A Filled Function for Non-smooth Box-constrained Global Optimization and Its Application
Wei-Xiang WANG¹,a, Duo WANG²,b, You-Lin SHANG²,*
¹Shanghai Second Polytechnic University, Shanghai 201209, China
²Henan University of Science and Technology, Luoyang 471003, China
ªzhesx@126.com, bwangduo9051@126.com, *mathshang@sina.com

Keywords: Nonlinear programming; Box-constrained global minimization; Filled function method; Global minimizer.

Abstract. This paper constructs a filled function for non-smooth box-constrained global minimization, investigates its properties and designs a corresponding algorithm. The constructed new filled function contains only one parameter and it has more merits than those with several parameters. Moreover, the iterative operations of the proposed filled function algorithm can be easily implemented. We also perform numerical experiments to show the efficiency of the proposed approach.

Introduction
With rapid progress being made in AI, economics and financing engineering, more and more complex global optimization models need to be solved, and studies on global optimization for the nonlinear programming problem \( \min_{x \in X} f(x) \), where \( X \) is a box set, become a hot topic today.

In recent years, lots of new theories and algorithms for global optimization have emerged, and they usually fall into two main categories: stochastic and deterministic method. Among those methods, filled function method put forward by Ge in [1] is a particularly useful deterministic method for global optimization. The filled function method is usually comprised of a sequence of cycles, each having two phases: a local minimization phase and a global descent phase. A local minimization phase aims at seeking one local minimizer of the original problem and the second phase focuses on finding a better initial point for the first phase. Those two phases are conducted repeatedly until the terminating criteria is met and one approximate global minimizer is obtained. The filled function in [1] has some shortcomings, and thus the papers in [2,3,4,5] reconsidered the filled function methods. Note that the above filled function methods are designed mainly for smooth global optimization. However, most of global optimization problems in real world are usually formulated into non-smooth ones. In this paper, we extend the filled function methods for smooth global optimization to include non-smooth case.

Generally speaking, there are two difficulties are needed to address by the global optimization. The first one is how to skip from the current optimizer, and the another how to judge the current optimizer is a global solution. This paper tackles only the first difficulty.

This paper is organized as follows: In Section 2, we make some assumptions on the problem \( (P) \), and construct a filled function and discuss its theoretical properties. In Section 3, we give an corresponding filled function algorithm. In Section 4, we provide several numerical results, including its application in solving NCP problem.

A Novel Filled Function and Its Properties
In this paper, we consider the following box-constrained global minimization problem \( (P): \min_{x \in X} f(x) \), where \( X \) is a box set. For simplicity, we denote the set of local minimizers of problem \( (P) \) by \( L(P) \). To introduce the filled function for non-smooth global optimization, we assume that the following are satisfied:
Assumption 1. \( f(x) \) is Lipschitz continuous on \( \mathbb{R}^n \) with a rank \( L > 0 \).

Assumption 2. The problem \( (P) \) has at least one global minimizer and has a finite number of different minimal function values.

The main tools used by filled function method for non-smooth global minimization is Clark generalized gradient. For more details of the Clark generalized gradient, please refer to [8].

Now, we give the definition of filled function for non-smooth global minimization problem \( (P) \).

Definition. A function \( \phi(x, x^*) \) is said to be a filled function of problem \( (P) \) at \( x^* \in L(P) \), if it has the following properties:

1. \( x^* \) is a strictly maximizer of \( \phi(x, x^*) \) on \( X \).
2. For any point \( x \neq x^* \in X \) satisfying \( \phi(x, x^*) \geq \phi(x, x^*) \), \( 0 \notin \partial \phi(x, x^*) \).
3. If \( x^* \) is not a global minimizer of \( f(x) \), then there exists \( x_0 \in S_2 = \{x \in X : f(x) < f(x^*) \} \) such that \( x_0 \) is a local minimizer of \( \phi(x, x^*) \).

Define

\[
\phi(t) = \begin{cases} 
1 & t \geq 0 \\
\frac{1}{2} \sin(\pi t + \frac{\pi}{2}) + \frac{1}{2} & -1 < t < 0 \\
0 & t \leq -1 
\end{cases}
\]

and its derivative is given below:

\[
\frac{d\phi(t)}{dt} = \begin{cases} 
\frac{1}{2} \pi \cos(\pi t + \frac{\pi}{2}) & -1 < t < 0 \\
0 & t \in (-\infty, -1] \cup [0, +\infty) 
\end{cases}
\]

Now, we construct a new filled function as follows:

\[
P(x, x^*, A) = \exp\left(-\|x - x^*\|\right) \phi(A f(x) - f(x^*)) + \frac{f(x)}{A}.
\]

where \( A > 0 \) is one parameter. Let \( D = \max_{x, x^* \in X} \|x - x^*\| \). The following theorems show that \( P(x, x^*, A) \) is a filled function.

Theorem 1. Let \( x^* \in L(P) \), then \( x^* \) is a strictly local maximizer of \( P(x, x^*, A) \).

Proof. Since \( x^* \in L(P) \), there exists a neighborhood \( N(x^*, \sigma) \) of \( x^* \), where \( \sigma > 0 \), such that \( f(x) - f(x^*) \geq 0 \), for all \( x \in N(x^*, \sigma) \cap X \). Therefore, when \( x \in N(x^*, \sigma) \cap X, x \neq x^* \), we have \( \phi(f(x) - f(x^*)) = 1 \). The above equation together with the assumption that \( \left| f(x) - f(x^*) \right| \leq L \|x - x^*\| \), implies that, when \( A > L \exp(-\sigma) \), we have

\[
F(x, x, A) - F(x, x^*, A) = \exp\left(-\|x - x^*\|\right) + \frac{f(x) - f(x^*)}{A} - 1
\]

\[
= \frac{1 - \exp\left(-\|x - x^*\|\right) + f(x) - f(x^*)}{A} \leq \frac{-\|x - x^*\| + L \|x - x^*\|}{A} < 0.
\]

Thus, \( x^* \) is a strict maximizer of \( F(x, x^*, A) \).

Theorem 2. Let \( x^* \in L(P) \), then, for any \( x \neq x^* \) and \( f(x) \geq f(x^*) \), we have \( 0 \notin \partial F(x, x^*, A) \).

Proof. By the conditions, it holds that

\[
\partial F(x, x^*, A) \subseteq -\frac{(x - x^*)}{\|x - x^*\|} \cdot (-1) + \frac{\partial f(x)}{A}.
\]

Thus, when \( A > L \exp D \), where \( D = \max \|x - x^*\| \), we have
\[
\langle x-x', \partial F(x,x', A) \rangle \leq \frac{\langle \partial f(x), x-x' \rangle}{A} - \left\| x-x' \right\| e^{-\frac{1}{A}} \leq \left\| x-x' \right\| \left( \frac{L}{A} - \frac{1}{\exp D} \right) < 0.
\]

Hence, we have \(0 \in \partial F(x,x', A)\).

Theorem 3. Assume that \(x^* \in L(P)\), but it is not a global minimizer, then there exists a point \(x_0 \in S_2 = \{x \in X : f(x) < f(x^*)\}\) such that \(x_0\) is a local minimizer of \(P(x,x^*, A)\).

**Proof.** Denote \(d = \min_{x_1, x_2 \in L(P), f(x_1) < f(x_2)} |f(x_1) - f(x_2)|\), \(0 < r < d\).

Since \(x^*\) is not a global minimizer of \((P)\), there exists a minimizer \(x_0\), such that \(f(x^*) > f(x_0)\). Thus, we have \(f(x_0) - f(x^*) \leq -d < r\). By the continuity of \(f(x)\), there exists one neighborhood \(N(x_0, \sigma)\) of \(x_0\), such that \(f(x) - f(x^*) < -r\), for any \(x \in N(x_0, \sigma)\). Therefore, when \(A > \frac{1}{r}\), it holds \(A(f(x) - f(x^*)) < -Ar < -1, \forall x \in N(x_0, \sigma)\). Thus, we have \(P(x,x^*, A) = \frac{f(x)}{A}\). Since \(x_0\) is a minimizer of \(f(x)\), it is also a minimizer of \(P(x,x^*, A)\).

Moreover, it holds \(f(x_0) < f(x^*)\). The proof of the theorem is completed.

**Filled Function Algorithm**

In the above section, we discussed some theoretical properties of the filled function. Now, we propose a filled function algorithm below.

**Filled function algorithm**

**Initialization step:**

Let \(A_0\) be the upper bound of parameter \(A\), \(x_1\) the initial point and \(e_1, e_2, \ldots, e_{2n}\) the positive and negative coordinate directions. Set \(k = 1\), and go to the main step.

**Main step**

1. Starting from \(x_1\), minimize \((P)\) by any non-smooth local minimization method to find a local minimizer \(x^*_k\) and go to 2.
2. Set \(A = 1\).
3. Construct a filled function \(P(x,x^*, A)\) and go to 4.
4. If \(k > 2n\), then go to 7; otherwise, set \(x = x^*_k + 0.1e_k\), and take \(x\) as an initial point to find a local minimizer \(x_k\) of the following problem: \(\min_{y \in X} P(y,x^*, A)\).
5. If \(x_k \in X\), then set \(k = k + 1\), and go to 4; otherwise, go to 6.
6. If \(f(x_k) < f(x^*_k)\), then, (a) set \(x = x_k\), \(k = 1\). (b) Use \(x\) as a new initial point and minimize \((P)\) to find its another local minimizer \(x_2^*\) with \(f(x_2^*) < f(x^*_k)\). (c) Set \(x_1^* = x_2^*\) and go to 2; Else if \(f(x_k) \geq f(x^*_k)\), then go to 7.
7. Increase \(A\) by setting \(A = 10A\). If \(A \leq A_0\), then set \(k = 1\), and go to 3; otherwise, take \(x_1^*\) as a global minimizer, and the algorithm stops.

**Remarks:**

1. The proposed filled function method is also suitable for smooth global optimization.
2. There are two phases in the filled function method: a local minimization phase and a global descent phase. In phase 1, a local minimizer \(x^*\) could be found by any non-smooth local minimization method, such as Hybrid Hooke and Jeeves-Direct Method for Non-smooth Optimization [7], Mesh Adaptive Direct Search Algorithms for Constrained Optimization [6],
Bundle methods, Powell’s method, etc. In phase 2, the constructed filled function \( P(x, x', A) \) is to be minimized. During the minimizing process, if a point \( x_k \) is identified such that \( f(x_k) < f(x') \), then phase 2 stops and the algorithm returns to phase 1 looking for a better optimizer for \( f(x) \). The above process repeats until the local minimization of this method does not yield a better point. The current local minimizer will then be taken as a global minimizer.

**Numerical Experiment**

In this section, we carry out some numerical tests including one application of the filled function method in solving NCP problem. All tests are programmed in Fortran 95. To find a local optimizer, in non-smooth case, we use Hybrid Hooke and Jeeves-Direct Method for Non-smooth Optimization [7] and the Mesh Adaptive Direct Search Algorithms for Constrained Optimization [6] and in smooth case, we use penalty function method and conjugate gradient method.

**Problem 1:**

\[
\min f(x) = \left| \frac{x - 1}{4} \right| + \sin(\pi(1 + \frac{x - 1}{4})) + 7|x| \leq 10.
\]

The algorithm successfully found a global solution: \( x^* = 1 \) with \( f(x^*) = 7 \). Table 1 records the numerical results of Problem 1.

<table>
<thead>
<tr>
<th>( k )</th>
<th>( x^0_k )</th>
<th>( f(x^0_k) )</th>
<th>( x^*_k )</th>
<th>( f(x^*_k) )</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>6.0000</td>
<td>8.9571</td>
<td>5.0000</td>
<td>8.0001</td>
</tr>
<tr>
<td>2</td>
<td>0.9678</td>
<td>7.0333</td>
<td>0.9998</td>
<td>7.0001</td>
</tr>
</tbody>
</table>

**Problem 2:**

\[
\min f(x) = \max\{5x_1 + x_2, -5x_1 + x_2, x_1^2 + x_2^2 + 4x_2\}, -4 \leq x_1, x_2 \leq 4.
\]

The algorithm successfully found a global solution: \( x^* = (0, -3) \) with \( f(x^*) = -3 \). Table 2 records the numerical results of Problem 2.

<table>
<thead>
<tr>
<th>( k )</th>
<th>( x^0_k )</th>
<th>( f(x^0_k) )</th>
<th>( x^*_k )</th>
<th>( f(x^*_k) )</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>(1,1)</td>
<td>6.0000</td>
<td>(0.0000,0.0000)</td>
<td>0.0000</td>
</tr>
<tr>
<td>2</td>
<td>(-0.0002,-0.9725)</td>
<td>-0.9715</td>
<td>(-0.0002,-0.9725)</td>
<td>-0.9715</td>
</tr>
<tr>
<td>3</td>
<td>(-0.0003,-2.5644)</td>
<td>-2.5487</td>
<td>(0.0000,-3.0000)</td>
<td>-3.0000</td>
</tr>
</tbody>
</table>

One of applications of the filled function method is using it to solve a NCP problem. Consider NCP(F). NCP(F) can be described as follows:

Given a vector function \( F(x): R^n \rightarrow R^n \), finds a point \( x^* \in R^n \) such that \( x^* \) satisfies

\[
x \geq 0, \quad F(x) \geq 0, \quad \langle x, F(x) \rangle = 0.
\]

Let \( x = (x_1, x_2, ..., x_n)^T \), \( F(x) = (f_1(x), f_2(x), ..., f_n(x))^T \),

\[
\phi(a,b) = \sqrt{a^2 + b^2} - a - b, \quad \forall a, b \in R.
\]

Then NCP(F) is equivalent to the following equation:

\[
\psi(x) = \left[ \phi(x_1, f_1(x)), \phi(x_2, f_2(x)), ..., \phi(x_n, f_n(x)) \right] = 0.
\]

Obviously, the above equation is also equivalent to the following global optimization problem.

\[
\min_{x \in R^n} f(x) = \|\psi(x)\|^2 = \sum_{k=1}^n \phi^2(x_k, f_k(x)).
\]

Thus, we can solve NCP via using the proposed filled function method.
Problem 3.

\[ f_1(x) = 3x_1 + 2x_1x_2 + 2x_2^2 + \left| x_3 - 3 \right| + 3x_4 - 3, \]
\[ f_2(x) = 2x_1^2 + x_1 + x_2^2 + 10x_3 + 2x_4 - 2, \]
\[ f_3(x) = 3x_1^2 + 2x_1x_2 + 2x_2^2 + 2x_3 + 9x_4 - 9, \]
\[ f_4(x) = x_1^2 + 3x_2^2 + 2x_3 + 3x_4 - 3. \]

The algorithm successfully found its solutions: \( x^* = (1,0,3,0)^T \) and \( x^* = (0,0,0,1)^T \) Table 3 records the numerical results of Problem 3.

<table>
<thead>
<tr>
<th>( x_0 )</th>
<th>( x_i^* )</th>
<th>( F(x_i^*) )</th>
<th>( F(x_i^<em>)^T x_i^</em> )</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>(0.000002)</td>
<td>(3.000273)</td>
<td>0.000804</td>
</tr>
<tr>
<td>0</td>
<td>0.000007</td>
<td>0.002068</td>
<td></td>
</tr>
<tr>
<td>0</td>
<td>0.000177</td>
<td>0.001686</td>
<td></td>
</tr>
<tr>
<td>0</td>
<td>1.000148</td>
<td>0.000798</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>(0.000001)</td>
<td>(3.000269)</td>
<td>0.000802</td>
</tr>
<tr>
<td>2</td>
<td>0.000003</td>
<td>0.002061</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>0.000176</td>
<td>0.001691</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>1.000149</td>
<td>0.000798</td>
<td></td>
</tr>
</tbody>
</table>

Acknowledgement

This work was supported by the NNSF of China (Nos: 11471102,11001248,51776116), the key discipline "Applied Mathematics" of SSPU (No. A30XK1322100).

References