

A New Type of Non-Parameter Filled Function and Its Application in Least Squares

Yulong Li¹, Shenggang Wang², Ying Zhang¹, Xiaoying Xie³

¹College of Mathematics and Computer Science, Zhejiang Normal University, Jinhua Zhejiang

²College of Agriculture and Biological Engineering, Jinhua Polytechnic, Jinhua Zhejiang

³College of Economics and Management, Zhejiang Normal University, Jinhua Zhejiang

Email: eke_li@163.com

Received: Apr. 6th, 2019; accepted: Apr. 21st, 2019; published: Apr. 28th, 2019

Abstract

In this paper, a novel non-parameter filled function for solving general constrained global optimization problem is proposed. Then the theoretical properties of the function are argued and corresponding algorithm is given in this paper. Numerical experiments using the Python programming language and comparison with previous results show that the proposed filled function algorithm is not only effective but also works better when dealing with certain function. Furthermore, the algorithm is tentatively combined with the treatment of the reaction rate and reactant concentration of the enzymatic reaction of chemical experiments, indicating that the algorithm also has good adaptability in practical cases.

Keywords

Global Optimization, Non-Parameter Filled Function, Enzymatic Reaction, Least Squares

一类新的无参数填充函数及其在最小二乘法中的应用

李玉龙¹, 王胜刚², 张莹¹, 谢笑盈³

¹浙江师范大学, 数学与计算机科学学院, 浙江 金华

²金华职业技术学院, 农业与生物工程学院, 浙江 金华

³浙江师范大学, 经济与管理学院, 浙江 金华

Email: eke_li@163.com

收稿日期: 2019年4月6日; 录用日期: 2019年4月21日; 发布日期: 2019年4月28日

摘要

针对一般约束优化问题, 本文提出了一类新的无参数填充函数, 分析该函数的理论性质并设计算法, 利用Python编程语言进行数值实验并且与前人结果进行比较, 表明该填充函数算法不仅是有效的而且在处理某些函数时效果更好。进一步地, 尝试将该算法与化学实验酶促反应的反应速度和反应浓度关系的处理结合, 实验结果表明该算法在实际应用中也有较好的适应性。

关键词

全局优化, 无参数填充函数, 酶促反应, 最小二乘

Copyright © 2019 by authors and Hans Publishers Inc.

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

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



Open Access

1. 引言

非线性规划, 作为数学规划的重要组成部分在经营管理、工程设计、科学研究、军事指挥等方面得到广泛的应用。近年来有关全局优化的理论和算法层出不穷。全局优化方法大致可分为三类: 1) 在局部极小点中搜寻全局极小点, 一般来说就是调用辅助函数跳出当前局部极小点得到更优的局部极小点; 2) 采用启发式或者随机搜索方法如模拟退火算法, 遗传算法、人工神经网络法等; 3) 针对特殊结构的问题采用特定的解决方法如 D.C 规划。而本文所研究的内容是属于第一类, 构造填充函数来求取全局极小点。填充函数法首先由西安交通大学葛仁溥教授[1]在 1984 年提出, 是一类较为有效的确定性全局优化方法。随后其在文献[2]中对最初提出的双参数填充函数进行改进, 提出了一类性能更优的单参数填充函数。

上海大学的张连生教授[3]对填充函数的定义进行改进, 并提出一系列性能更优的填充函数; 而且在张教授科研团队的深入研究下, 填充函数内容越来越丰富。如: 针对非线性等式约束问题, 杨永健教授在文献[4]中提出了一类单参数填充函数; 针对一般约束性优化问题, 王伟祥教授在文献[5]中提出了一类单参数填充函数; 张莹在文献[6]中针对非光滑约束问题提出了一类单参数填充函数。

在分析现有填充函数时发现多数研究者所构造的填充函数或者带有指数项[7], 或者结构比较复杂[8], 本文提出一种新的不带指数项的无参数填充函数, 具有结构简单的优点。通过分析该填充函数的性质知: 若填充函数 $p(x, x^*) > 0$, 则说明在 x 的邻域内一定存在比当前局部极小点的值更小的局部极小点。这个优良性质应用到算法设计中, 可以提升算法的实用性。通过几个数值算例, 表明该算法是有效的。最后通过对化学实验酶促反应中反应速度和反应浓度的关系数据的处理得知: 该算法在实际应用中也有较好的适应性。

2. 新的无参数填充函数

考虑如下问题:

$$(P) \begin{cases} \min f(x), \\ \text{s.t. } x \in R^n. \end{cases} \quad (2.1)$$

其中 $f: R^n \rightarrow R$ 是连续可微函数。

假设 2.1: 函数 $f(x)$ 是强制性的, 也就是当 $\|x\| \rightarrow +\infty$ 时, 有 $f(x) \rightarrow +\infty$ 。

由假设 2.1 可以推出, 一定存在一个紧集 $X \subset R^n$, 它的内部包含了 $f(x)$ 的所有全局极小点和函数值较小的局部极小点, 而且还可以认为目标函数在 X 的边界上的函数值大于在其内部的函数值, 因此, 问题(2.1)可等价于如下问题:

$$\begin{cases} \min f(x), \\ \text{s.t. } x \in X \end{cases} \quad (2.2)$$

假设 2.2: 问题(2.2)的不同局部极小点的个数是有限个。

下面给出葛仁浦教授在文献[1]中提出的填充函数的定义。

定义 2.1: 函数 $p(x, x^*)$ 称为 $f(x)$ 在局部极小点 x^* 处的填充函数, 如果满足:

- 1) x^* 是 $p(x, x^*)$ 的一个严格局部极大点, $f(x)$ 在点 x^* 处的盆谷 B^* 成为 $p(x, x^*)$ 的峰的一部分。
- 2) $p(x, x^*)$ 在比 B^* 高的盆谷里没有平稳点, 即 $\nabla p(x, x^*) \neq 0$ 。
- 3) 如果存在比 B^* 低的盆谷 B_1^* , 则在 x' 和 x^* 的连线上极小化 $p(x, x^*)$ 得到极小点 $x'' \in B^*$, 其中, $N(x_1^*, \delta_1)$ ($\delta_1 > 0$), $x' \in N(x_1^*, \delta_1)$ 。

定义 2.2: 函数 $f(x)$ 在一局部极小点 x^* 处的盆谷是指一连通域 B^* , 具有下列性质:

- 1) $x^* \in B^*$;
- 2) 对于任意一点 $x \in B^*$ 使得 $x \neq x^*$ 及 $f(x) > f(x^*)$, 存在一条从 x 到 c 的下降路径。

针对问题(2.1), 提出了一类无参数填充函数如下:

$$p(x, x^*) = -\ln(1 + \|x - x^*\|^2)(f(x) - f(x^*)). \quad (2.3)$$

其中 x^* 是目标函数 $f(x)$ 的当前局部极小点。

下面来证明该函数是符合定义(2.1)的填充函数。

定理 2.1: x^* 是 $p(x, x^*)$ 的严格局部极大点。

证明: 因为 x^* 是 $f(x)$ 的一个局部极小点, 则存在它的一个邻域 $N(x^*, \delta^*)$ ($\delta^* > 0$), 使得对于任意 $x \in N(x^*, \delta^*) \setminus \{x^*\}$, 有 $f(x) > f(x^*)$ 。则有,

$$\begin{aligned} p(x, x^*) &= -\ln(1 + \|x - x^*\|^2)(f(x) - f(x^*)) \\ &< 0 = p(x^*, x^*) \end{aligned}$$

证毕。

定理 2.2: $p(x, x^*)$ 在比 B^* 高的盆谷 B_1^* 里没有平稳点, 即 $\nabla p(x, x^*) \neq 0$ 。

证明: 由定理条件可知: 对 $\forall x \in B_1^*$, 有 $f(x) > f(x^*)$, 又因为 $f(x)$ 在 R^n 上连续可微, 可以得到 $f(x)$ 在 x^* 处的泰勒展开式为:

$$f(x) = f(x^*) + (x - x^*)^T \nabla f(x) + o(\|x - x^*\|)$$

所以有:

$$f(x) - f(x^*) = (x - x^*)^T \nabla f(x) + o(\|x - x^*\|) > 0$$

则有: $(x - x^*)^T \nabla f(x) > 0$ 。 $p(x, x^*)$ 梯度如下:

$$\nabla p(x, x^*) = - \left[\frac{2(x - x^*)}{1 + \|x - x^*\|^2} (f(x) - f(x^*)) + \ln(1 + \|x - x^*\|^2) \nabla f(x) \right]$$

于是

$$\nabla p(x, x^*)^T (x - x^*) = \left[\frac{2\|x - x^*\|^2}{1 + \|x - x^*\|^2} (f(x) - f(x^*)) + \ln(1 + \|x - x^*\|^2) \nabla f(x)^T (x - x^*) \right] < 0$$

所以 $\nabla p(x, x^*) \neq 0$ 。证毕。

定理 2.3: 如果 x^* 不是全局极小点, 并且 x_1^* 是离 x^* 最近的一个局部极小点并且满足 $f(x_1^*) < f(x^*)$, 则对 $\forall x' \in N(x_1^*, \delta_1^*) (\delta_1^* > 0)$ 则在 x' 和 x^* 的连线上极小化 $p(x, x^*)$, 得到 $p(x, x^*)$ 局部极小点 $x'' \in B^*$ 。

证明: 因为 x^* 是 $f(x)$ 的局部极小点, 则存在 x^* 的小邻域 $N(x^*, \delta^*) (\delta^* > 0)$, 对任意的 $x \in N(x^*, \delta^*)$, 有 $f(x) \geq f(x^*)$, 则有

$$p(x, x^*) = -\ln(1 + \|x - x^*\|^2) (f(x) - f(x^*)) < 0$$

其中 $x \in N(x^*, \delta^*)$ 且 $x \neq x^*$ 。

相似地, 对于 x_1^* , 则存在它的一个小邻域 $N(x_1^*, \delta_1^*) (\delta_1^* > 0)$, 对任意的 $x' \in N(x_1^*, \delta_1^*)$, 有 $f(x^*) > f(x') \geq f(x_1^*)$, 显然有:

$$p(x', x^*) = -\ln(1 + \|x' - x^*\|^2) (f(x') - f(x^*)) > 0$$

通过分析 x' 和 x^* 的轨迹可知: 在最初 $p(x^*, x^*) = 0$, x 从 x^* 处开始迭代搜索, 有 $p(x, x^*) < 0$, 而当 x 接近 x' 时, 可知 $p(x, x^*) > 0$, 所以可以知道在 $f(x) \geq f(x^*)$ 时, $p(x, x^*)$ 递减, 在 $f(x) < f(x^*)$ 时, $p(x, x^*)$ 递增。由此可以知道在 x' 和 x^* 的连线上存在 x'' 极小化 $p(x, x^*)$ 。证毕。

定理 2.1~定理 2.3 表明 $p(x, x^*)$ 是符合定义 2.1 的填充函数。下面的定理 2.4 表明该填充函数具有一个很好的性质, 能在算法执行中快速地找到更优的局部极小点。

定理 2.4: 若 x^* 是 $f(x)$ 的全局极小点, 则对任意的 $x \in X$ 都有 $p(x, x^*) \leq 0$ 成立; 否则存在点 x^{**} 以及它的邻域 $N(x^{**}, \delta) (\delta > 0)$, 使得对任意的 $x \in N(x^{**}, \delta)$, 成立 $p(x, x^*) > 0$ 。

证明: 若 x^* 是 $f(x)$ 的全局极小点, 则对任意 $x \in X$, 都有 $f(x) \geq f(x^*)$, 所以 $p(x, x^*) = -\ln(1 + \|x - x^*\|^2) (f(x) - f(x^*)) \leq 0$; 若 x^* 不是 $f(x)$ 的全局极小点, 则存在点 x^{**} 使得 $f(x^{**}) < f(x^*)$, 由 $f(x)$ 的连续性可知存在 x^{**} 的邻域 $N(x^{**}, \delta^{**}) (\delta^{**} > 0)$, 则 $\forall x \in N(x^{**}, \delta^{**})$ 有 $f(x) < f(x^*)$ 。所以有 $p(x, x^*) > 0$ 。证毕。

注: 由定理 2.2 的证明可以发现 $d = x - x^*$ 是填充函数 $p(x, x^*)$ 的一个下降方向; 通过定理 2.4 可以发现, 当遇到 $p(x, x^*) > 0$ 时, 可以知道 $f(x) < f(x^*)$, 则在算法实施过程中说明找到一个比当前局部极小点的值更小的点。

3. 填充函数算法

填充函数法是求解全局优化问题的有效方法, 该方法的基本思想是: 在可行域 X 中选取一点 x_0 , 利用已有的局部优化算法如最速下降法、共轭梯度法和拟牛顿法等来求出目标函数 $f(x)$ 的一个局部极小点 x^* , 然后在 x^* 处构造一个填充函数, 在 x^* 的某个邻域内选取一点作为初始点, 极小化填充函数, 得到一点 x_1^* , 使得 $f(x_1^*) < f(x^*)$; 然后用 x_1^* 代替 x_0 并重复上述过程, 直到找到全局极小点。根据这种思想, 结合本文所提出填充函数的特点以及文献的算法[9], 提出相应改进算法。步骤如下:

步骤 0: 初始化数据, 设定搜索步长 $\delta > 0, k = 1$, 选择方向 $d_i (i = 1, 2, \dots, m), m \geq 2n$, 其中 n 为变量

的维度，选取一个初始点 $x_0 \in X$ 。

步骤 1: 以 x_0 为初始点，应用局部优化算法如拟牛顿法求出 $f(x)$ 的一个局部极小点 x^* ，并求出局部极小值 $f(x^*)$ 。

步骤 2: 设置 $i=1$ 。

步骤 3: 如果 $i \leq m$ ，转步骤 4；否则 x^* 作为全局极小点，停止计算。

步骤 4: 令 $x_1 = x^* + \delta d_i$ ，构造 $f(x)$ 在 x^* 处的填充函数 $P(x, x^*)$ 。然后从 x_1 开始使用局部优化算法得到 $P(x, x^*)$ 的局部极小点 x_1^* 。

步骤 5: 令 $x = x_1^*$ 。

步骤 6: 若 $p(x, x^*) > 0$ ，则令 $k = k + 1$ 且 $x_0 = x$ ，转步骤 1，否则令 $i = i + 1$ ，转步骤 2，若 $x \notin X$ ，则令 $i = i + 1$ ，转到步骤 3。

在步骤 4 中的 x_1 一般是这样选取的： $m = 2n$ ，令 x_1 的选取是对称的。例如对 $n = 2$ ， x_1 为：

$$x^* + \delta(1, 0), x^* + \delta(0, 1), x^* - \delta(1, 0), x^* - \delta(0, 1)$$

4. 数值实验

为了验证本文所提出的无参数填充函数算法是有效的，应用上述算法，使用常用的几个测试函数，使用 Python3.5.6 进行编程计算。

4.1. 测试函数

1) 6-Hump Back Camel 函数[10]

$$\begin{cases} \min f(x) = 4x_1^2 - 2.1x_1^4 + \frac{1}{3}x_1^6 - x_1x_2 - 4x_2^2 + 4x_2^4 \\ \text{s.t. } -3 \leq x_1, x_2 \leq 3 \end{cases}$$

2) Restrign 函数[10]

$$\begin{cases} \min f(x) = x_1^2 + x_2^2 - \cos(18x_1) - \cos(18x_2) \\ \text{s.t. } -1 \leq x_1, x_2 \leq 1 \end{cases}$$

3) 文献[11]中的一个二维函数($c = 0.05$)

$$\begin{cases} \min f(x) = [1 - 2x_2 + c \sin(4\pi x_2) - x_1]^2 + [x_2 - 0.5 \sin(2\pi x_1)]^2 \\ \text{s.t. } -10 \leq x_1, x_2 \leq 10 \end{cases}$$

4.2. 数值结果

以表格形式给出数值结果，表 1~3 分别对应测试函数(1)~(3)的数值结果，表 4 是文献[11]算例与本文算例的数值对照表。下面是表格中符号的解释。

k ：求解第 k 次局部极小点的迭代步数；

x_k ：满足 $x_k \in X$ 的第 k 次初始点；

x_k^* ：第 k 个局部极小点；

$f(x_k^*)$ ：第 k 个局部极小点的目标函数值；

iter：算法迭代步数；

x^* ：算法最终得到的全局极小点；

$f(x^*)$: 算法最终得到的全局极小值。

Table 1. Original value $x = (1, 2)^T$

表 1. 初始点为 $x = (1, 2)^T$

k	x_k	x_k^*	$f(x_k^*)$
1	$\begin{pmatrix} 1 \\ 2 \end{pmatrix}$	$\begin{pmatrix} 1.70360672 \\ 0.79608357 \end{pmatrix}$	-0.21546382438372103
2	$\begin{pmatrix} 0.96346164 \\ 0.76342669 \end{pmatrix}$	$\begin{pmatrix} 0.08984232 \\ 0.71265641 \end{pmatrix}$	-1.0316284534895122

Table 2. Original value $x = (1, 1)^T$

表 2. 初始点为 $x = (1, 1)^T$

k	x_k	x_k^*	$f(x_k^*)$
1	$\begin{pmatrix} 1 \\ 1 \end{pmatrix}$	$\begin{pmatrix} 1.04075871 \\ 1.04075871 \end{pmatrix}$	0.17977496636886192
2	$\begin{pmatrix} 0.69788109 \\ 0.69788109 \end{pmatrix}$	$\begin{pmatrix} 8.75480363e-12 \\ 8.75480363e-12 \end{pmatrix}$	-2.0

Table 3. $c = 0.05$, Original value $x = (6, -2)^T$

表 3. $c = 0.05$, 初始点为 $x = (6, -2)^T$

k	x_k	x_k^*	$f(x_k^*)$
1	$\begin{pmatrix} 6 \\ -2 \end{pmatrix}$	$\begin{pmatrix} 5.71982142 \\ -1.9162353 \end{pmatrix}$	2.7433787691978955
2	$\begin{pmatrix} 1.60870902 \\ -0.29679585 \end{pmatrix}$	$\begin{pmatrix} 1.59746304 \\ -0.2874076 \end{pmatrix}$	4.959859574394102e-15

Table 4. Compared with paper [11]

表 4. 本文算例与文献[11]算例对照表

	iter	x^*	$f(x^*)$
本文算例 1	2	$(0.089842, 0.71265641)$	-1.3016284534898765
文献[11]	2	$(0.0899, 0.7127)$	-1.0316
本文算例 2	2	$\begin{pmatrix} 8.75480363e-12 \\ 8.75480363e-12 \end{pmatrix}$	-2.0
文献[11]	6	$(0.2718, 0.2718) \times e-4$	-2.0000
本文算例 3	2	$\begin{pmatrix} 1.59746304 \\ -0.2874076 \end{pmatrix}$	4.959859574394102e-15
文献[11]	5	$(1.8513, -0.4021)$	4.8751e-9

本文算例与文献[11]算例对比表明, 应用本文算法所得结果精度更高, 迭代步数更少, 特别对算例 3 而言, 更好的迭代步数得到更优的全局极小点。

5. 填充函数在实验数据处理中的应用

5.1. 背景和问题

酶是一种具有特异性的高效生物催化剂, 绝大多数的酶是活细胞产生的蛋白质。酶的催化条件温和, 在常温、常压下即可进行。酶催化的反应称为酶促反应, 要比相应的非催化反应要快 $10^3 \sim 10^{17}$ 倍。酶促反应动力学简称酶促动力学, 主要研究酶促反应的速度和底物(即反应物)浓度以及其他因素的关系。在底物浓度很低时酶促反应是一级反应; 当底物浓度处于中间范围时, 是混合级反应; 当底物浓度增加时, 向零级反应过度[12]。

某生化系学生为了研究嘌呤霉素在某项酶促反应中对反应速度与底物浓度之间关系的影响, 设计了两个实验, 一个实验中所使用的酶是经过嘌呤霉素处理的, 而另一个实验所用的酶是未经嘌呤霉素处理过的, 所得的实验数据见表 5。试根据问题的背景和这些数据建立一个合适的数学模型, 来反映这项酶促反应的速度与底物浓度以及嘌呤霉素处理与否之间的关系。

Table 5. Reaction rate and substrate concentration data in puromycin experiments

表 5. 嘌呤霉素实验中的反应速度与底物浓度数据

底物浓度/ppm		0.02		0.06		0.11		0.22		0.56	
反应速度	处理	76	47	97	107	0.11	139	159	152	191	201
	未处理	67	51	84	86	98	115	131	124	144	158

注: ppm = 0.001%。

本文仅对嘌呤霉素处理过的数据进行拟合, 为了方便处理, 对数据取平均值, 数据如表 6:

Table 6. Average the data after treatment with scorpion toxin

表 6. 对嘌呤霉素处理后数据取平均值处理

反应物浓度	0.02	0.06	0.11	0.22	0.56
反应速度	61.5	102	131	155.5	196

5.2. 分析与假设

记酶促反应速度为 y , 底物浓度为 x , 二者之间的关系写作 $y = f(x, \beta)$, 其中 β 为参数。由酶促反应的基本性质可知, 当底物浓度较小时, 反应速度大致与浓度成正比(即一级反应); 当底物浓度很大, 渐进饱和时, 反应速度将趋于一个固定值 - 最终反应速度(即零级反应)。下面是针对该种性质提出的模型:

指数增长模型

$$y = f(x, \beta) = \beta_1 (1 - e^{-\beta_2 x})$$

本文将求解系数 β 转换为下面无约束优化问题:

$$\begin{aligned} \min f(x) &= \sum_{i=1}^n (y_i - \hat{y}_i)^2 \\ &= \sum_{i=1}^n (\beta_1 (1 - e^{-\beta_2 x}) - \hat{y}_i)^2 \end{aligned}$$

给定初始值 $\beta_0 = (155, 14)^T$, 经过 2 次迭代得到 $\beta^* = (181.54213939, 13.10716834)^T$, 而文献[12]中所给出参数估计 $\beta_1 = (192.0969, 11.3846)^T$ 。本文目标函数值为 918.2109, 剩余标准差 RMSE = 17.4949, 而

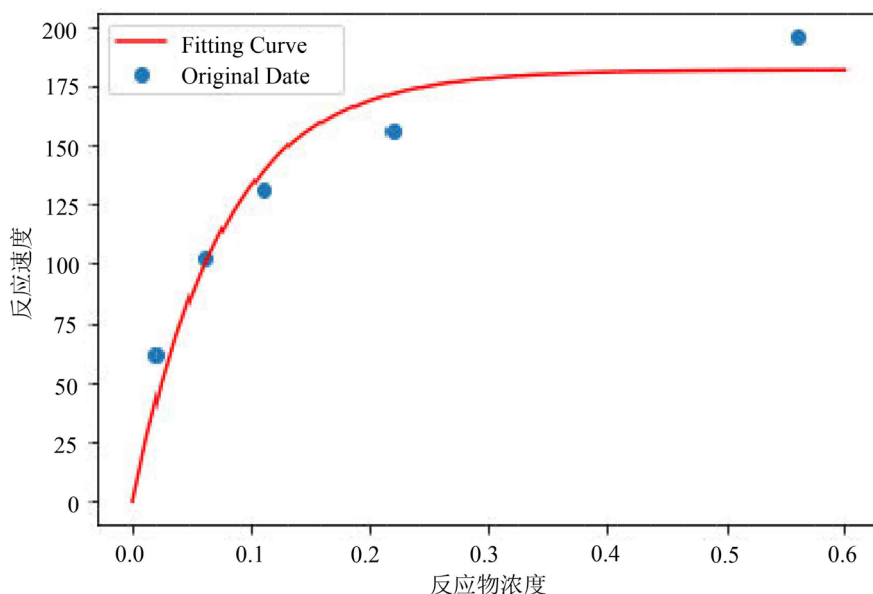


Figure 1. Fitting curve
图 1. 拟合曲线

文献[12]所得的目标函数值为1041.9798，剩余标准差 $RMSE = 18.6367$ 。由此可知，使用本文算法所得的全局极小值更小，拟合效果更好。所得拟合曲线如图 1 所示。

6. 结语

本文给出了一个形式简单，性能更优的填充函数，分析其性质并根据该性质设计算法，数值实验表明该算法是有效的，与文献[11]数值结果比较可知，该算法在处理某些函数时，有着较好的适应性。同时，本文也尝试使用该算法求解最小二乘问题，实验结果表明该算法在实际应用中也有较好的适应性。

基金项目

浙江省科技厅资助项目(LGN19C040001)；浙江省教育厅科研项目(Y201636028)；国家社科基金资助项目(17BTJ028)。

参考文献

- [1] Ge, R.P. (1990) A Filled Function Method for Finding a Global Minimizer of a Function of Several Variables. *Mathematical Programming*, **46**, 191-204. <https://doi.org/10.1007/BF01585737>
- [2] Ge, R.P. and Qin, Y.F. (1987) A Class of Filled Functions for Finding Global Minimizers of a Function of Several Variables. *Journal of Optimization Theory and Applications*, **54**, 241-252. <https://doi.org/10.1007/BF00939433>
- [3] Zhang, L.S., Ng, C.-K., Li, D. and Tian, W.W. (2004) A New Filled Function Method for Global Optimization. *Journal of Global Optimization*, **28**, 17-43. <https://doi.org/10.1023/B:JOGO.0000006653.60256.f6>
- [4] Lin, Y.J. and Yang, Y.J. (2010) Filled Function Method for Nonlinear Equations. *Journal of Computational and Applied Mathematics*, **234**, 695-702. <https://doi.org/10.1016/j.cam.2010.01.007>
- [5] 王伟祥, 尚有林, 张连生. 约束全局优化问题的一个单参数填充函数方法[J]. 工程数学学报, 2008, 25(5): 795-803.
- [6] Zhang, Y. and Xu, Y.T. (2009) A One-Parameter Filled Function Method Applied to Nonsmooth Constrained Global Optimization. *Computers & Mathematics with Applications*, **58**, 1230-1238. <https://doi.org/10.1016/j.camwa.2009.07.038>
- [7] Yang, Y.J. and Shang, Y.L. (2006) A New Filled Function for Unconstrained Global Optimizaton. *Applied Mathemat-*

ics and Computation, **173**, 501-512. <https://doi.org/10.1016/j.amc.2005.04.046>

- [8] He, S.X., Chen, W.L. and Wang, H. (2011) A New Filled Function Algorithm for Constrained Global Optimization Problems. *Applied Mathematics and Computation*, **217**, 5853-5859. <https://doi.org/10.1016/j.amc.2010.12.070>
- [9] 冉慧, 宋雪. 求解约束优化问题的无参数填充函数算法[J]. 重庆理工大学学报(自然科学版), 2013, 27(5): 132-136.
- [10] Lin, H.W., Wang, Y.P., Wang, X.L. and Gao, Y.L. (2012) A New Filled Function Method for Global Optimization with Box Constraint. *Journal of Information & Computational Science*, **10**, 2843-2853.
- [11] An, L., Zhang, L.S. and Chen, M.L. (2004) A Parameter-Free Filled Function for Global Unconstrained Optimization. *Journal of Shanghai University (English Edition)*, **215**, 3610-3619.
- [12] 姜启源, 谢金星, 叶俊. 数学模型 [M]. 第四版. 北京: 高等教育出版社, 2011.

知网检索的两种方式:

1. 打开知网页面 <http://kns.cnki.net/kns/brief/result.aspx?dbPrefix=WWJD>
下拉列表框选择: [ISSN], 输入期刊 ISSN: 2324-7991, 即可查询
2. 打开知网首页 <http://cnki.net/>
左侧“国际文献总库”进入, 输入文章标题, 即可查询

投稿请点击: <http://www.hanspub.org/Submission.aspx>

期刊邮箱: aam@hanspub.org