

求解双层规划问题的填充函数法

袁柳洋^{1,2}, 李青¹

(1. 武汉科技大学理学院, 湖北 武汉 430065)

(2. 冶金工业过程系统科学湖北省重点实验室, 湖北 武汉 430065)

摘要: 本文研究了一类带等式和不等式约束的双层规划问题, 首先利用下层问题的 KKT 条件将双层规划转化为单层约束规划问题; 其次结合罚函数法, 构造了一种新的填充函数, 并探讨了它的性质; 最后基于构造的填充函数, 获得了一种求解双层规划问题的填充函数法, 并通过数值实验说明了该算法的可行性.

关键词: 双层规划; 单层约束规划; 罚函数法; 填充函数; KKT 条件

MR(2010) 主题分类号: 90C29; 90C99 中图分类号: O224

文献标识码: A 文章编号: 0255-7797(2022)02-0153-09

1 引言

双层规划是一类具有上下两层递阶结构的系统决策优化问题, 其中上层问题的目标函数和约束条件不仅与上层决策变量有关, 而且还依赖于下层问题的最优解, 而下层问题的最优解又受上层决策变量的影响. 双层规划已被应用到诸多领域中, 比如交通运输^[1]、资源分配^[2]、价格问题、供应链管理^[3]和生产计划^[3]等. 尽管双层规划问题的应用广泛, 但对双层规划问题的求解仍然是非常困难的, 即使是线性双层规划问题, 也是强 NP - 难问题, 比如 Jeroslow^[4]、Bard^[5]、Hansen^[6]、Vicente 等人^[7], 给出了相应的证明. 另一个造成双层规划问题求解困难的原因是非凸性, 即使最简单的双层线性规划, 一般来说也是一个非凸优化问题, 尽管能找到问题的解, 通常只能是局部最优解而非全局最优解. 虽然双层规划问题求解困难, 但到目前为止, 仍有不少算法被用来求解该问题, 比如极点搜索算法^[8]、分枝定界算法^[9,10]、罚函数算法^[11]、下降方法、互补旋转算法、非数值优化算法等. 其中绝大多数算法都只能求解双层规划问题的局部最优解, 而本文提出了一种填充函数法来求解双层规划问题的全局最优解.

填充函数法最早是由西安交通大学的葛仁溥教授^[12]首次提出的. 该算法的基本思想是: 先将目标函数局部极小化, 找到原始问题的一个局部最优解, 然后在找到的这个解处构造一个填充函数, 再对此填充函数进行局部极小化, 使得填充函数的局部极小点处的原始问题的目标函数值比之前的局部极小值小, 多次进行上述局部极小化和填充过程, 即可找到全局最优解. 该方法是求解优化问题的一种有效的确定性全局优化算法. 在利用该方法求解优化问题时, 关键是要构造一个填充函数以保证算法在实现过程中, 避免陷入局部最优, 从而找到更好的局部极小点.

*收稿日期: 2021-03-03 接收日期: 2021-04-27

基金项目: 国家自然科学基金资助 (11401450; 11401126).

作者简介: 袁柳洋 (1988-), 女, 湖北武汉, 副教授, 最优化理论与算法.

填充函数法最初是被用来求解无约束优化问题的 [13--16], 后来又被扩展到用于求解其他优化问题, 比如, 非线性全局优化问题 [17]、等式约束优化问题 [18,19]、非线性方程组等 [20]. 本文对填充函数的应用进行扩展, 用来求解双层规划问题.

本文余下的内容可分为如下几个部分: 第二部分, 双层规划问题的转化, 利用下层问题的 KKT 条件将双层规划转化为单层约束规划问题; 第三部分, 结合罚函数法, 构造了一种新的填充函数, 并探讨了它的性质; 第四部分, 基于构造的新的填充函数, 提出了一种求解双层规划问题的填充函数法; 第五部分, 对两个例子进行了数值实验; 最后一部分是总结.

2 双层规划问题的转化

本文考虑上、下两层都带有等式和不等式约束的双层规划问题, 其数学模型可描述为:

$$\begin{aligned} & \min_x F(x, y) \quad \text{s.t.} \\ & G_i(x, y) \leq 0, i = 1, 2, \dots, m, \quad H_j(x, y) = 0, j = 1, 2, \dots, n, \quad x \in X, \\ & \min_y f(x, y) \quad \text{s.t.} \\ & g_a(x, y) \leq 0, a = 1, 2, \dots, s, \quad h_b(x, y) = 0, b = 1, 2, \dots, t, \quad y \in Y. \end{aligned} \quad (2.1)$$

其中 $x \in \mathbb{R}^n$ 是上层问题的决策变量, $y \in \mathbb{R}^n$ 是下层问题的决策变量, $F(x, y), G_i(x, y), H_j(x, y)$ 分别是上层的目标函数和约束函数, $f(x, y), g_a(x, y), h_b(x, y)$ 分别是下层的目标函数和约束函数. 对问题 (2.1), 本文做如下假设.

假设 2.1 假设 $f(x, y), g_a(x, y), a = 1, 2, \dots, s$. 和 $h_b(x, y), b = 1, 2, \dots, t$. 是光滑函数, f 和 $g_a(x, y), a = 1, 2, \dots, s$. 是凸函数, $h_b(x, y), b = 1, 2, \dots, t$. 是关于 y 的线性函数.

由假设 2.1 可知, 利用下层问题的 KKT 最优性条件, 可将双层规划问题等价转化为连续的单层约束优化问题.

假设 2.2 假设问题 (2.1) 的局部极小点存在且个数有限. 设下层问题的拉格朗日函数为:

$$L(x, y, \lambda, \mu) = f(x, y) + \sum_{a=1}^s \lambda_a g_a(x, y) + \sum_{b=1}^t \mu_b h_b(x, y). \quad (2.2)$$

则下层问题的 KKT 最优性条件为:

$$\begin{aligned} & \frac{\partial f}{\partial y} + \sum_{a=1}^s \lambda_a \frac{\partial g_a}{\partial y} + \sum_{b=1}^t \mu_b \frac{\partial h_b}{\partial y} = 0, \\ & g_a(x, y) \leq 0, a = 1, 2, \dots, s, \\ & \lambda_a \geq 0, a = 1, 2, \dots, s, \\ & \lambda_a g_a(x, y) = 0, a = 1, 2, \dots, s, \\ & h_b(x, y) = 0, b = 1, 2, \dots, t, \\ & \mu_b \neq 0, b = 1, 2, \dots, t, \\ & y \in Y. \end{aligned} \quad (2.3)$$

那么双层规划问题 (2.1) 可转化为如下的单层约束规划问题:

$$\begin{aligned}
 & \min_{x,y} F(x,y) \quad \text{s.t.} \\
 & G_i(x,y) \leq 0, i = 1, 2, \dots, m, \quad H_j(x,y) = 0, j = 1, 2, \dots, n, \\
 & \frac{\partial f}{\partial y} + \sum_{a=1}^s \lambda_a \frac{\partial g_a}{\partial y} + \sum_{b=1}^t \mu_b \frac{\partial h_b}{\partial y} = 0, \\
 & g_a(x,y) \leq 0, a = 1, 2, \dots, s, \\
 & \lambda_a \geq 0, a = 1, 2, \dots, s, \\
 & \lambda_a g_a(x,y) = 0, a = 1, 2, \dots, s, \\
 & h_b(x,y) = 0, b = 1, 2, \dots, t, \\
 & \mu_b \neq 0, b = 1, 2, \dots, t, \\
 & x \in X, \quad y \in Y.
 \end{aligned} \tag{2.4}$$

令 $z := (x, y, \lambda, \mu)$, $Z := \{z = (x, y, \lambda, \mu) \mid x \in X, y \in Y, \lambda_a \geq 0, \mu_b \neq 0\}$, $L(z) = \lambda_a g_a(x, y)$, $\varphi(z) = \frac{\partial f}{\partial y} + \sum_{a=1}^s \lambda_a \frac{\partial g_a}{\partial y} + \sum_{b=1}^t \mu_b \frac{\partial h_b}{\partial y}$. 则单层约束规划问题 (2.4) 可化为

$$\begin{aligned}
 & \min F(z) \quad \text{s.t.} \\
 & G_i(z) \leq 0, i = 1, 2, \dots, m, \quad H_j(z) = 0, j = 1, 2, \dots, n, \\
 & \varphi(z) = 0, \\
 & g_a(z) \leq 0, a = 1, 2, \dots, s, \quad h_b(z) = 0, b = 1, 2, \dots, t, \\
 & L(z) = 0, \\
 & z \in Z.
 \end{aligned} \tag{2.5}$$

此时问题 (2.5) 等价于问题 (2.1), 通过求解问题 (2.5) 的最优解就可以求解原始问题 (2.1) 的最优解.

3 填充函数的构造

定义 3.1 ^[20] 函数 $P(z, z^*) : Z \rightarrow \mathbb{R}$ 被称为 $F(z)$ 在局部极小点 z^* 处的填充函数, 如果它满足如下性质:

- (1) z^* 是 $P(z, z^*)$ 在 Z 上的一个严格局部极大点;
- (2) 对任意的 $z \in S_1$, 有 $\nabla P(z, z^*) \neq 0$, 其中 $S_1 = \{z \mid F(z) \geq F(z^*), z \in Z \setminus \{z^*\}\}$;
- (3) 若 z^* 不是全局极小点, 即 $F(z)$ 存在另一个局部极小点 $z^{**} \in \text{int}(Z)$, 满足 $F(z^{**}) < F(z^*)$, 则 $P(z, z^*)$ 存在一个极小点 $z' \in N(z^{**}, \delta) \subset Z$. 其中 $N(z^{**}, \delta)$ 是点 z^{**} 处的 δ 邻域.

在下文中, 介绍满足定义 3.1 的填充函数 $F_f(z, z^*, q, u)$. 首先, 本文设计了两个不同的连续函数 $\phi_r(t)$ 和 $\varphi_r(t)$. 函数 $\phi_r(t)$ 具有如下的性质: 当 $t \geq 0$ 时, $\phi_r(t) = 1$; 当 $t \leq -r$ 时, $\phi_r(t) = 0$; 另外它是单调递增的. $\varphi_r(t)$ 具有如下的性质: 当 $t \geq r$ 时, $\varphi_r(t) = t + r$; 当 $t \leq 0$ 时, $\varphi_r(t) = 0$; 另外它是单调递增的.

对于任何给定的 $r > 0$,

$$\phi_r(t) = \begin{cases} 1, & t \geq 0, \\ \log_{1+r^3} (-2(t+r)^3 + 3r(t+r)^2 + 1), & -r < t < 0, \\ 0, & t \leq -r. \end{cases} \tag{3.1}$$

$$\varphi_r(t) = \begin{cases} t+r, & t \geq r, \\ (t+r) \log_{1+r^3}(-2t^3+3rt^2+1), & 0 < t < r, \\ 0, & t \leq 0. \end{cases} \quad (3.2)$$

则有

$$\phi'_r(t) = \begin{cases} 0, & t \geq 0, \\ \frac{-6(t+r)^2+6r(t+r)}{-2(t+r)^3+3r(t+r)^2+1}, & -r < t < 0, \\ 0, & t \leq -r. \end{cases} \quad (3.3)$$

$$\varphi'_r(t) = \begin{cases} 1, & t \geq r, \\ \log_{1+r^3}(-2t^3+3rt^2+1) + (t+r) \frac{-6t^2+6rt}{-2t^3+3rt^2+1}, & 0 < t < r, \\ 0, & t \leq 0. \end{cases} \quad (3.4)$$

显然 $\phi_r(t)$ 和 $\varphi_r(t)$ 在 \mathbb{R} 上是连续可微的. 当 $r \rightarrow 0^+$ 时, $\varphi_r(t) \rightarrow \max(t, 0)$. 为此, 本文构造了一个填充函数:

$$F_f(z, z^*, q, u) = \frac{1}{\|z - z^*\|^2 + 1} \phi_q \left(F(z) - F(z^*) + \frac{1}{2} u \left(\sum_{i=1}^m \varphi_{\frac{1}{u}}^2(G_i(z)) + \sum_{a=1}^s \varphi_{\frac{1}{u}}^2(g_a(z)) + \sum_{j=1}^n H_j^2(z) + \sum_{b=1}^t h_b^2(z) + L^2(z) + \wp^2(z) \right) \right). \quad (3.5)$$

其中 $q > 0, u > 0$ 是两个参数, $u > 0$ 是惩罚参数, $\phi_r(t)$ 和 $\varphi_r(t)$ 由式 (3.1)、(3.2) 定义.

以下定理表明 $F_f(z, z^*, q, u)$ 是满足定义 3.1 的一类填充函数.

定理 3.1 假设 $q > 0$ 充分小, 且 $u > 0$ 足够大, 则 z^* 是 $F_f(z, z^*, q, u)$ 在 Z 上的一个全局极大点. 证明: 由式 (3.5) 可得,

$$F_f(z^*, z^*, q, u) = \frac{1}{\|z^* - z^*\|^2 + 1} \phi_q \left(F(z^*) - F(z^*) + \frac{1}{2} u \left(\sum_{i=1}^m \varphi_{\frac{1}{u}}^2(G_i(z^*)) + \sum_{a=1}^s \varphi_{\frac{1}{u}}^2(g_a(z^*)) + \sum_{j=1}^n H_j^2(z^*) + \sum_{b=1}^t h_b^2(z^*) + L^2(z^*) + \wp^2(z^*) \right) \right). \quad (3.6)$$

又根据 $\phi_r(t)$ 的性质知, 当 $t \geq 0$ 时, $\phi_r(t) = 1$, 有 $F_f(z, z^*, q, u) \leq F_f(z^*, z^*, q, u)$. 因此 z^* 是 $F_f(z, z^*, q, u)$ 在 Z 上的一个全局极大点.

定理 3.1 表明函数 $F_f(z, z^*, q, u)$ 满足定义 3.1 的条件 (1). 一般情况下, 要寻求函数的全局极大点是非常困难的事情. 然而, 此处构造的函数 $F_f(z, z^*, q, u)$ 的全局极大点非常容易求得, 就为 z^* .

定理 3.2 假设 $q > 0$ 充分小, 且 $u > 0$ 足够大, 则对于任何 $z \in S_1 \setminus \{z^*\}$ 有 $\nabla F_f(z, z^*, q, u) \neq 0$, 其中 $S_1 = \{z \in S \mid F(z) \geq F(z^*)\}$.

证明: 假设 $z \in S_1$, 则 $F(z) > F(z^*)$, 得出

$$F_f(z, z^*, q, u) = \frac{1}{\|z - z^*\|^2 + 1}, \quad \nabla F_f(z, z^*, q, u) = -\frac{2(z - z^*)}{(\|z - z^*\|^2 + 1)^2}.$$

又 $z \neq z^*$ 则 $\nabla F_f(z, z^*, q, u) \neq 0$.

定理 3.2 表明函数 $F_f(z, z^*, q, u)$ 满足定义 3.1 的条件 (2).

定理 3.3 假设 $q > 0$ 充分小, 且 $u > 0$ 足够大, 若 z^* 不是全局极小点, 即 $F(z)$ 存在另一个极小点 $z^{**} \in \text{int}(Z)$, 使得 $F(z^{**}) < F(z^*)$, 则当 $0 < q < (F(z^*) - F(z^{**}))$ 时, $F_f(z, z^*, q, u)$ 一定存在一个极小点 z' , 使得 $z' \in N(z^{**}, \delta) \subset Z$, 其中 $N(z^{**}, \delta)$ 是点 z^{**} 处的 δ 邻域.

证明: 因 $F(z)$ 是连续的, 则一定存在点 \bar{z}^* , $z' \in N(z^{**}, \delta)$, 满足 $F(\bar{z}^*) = F(z^*)$, $F(\bar{z}^*) > F(z') > F(z^{**})$, 其中 $z' = \bar{z}^* + \varepsilon(\bar{z}^* - z^*)$. 当 $0 < q < (F(z^*) - F(z^{**}))$ 时, 令 $F(z') \leq F(z^*) - q$, 则 $F_f(z', z^*, q, u) = 0$. 又 $F_f(z, z^*, q, u) \geq 0$ 对所有的 $z \in Z$ 都成立. 所以 z' 是 $F_f(z, z^*, q, u)$ 的极小点, 则结论成立.

定理 3.3 表明函数 $F_f(z, z^*, q, u)$ 满足定义 3.1 的条件 (3).

4 填充函数算法

本节基于上一节构造的填充函数 $F_f(z, z^*, q, u)$, 提出了一种求解双层规划问题的填充函数法, 下面介绍相应的求解问题 (2.5) 的填充函数算法 4.1.

步 0 选择充分小的正数 q, λ_L 和充分大的正数 u , 选择一个正的整数 K 和方向 $e_i, i = 1, \dots, K$. 选择一个初始点 $z_0 \in Z$. 置 $k := 0$;

步 1 从 z_k 开始, 通过局部搜索方法找到问题 (2.5) 的一个局部极小点 z_k^* ;

步 2 令

$$F_f(z, z_k^*, q, u) = \frac{1}{\|z - z_k^*\|^2 + 1} \phi_q \left(F(z) - F(z_k^*) + \frac{1}{2} u \left(\sum_{i=1}^m \varphi_{\frac{1}{u}}^2(G_i(z)) + \sum_{a=1}^s \varphi_{\frac{1}{u}}^2(g_a(z)) + \sum_{j=1}^n H_j^2(z) + \sum_{b=1}^t h_b^2(z) + L^2(z) + \varphi^2(z) \right) \right). \quad (4.1)$$

其中 $\phi_r(t)$ 和 $\varphi_r(t)$ 由式 (3.1)、(3.2) 定义, 置 $l = 1$ 和 $\lambda = 1$;

步 3

(a) 若 $l \leq K$, 转 (b); 否则, 转步 5;

(b) 若 $\lambda \geq \lambda_L$, 则置 $w_k^l := z_k^* + \lambda e_l$, 转 (c); 否则, 置 $l := l + 1, \lambda = 1$, 转 (a);

(c) 若 $w_k^l \in Z$, 转 (d); 否则, 置 $\lambda := \frac{\lambda}{2}$, 转 (b);

(d) 若 $f(w_k^l) < f(z_k^*)$, 则置 $z_{k+1} := w_k^l, k := k + 1$, 转步 1; 否则, 转步 4;

步 4 从 w_k^l 出发, 搜索如下填充函数问题的局部极小点:

$$\min_{z \in Z} F_f(z, z_k^*, q, u), \quad (4.2)$$

设 w_k^* 是问题 (4.2) 一个已获得的局部极小点, 若它满足 $f(w_k^*) < f(z_k^*)$, 则置 $z_{k+1} := w_k^*, k := k + 1$, 转步 1; 否则, 置 $l = l + 1, \lambda = 1$, 转步 3(a);

步 5 令 $z_s = z_k^*$, 停止.

算法 4.1 的主要思想:

该算法按以下方法选取步 0 中的方向 $e_i, i = 1, \dots, K$. 例如, 当 $n = 2$ 时, 取 $K = 6n$, 方向 e_i 被选为:

$$e_i = \left(\cos \frac{2\pi(i-1)}{K}, \sin \frac{2\pi(i-1)}{K} \right), i = 1, \dots, K.$$

当 $n \geq 3$ 时, 取 $K = 2n, i = 1, \dots, n$. e_i 中的第 i 个分量为 1, 其他分量为 0; $i = n+1, \dots, K$. e_i 中的第 i 个分量为 -1, 其他分量为 0.

在步 0 中选取一个合适的初始点, 从这个初始点开始, 利用局部极小化方法, 可以获得问题 (2.5) 的一个局部极小点 z_k^* . 接下来的任务是确定 z_k^* 是全局极小点或者找到比 $F(z_k^*)$ 函数值更小的局部极小点.

步 3 是选取合适的初始点, 去极小化填充函数问题 (4.2). 若方向集 $\{e_i, i = 1, \dots, K\}$ 足够大, 当无论怎么选初始点时, 问题 (4.2) 的局部极小点 w_k^* 均在 Z 的边界处获得, 且 $F_f(w_k^*, z_k^*, q, u) > 0$, 则称步 1 中找到的局部极小点 z_k^* 为问题 (2.5) 的近似全局极小点. 当 q 取得足够小时, 该点即被认为是问题 (2.5) 的全局极小点.

定理 4.1 算法 4.1 产生的序列 $\{z_k^*\}$ 收敛于问题 (2.5) 的全局极小点.

证明: 由假设 2.2 可知, 问题 (2.1) 的局部极小点存在且个数有限, 同时算法 4.1 中产生的迭代序列 $\{z_k^*\}$ 满足 $F(z_{k+1}^*) < F(z_k^*)$, 所以 $\{z_k^*\}$ 是一个使得函数值下降的序列, 由此可知, 算法 4.1 产生的序列 $\{z_k^*\}$ 收敛于问题 (2.5) 的全局极小点.

5 数值实验

本节应用填充函数算法来求解两个数值例子, 以揭示算法的可行性, 并与文献中的数值结果进行比较. 所有的数值实验均在 Matlab2018a 中执行. 算法使用 Matlab2018a 工具箱中的 fmincon 函数来局部极小化问题. 在整个数值实验中, 算法中的参数为: $q = 10^{-6}$, $u = 4^5$, $\lambda_L = \frac{1}{2^5}$.

例 1 见文献 [22].

$$\begin{aligned} \min_x F &= x^2 + y^2 - 16x - 5xy \quad \text{s.t.} \quad 0 \leq x \leq 20, \\ \min_y f &= -y \quad \text{s.t.} \quad x + y - 20 \leq 0, \quad 0 \leq y \leq 10. \end{aligned} \quad (5.1)$$

该问题的全局最优解为 $(x, y) = (11.1429, 8.8571)^T$, 其对应的目标函数值为 -469.1429 . 例 1 的数值计算结果见表 1. 由表 1 可知, 通过代入不同的初始点, 算法经过一次迭代, 即可找到全局最优解. 数值结果表明该算法是可行的.

例 2 见文献 [24].

$$\begin{aligned} \min_{x_1, x_2} F &= (x_1 \ x_2) \begin{pmatrix} 2 & 3 \\ 4 & 1 \end{pmatrix} \begin{pmatrix} y_1 \\ y_2 \end{pmatrix} \quad \text{s.t.} \quad x_1 + x_2 = 1, \quad x_1, x_2 \geq 0, \\ \min_{y_1, y_2} f &= (x_1 \ x_2) \begin{pmatrix} -1 & -4 \\ -3 & -2 \end{pmatrix} \begin{pmatrix} y_1 \\ y_2 \end{pmatrix} \quad \text{s.t.} \quad y_1 + y_2 = 1, \quad y_1, y_2 \geq 0. \end{aligned} \quad (5.2)$$

该问题的全局最优解为 $(x_1, x_2, y_1, y_2) = (0.0048, 0.0576, 0.1057, 0.8943)^T$, 其对应的目标函数值为 0.0899. 例 2 的数值计算结果见表 2. 由表 2 可知, 通过六次迭代, 即可找到全局最优解. 数值结果表明该算法是可行的.

表 1 例 1 的计算结果

k	z_k^0	$F(z_k^*)$	w_k^l	$F(w_k^l)$	$t(s)$
1	$\begin{pmatrix} 0.0000 \\ 0.0000 \end{pmatrix}$	0	$\begin{pmatrix} 11.1429 \\ 8.8571 \end{pmatrix}$	-469.1429	0.0782
1	$\begin{pmatrix} 5.0000 \\ 5.0000 \end{pmatrix}$	-105	$\begin{pmatrix} 11.1429 \\ 8.8571 \end{pmatrix}$	-469.1429	0.1094
1	$\begin{pmatrix} 11.0000 \\ 9.0000 \end{pmatrix}$	-469	$\begin{pmatrix} 11.1429 \\ 8.8571 \end{pmatrix}$	-469.1429	0.2969
1	$\begin{pmatrix} 10.0000 \\ 10.0000 \end{pmatrix}$	-460	$\begin{pmatrix} 11.1429 \\ 8.8571 \end{pmatrix}$	-469.1429	0.0156

表 2 例 2 的计算结果

k	z_k^0	$F(z_k^*)$	w_k^l	$F(w_k^l)$
1	$\begin{pmatrix} 0.5000 \\ 0.5000 \\ 0.5000 \\ 0.5000 \end{pmatrix}$	2.5000	$\begin{pmatrix} 0.3173 \\ 0.5576 \\ 0.1368 \\ 0.8632 \end{pmatrix}$	1.6900
2	$\begin{pmatrix} 0.3173 \\ 0.5576 \\ 0.1262 \\ 0.8738 \end{pmatrix}$	1.6808	$\begin{pmatrix} 0.1923 \\ 0.5576 \\ 0.1262 \\ 0.8738 \end{pmatrix}$	1.3216
3	$\begin{pmatrix} 0.1923 \\ 0.5576 \\ 0.1172 \\ 0.8828 \end{pmatrix}$	1.3081	$\begin{pmatrix} 0.1298 \\ 0.5576 \\ 0.1172 \\ 0.8828 \end{pmatrix}$	1.1280
4	$\begin{pmatrix} 0.1298 \\ 0.5576 \\ 0.1131 \\ 0.8869 \end{pmatrix}$	1.1217	$\begin{pmatrix} 0.0673 \\ 0.5576 \\ 0.1131 \\ 0.8869 \end{pmatrix}$	0.9412
5	$\begin{pmatrix} 0.0673 \\ 0.5576 \\ 0.1093 \\ 0.8907 \end{pmatrix}$	0.9351	$\begin{pmatrix} 0.0048 \\ 0.5576 \\ 0.1093 \\ 0.8907 \end{pmatrix}$	0.7545
6	$\begin{pmatrix} 0.0048 \\ 0.5576 \\ 0.1057 \\ 0.8943 \end{pmatrix}$	0.7485	$\begin{pmatrix} 0.0048 \\ 0.0576 \\ 0.1057 \\ 0.8943 \end{pmatrix}$	0.0899

6 小结

本文根据定义 3.1, 结合罚函数提出了一种新的填充函数并探讨了它的性质. 通过下层问题的 KKT 条件将双层规划转化为单层约束规划问题; 最后基于构造的新的填充函数, 提出了一种求解双层规划问题的填充函数法, 并对两个例子进行了数值实验, 结果表明填充函数算法是可行的. 但在算法中, 由于初始点选取的不同, 就会出现找到最优解需要的迭代次数不同, 简言之, 算法对于初值的选取太过于依赖, 这在后续的研究中会进一步改进.

参 考 文 献

- [1] 代如玉. 基于累积前景理论的城市轨道交通票价双层规划模型及算法 [D]. 南昌: 华东交通大学, 2018.
- [2] 吴睿. 管理系统中双层优化问题的算法研究 [D]. 西安: 西安建筑科技大学, 2009.
- [3] 高鹏. 基于双层线性规划模型的供应链战略伙伴物流规划研究 [D]. 杭州: 浙江大学, 2006.
- [4] Jeroslow R G. The polynomial hierarchy and a simple model for competitive analysis[J]. *Mathematical Programming*, 1985, 32(2): 146–164.
- [5] Bard J F. Some properties of the bilevel programming problem[J]. *Existence and Applications*, 1991, 68(2): 371–378
- [6] Hansen P. New branch-and-bound rules for liner bilevel programming[J]. *Siam Journal on Scientific and Satsistical Computing*, 1992, 13(5): 273–292.
- [7] Savard G, Gauvin J. The steepest descent direction for the nonlinear bilevel programming problem[J]. *Operations Research Letters*, 1994, 15(5): 265–272.
- [8] 贾飞. 解非线性双层规划的算法研究 [D]. 西安: 西安电子科技大学, 2014.
- [9] 贾新花, 赵茂先, 胡宗国. 一种 0-1 型双层线性规划的分支 - 定界法 [J]. *山东理工大学学报*, 2008, 22(6): 105–107
- [10] 高莹莹. 二次双层规划问题全局最优解的有效算法研究 [D]. 长春: 长春工业大学, 2014.
- [11] 孟志青, 沈瑞, 徐新生. 一种双层规划的光滑化目标罚函数算法 [J]. *运筹学学报*, 2015, 19(3): 26–33.
- [12] Ge Renpu. A filled function method for finding a global minimizer of a function of several variables[J]. *Mathematics Programming*, 1990, 46(1–3): 191–204.
- [13] Xu Zheng, Xu Chengxian. Filled functions for unconstrained global optimization[J]. *Journal of Global Optimization*, 2000, 15(3): 307–318.
- [14] Zhang LianSheng. A new filled function method for global optimization[J]. *Journal of Global Optimization*, 2004, 28(1): 17–43.
- [15] Wu Ziyou. A filled function method for nonlinear equations[J]. *Applied Mathematics and Computation*, 2007, 189(2): 1196–1204.
- [16] Zhu Wenxing. A class of filled functions for box constrained continuous global optimization[J]. *Applied Mathematics and Computation*, 2004, 169(1): 129–145.
- [17] 曹建辉. 约束全局优化问题的填充函数法 [D]. 兰州: 西北师范大学, 2010.
- [18] 连淑君, 杜爱华, 唐加会. 等式约束优化问题的一类新的简单光滑精确罚函数 [J]. *运筹学学报*, 2017, 21(1): 33–43.
- [19] Wu Ziyou et al. A filled function method for constrained global optimization[J]. *Journal of Global Optimization*, 2007, 39(4): 495–507.
- [20] 袁柳洋, 唐秋华, 贾世会. 求解非线性 P0 互补问题的填充函数法 [J]. *武汉科技大学学报*, 2016, 39(3): 236–240.

- [21] 邱根胜, 王金凤. 一类非线性双层规划问题的罚函数 [J]. 南昌航空工业学院学报, 2009, 23(3): 61–64.
- [22] 郑跃, 万仲平, 吕一兵. 非线性二层规划问题的全局优化方法 [J]. 系统科学与数学, 2012, 32(5): 513–521.
- [23] 双层规划及其应用 [DB/OL]. <https://wenku.baidu.com/view/2737b14fa1c7aa00b52acb52acbcc.html>.
- [24] 范成礼, 邢清华, 付强, 等. 求解非线性双层规划问题的混合变邻域粒子群算法 [J]. 系统工程理论与实践, 2015, 35(2): 473–480.

A FILLED FUNCTION METHOD FOR SOLVING BI-LEVEL PROGRAMMING PROBLEMS

YUAN Liu-yang^{1,2}, LI Qing¹

(1. College of Science, Wuhan University of Science and Technology, Wuhan 430065, China)

(2. Hubei Provincial Key Laboratory of Metallurgical Industry Process System Science, Wuhan 430065, China)

Abstract: This paper studies a type of bi-level programming problem with equality and inequality constraints. Firstly, the KKT condition of the lower problem is used to convert the bi-level programming into a single-level constraint programming problem; secondly, a new filled function is constructed by combining the penalty function method. And we discussed its properties; finally, based on the constructed filled function, a filled function method for solving bi-level programming problems is obtained, and the feasibility of the algorithm is demonstrated through numerical experiments.

Keywords: bi-level programming; single-level constrained programming; penalty function method; filled function; KKT condition

2010 MR Subject Classification: 90C33; 90C30