DOI: 10.13656/j.cnki.gxkx.2005.02.001
广西科学 Guangxi Sciences 2005, 12(2): 81~ 84

Hooke-Jeeves方法在简单约束优化中的推广*

Extensions of Hooke-Jeeves Method to Optimization with Simple Constraints

简金宝[1],罗 雁[2],徐庆娟[1]
Jian Jinbao[1], Luo Yan[2], Xu Qingjuan[1]

(1.广西大学数学与信息科学学院,广西南宁 530004; 2.钦州师范高等专科学校数学与计算机科学系,广西钦州 535000)
(1. Coll. of Math. & Info. Sci., Guangxi Univ., Nanning, Guangxi, 530004, China; 2. Dept. of Math. & Comp. Sci., Qinzhou Teachers Coll., Qinzhou, Guangxi, 535000, China)

摘要: 分别将无约束优化的线搜索和离散步 Hooke-Jeeves算法推广到带广义界的简单约束优化,产生 2个新算法,得到可行区间的计算公式.在适当条件下,证明线搜索的 Hooke-Jeeves算法推广后仍具有全局收敛性,算法有效数值试验表明 2个算法均是有效的.

关键词: 简单约束 最优化 Hooke-Jeeves方法 线搜索 离散步
中图法分类号: O221.2 文献标识码: A 文章编号: 1005-9164(2005)02-0081-04

Abstract The algorithm of Hooke-Jeeves using line searches and the algorithm of Hooke-Jeeves with discrete steps are extended such that they can solve simple constraints optimization with general bound. Two new algorithms are proposed. Under some suitable conditions, we prove that the extended algorithm of Hooke-Jeeves with line searches possesses global convergence. Some efficient numerical experiments are given.

Key words simple constraints, optimization, the method of Hooke-Jeeves, line search, discrete step

本文所研究的 Hooke-Jeeves方法也称步长加速法[1],它是无约束最优化方法中处理不可微函数时常用的一种方法.文献[2,3]将这种方法推广到约束优化.本文将 Hooke-Jeeves方法推广到带广义界的简单约束优化,并产生2个算法.这2个算法只需每次迭代时沿坐标方向或加速方向在可行区间内执行线搜索即可.所谓可行区间,就是为保证迭代点的可行性导出的步长或加速因子的所在区间.本文给出可行区间的计算公式.在适当条件下,证明推广的线搜索 Hooke-Jeeves算法的全局收敛性,最后通过数值试验说明文中算法的有效性.

1 线搜索算法和离散步算法

考虑如下一般形式的 n 维简单约束问题:

(P) $\min\{f(x) \mid x \in R\}$,

其中,可行集 $R = \{x \in E^n \mid l \leqslant x \leqslant u\}$, $l_i \in E^1 \cup \{-\infty\}$, $u_i \in E^1 \cup \{+\infty\}$, $i = 1,\cdots,n$. 记 $L^\infty = \{j \mid l_j = -\infty\}$, $U^\infty = \{j \mid u_j = +\infty\}$.

1.1 线搜索算法

步骤 0 取终止参数 $\chi > 0$,初始可行点 x^1, $l \leqslant x^1 \leqslant u$, $d^1,\cdots,d^n$ 为坐标方向,令 $y^1 = x^1$, $k = j = 1$.

步骤 1 计算可行区间 $[\underline{\lambda_j^*}, \overline{\lambda_j^*}]$

$$[\underline{\lambda_j^*}, \overline{\lambda_j^*}] = \begin{cases} [l_j - y_j^j, u_j - y_j^j], j \notin L^\infty, j \notin \infty; \\ (-\infty, u_j - y_j^j], j \in L^\infty; \\ [l_j - y_j^j, +\infty), j \in U^\infty. \end{cases} \quad (1.1)$$

求 $\min\{f(y^j + \lambda d^j) \mid \lambda \in [\underline{\lambda_j^*}, \overline{\lambda_j^*}]\}$的最优解 λ_j, 令 $y^{j+1} = y^j + \lambda_j d^j$. 如果 $j < n$, 令 $j := j + 1$, 重复步骤 1. 如果 $j = n$, 令 $x^{k+1} = y^{n+1}$. 如果 $\| x^{k+1} - x^k \|$

收稿日期: 2004-09-28
修回日期: 2004-10-08
作者简介: 简金宝(1964-),男,广西东兰人,教授,主要从事最优化理论与方法研究。
* 国家自然科学基金(10261001)和广西科学基金(0236001, 0249003)联合资助项目。

$< X$,停;否则,转步骤 2.

步骤 2 令 $d = x^{k+1} - x^k$,计算可行区间 $[\underline{\lambda}^*, \overline{\lambda}^*]$

$$[\underline{\lambda}^*, \overline{\lambda}^*] = \begin{cases} [\underline{\lambda}, \overline{\lambda}], j \notin L^{\infty}, j \notin U^{\infty}; \\ [\underline{\lambda}_L, \overline{\lambda}_L], j \in L^{\infty}; \\ [\underline{\lambda}_U, \overline{\lambda}_U], j \in U^{\infty}. \end{cases} \quad (1.2)$$

$$\underline{\lambda} = \max\{\frac{l_j - x_j^{k+1}}{d_j}, \text{if } d_j > 0, j = 1,\cdots,n; \frac{u_j - x_j^{k+1}}{d_j}, \text{if } d_j < 0, j = 1,\cdots,n\}. \quad (1.3)$$

$$\overline{\lambda} = \min\{\frac{u_j - x_j^{k+1}}{d_j}, \text{if } d_j > 0, j = 1,\cdots,n; \frac{l_j - x_j^{k+1}}{d_j}, \text{if } d_j < 0, j = 1,\cdots,n\}. \quad (1.4)$$

$$\underline{\lambda}_L = \max\{\frac{u_j - x_j^{k+1}}{d_j}, \text{if } d_j < 0, j = 1,\cdots,n\},$$

$$\overline{\lambda}_L = \min\{\frac{u_j - x_j^{k+1}}{d_j}, \text{if } d_j > 0, j = 1,\cdots,n\}. \quad (1.5)$$

$$\underline{\lambda}_U = \max\{\frac{l_j - x_j^{k+1}}{d_j}, \text{if } d_j > 0, j = 1,\cdots,n\},$$

$$\overline{\lambda}_U = \min\{\frac{l_j - x_j^{k+1}}{d_j}, \text{if } d_j < 0, j = 1,\cdots,n\}. \quad (1.6)$$

求 $\min\{f(x^{k+1} + \lambda d) | \lambda \in [\underline{\lambda}^*, \overline{\lambda}^*]\}$的最优解 $\hat{\lambda}$,令 $y^1 = x^{k+1} + \hat{\lambda} d$.令 $j = 1, k := k + 1$,转步骤 1.

1.2 离散步算法

步骤 0 选 $d^1,\cdots,d^n$ 为坐标方向,终止参数 $X > 0$,初始步长 $\Delta \geqslant X$,初始可行点 x^1,$l \leqslant x^1 \leqslant u$,令 $y^1 = x^1, k = j = 1$.

步骤 1 计算可行区间 $[\underline{\lambda}_j^*, \overline{\lambda}_j^*]$((1.1)式).如果 $\Delta \in [\underline{\lambda}_j^*, \overline{\lambda}_j^*]$,步长 Δ 称为可行的.此时,如果 $f(y^j + \Delta d^j) < f(y^j)$,试验称为成功的,令 $y^{j+1} = y^j + \Delta d^j$,转步骤 2.如果 $f(y^j + \Delta d^j) \geqslant f(y^j)$,试验称为失败的,计算 $[\underline{\Delta}, \overline{\Delta}]$

$$[\underline{\Delta}, \overline{\Delta}] = \begin{cases} [y_j^j - u_j, y_j^j - l_j], j \notin L^{\infty}, j \notin U^{\infty}; \\ [y_j^j - u_j, +\infty), j \in L^{\infty}; \\ (-\infty, y_j^j - l_j], j \in U^{\infty}. \end{cases} \quad (1.7)$$

如果 $\Delta \in [\underline{\Delta}, \overline{\Delta}]$,步长 Δ 称为可行的,此时,如果 $f(y^j - \Delta d^j) < f(y^j)$,令 $y^{j+1} = y^j - \Delta d_j$,转步骤 2;如果 $f(y^j - \Delta d^j) \geqslant f(y^j)$,令 $y^{j+1} = y^j$,转步骤 2.如果 $\Delta \notin [\underline{\Delta}, \overline{\Delta}]$,步长 Δ 称为不可行的,令 $y^{j+1} = y^j$,转步骤 2.

步骤 2 如果 $j < n$,令 $j := j + 1$,返回步骤 1.否则,如果 $f(y^{n+1}) < f(x^k)$,转步骤 3;如果 $f(y^{n+1}) \geqslant f(x^k)$,转步骤 4.

步骤 3 令 $x^{k+1} = y^{n+1}$, $d = x^{k+1} - x^k$,计算可行区间 $[\underline{\lambda}^*, \overline{\lambda}^*]$(见(1.2)式~(1.6)式).如果 $\underline{\lambda}^* < 0.618\overline{\lambda}^*$,则取 $\lambda = 0.618\overline{\lambda}^*$;否则,取 $\lambda = \overline{\lambda}^*$.令 $y^1 = x^{k+1} + \lambda d$, $j = 1, k := k + 1$,转步骤 1.

步骤 4 如果 $\Delta \leqslant X$,停,x^k 为近似解;否则,令 $\Delta := \frac{1}{2}\Delta$, $y^1 = x^k$, $x^{k+1} = x^k$, $k := k + 1$, $j = 1$,转步骤 1.

2 线搜索 Hooke-Jeeves算法的收敛性分析

本节证明推广的线搜索 Hooke-Jeeves算法的全局收敛性.为方便起见,以下论证只考虑 $l_i \in E^1$, $u_i \in E^1$的情形.

假设 2.1 设 $f: R \to E^1$可微,且 f沿任一坐标方向在闭区间上的极小是唯一的.

引理 2.2 若假设 2.1成立,且求 $\min\{f(x) | x \in R\}$的算法映射 B定义如下: $y \in B(x)$意味着 y是由 x开始逐次沿坐标方向在闭区间上极小化得到的,则 B是闭的.

证明 参见文献[4]定理 7.3.5可知本结论成立.

引理 2.3 若 $y + \hat{\lambda} d$是问题 $\min\{f(y + \lambda d) | l \leqslant y + \lambda d \leqslant u\}$的一个最优解,则 $y + \hat{\lambda} d$是该问题的 KKT点.

证明 由于 $\{\lambda | l \leqslant y + \lambda d \leqslant u\}$是线性约束集,故 Abadie约束规格恒成立.由文献[4]定理 5.3可知,$y + \hat{\lambda} d$是该问题的 KKT点.

假设 2.4 设 X是 E^n中非空闭集,$K \subset X$是非空解集,$T: E^n \to E^1$是连续函数.给定初始点 $x^1 \in X$,集合 $\wedge = \{x : T(x) \leqslant T(x^1)\}$是紧致的.

假设 2.5 映射 $B: X \to 2^X$满足: $\forall x \in X \backslash K$, $\forall y \in B(x)$,有 $T(y) < T(x)$; B在 $X \backslash K$上是闭的.映射 $C: X \to 2^X$满足: $\forall x \in X$, $\forall y \in C(x)$,有 $T(y) \leqslant T(x)$.

引理 2.6 若假设 2.4和假设 2.5成立,则合成映射 $A = CB$定义的算法 A是全局收敛的,即算法 A或有限步终止于 K中的点,或产生无穷点列 $\{x^k\}$,使得 $\{x^k\}$的任意聚点属于 K.

证明 详见文献[4]定理 7.3.4的论证.

定理 2.7 若假设 2.1成立,给定初始点 $x^1 \in R$,集合 $\wedge = \{x : f(x) \leqslant f(x^1)\}$是紧致的,则线搜索算法或有限步终止于 (P)的 KKT点,或产生无穷点列 $\{x^k\}$,使得 $\{x^k\}$的任意聚点为 (P)的 KKT点.

证明 验证引理 2.6中的假设条件成立,从而完成本题证明.

设 $X = R$, $K = \{\bar{x} | \bar{x}$是(P)的 KKT点$\}$,将线搜索算法表示成映射 $A = CB$,其中 B是循环坐标映射,C为模式搜索映射(步骤 2).$T(x) = f(x)$,由引理

2.2可知 B 是闭的,由 C的定义知,对 $\forall z\in C(y)$, $\mathrm{T}(z)\leqslant \mathrm{T}(y)$.

(1) 证 $\forall x\in X\backslash \mathrm{K}$, $\forall y\in B(x)$, 有 $\mathrm{T}(y) < \mathrm{T}(x)$. 即证 $f(y) < f(x)$.

设 x^k 为当前迭代点, 且 $x^k\in X\backslash\mathrm{K}$, $x^{k+1}\in B(x^k)$.

往证: $f(x^{k+1}) < f(x^k)$.

令 $x^k = y^1, y^{j+1} = y^j + \lambda_j d^j, j = 1,\cdots,n$. 由 B的定义可知

$$x^{k+1} = y^{n+1}. \tag{2.1}$$

$f(x_{k+1}) = f(y^{n+1}) = f(y^n + \lambda_n d^n)\leqslant f(y^n) = f(y^{n-1} + \lambda_{n-1}d^{n-1})\leqslant f(y^{n-1}) = f(y^{n-2} + \lambda_{n-2}d^{n-2})\leqslant\cdots\leqslant f(y^2) = f(y^1 + \lambda_1 d^1)\leqslant f(y^1) = f(x^k)$. $\qquad (2.2)$

即 $f(x^{k+1})\leqslant f(x^k)$. 下证 $f(x^{k+1})\neq f(x^k)$.

反证法 假设 $f(x^{k+1}) = f(x^k)$,由已知条件可知, $x^{k+1} = x^k$. 由 (2.2) 式可知,

$f(x^{k+1}) = f(y^{n+1}) = f(y^n + \lambda_n d^n) = f(y^n) = f(y^{n-1} + \lambda_{n-1}d^{n-1}) = f(y^{n-1}) = f(y^{n-2} + \lambda_{n-2}d^{n-2}) = \cdots = f(y^2) = f(y^1 + \lambda_1 d^1) = f(y^1) = f(x^k)$.

故 $\quad \lambda_1 = \lambda_2 = \cdots = \lambda_n = 0. \qquad (2.3)$

另一方面,由引理 2.3可知, $y^j + \lambda_j d^j$ 是问题 $\min\{f(y+\lambda d)\mid l\leqslant y+\lambda d\leqslant u\}$的 KKT点,故满足该问题的 KKT条件:

$$\begin{cases} 5 f(y^j + \lambda_j d^j)^T d^j + \sum_{i=1}^{n}\mathrm{T}_i d_i^j - \sum_{i=1}^{n}\mathrm{U}_i d_i^j = 0, \\ \mathrm{T}_i(y_i^j - u_i) = 0, \quad i = 1,\cdots,n; \\ \mathrm{U}_i(l_i - y_i^j) = 0, \quad i = 1,\cdots,n; \\ \mathrm{T}_i\geqslant 0; \mathrm{U}_i\geqslant 0; \quad i = 1,\cdots,n. \end{cases} \tag{2.4}$$

其中, d_i^j 为向量 d^j 的第 i 个分量. 由 (2.3) 式可将 (2.4) 式化为:

$$\begin{cases} 5 f(y^j)^T d^j + \sum_{i=1}^{n}\mathrm{T}_i d_i^j - \sum_{i=1}^{n}\mathrm{U}_i d_i^j = 0, \\ \mathrm{T}_i(y_i^j - u_i) = 0, \quad i = 1,\cdots,n; \\ \mathrm{U}_i(l_i - y_i^j) = 0, \quad \mathrm{T}_i\geqslant 0; \mathrm{U}_i\geqslant 0, \quad i = 1,\cdots,n. \end{cases} \tag{2.5}$$

在 (2.5) 式中分别令 $j = 1,\cdots,n$,得

$$\begin{cases} \dfrac{\partial f(y^i)}{\partial y_i^j} + \mathrm{T}_i - \mathrm{U}_i = 0, \mathrm{T}_i(y_i^j - u_i) = 0, i = 1,\cdots,n; \\ \mathrm{U}_i(l_i - y_i^j) = 0, \quad \mathrm{T}_i\geqslant 0; \mathrm{U}_i\geqslant 0, \ i = 1,\cdots,n. \end{cases} \tag{2.6}$$

(2.6) 式恰是问题 (P) $\min\{f(x)\mid x\in R\}$的 KKT条件,因此 y^j $(j = 1,\cdots,n)$ 是其 KKT点. 另一方面,由 (2.1) 式及 (2.3) 式可知 $x^{k+1} = y^{n+1} = \cdots = y^2 = y^1 = x^k$. 因此, x^k是问题 (P) 的 KKT点,这与 $x^k\in X\backslash\mathrm{K}$矛盾. 故假设不成立.

(2) 由已知条件及上述论证可知假设 2.4和假设 2.5成立,故由引理 2.6知线搜索算法或有限步终止于 (P) 的 KKT点,或产生无穷点列 $\{x^k\}$,使得 $\{x^k\}$的任意聚点为 (P) 的 KKT点. 证毕.

3 数值试验

为了进一步从数值计算的角度分析算法的有效性,本文将线搜索算法和离散步算法用 MATLAB制成软件,在一定范围内对若干实例进行试算,其结果说明算法是有效的. 在此选几个简单的例子[5]加以说明,并略去其详细迭代过程. 需要指出的是,表 1中的 h 为线搜索算法采用黄金分割法进行线搜索的终止参数.

表 1 算法的数值试验结果

Table 1 Numerical experiment results of the algorithms

	例 1线搜索算法 Line search algorithm for Example 1	例 2离散步算法 Discrete step algorithm for Example 2
k	X= 1e− 5	X= 5e− 2
$h\mid\Delta$	h = 5e− 6	Δ = 5
x^1	x_1^1	x_2^1
迭代次数 k Number of iteration	3	3
近似解 x^* Approximate solution x^*	x_1^*	x_2^*
$f(x^*)$	98.2927	− 45.7784
	例 3线搜索算法 Line search algorithm for Example 3	**例 3离散步算法 Discrete step algorithm for Example 3**
X	X= 1e− 2	X= 1e− 2
$h\mid\Delta$	h = 1e− 2	Δ = 75
x^1	x_3^1	x_4^1
迭代次数 k Number of iteration k	11	15
近似解 x^* Approximate solution x^*	x_3^*	x_4^*
$f(x^*)$	− 2.8992e+ 025	− 2.8991e+ 025

Note x_1^1 = (20, 55, 15, 20, 60, 20, 20, 60, 20, 20, 60, 20, 20, 60, 20); x_2^1 = (9, 9, 9, 9, 9, 9, 9, 9, 9, 9); x_1^* = (8, 43, 3, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0); x_2^* = (9.3516, 9.3516, 9.3516, 9.3516, 9.3516, 9.3516, 9.3516, 9.3516, 9.3516, 9.3516).

x_3^1 = (0, 0, 0, 0, 0, 0, 0, 0, 0, 0); x_4^1 = (0, 0, 0, 0, 0, 0, 0, 0, 0, 0);

x_3^* = (31.2578, 42.3062, 59.1992, 31.0149, 49.8640, 40.1319, 49.2428, 35.8561, 51.8045, 47.3226);

x_4^* = (31.2927, 38.0178, 59.1980, 31.0730, 49.8779, 38.0178, 49.2462, 35.8887, 52.0162, 47.4609).

例 **1** 取自文献[5]第 137页 hs118,稍作修改.

解 $\min\ 2.3(x_1+x_4+x_7+x_{10}+x_{13})+0.0001(x_1^2+x_4^2+x_7^2+x_{10}^2+x_{13}^2)+1.7(x_2+x_5+x_8+x_{11}+x_{14})+0.0001(x_2^2+x_5^2+x_8^2+x_{11}^2+x_{14}^2)+2.2(x_3+x_6+x_9+x_{12}+x_{15})+0.00015(x_3^2+x_6^2+x_9^2+x_{12}^2+x_{15}^2)$,

s.t $l \leqslant x \leqslant u$;

$l=[8,43,3,0,0,0,0,0,0,0,0,0,0,0,0]$;

$u=[21,57,16,90,120,60,90,120,60,90,120,60,90,120,60]$.

例 **2** 取自文献[5]第 125页 hs110.

解 $\min \sum_{i=1}^{10} \ln^2(x_i-2)+\ln^2(10-x_i))-(x_1x_2\cdots x_{10})^{0.2}$,

s.t $2.001 \leqslant x_i \leqslant 9.999, i=1,\cdots,10$.

例 **3** 取自文献[5]第 126页 hs111,稍作修改.

解 $\min \sum_{i=1}^{10}[\exp(x_i)(c_i+x_i-\lg(\sum_{k=1}^{10}\exp(x_k)))]$,

s.t $-100 \leqslant x_i \leqslant 100, i=1,\cdots,10$.

$c=[-6.089,-17.164,-34.054,-5.914,-24.721,-14.986,-24.100,-10.708,-26.662,-22.179]$.

4 结束语

Hooke-Jeeves方法是无约束最优化方法中处理不可微函数时常用的一种方法,本文将 Hooke-Jeeves方法推广到带广义界的简单约束优化,并产生 2个算法.在适当条件下,证明了推广后的线 Hooke-Jeeves算法具有全局收敛性.本文还将线搜索算法和离散步算法在一定范围内对 3个例子进行试算,结果表明推广后的 Hooke-Jeeves算法是有效的.

参考文献:

[1] Hadley G. Nonlinear and Dynamic Programming[M]. Addison-Wesley, 1964.

[2] Klingman W R, Himmelblau D M. Nonlinear programming with the aid of multiplier gradient summation technique[J]. J Asscciation for Computing Machinery, 1964, 11: 400-415.

[3] Glass H, Copper L. Sequential search: a method for solving constrained optimization problems [J]. J Association Computing Machinery, 1965, 12: 71-82.

[4] Bazaraa M S, Sherali H D, Shetty C M. Nonlinear Programming: Theory and Algorithms [M]. New York: Wiley, 1993.

[5] Hock W, Schittowski K. Test Examples for Nonlinear Programming Codes in Lecture Notes. In: Economics and Mathhematical Systems Springer-verlag [M]. Berlin: Heidelberg and New York, 1981. 125-137.

(责任编辑: 黎贞崇)

美国科学家研制出“夸克胶子等离子体”

美国能源部下属的布鲁克黑文国家实验室的科学家研制出“夸克胶子等离子体”。这是一种全新的物质形态,曾广泛存在于宇宙诞生后的百万分之几秒内。

现有物理学理论认为,宇宙诞生后的百万分之几秒内,宇宙中曾存在过一种被称为“夸克胶子等离子体”的物质。在“夸克胶子等离子体”中,夸克和胶子(一种理论上假设的无质量粒子)等基本粒子以自由状态存在。它们随宇宙的冷却,结合形成质子和中子等亚原子粒子,后者又形成原子核,最终产生原子以及今天的宇宙万物。

在宇宙现有物质中,夸克等被约束在质子和中子内,无法独立存在。布鲁克黑文国家实验室的研究人员从 2000年 6月起,让金原子核以接近光速的速度相撞,试图以相撞产生的巨大能量和温度“融解”质子和中子,使夸克胶子以自由形态释放出来。结果发现,相撞产生的原始粒子,根据金原子核相撞产生的不同压力变化在“集体移动”,好像一个鱼群在游动。由于这种运动方式与液体运动的性质十分相似,研究人员将其称为“流动”。这种运动接近一种“完美”状态,符合“完美液体”的特征。

夸克胶子等离子体呈液体状态,为研究宇宙在诞生后的最初形态提供了新的见解,宇宙在诞生后的百万分之几秒内可能就是一种“完美的液态”。美国能源部认为布鲁克黑文国家实验室的这项成果是物理学界一次具有历史意义的重大进展。

据《科学时报》