

无约束全局优化问题的单参数打洞函数法

刘飞燕^{1*}, 许锦华²

¹浙江师范大学数学科学学院, 浙江 金华

²中国电信股份有限公司金华分公司, 浙江 金华

收稿日期: 2026年3月9日; 录用日期: 2026年4月3日; 发布日期: 2026年4月10日

摘要

打洞函数法是求解全局优化问题的一种确定性算法。本文针对无约束全局优化问题, 提出一种新的单参数打洞函数。在合理假设条件下, 对该函数的性质及其相关理论性质进行了严格证明, 并据此设计了相应的求解算法。通过选取经典算例进行数值实验, 结果表明所提出的算法具有良好的有效性与可行性。

关键词

全局优化, 打洞函数, 单参数

A Tunneling Function Method with an One-Parameter for Unconstrained Global Optimization Problem

Feiyan Liu^{1*}, Jinhua Xu²

¹School of Mathematical Sciences, Zhejiang Normal University, Jinhua Zhejiang

²China Telecom Corporation Limited Jinhua Branch, Jinhua Zhejiang

Received: March 9, 2026; accepted: April 3, 2026; published: April 10, 2026

Abstract

The Tunneling Function Method is a deterministic algorithm for solving global optimization problems. This paper proposes a novel one-parameter tunneling function for unconstrained global optimization. Under reasonable assumptions, the properties of this function and its related theoretical foundations are rigorously proved. Based on this, a corresponding solution algorithm is designed. Numerical experiments conducted on selected classical test problems demonstrate that the proposed

*通讯作者。

algorithm exhibits good effectiveness and feasibility.

Keywords

Global Optimization, Tunneling Function, One-Parameter

Copyright © 2026 by author(s) and Hans Publishers Inc.

This work is licensed under the Creative Commons Attribution International License (CC BY 4.0).

<http://creativecommons.org/licenses/by/4.0/>



Open Access

1. 引言

随着计算科学和工程技术的发展, 全局优化算法在科学研究、工程设计、经济管理等诸多领域得到了广泛应用。然而, 由于其复杂性和多样性, 全局优化问题一直是优化领域的一个重大挑战。与传统的局部优化算法不同, 全局优化算法需要在复杂的搜索空间中找到全局最优解, 而不是局部最优解。为了达到这一目标, 研究者们提出了各种方法和策略。目前, 对于求解全局优化问题的方法主要分为两类: 确定性算法和随机性算法。打洞函数法是求解优化问题的一类有效确定性算法, 其关键在于打洞函数的设计。当算法陷入局部极小点时, 通过构造一个在当前局部极小点处符合特定性质的打洞函数, 并使用局部优化算法来求得这个打洞函数的局部极小点 x^* , 再从 x^* 附近出发, 利用局部优化算法, 求目标函数的下一个局部极小点, 算法就能跳出当前的局部极小点, 并寻找到更优的局部极小点, 重复以上步骤, 直到满足终止条件, 输出全局极小点或近似全局极小点。

打洞函数最早是 Levy 和 Montalvo 在文献[1]中提出, 其形式如下:

$$T(x, x^*) = \frac{f(x) - f(x^*)}{\left[(x - x^*)^T (x - x^*) \right]^\alpha},$$

其中 α 是 $(x - x^*)^T (x - x^*)$ 的强度, 然而实际应用中发现, 当强度 α 足够大时, 算法才会有效, 但这样不仅需要重新初始化计算流程导致计算效率降低, 还可能因函数形态过度平坦化而出现假的平稳点问题。为克服上述缺陷, 后续研究提出了多种改进方案。Yao 在文献[2]中提出了动态打洞算法, 通过能量函数来完成打洞过程, 但需要求解非线性动力系统显著增加了计算复杂度。Oblov 提出了随机打洞算法[3], 提升了全局搜索能力, 但常微分方程组的求解仍带来额外计算负担。然后, 李忠豪在文献[4]中提出了变形打洞函数算法, 该算法避免了打洞函数的缺点, 同时也不需要求解常微分方程, 然而收敛效率仍有优化空间。后来, Xu 等在文献[5]中进一步改进了变形打洞函数的定义, 并将定义推广到研究非光滑规划问题中。针对文献[5]所提出的定义, 本文构造了一种单参的打洞函数, 并将其应用于操作系统参数调优中, 结果表明该算法是有效的。对于问题有如下假设:

2. 定义与假设

本文考虑以下的无约束全局优化问题:

$$\begin{aligned} \min & f(x) \\ \text{s.t.} & x \in \mathbb{R}^n, \end{aligned}$$

对于问题有如下假设:

假设 1 目标函数 $f(x)$ 在 R^n 上是局部 Lipschitz。

假设 2 目标函数 $f(x)$ 是强制性的, 即当 $\|x\| \rightarrow +\infty$ 时, 有 $f(x) \rightarrow +\infty$ 。

注 1 由假设 2 可知, 一定存在一个有界闭箱 Ω , 使得目标函数 $f(x)$ 的全局极小点在其内部。因此, 问题可以重新表述为:

$$\begin{aligned} \min & f(x) \\ \text{s.t.} & x \in \Omega. \end{aligned}$$

假设 3 目标函数 $f(x)$ 有有限个局部极小值, 但其局部极小点可以是无限个。

为了方便起见, 记 $L(P)$ 为目标函数 $f(x)$ 的所有局部极小点的集合, $G(P)$ 为目标函数 $f(x)$ 的所有全局极小点的集合。本文采用的是文献[5]中的定义:

定义 1 令 $x_k^* \in L(P)$ 是 $f(x)$ 的一个局部极小点, 定义函数 $T(x, x_k^*, r)$ 是 $f(x)$ 在点 x_k^* 处的打洞函数, 若 $T(x, x_k^*, r)$ 满足:

- (1) $T(x, x_k^*, r) = 0$ 当且仅当 $f(x) - f(x_k^*) + r < 0$ 成立, 其中 $r > 0$ 充分小;
- (2) 对于任意的 $x \in U_1 = \{x \in \Omega: f(x) \geq f(x_k^*), x \neq x_k^*\}$, 有 $0 \notin \partial T(x, x_k^*, r)$;
- (3) 若 $U_2 = \{x \in \Omega: f(x) < f(x_k^*)\}$ 非空, 则一定存在 $x_2^* \in U_2$ 是 $T(x, x_k^*, r)$ 的局部极小点。

3. 一个新的单参数打洞函数及其性质

下面引入一个在定义 1 下的打洞函数。对于 $x_1^* \in L(P), x_0 \notin \Omega$, 定义:

$$T(x, x_k^*, r) = \|x - x_0\|^2 \cdot \frac{\max\{0, f(x) - f(x_k^*) + r\}}{f(x) - f(x_k^*) + r},$$

其中, r 是一个足够小的正数。

注 1 x_0 位于有界闭箱外部, 而 x 位于有界闭箱内部, 因此选取 $\|x - x_0\|^2$ 作为外部乘子项可以保证函数的有效性。

接下来, 本文将详细证明所提出的打洞函数满足这三条性质:

定理 1 $T(x, x_k^*, r) = 0$ 当且仅当 $f(x) - f(x_k^*) + r < 0$ 成立, 其中 $r > 0$ 充分小。

证明: $T(x, x_k^*, r) = 0$ 当且仅当 $\max\{0, f(x) - f(x_k^*) + r\} = 0$, 则当且仅当 $f(x) - f(x_k^*) + r < 0$ 。证毕。

定理 2 对于任意的 $x \in U_1 = \{x \in \Omega: f(x) \geq f(x_k^*), x \neq x_k^*\}$, 有 $0 \notin \partial T(x, x_k^*, r)$ 。

证明: 若 $x \in U_1 = \{x \in \Omega: f(x) \geq f(x_k^*), x \neq x_k^*\}$, 则 $f(x) - f(x_k^*) + r > 0$ 。从而

$$T(x, x_k^*, r) = \|x - x_0\|^2,$$

所以

$$\partial T(x, x_k^*, r) = \nabla T(x, x_k^*, r),$$

由此可知

$$\nabla T(x, x_k^*, r) \cdot \frac{x - x_0}{\|x - x_0\|^2} = 2.$$

故 $0 \notin \partial T(x, x_k^*, r)$ 。证毕。

定理 3 若 $U_2 = \{x \in \Omega: f(x) < f(x_k^*)\}$ 非空, 则一定存在 $x^* \in U_2$ 是 $T(x, x_k^*, r)$ 的局部极小点。

证明: 因为 $x_k^* \in L(P)$ 但 $x_k^* \notin G(P)$, 则存在 $f(x)$ 的另一个局部极小点 x^* 且 $f(x^*) < f(x_k^*)$ 。由于函数 $f(x)$ 在 R^n 上是局部 Lipschitz 函数, 则存在 x^* 的一个邻域 $N(x^*, \delta)$, 当 $x \in N(x^*, \delta)$ 时, 有

$f(x^*) < f(x) < f(x_k^*)$ 。令 r 满足 $f(x^*) - f(x_k^*) + r \leq 0$, 且 $f(x) - f(x_k^*) + r > 0$ 。则有 $T(x, x_k^*, r) = \|x - x_0\|^2 \cdot 0 = 0$ 和 $T(x, x_k^*, r) = \|x - x_0\|^2 > 0$ 。可知 x^* 是 $T(x, x_k^*, r)$ 的局部极小点。证毕。

综上, 由定理 1~3 证得 $T(x, x_k^*, r)$ 是一个打洞函数。

注 2 本文所构造的打洞函数选取 $\|x - x_0\|^2$ 作为外部乘子项, 可以兼顾函数的光滑性与凸性, 同时规避传统打洞函数因乘子项选择不当导致的假平稳点、函数形态平坦化问题。此外 $\|x - x_0\|^2$ 是连续可微的二次函数, 与目标函数结合后, 能保证打洞函数整体的光滑性(若目标函数光滑), 则适配拟牛顿法等局部优化算法的求解要求; 即便目标函数非光滑(局部 Lipschitz), 平方范数的非负性也能维持打洞函数的局部 Lipschitz 性质, 符合论文假设 1。

注 3 二次项天然具备强凸性, 可有效抵消目标函数局部非凸带来的影响, 确保打洞函数在邻域内具有唯一极小点; 该性质直接支撑定理 2 中关于逃离已知极小点的严格数学保证, 避免优化过程陷入冗余停滞。

4. 打洞函数算法

下面给出一个单参数的打洞函数算法, One-Parameter Tunneling Function Method, 以下简称 OTFM 算法:

步骤 0 初始化。选择参数 $r_0 > 0$ 作为 r 的下界。选取步长 $\alpha > 0$, d_k 是随机方向, 其中 $k = 1, 2, \dots, k_0$, $k_0 \geq 2n$, 这里 n 是变量的维数。取 m 、 K 分别为内外循环次数的上界。令 $i := 1, k := 1, r := 1$;

步骤 1 从初始点 x_k^0 开始用局部优化算法求得目标函数 $f(x)$ 的局部极小点 x_k^* 。

步骤 2 构造在点 x_k^* 处的打洞函数

$$T(x, x_k^*, r) = \|x - x_0\|^2 \cdot \frac{\max\{0, f(x) - f(x_k^*) + r\}}{f(x) - f(x_k^*) + r}.$$

步骤 3 若 $i \leq m$, 则选取点 $x'_k = x_k^* + \alpha d_i \in \Omega$ 为新的初始点, 并极小化打洞函数 $T(x, x_k^*, r)$, 得到 $T(x, x_k^*, r)$ 的一个局部极小点为 x_{rk}^* , 转步骤 4; 否则转步骤 6。

步骤 4 若 $x_{rk}^* \in \Omega$, 转步骤 5; 否则, 令 $i := i + 1$, 转步骤 3。

步骤 5 从点 x_{rk+1}^* 出发, 利用局部优化算法求得目标函数 $f(x)$ 的一个局部极小点 x_{k+1}^* , 若函数值 $f(x_{k+1}^*) < f(x_k^*)$, 令 $f(x_{k+1}^*) := f(x_k^*)$, $x_k^* := x_{k+1}^*$, $k := k + 1$, 转步骤 2; 否则, 转步骤 6。

步骤 6 若 $k \leq K$, 令 $r = 0.1r$, 令 $i := 1$; 若 $r < r_0$, 令 $r := r_0$, 转步骤 3; 否则算法停止, 并输出 x_k^* 为近似全局极小点, $f(x_k^*)$ 为近似全局极小值。

注 4 在求解光滑问题时, 局部优化算法可以使用拟牛顿法; 在求解非光滑问题时, 局部优化算法可以使用次梯度法。

5. 数值算例及其实验结果

下面给出几个数值算例, 用来验证此算法的有效性。该算法是基于 Python 3.11 实现的。符号的表达意义如下:

k : 求解第 k 次局部极小点时的迭代次数;

x_k : 第 k 次迭代时的初始点;

$f(x_k)$: 目标函数在点 x_k 处的函数值;

x_k^* : 第 k 次迭代时求得的局部极小点;

$f(x_k^*)$: 目标函数在点 x_k^* 处的函数值。

算例 1 [6]

$$\begin{aligned} \min f(x) &= 2x^2 - 1.05x^4 + \frac{1}{6}x - |x| \\ \text{s.t. } &-0.8 \leq x \leq 1. \end{aligned}$$

可以得到函数的全局极小点为 -0.329111 , 全局极小值为 -0.179652 。选取不同的初始点分别为 0.5 与 -0.6 , 计算结果见下表表 1。

Table 1. The experimental results of Example 1**表 1.** 算例 1 实验结果

k	x_k	$f(x_k)$	x_k^*	$f(x_k^*)$
1	0.5	0.208333	0.3192	-0.073106
2	-0.1634	-0.138025	-0.3291	-0.179652
1	-0.6	-0.087600	-0.329111	-0.179652

算例 2 [7]

$$\begin{aligned} \min f(x) &= \max \{5x_1 + x_2, -5x_1 + x_2, x_1^2 + x_2^2 + 4x_2\} \\ \text{s.t. } &-4 \leq x_1 \leq 4, -4 \leq x_2 \leq 4. \end{aligned}$$

算法可以得到全局极小点为 $(0, -3)^T$, 对应的全局最优值为 $f(x^*) = -3$ 。计算结果见下表 2。

Table 2. The experimental results of Example 2**表 2.** 算例 2 实验结果

k	x_k	$f(x_k)$	x_k^*	$f(x_k^*)$
1	$(1, 1)^T$	6	$(-0.2548, -2.8476)^T$	-2.846372
2	$(0.0014, -2.9999)^T$	-2.999999	$(0, -3)^T$	-3

算例 3 [8]

$$\begin{aligned} \min f(x) &= \sum_{i=1}^5 i [\cos[(i+1)x + i]] + 5 \\ \text{s.t. } &-10 \leq x \leq 10. \end{aligned}$$

算法成功找到了一个近似全局极小点为 -4.255752 , 对应的近似全局最优值为 6.699793 。选取不同初始点分别为 -5 与 10 , 具体数值见下表 3。

Table 3. The experimental results of Example 3**表 3.** 算例 3 实验结果

k	x_k	$f(x_k)$	x_k^*	$f(x_k^*)$
1	-5	15.698445	-4.0975	15.495723
2	-3.0697	14.728630	-1.6087	13.138000
3	-4.2550	6.706294	-4.2557	6.699793
1	10	15.011024	9.0550	14.246819
2	8.3105	6.700388	8.3106	6.699793

算例 4 [8]

$$\min f(x) = \sum_{i=1}^n |x_i - 0.5|$$

$$\text{s.t. } -5 \leq x_i \leq 5, i = 1, 2, \dots, n.$$

对任意的 n , 算法都成功找到了全局极小点 $(0.5, 0.5, \dots, 0.5)^T$, 全局极小值 0。取 $n = 4$, 得到具体数值见表 4。

Table 4. The experimental results of Example 4

表 4. 算例 4 实验结果

k	x_k	$f(x_k)$	x_k^*	$f(x_k^*)$
1	$\begin{pmatrix} 3 \\ 3 \\ 3 \\ 3 \end{pmatrix}$	10	$\begin{pmatrix} 2.0782 \\ 1.0788 \\ 1.1703 \\ -0.0727 \end{pmatrix}$	5.63944075
2	$\begin{pmatrix} 0.6180 \\ 1.0954 \\ 0.5987 \\ 0.8046 \end{pmatrix}$	1.11699193	$\begin{pmatrix} 0.5 \\ 0.5 \\ 0.5 \\ 0.5 \end{pmatrix}$	0

算例 5 [9]

$$\min f(x) = \sum_{i=1}^n [x_i^2 - 10\cos(2\pi x_i) + 10]$$

$$\text{s.t. } -5 \leq x_i \leq 5, i = 1, 2, \dots, n.$$

选取 $n = 8$, 得到结果见表 5。

Table 5. The experimental results of Example 5

表 5. 算例 5 实验结果

k	x_k	$f(x_k)$	x_k^*	$f(x_k^*)$
1	$\begin{pmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \end{pmatrix}$	8	$\begin{pmatrix} -0.07277745 \\ -0.02831491 \\ -0.07277745 \\ -0.07277745 \\ -0.02831491 \\ -0.07277745 \\ -0.02831491 \\ -0.07277745 \end{pmatrix}$	5.639440
2	$\begin{pmatrix} -0.07277745 \\ -0.02831491 \\ -0.07277745 \\ -0.02831491 \\ -0.07277745 \\ -0.02831491 \\ -0.07277745 \end{pmatrix}$	5.639440	$\begin{pmatrix} -7.31283908e-09 \\ -7.38194029e-09 \\ -7.43364342e-09 \\ -7.43364429e-09 \\ -7.29981432e-09 \\ -7.29981431e-09 \\ -7.29981431e-09 \\ -7.29981431e-09 \end{pmatrix}$	0

可以看出, 在求解可微全局优化问题时, OTFM 算法同样有效。

将 OTFM 算法与过往文献进行比较, 比较的结果见表 6。

Table 6. Comparison of results

表 6. 结果对比

算例	对比	迭代次数	初始点	全局极小点	全局极小值
1	OTFM 算法	2	0.5	-0.329111	-0.179652
1	OTFM 算法	1	-0.6	-0.329111	-0.179652
1	文献[6]算法	2	0.5	-0.329111	-0.179652
2	OTFM 算法	2	$\begin{pmatrix} 2 \\ 2 \end{pmatrix}$	$\begin{pmatrix} 0 \\ -3 \end{pmatrix}$	-3
2	文献[7]算法	3	$\begin{pmatrix} 2 \\ 2 \end{pmatrix}$	$\begin{pmatrix} 0 \\ -3 \end{pmatrix}$	-3
3	OTFM 算法	3	-5	-4.255752	6.699793
3	OTFM 算法	2	10	8.310618	6.699793
3	文献[8]算法	3	3	2.027586	6.699800

将表6中的OTFM算法与文献进行对比, 可以看出, 算例1中算法的迭代次数要少于文献[6]算法的迭代次数, 算例2中算法的迭代次数少于文献[7]算法的迭代次数; 此外算例3的OTFM算法找到了比文献[8]更好的近似全局极小点。

6. 结语

本文提出一种用于求解无约束全局优化问题的打洞函数。该函数也可适用于不可微问题的求解, 且仅含单一参数, 从而简化了参数调节过程。基于此构造的打洞函数算法形式简洁, 便于计算实现。数值算例结果表明, 该算法具有较好的有效性与可行性。在此基础上, 可进一步将该方法与机器学习算法相结合, 以展开更深入的理论探讨与应用研究。

参考文献

- [1] Levy, A.V. and Montalvo, A. (1985) The Tunneling Algorithm for the Global Minimization of Functions. *SIAM Journal on Scientific and Statistical Computing*, **6**, 15-29. <https://doi.org/10.1137/0906002>
- [2] Yao, Y. (1989) Dynamic Tunneling Algorithm for Global Optimization. *IEEE Transactions on Systems, Man, and Cybernetics*, **19**, 1222-1230. <https://doi.org/10.1109/21.44040>
- [3] Oblow, E.M. (2001) SPT: A Stochastic Tunneling Algorithm for Global Optimization. *Journal of Global Optimization*, **20**, 191-208. <https://doi.org/10.1023/A:1011265010691>
- [4] 李忠豪. 全局优化的填充函数和变形打洞函数法[D]: [博士学位论文]. 上海: 上海大学, 2012.
- [5] Xu, Y.T., Zhang, Y. and Wang, S.G. (2015) A Modified Tunneling Function Method for Non-Smooth Global Optimization and Its Application in Artificial Neural Network. *Applied Mathematical Modelling*, **39**, 6438-6450. <https://doi.org/10.1016/j.apm.2015.01.059>
- [6] 蔡珍珍. 非线性全局优化问题填充函数法的研究[D]: [硕士学位论文]. 重庆: 重庆大学, 2016.
- [7] 王伟祥, 孙广磊. 求解非光滑全局优化问题的单参数填充函数算法[J]. 上海第二工业大学学报, 2022, 39(3): 251-255.
- [8] 张莹. 非光滑规划全局优化的填充函数法[D]: [博士学位论文]. 上海: 上海大学, 2009.
- [9] 李佳媛. 基于填充函数的神经网络优化算法[D]: [硕士学位论文]. 金华: 浙江师范大学, 2024.