MATEC Web of Conferences **139**, 00223 (2017) DOI: 10.1051/matecconf/201713900223
ICMITE 2017

A novel global optimization algorithm based on the smoothing function

Tian Jingjing[1,⋆] and *Yuan* Liu-yang[1,⋆⋆]

[1] *College of Science, Wuhan University of Science and Technology, Wuhan, Hubei, P.R. China, 430065*

Abstract. In this paper, a new global optimization algorithm based on the smoothing function is suggested. Firstly, one of the minimizer x_1^* of the smooth function F is found by employing any local minimizer finder, then the function $m(x, x_1^*) := \min\{F(x), F(x_1^*)\}$ is considered instead of the objective function F and the minimizer x_2^* of the function m is searched. Because of the non-smooth of the function m, the smoothing function for m is used in order to make the local minimizer finding algorithms for smooth optimization available. Finally, a global optimization algorithm based on the smoothing function is presented. The implementation of the algorithm on several test problems is reported with numerical results.

1 Introduction

We consider the box constrained global optimization problem as follows:

$$\begin{array}{ll} \min & F(x) \\ \text{s.t.} & x \in D, \end{array} \tag{1}$$

here $F(x) : R^n \to R$ is a continuously differentiable function and D is a box. As $F : R^n \to R$ is a coercive function, i.e., $F(x) \to +\infty$ when $\|x\| \to +\infty$, then the following problem:

$$\min_{x \in R^n} F(x)$$

can be reformulated as the box constrained optimization problem in (1).

Global optimization has many applications in almost all fields of science and technology. Because a general non-convex objective function has many local minima, then it makes global optimization hard to solve ([2, 6, 13]). Currently, Global optimization makes great progresses on theories and practical applications, and there are many methods have been presented, for example the integration-level set methods [1], the branch and bound methods [6], the outer approximation methods [7], the tunneling function methods [8], the interval methods [12], the filled function methods [3–5, 9–11, 14–22]. Here, a novel global optimization algorithm based on the smoothing function is presented.

Most of global optimization algorithms must use local searches. The main difficulty for these types global optimization algorithms is to pass lower minimizers from the current one. Our minimization algorithm is also based on finding the lower local minimizer than the current local minimizer. Indeed, one of the minimizer x_1^* of the smooth function F is found by employing any local minimizer finder, then the function $m(x) := \min\{F(x), F(x_1^*)\}$ is considered instead of the objective function F and the minimizer x_2^* of the function $m(x)$ is searched. This loop proceeds until the global minimizer is found. It has to be stated that, the function $\min\{F(x), F(x_1^*)\}$ has the same minimizers with the function F except x_1^*. Therefore, one of the main difficulties of global optimization process is eliminated.

On the other hand, the function m can be non-smooth because of the presence of "min" operator. Therefore, we propose using the smoothing functions for m in order to make the local minimizer finding algorithms for smooth optimization available. After smoothing process, the next local minimizer is found by considering any of local minimizer finder for smooth optimization.

The structure of the paper is as follows. After this introduction, a smoothing function for the function $m(x, x_1^*) = \min\{F(x), F(x_1^*)\}$ is designed in section 2, and the properties of the smoothing function are analysed and discussed. Moreover, in section 3, a global optimization algorithm based on the smoothing function is proposed. Finally, the numerical experiment results are introduced in section 4. Compared with the numerical experiment results of [21], the results indicate that the global optimization algorithm in the paper is effective. Section 5 is the conclusion.

2 The smoothing function

Suppose that x_1^* is a current local minimizer of problem (1), then the smoothing function $m(x, x_1^*) := \min\{F(x), F(x_1^*)\}$ is considered in stead of the objective function F and the minimizer x_2^* of the function m is searched. It can be stated that, the function m has the same minimizers with the function F except x_1^*. However, the function m can be non-smooth because of the presence of "min" operator. Therefore, we propose using the smooth-

⋆e-mail: 646864258@qq.com
⋆⋆Corresponding author, e-mail: yangly0601@126.com

DOI: 10.1051/matecconf/201713900223

ing functions for m in order to make the local minimizer algorithms for smooth optimization available.

For the function $m(x, x_1^*) := \min\{F(x), F(x_1^*)\}$, we have the following:

$$m(x, x_1^*) := F(x_1^*) + \min\{F(x) - F(x_1^*), 0\}.$$

Thus, The following smoothing function is designed:

$$S(x, x_1^*) := F(x_1^*) + q(F(x) - F(x_1^*), u),$$

where $u > 0$ and

$$q(t,u) = \begin{cases} t, & t \le -u, \\ -\frac{(t-u)^2}{4u}, & -u < t < u, \\ 0, & t \ge u. \end{cases}$$

Proposition 2.1. For any $(t, u) \in R^2$, we have the following results.

(1) $q(\cdot, \cdot)$ is continuously differentiable at any $(t, u) \in R^2$ with $u > 0$;

(2) $q(t, 0) = \min\{t, 0\}$;

(3) $1 \ge \frac{\partial}{\partial t} q(t, u) \ge 0$ at any $(t, u) \in R^2$ with $u > 0$.

Theorem 2.1. For every $x \in R^n$,

$$-\frac{u}{4} \le S(x, x_1^*) - m(x, x_1^*) \le 0$$

Proof. For every $x \in R^n$,

$$S(x, x_1^*) - m(x, x_1^*) = q(F(x) - F(x_1^*), u) - \min\{F(x) - F(x_1^*), 0\},$$

(1) When $|F(x) - F(x^*)| \ge u$,

$$S(x, x_1^*) - m(x, x_1^*) = 0.$$

(2) When $0 \le F(x) - F(x^*) \le u$,

$$S(x, x_1^*) - m(x, x_1^*) = -\frac{(F(x) - F(x_1^*) - u)^2}{4u}.$$

Therefore,

$$-\frac{u}{4} \le S(x, x_1^*) - m(x, x_1^*) \le 0.$$

(3) When $-u \le F(x) - F(x^*) \le 0$,

$$S(x, x_1^*) - m(x, x_1^*) = -\frac{(F(x) - F(x_1^*) - u)^2}{4u} - (F(x) - F(x_1^*)).$$

Therefore,

$$-\frac{u}{4} \le S(x, x_1^*) - m(x, x_1^*) \le 0.$$

From the theorem above, we can see that S is the smoothing function of m when u is sufficiently small.

3 Global optimization algorithm

In this part, a novel global optimization algorithm for solving problem (1) is proposed based on the smoothing function $S(x, x_1^*)$, and the algorithm can find a global optimal solution or an approximate global optimal solution of problem (1).

The novel global optimization algorithm for problem (1) is described as follows. we refer the algorithm as SMGP(the global optimization algorithm based on the smoothing function for problem (1)).

Algorithm SMGP

Step 0: Choose sufficiently small numbers $u_L > 0$, $\lambda_L > 0$. Choose a positive integer number I and directions e_i, $i = 1, \dots, I$. Choose a starting point $x_0 \in D$. Set $i := 0$;

Step 1: Seek for a local minimizer x_i^* of problem (1) by using local search methods starting from x_i;

Step 2: Let

$$S(x, x_k^*) := F(x_k^*) + q(F(x) - F(x_k^*), u),$$

where $u > 0$ and

$$q(t,u) = \begin{cases} t, & t \le -u, \\ -\frac{(t-u)^2}{4u}, & -u < t < u, \\ 0, & t \ge u. \end{cases}$$

Set $l = 1$ and $\lambda = 2$;

Step 3: (a)If $l \le I$, then go to 3(b); Otherwise, go to Step 5;

(b)If $\lambda \ge \lambda_L$, then let $y_i^l := x_i^* + \lambda e_l$, go to 3(c); otherwise, let $l := l + 1$, $\lambda = 2$, go to 3(a);

(c)If $y_i^l \in D$, then go to 3(d); otherwise, let $\lambda := \frac{\lambda}{2}$, go to 3(b);

(d)If $F(y_i^l) < F(x_i^*)$, then let $x_{i+1} := y_i^l$, $i := i + 1$, go to Step 1; otherwise, go to Step 4;

Step 4: Seek for a local minimizer for the following optimization problem from the starting point y_i^l:

$$\min_{x \in D} S(x, x_i^*), \tag{2}$$

Let y_i^* be a found local minimizer of problem (2). If y_i^* satisfies $F(y_i^*) < F(x_i^*)$, then let $x_{i+1} := y_i^*$, $i := i + 1$, go to Step 1; otherwise, let $l = l + 1$, $\lambda = 2$, go to Step 3(a);

Step 5: Let $x_s = x_i^*$ and stop.

Next, we give the explanation of the motivation and mechanism of the algorithm.

In Step 0, we choose the directions e_i, $i = 1, \dots, I$ as follows. For example, as $n \ge 2$, set $I = 2n$, for $i = 1, \dots, n$, and the i-th component of e_i is one, the other component of e_i is zero; for $i = n + 1, \dots, I$, the i-th component of e_i is -1, the other component of e_i is zero.

In Step 0, we choose a starting point. Then search for a local minimizer x_i^* of problem (1) starting from the initial point by using a local minimization method. If x_i^* is not a global minimizer, then to find deeper local minimizers of problem (1) is our main task.

In step 3, we try to choose an appropriate initial point to find a local minimizer of problem (2) by using a local minimization method. If the set of directions $\{e_i, i = 1, \dots, I\}$ is large enough, no matter how to choose the initial points, local minimizers y_i^* of problem (2) will equal to x_i^*, and $S(y_i^*, x_i^*, u) = f(x_i^*)$, then we can say that the local minimizer x_i^* in Step 1 is an approximate global optimal solution of problem (1). When u is sufficiently small, the global optimal solution of problem (1) can be found.

4 Numerical experiments

In this part, several sets of numerical experiments are implemented and the results can show that Algorithm SMGP is effective. All the numerical experiments are implemented in Matlab2010b. We use The Matlab function 'fmincon' in the algorithm to search for local minimizers of problem (1) and problem (2).

The experiment results of each example are summarized in tables. The following symbols used in the tables are given:

i The iteration number in finding the i-th local minimizer;

x_i^0 The initial point in the i-th iteration for searching for the i-th local minimizer of problem (1);

x_i^* The i-th local minimizer for problem (1);

$F(x_i^*)$ The objective function value for x_i^*;

y_i^0 The initial point in the i-th iteration for searching the i-th local minimizer of problem (2);

y_i^* The i-th local minimizer for problem (2);

$F(y_i^*)$ The objective function value for y_i^*;

$CPU(s)$ The CPU time in seconds for the algorithms.

and throughout our computational experiments, we set the parameters in Algorithm SMGP as follows:

$$u_L = 10^{-6},\ \lambda_L = \frac{1}{2^3}.$$

Example 1.

$$\begin{array}{ll} \min & F(x) = \frac{x^6}{6} - \frac{5x^4}{4} + 2x^2 \\ \text{s.t.} & -2.5 \le x \le 2.5, \end{array}$$

This example has three local minimizer $x_1^* = -2$, $x_2^* = 0$, $x_3^* = 2$, where x_1^*, x_3^* is the global minimizer with $f(2) = f(-2) = -1.3333$. The computational results are given in Table I.

Example 2. (The Goldstein and Price function in [3, 22]).

$$\begin{array}{ll} \min & F(x) = g(x)h(x) \\ \text{s.t.} & -3 \le x_1, x_2 \le 3, \end{array}$$

where

$$\begin{array}{l} g(x) = 1+ \\ (x_1 + x_2 + 1)^2(19 - 14x_1 + 3x_1^2 - 14x_2 + 6x_1x_2 + 3x_2^2), \\ h(x) = 30+ \\ (2x_1 - 3x_2)^2(18 - 32x_1 + 12x_1^2 + 48x_2 - 36x_1x_2 + 27x_2^2). \end{array}$$

The Example has the global optimal solution $x^* = (0, -1)^T$ and $f(x^*) = 3$. The computational results are summarized in Table II.

Example 3. (Six-hump back camel function in [3, 22]).

$$\begin{array}{ll} \min & f(x) = 4x_1^2 - 2.1x_1^4 + x_1^6/3 - x_1x_2 - 4x_2^2 + 4x_2^4, \\ \text{s.t.} & -3 \le x_1, x_2 \le 3. \end{array}$$

This problem has the global optimal solutions: $x^* = (0.0898, 0.7127)^T$ or $(-0.0898, -0.7127)^T$, where $f(x^*) = -1.0316$. The computational results are summarized in Table III.

5 Conclusion

In this paper, A new global optimization algorithm based on the smoothing function is proposed. Firstly, one of the minimizer x_1^* of the smooth function F is found by employing any local minimizer finder, then the function $m(x, x_1^*) := \min\{F(x), F(x_1^*)\}$ is considered instead of the objective function F and the minimizer x_2^* of the function m is searched. Because of the non-smooth of the function m, the smoothing function for m is used in order to make the local minimizer finding algorithms for smooth optimization available. Finally, a global optimization algorithm based on the smoothing function is proposed. The numerical experiments are carried out on several test problems. Compared with experiment results of [21], the results indicate Algorithm SMGP and Algorithm FFGP in [21] have the similar results, However the Cpu time of Algorithm SMGP is less than Algorithm FFGP in most case. Consequently Algorithm FFGP is effective. Nevertheless, the set of directions $\{e_i, i = 1, \dots, I\}$ in Algorithm SMGP is crucial. so without appropriate directions, The local minimizers of problem (2) can not be found. Next, we will pay attention to how to modify it.

Acknowledgment

The research was financially supported by the National Natural Science Foundation of China (NSFC)(11401450 and 11401126).

References

[1] S. H. Chew and Q. Zheng, Integral Global Optimization, Volume 298 of Lecture Notes in Economics and Mathematical Systems, Springer–Verlag, 1988.

[2] L. C. W. Dixon, J. Gomulka and S. E. Herson, "Reflection on global optimization problems", in *Optimization in Action* (Dixon, L.C.W. etc.), Academic Press, New York, pp. 398-435, 1976.

 DOI: 10.1051/matecconf/201713900223

[3] R. P. Ge, "A filled function method for finding a global minimizer of a function of several variables," Mathematical Programming, vol.46, pp. 191-204, 1990.
[4] R. P. Ge and Y. F. Qin, "A class of filled functions for finding a global minimizer of a function of several variables," Journal of Optimization Theory and Applications, vol.54, pp. 241-252, 1987.
[5] R. P. Ge and Y. F. Qin, "The globally convexized filled functions for global optimization," Applied Mathematics and Computation, vol.35, pp. 131-158, 1990.
[6] R. Horst, P. M. Pardalos and N. V. Thoai, Introduction to Global Optimization, 2^{nd} edition, Kluwer Academic Publishers, Dordrecht, 2001.
[7] R. Horst, N. V. Thoai and H. Tuy, "Outer approximation by polyhedral convex sets," Operations Research Spektrum, vol.9, pp. 153-159, 1987.
[8] A. Levy and A. Montalvo, "The tunneling algorithm for the global minimization of functions," SIAM Journal on Scientific and Statistical Computing, vol.6, pp. 15-29, 1986.
[9] X. Liu, "Finding global minima with a computable filled function," Journal of Global Optimization, vol.19, pp. 151-161, 2001.
[10] H. W. Lin, Y. P. Wang, L. Fan and Y. L. Gao, "A new discrete filled function method for finding global minimizer of the integer programming," Applied Mathematics and Computation, vol.219, pp. 4371-4378, 2013.
[11] H. W. Lin, Y. L. Gao and Y. P. Wang, "A continuously differentiable filled function method for global optimization," Numerical Algorithms, vol.66, pp. 511-523, 2014.
[12] R. E. Moore, Enterbal Analysis, Prentice-Hall, NJ: Englewood Cliffs, 1966.
[13] P. M. Pardalos, H. E. Romeijn and H. Tuy, "Recent development and trends in global optimization," Journal of Computational and Applied Mathematics, vol.124, pp. 209-228, 2000.
[14] Z. Wan, L. Y. Yuan and J. W. Chen, "A filled function method for nonlinear systems of equalities and inequalities," Computational&Applied Mathematics, vol.31, pp. 391-405, 2012.
[15] W. X. Wang, Y. L. Shang, L. S. Zhang and Y. Zhang, "Global minimization of non-smooth unconstrained problems with filled function," Optimization Letters, vol.7, pp. 435-446, 2013.
[16] F. Wei and Y. P. Wang, "A new filled function method with one parameter for global optimization," Mathematical Problems in Engineering, vol.2013, pp. 1-12, 2013.
[17] F. Wei, Y. P. Wang and H. W. Lin, (2014), "A new filled function method with two parameters for global optimization," Journal of Optimization Theory and Applications, vol.163, pp. 510-527, 2014.
[18] Y. J. Yang and Y. L. Shang, "A new filled function method for unconstrained global optimization," Applied Mathematicas Computation, vol.173, pp. 501-512, 2006.
[19] Y. J. Yang, Z. Y. Wu and F. S. Bai, "A filled function method for constrained nonlinear integer programming," Journal of Industrial and Management Optimization, vol.4, pp. 353-362, 2008.
[20] L. Y. Yuan, Z. Wan, J. J. Zhang and B. Sun, "A filled function method for solving nonlinear complementarity problems," Journal of Industrial and Management Optimization, vol.5, pp. 911-928, 2009.
[21] L. Y. Yuan, Z. Wan and Q. H. Tang, "A criterion for an approximation global optimal solution based on the filled functions," Journal of Industrial and Management Optimization, vol.12, pp. 375-387, 2016.
[22] L. S. Zhang, C. NG, D. Li and W. Tian, "A new filled function method for global optimization," Journal of Global Optimization, vol.28, pp. 17-43, 2004.
[23] Q. Zheng and D. Zhuang, "Integral global minimization: Algorithms, implementations and numerical tests," Journal of Global Optimization, vol.7, pp. 421-454, 1995.

Table 1. Experiment results for Example 1

i	x_i^0	x_i^*	$F(x_i^*)$	CPU(s)
0	-0.5	0	0	
1	2	2	-1.3333	SMGP: 0.0156
0	-0.5	0	0	
1	2	2	-1.3333	FFGP: 0.0468

Table 2. Experiment results for Example 2

i	x_i^0	x_i^*	$F(x_i^*)$	CPU(s)
0	$\begin{pmatrix} -1 \\ -1 \end{pmatrix}$	$\begin{pmatrix} 0 \\ -1 \end{pmatrix}$	3	SMGP: 0.0156
0	$\begin{pmatrix} -1 \\ -1 \end{pmatrix}$	$\begin{pmatrix} 0 \\ -1 \end{pmatrix}$	3	FFGP: 0.0312

Table 3. Experiment results for Example 3

i	x_i^0	x_i^*	$F(x_i^*)$	y_i^0	y_i^*	$F(y_i^*)$
0	$\begin{pmatrix} 2 \\ -1 \end{pmatrix}$	$\begin{pmatrix} -1.7 \\ -0.8 \end{pmatrix}$	-0.2155	$\begin{pmatrix} 0.3 \\ -0.8 \end{pmatrix}$	$\begin{pmatrix} -0.09 \\ -0.71 \end{pmatrix}$	-1.0316
0	$\begin{pmatrix} -2 \\ 1 \end{pmatrix}$	$\begin{pmatrix} 1.7 \\ 0.8 \end{pmatrix}$	-0.2155	$\begin{pmatrix} -0.3 \\ 0.8 \end{pmatrix}$	$\begin{pmatrix} 0.09 \\ 0.71 \end{pmatrix}$	-1.0316
0	$\begin{pmatrix} 2 \\ -1 \end{pmatrix}$	$\begin{pmatrix} -1.7 \\ -0.8 \end{pmatrix}$	-0.2155	$\begin{pmatrix} 0.3 \\ -0.8 \end{pmatrix}$	$\begin{pmatrix} 0.3 \\ -0.8 \end{pmatrix}$	-0.3571
1	$\begin{pmatrix} 0.3 \\ -0.8 \end{pmatrix}$	$\begin{pmatrix} -0.09 \\ -0.71 \end{pmatrix}$	-1.0316	-	-	-