate the global minimizer. These cases are shown in Table 4 by "*". The symbol "–" has the same meaning as in Table 3.

5 Conclusion

In this paper the one-dimensional global optimization problem where the objective function satisfies Hölder condition over a closed interval has been considered. Recently, Gourdin et al. (see [4]) have introduced an extension of the Piyavskii method proposed for Lipschitz functions to Hölder optimization. It uses an a priori given Hölder constant h and requires solution to an equation of degree N at each iteration.

In this paper a new scheme for solving this problem has been introduced. Three algorithms have been proposed. All of them work without solving equations of degree N by using a simpler procedure. The first algorithm uses an a priori given Hölder constant h. Then, the case (very often arising in applications) when Hölder constant is not given a priori was considered. In this case the second proposed method adaptively estimates Hölder constant during the search.

It was shown that local information about the objective function used inside the global procedure can accelerate the search significantly. In fact, the third proposed algorithm introduces a procedure for adaptive estimating local Hölder constants over different subregions of the search domain.

Numerical experiments show quite promising performance of the new algorithms adaptively estimating Hölder constants.

N	*Optimal point*	*Optimal value*	H_N	*GJE*	*A0*	*A1*	*A2*	*A3*
5	2.82909266	1.15879294	77	1886	2530	1995	258	103
10	2.83390034	1.15176044	58	208	1761	1295	82	85
20	2.83390034	1.14960372	51	- -	760	518	171	102
40	2.83390034	1.14946908	48	- -	220*	69*	949	156
60	2.83390034	1.14956783	47	- -	53*	87*	581	161
80	2.83390034	1.14964447	47	- -	41*	94*	241	169
100	2.83390034	1.14969913	47	- -	34*	71*	261	167

Table 4

REFERENCES

1. Butz A.R., *Space filling curves in mathematical programming*, Inform. Control 12(4), 314-330 (1968).
2. Butz A.R., *Alternative algorithm for Hilbert's space filling curve*, IEEE Trans. Computers C-20(4), 424-426 (1971).
3. Gelfand I., Raikov D., and Shilov G., *Commutative Normed Rings*, AMS Chelsea Publishing, (1991).
4. Gourdin E., Jaumard B., and Ellaia R., *Global optimization of Hölder functions*, Journal of Global Optimization 8, 323-348 (1996).
5. Hanjoul P., Hansen P., Peeters D., and Thisse J.F., *Uncapacitated plant location under alternative space price policies*, Management Science 36, 41-47 (1990).
6. Kiatsupaibul S. and Smith, R.L. *On the solution of infinite horizon optimization problems through global optimization algorithms.* Tech. Report 98-19, DIOE, University of Michigan, Ann Arbor, (1998).
7. Piyavskii S.A., *An algorithm for finding the absolute minimum of a function*, Theory of Optimal Solutions 2, Kiev, IK AN USSR (in Russian), 13-24 (1967).
8. Piyavskii S.A., *An algorithm for finding the absolute extremum of a function*, USSR Comp. Maths. Math. Phys. 12, 57-67 (1972).
9. Press W.H., Teukolsky S.A., Vettering W.T., and Flannery B.P., *Numerical Recipes in Fortran, The Art of Scienific Computing*, Second Edition, Cambridge University Press, (1992).
10. Sergeyev Ya.D., *An information global optimization algorithm with local tuning*, SIAM Journal on Optimization 5(4), 858-870 (1995).
11. Strongin R.G., *Numerical Methods in Multiextremal Problems*, Nauka, Moskow (1978). (In Russian).
12. Strongin R.G., Sergeyev Ya.D., *Global optimization with non-convex constraints: Sequential and parallel algorithms*, Kluwer Academic Publishers, Dordrecht, 2000.