2008 International Symposium on Nonlinear Theory and its Applications
NOLTA'08, Budapest, Hungary, September 7-10, 2008

An Algorithm for Globally Minimizing a Function with Many Local Minimal Values whose Sequence is Unimodal

Hideo KANEMITSU†, Hideaki KONNO† and Masaaki MIYAKOSHI‡

†Hakodate Campus, Hokkaido University of Education, Hakodate 040-8567, Japan
‡Graduate School of Engineering, Hokkaido University, Sapporo 060-8628, Japan
Email: † {hkanemt, konno}@cc.hokkyodai.ac.jp, ‡ miyakosi@main.eng.hokudai.ac.jp

Abstract—We present an algorithm for finding the global minimum of a function with many isolated local minimal function values whose (strict lower) sequence is unimodal. First, we define a univariate function such that local minimal values of the function are a unimodal sequence. Next, we introduce a new algorithm for finding the global minimum of these functions and investigate the convergence of the algorithm. We also present an algorithm for finding the global minimum of multivariate functions. We show using a numerical example that the algorithm effectively finds the global minimum with only a few function evaluations.

1. Introduction

In recent years, many deterministic and stochastic algorithms have been proposed for solving a global optimization (minimization) problem (P) of a real valued function f of n-variables with bounded constraints:

$$\begin{aligned} \text{min. } & f(\boldsymbol{x}) \equiv f(x_1, x_2, \ldots, x_n),\ \boldsymbol{x} \in D^n \\ & D^n = \{\, \boldsymbol{x} \in \boldsymbol{R}^n \,|\, x_j \in [a_j, b_j] \,\}\,. \end{aligned} \tag{P}$$

It is assumed that the function $f(\boldsymbol{x}) \in C^2$ has a finite number of isolated local minima $\boldsymbol{x}_k^* \in D^n$ $(k = 1, 2, \ldots, M)$.

Deterministic methods [3] repeatedly divide a given region into subregions, select a subregion in which a global optimum is included, and give a guarantee of successfully finding the global optimum under highly restrictive conditions on objective functions (for example, Lipschitz continuity with a known Lipschitz constant). On the other hand, stochastic methods [4] involve random sampling or a combination of random sampling and local search. The latter algorithms, called *multistart based* methods, can find the global optimum with a high degree of accuracy.

However, searching spaces of those methods exponentially increase with increase in the number of dimensions in the problem (P). This phenomenon, known as the "curse of dimensionality", led to the abandonment of those search methods in favour of ones using some *a priori* knowledge or *priori* structure of the function f.

In this paper, we consider a special structure of functions f such that the sequence of local minimal function values is unimodal. The basic concept of the similar type of functions has been described in our previous paper [6]. However, the type of those functions was not defined.

The purpose of this paper is to define of the type of those univariate functions, to propose a more effective algorithm, and prove convergence of the algorithm.

The remainder of the paper is organized as follows. A problem, definitions and an example of the problem are given as preliminaries in section 2. In section 3, algorithms for univariate functions, their convergence properties and a numerical example are presented. An algorithm for multivariate functions using one of algorithms for univariate functions and the results of a numerical experiment are presented in section 4. Finally, concluding remarks are given.

2. Preliminary

2.1. Problem and Assumption

In this section and the next section, we consider a univariate minimization problem (P1):

$$\text{min. } f(x), \quad x \in D^1 = [a_1, b_1] \subset \boldsymbol{R}. \tag{P1}$$

Similar to the assumption of the problem (P), f is a twice continuous function, and all local minima of f in $[a, b]$ are isolated. These minima are denoted by $x_*^1 < x_*^2 < \cdots < x_*^M$, and these function values are denoted by $f_i^* \equiv f(x_*^i)$ $(i = 1, 2, \ldots, M)$.

2.2. Definitions

Definition 1 In the problem (P1), it is said to be a *strictly lower unimodal sequence* (hereafter called a *unimodal sequence*) in the sequence of local minimal function values if there exists $k \in [2, M-1]$ for such that

$$\left\{ \begin{array}{l} x_*^1 < x_*^2 < \cdots < x_*^{k-1} < x_*^k < x_*^{k+1} < \cdots < x_*^M \\ f_*^1 > f_*^2 > \cdots > f_*^{k-1} > f_*^k < f_*^{k+1} < \cdots < f_*^M. \end{array} \right. \tag{1}$$

From the above definition, note that the function f has a unique global minimal value $f^{**} \equiv f_*^k$ at the point x_*^k.

Definition 2 Since we deal with a minimization problem, we will call the function $u(x)$ a *lower unimodal function* (hereafter called a *unimodal function*) with the unique minimum x_{**} in the closed interval $[a_1, b_1]$ if the following condition holds.

$$\begin{cases} a_1 \le \forall x^1 < \forall x^2 \le x_{**} \Longrightarrow u(x^1) > u(x^2) \\ x_{**} \le \forall x^1 < \forall x^2 \le b_1 \Longrightarrow u(x^1) < u(x^2). \end{cases} \quad (2)$$

Definition 3 The *unimodal region* [5] $R_u(x_*^i)$ of a local minimum x_*^i of a function f is defined as the maximum region (interval) such that the function f is unimodal:

$$R_u(x_*^i) = \{ [a_*^i, b_*^i] \mid \text{maximum interval } [a_*^i, b_*^i], \text{ s.t. } f(x) \text{ is unimodal in } [a^i, b^i] \} \}. \quad (3)$$

Definition 4 The radius $r(x_*^i)$ of the unimodal region [5] $R_u(x_*^i) = [a_*^i, b_*^i]$ at a local minimum x_*^i is defined as the minimum of two distances from the local minimum x_*^i to both sides a_*^i and b_*^i of the interval:

$$r(x_*^i) = \min\{ x_*^i - a_*^i,\ b_*^i - x_*^i \}. \quad (4)$$

2.3. A Simple Problem and an Example

For simplifying investigation, we focus on the following problem (P1S) as a special case of the problem (P1):

$$\text{min. } f(x) = u(x) + p(x), \quad x \in D^1 = [a_1, b_1] \quad \text{(P1S)}$$

where $u(x)$ is a unimodal function, and $p(x)$ is a periodic function with period T^p and local minima. In order for $f(x)$ to have many local minima, we assume the following conditions hold for $p(x)$ and $u(x)$.

Condition 1. The period T^p of $p(x)$ is much less than the width $b_1 - a_1$ of the interval $D^1 = [a_1, b_1]$, that is

$$T^p << b_1 - a_1. \quad (5)$$

Condition 2. The maximum absolute value of derivatives of $u(x)$ is less than that of $p(x)$, that is

$$\max_{x \in [a_1, b_1]} \{ |\mathrm{d}u(x)/\mathrm{d}x| \} < \max_{x \in [a_1, b_1]} \{ |\mathrm{d}p(x)/\mathrm{d}x| \}. \quad (6)$$

Condition 3. $u(x)$ has the unique minimum x_{**}, and the point x_{**} also becomes one of the minima of $p(x)$.

From $x_{**} = \operatorname{argmin} u(x) = \operatorname{argmin} p(x)$ in **Condition 3**, the global minimum of $f(x) = u(x) + p(x)$ is x_{**}.

For example, let $u(x) = 0.1x^2 + 0.01$, $p(x) = -0.01\cos(40\pi x)$ and $D^1 = [-1, 1]$. We present the following univariate problem (P1SE):

$$\begin{aligned} \text{min. } \quad f(x) &\equiv u(x) + p(x) \\ &= 0.1x^2 + 0.01 - 0.01\cos(40\pi x), \\ & \quad x \in D^1 = [-1, 1]. \end{aligned} \quad \text{(P1SE)}$$

The problem satisfies **Condition 1** and **Condition 2**, and the function has 41 local minima ($\approx (b_1 - a_1)/T^p = 2/0.05 = 40$) in the interval $[-1, 1]$. Since the unique minimum of $u(x)$ is 0 and 0 is also one minimum of $p(x)$, **Condition 3** is satisfied with $x_{**} = \operatorname{argmin}_{x \in [-1,1]} f(x) = 0$ [6].

3. Algorithm for Univariate Functions

3.1. Outline of the previous algorithm

The idea of our previously proposed algorithm [6] is to use a two-stage minimizer, 1) a *large-step minimizer* and 2) a *small-step local minimizer*, in each iteration. An outline of the two steps is:

1) The *large-step minimizer* generates new points $x^0, x^1, \ldots$ such that any two points are included in the different unimodal region, and then
2) the *small-step local minimizer* finds a local minimum x_*^i in a unimodal region $R_u(x_*^i)$ from a starting point $x^k \in R_u(x_*^i)$ generated by the *large-step minimizer*.

The condition of generated points $x^0, x^1, \ldots$ at the large-step minimizer 1) can be formulated as follows:

$$\begin{aligned} \forall x^m \ne \forall x^n \text{ and } x^m \in R_u(x_*^i),\ x^n \in R_u(x_*^j) \\ \Longrightarrow \quad R_u(x_*^i) \cap R_u(x_*^j) = \emptyset. \end{aligned} \quad (7)$$

The small-step local minimizer of 2) that finds a local minimum x^* is realized by a procedure $ML(x^k, \overline{\delta})$ with a starting point x^k and small-step $\overline{\delta}$ such that:

$$\text{for } x^k \in R_u(x_*^i),\ x_*^{(k)} \leftarrow ML(x^k, \overline{\delta}) \Longrightarrow x_*^{(k)} = x_*^i. \quad (8)$$

By the above investigation, it is concluded that points $x^0, x^1, \ldots$ generated by the large-step minimizer converge to different local minima $x_*^{(0)}, x_*^{(1)}, \ldots$ by the small-step local minimizer, that is

$$x^m \ne x^n,\ x_*^{(m)} \leftarrow ML(x^m, \overline{\delta}),\ x_*^{(n)} \leftarrow ML(x^n, \overline{\delta}) \Longrightarrow x_*^{(m)} \ne x_*^{(n)}. \quad (9)$$

From equation (1) as the definition of a unimodal sequence, for three points $x_*^{(p)} > x_*^{(q)} > x_*^{(r)}$, the following condition of encloseing the global minimum x_{**} holds.

$$f(x_*^{(p)}) > f(x_*^{(q)}) < f(x_*^{(r)}) \Longrightarrow x_{**} \in (x_*^{(p)}, x_*^{(r)}) \quad (10)$$

An outline of the previous algorithm is as follows.

Step 1. Bracketing the global minimum x_{**} by an interval $[x_*^{(p)}, x_*^{(r)}]$ such that $f(x_*^{(p)}) > f(x_*^{(q)}) < f(x_*^{(r)})$.

Step 2. Reducing the interval $[x_*^{(p)}, x_*^{(r)}]$ such that $x_{**} \in [x_*^{(p)}, x_*^{(r)}]$ until the following stop condition holds.

$$x_{**} \in (x_*^{(p)}, x_*^{(r)}) \text{ and } \forall x_*^j \ne x_{**},\ x_*^j \notin (x_*^{(p)}, x_*^{(r)}) \quad (11)$$

In each step, a new interval $[x_*^{(p)}, x_*^{(r)}]$ is determined using the *large-step minimizer* and *small-step local minimizer*.

3.2. Small-step Local Minimizer

For an algorithm satisfying equation (8), it is desirable that the sequence $\{x^{(k)}\}$ of points always be included in *the unimodal region* $R_u(x_*)$. In order to satisfy this condition (8), the step length $\delta^{(k)}$ that generates an ordinary local minimizer is restricted to $\overline{\delta}$ as follows:

$$\begin{cases} \overline{\delta}^{(k)} \leftarrow \min\{\delta^{(k)}, \overline{\delta}\}; \\ x^{(k+1)} \leftarrow x^{(k)} + \overline{\delta}^{(k)}, \quad (k = 0, 1, 2, \ldots), \end{cases} \quad \text{(d)}$$

where $\overline{\delta}$ is set to

$$\overline{\delta} \leq \frac{1}{2} r_{**}, \quad \text{where } r_{**} = \min_{1 \leq i \leq M} r(x_*^i). \tag{12}$$

Thus, the specification of a small-step minimizer algorithm $MLuf$ that finds the local minimum $x_*^{(i)}$ and its function value $f_*^{(i)}$ from a starting point x^i and its function value f^i with an upper limit of step $\overline{\delta}$ and a tolerance ε is

$$(f_*^{(i)},\ x_*^{(i)}) \leftarrow MLuf(\ f^i,\ x^i,\ \overline{\delta},\ \varepsilon\ ).$$

3.3. Modification of the previous algorithm and its convergence

3.3.1. Step 1 (Bracketing the global minimum)

If points $x^0, x^1 \ldots, x^N$ $(N+1 < M)$ are determined:

$$\begin{cases} a_1 \leq x^0 < x^1 < \cdots < x^N \leq b_1 \\ x^{i+1} - x^i = \Delta > T^p \quad (i = 0, 1 \ldots, N-1) \end{cases}$$

by the large-step minimizer, then from (9) these local minima by the small-step local minimizer are satisfied as follows:

$$a_1 < x_*^{(0)} < x_*^{(1)} < \cdots < x_*^{(N)} < b_1. \tag{13}$$

From the above relationship and (1), the following encloseing conditions of **Step 1** in **3.2** hold:

$$\begin{cases} f(x_*^{(0)}) \leq f(x_*^{(1)}) \implies x^{**} \in [a_1, x_*^{(1)}] \\ f(x_*^{(i)}) > f(x_*^{(i+1)}) \text{ and } f(x_*^{(i+1)}) \leq f(x_*^{(i+2)}) \\ \qquad\qquad \implies x^{**} \in [x_*^{(i)}, x_*^{(i+2)}] \\ f(x_*^{(0)}) > \cdots > f(x_*^{(N)}) \implies x^{**} \in [x_*^{(N-1)}, b_1] \end{cases} \tag{14}$$

This algorithm uses all conditions for enclosing a global minimum, while the previous algorithm used only the second condition of the above conditions.

It is inefficient that the procedure $ML(\cdot)$ is always performs for each generated point x^i for checking the conditions (14). To overcome this problem, we usually apply the following moderated conditions instead of (14):

$$\begin{cases} f(x^0) \leq f(x^1) & \cdots\ 1) \\ f(x^i) > f(x^{i+1}) \text{ and } f(x^{i+1}) \leq f(x^{i+2}) & \cdots\ 2) \\ f(x^0) > \cdots > f(x^N). & \cdots\ 3) \end{cases} \tag{14'}$$

If the above conditions hold, then more reliable conditions (14) are applied.

3.3.2. Step 2 (Reducing the encloseing interval)

If the stop condition (11) holds in the reducing interval of **Step 2**, this implies that the interval is the smallest interval $[x_*^{(p)}, x_*^{(r)}]$ including the global minimum x_{**}. From the above fact and equation (1), we have

$$\begin{cases} x_{**} \in [x_*^{(p)}, x_*^{(r)}] \iff x_*^k \in [x_*^{k-1}, x_*^{k+1}] \\ f(x_*^{(p)}) > f(x_{**}) < f(x_*^{(r)}) \iff f_*^{k-1} > f_*^k < f_*^{k+1}. \end{cases}$$

Therefore, the following relationships hold.

$$\begin{cases} x_{**} = x_*^k, \quad x_*^{(p)} = x_*^{k-1}, \quad x_*^{(r)} = x_*^{k+1} \\ f_{**} = x_*^k, \quad f_*^{(p)} = f_*^{k-1}, \quad f_*^{(r)} = f_*^{k+1} \end{cases} \tag{15}$$

We have considered minimizing $f(x) = u(x) + p(x)$, and $u'(0) \approx 0$ around the global minimum x_{**}. In this case $f'(x) \approx p'(x)$ and $p(x)$ is a periodical function with period T^p. Therefore, if x_*^{k-1}, $x_*^k (= x_{**})$, and x_*^{k+1} exist around local minima, then

$$\begin{cases} x_{**} - x_*^{k-1} = x_*^k - x_*^{k-1} \approx x_*^{k+1} - x_{**} = x_*^{k+1} - x_*^k \approx T^p \\ x_*^{(r)} - x_*^{(p)} = x_*^{k+1} - x_*^{k-1} \approx 2T^p. \end{cases}$$

From this result, we can replace the stop condition (11) of **Step 2** by s simpler condition:

$$f_*^{(p)} > f_*^{(q)} < f_*^{(p)}, \qquad x_*^{(r)} - x_*^{(p)} \approx 2T^p. \tag{16}$$

3.4. Main Algorithm

Based on investigations in **3.1**-**3.3**, we show the main algorithm that finds the global minimum x_{**} and its function value f_{**} of a function $f(x)$ in a searching region $D^1 = [a_1, b_1]$ for a given initial point $x^0 (= a)$, its function value f^0, an initial step size Δ, an upper limit of step size $\overline{\delta}$, a period T^p and a tolerance ε. The algorithm consists of the following steps.

$(f_{**}, x_{**}) \leftarrow MGufm(\ f,\ D^1, f^0,\ x^0,\ \Delta,\ \overline{\delta}, T^p,\ \varepsilon\ )$

G1. (Initial Step)
  $x^1 \leftarrow x^0 + \Delta;\ \ f^1 \leftarrow f(x^1);\ \ brct \leftarrow 3;$

G2. (Bracketing the global minimum)
  if $f^0 \leq f^1$ **then** $brct \leftarrow 1$; **break**; // 1) of (14')
  for $i \leftarrow 0$ **to** $N-2$ **do**
    $x^{i+2} \leftarrow a_1 + (i+2) \cdot \Delta;\ \ f^{i+2} \leftarrow f(x^{i+2});$
    if $f^i > f^{i+1}$ **and** $f^{i+1} \leq f^{i+2}$
      then $brct \leftarrow 2$; **break**; // 2) of (14')
  if $brct = 1$ **then** $p \leftarrow 0;\ q \leftarrow 1;\ r \leftarrow 2;$
    $(f_*^{(p)}, f_*^{(q)}, f_*^{(r)}, x_*^{(p)}, x_*^{(q)}, x_*^{(r)}) \leftarrow brkt3p(f^0, f^1, x^0, x^1);$
  if $brct = 2$ **then**
    $(f_*^{(j)}, x_*^{(j)}) \leftarrow MLuf(f^j, x^j, \overline{\delta}, \varepsilon);\ \ (j = i, i+1, i+2);$
  else if $brct = 3$ **then** $p \leftarrow N-1;\ q \leftarrow N;\ r \leftarrow N+1;$
    $(f_*^{(p)}, f_*^{(q)}, f_*^{(r)}, x_*^{(p)}, x_*^{(q)}, x_*^{(r)}) \leftarrow brkt3p(f^{N-1}, f^N, x^{N-1}, x^N);$

G3. (Reducing interval enclosing the global minimum)
  Reduce $[x_*^{(p)}, x_*^{(r)}]$ until $f_*^{(p)} > f_*^{(q)} < f_*^{(p)}$, $x_*^{(r)} - x_*^{(p)} \approx 2T^p$.

G4. (Last setting) $x_{**} \leftarrow x_*^{(q)};\ \ f_{**} \leftarrow f_*^{(q)};$

Here, $brkt3p(\cdot)$ is a procedure that encloses the global minimum by three local minima such that $x_*^{(p)} < x_*^{(q)} < x_*^{(r)}$, $f_*^{(p)} < f_*^{(q)} < f_*^{(r)}$ from two initial points: x^{N-1}, x^N.

3.5. Numerical Experiment

A numerical experiment was performed for the problem (P1SE), and the conditions of the experiment were

$$T^p = 0.05,\ \overline{\delta} = 0.2T^p\ \varepsilon = 10^{-5}, \text{ and } \Delta = 4T^p.$$

The algorithm found the global minimum of this function $f(x)$ with 57 function evaluations, about 16% fewer function evaluations than the 68 function evaluations in the previous study [6].

4. Algorithm for Multivariate Functions

The minimization problem (Q) of a multivariate function $f(\boldsymbol{x})$ that is expressed by the sum of a unimodal function $u(\boldsymbol{x})$ and a periodic function $p(\boldsymbol{x})$ is as follows:

$$\text{min. } f(\boldsymbol{x}) \equiv u(\boldsymbol{x}) + p(\boldsymbol{x}), \quad \boldsymbol{x} \in D^n \equiv \prod_{j=1,\ldots,n} [a_j, b_j]. \quad \text{(Q)}$$

To solve this problem, a new point $\boldsymbol{x}^{(k+1)}$ is repeatedly generated using step length $\alpha^{(k)}$ that is determined by minimizing from the previous point $\boldsymbol{x}^{(k)}$ in the direction $\boldsymbol{d}^{(k)}$:

$$\begin{cases} \alpha^{(k)} = \underset{\alpha}{\text{argmin}}\{\, \phi(\alpha) \equiv f(\boldsymbol{x}^{(k)} + \alpha \boldsymbol{d}^{(k)}) \,\}, \\ \boldsymbol{x}^{(k+1)} = \boldsymbol{x}^{(k)} + \alpha^{(k)} \boldsymbol{d}^{(k)}, \quad (k = 0, 1, 2, \ldots), \end{cases} \tag{17}$$

where this minimization step is called *line search*.

Generally, line search procedures can only find a local minimum of the function $\phi(\alpha)$. Thus, even though the line search is repeatedly applied, the possibility of finding the global minimum of function f is low.

In order to overcome this problem, we modify the specification of algorithm *MGufm* into the following specification so that the algorithm can find the global minimum $\boldsymbol{x}^{(k+1)}$ and its function value $f^{(k+1)}$ from a starting point $\boldsymbol{x}^{(k)} \in D^n$ along the direction $\boldsymbol{d}^{(k)}$:

$$(f^{(k+1)}, \boldsymbol{x}^{(k+1)}) \leftarrow MGufn\,(\, f, D^n, f^{(k)}, \boldsymbol{x}^{(k)}, \boldsymbol{d}^{(k)}, \Delta^{(0)}, \overline{\delta}, T^p, \varepsilon\,).$$

In the problem (Q), if $u(\boldsymbol{x})$ is a *strongly quasi-convex function*, then the function $\phi(\alpha) = u(\boldsymbol{x}^{(k)} + \alpha \boldsymbol{d}^{(k)})$ becomes unimodal for all $\boldsymbol{x}^{(k)} \in D^n$ and $\boldsymbol{d}^{(k)}$. If conditions similar to **Conditions 1–3** hold, then the possibility of finding the global minimum of the function f will become high.

In particular, if functions $u(\boldsymbol{x})$ and $p(\boldsymbol{x})$ are separable with respect to each variable x_i, that is,

$$f(\boldsymbol{x}) \equiv u(\boldsymbol{x}) + p(\boldsymbol{x}) = \sum_{i=1}^{n} \{u_i(x_i) + p_i(x_i)\}, \tag{18}$$

then the global minimum of this function can be found by applying the line search procedure *MGufn* along the i-th coordinate vector $\boldsymbol{e}^i$ only n-times.

We show an algorithm for finding the global minimum $\boldsymbol{x}_{**}$ and its function value f_{**} from an initial point $\boldsymbol{x}^{(0)}$ for the above function f with periods $\boldsymbol{T}^p = (T_1^p, T_2^p, \ldots, T_n^p)$. The algorithm consists of the following steps.

$(f_{**}, \boldsymbol{x}_{**}) \leftarrow MGnf(\, f,\, D^n, f^{(0)},\, \boldsymbol{x}^{(0)},\, \Delta,\, \overline{\boldsymbol{\delta}}, \boldsymbol{T}^p,\, \varepsilon\,)$
for $k = 1$ **to** n
  $(f^{(k+1)}, \boldsymbol{x}^{(k+1)}) \leftarrow MGufn(\, f, D^n, f^{(k)},\, \boldsymbol{x}^{(k)}, \boldsymbol{e}^k, \Delta,\, \overline{\delta}_k, T_k^p, \varepsilon\,)$;
$f_{**} \leftarrow f^{(n+1)}$; $\boldsymbol{x}_{**} \leftarrow \boldsymbol{x}^{(n+1)}$;

4.1. Numerical Experiment

We used the following Rastrign's function as a numerical experiment:

$$f(\boldsymbol{x}) = 100 + \sum_{i=1}^{10} \left(x_i^2 - 10\cos(2\pi x_i)\right),$$
$$x_i \in [-5.12, 5.12], \ (i = 1, 2, \ldots, 10)$$
$$\boldsymbol{x}_{**} = (0, 0, \ldots, 0), \quad f_{**} = 0.$$

The conditions of the experiment were as follows.

- The period of the i-th coordinate was $T_i^p = \sqrt{i}\pi$, $\overline{\delta}_i = 0.2T_i^p$ $(i = 1, 2, \ldots, 10)$, and the initial point was $\boldsymbol{x}^{(0)} = (-5, -5, \ldots, -5)$.

The algorithm found a global minimum $\boldsymbol{x}_{**} = (0, 0, \ldots, 0)$ with 380 function evaluations (480 f.e.). This result (380 f.e.) is about 0.056% rate of the number of function evaluations required in [4] (683,875 f.e.).

5. Conclusions

We have mainly studied an algorithm for finding the global minimum of a univariate function $f(x)$ with many local minima whose sequence is unimodal. This algorithm is characterized by the use of a large-step minimizer and a small-step local minimizer. More clear and effective algorithm was able to propose than previous algorithm. Numerical result showed that the algorithm efficiently finds the global minimum with only a few function evaluations.

Moreover, we considered a multivariate function $f(\boldsymbol{x})$ similar to the above unimodal function and showed an algorithm for minimizing a certain class of the function. The results of a numerical example showed that the algorithm effectively finds the global minimum.

References

[1] Avriel, M. : "Nonlinear Programming – Analysis and Methods," Prentice-Hall, pp. 225–226, 1976.

[2] Brent, R. P. : "Algorithm for Minimization without Derivatives," Prentice-Hall:New York 1973.

[3] Floudas, C. A. : "Deterministic Global Optimization –Theory, Methods and Applications–," Kluwer A.P., 2000.

[4] Dzemyda, G. , Šaltenis, V. and Žilinskas, A. : "Stochastic and Global Optimization," Kluwer Acad. Publ., Dordrecht, 2002.

[5] Kanemitsu, H. and Shimbo, M. : "Two Methods for Finding Local and Global Maxima of a Multimodal Function of One Variable Based on the Concept of a Unimodal Region," *IPSJ*, 37: 1646–1656, 1996.

[6] Kanemitsu, H., Konno, H., Miyakoshi, M. and Shimbo, M. : "An Algorithm for Globally Minimizing a Function with Many Local Minima whose Lower Envelope is Unimodal," Proc. of NOLTA'2002, pp.443-446, 2002.