

DOI:10.13718/j.cnki.xsxb.2018.06.008

一种求解随机线性二层规划问题的 分支定界-粒子群混合算法^①

张 涛

长江大学 信息与数学学院, 湖北 荆州 434023

摘要: 将一类随机线性二层规划模型转换为带期望约束的确定性线性二层协方差规划模型, 并进一步利用 KKT 条件将二层协方差规划模型转化为单层规划模型, 然后利用分支定界-粒子群混合算法对该模型进行求解. 与传统分支定界算法的对比实验表明, 该算法有效改善了上层问题的方差结果, 且计算效率得到了较显著提高.

关 键 词: 随机线性二层规划; 带期望约束的二层协方差模型; 粒子群优化算法; 分支定界算法

中图分类号: O211.5

文献标志码: A

文章编号: 1000-5471(2018)06-0037-09

二层规划是一种具有递阶结构的系统优化问题. 在二层规划模型中, 一般可分为上下 2 层, 上层是一个含有下层最优决策变量(或最优目标函数值)的复合最优化问题, 而下层则是一个以上层决策变量为参数的参数规划^[1-3]. 作为描述管理部门的多个阶层关系的一种有力工具, 二层规划模型被广泛应用于优化控制、博弈策略、网络信号控制、客运专线开行方案及油田开发规划等问题. 目前, 求解二层规划问题的传统算法大致可以分为 4 类: Karus-Kuhn-Tucker 方法^[4-7]、分支定界法^[8]、罚函数法^[9-12] 以及下降法^[13-14].

在实际问题中, 由于各种不确定因素普遍存在, 使得二层规划问题变得更为复杂. 如资源的数量、价格以及客户的需求等都是变动的和不确定的. 为了尽量按照决策问题的本来面目去认识它、描述它并解决它, 在二层规划中引入不确定信息(如随机变量、模糊变量)已显得非常必要. 一旦随机变量引入二层规划问题, 就可以得到相应的随机二层规划问题. 然而, 最近二十几年, 大部分研究者将注意力集中在确定性环境中二层规划问题的算法与应用方面, 而对随机二层规划问题研究相对较少^[15-22].

针对一类目标函数和约束函数系数均为随机变量的线性二层规划模型, 本文设计了求解该问题的分支定界-粒子群算法: 首先, 将该随机二层规划问题转换为带期望约束的确定性线性二层协方差规划问题, 然后利用 KKT 条件将二层协方差规划问题转化为单层规划问题; 接着, 利用分支定界技术处理单层规划模型中的互补松弛条件, 从而将单层规划问题分解为系列带线性约束的子规划问题, 并用粒子群优化算法对子规划问题进行求解.

1 随机线性二层规划模型与基本概念

假设 $x \in \mathbb{R}^{n_1}$, $y \in \mathbb{R}^{n_2}$, $F: \mathbb{R}^{n_1} \times \mathbb{R}^{n_2} \longrightarrow \mathbb{R}$, $f: \mathbb{R}^{n_1} \times \mathbb{R}^{n_2} \longrightarrow \mathbb{R}^m$, $A \in \mathbb{R}^{m \times n_1}$, $B \in \mathbb{R}^{m \times n_2}$, $b \in \mathbb{R}^m$, 随机二层规划可表述为:

① 收稿日期: 2017-12-04

基金项目: 国家自然科学基金资助项目(61673006); 国家留学基金委资助出国留学项目(201708420111).

作者简介: 张 涛(1978-), 男, 湖北随州人, 博士, 副教授, 主要从事最优化理论与方法、智能计算等方面的研究.

$$\begin{aligned}
\min_{\mathbf{x}, \mathbf{y}} F(\mathbf{x}, \mathbf{y}) &= \mathbf{c}_1^T \mathbf{x} + \mathbf{d}_1^T \mathbf{y} \\
\min_{\mathbf{y}} f(\mathbf{x}, \mathbf{y}) &= \mathbf{c}_2^T \mathbf{x} + \mathbf{d}_2^T \mathbf{y} \\
\text{s. t. } \mathbf{Ax} + \mathbf{By} &\leq \mathbf{b} \\
\mathbf{x} &\geq \mathbf{0}, \mathbf{y} \geq \mathbf{0}
\end{aligned} \tag{1}$$

其中: $F(\mathbf{x}, \mathbf{y})$ 和 $f(\mathbf{x}, \mathbf{y})$ 分别表示上层目标函数和下层目标函数; $\mathbf{c}_1, \mathbf{c}_2$ 为 n_1 维随机系数; $\mathbf{d}_1, \mathbf{d}_2$ 为 n_2 维随机变量.

假设 α_i 表示约束可以违背的概率, $P(\cdot)$ 表示概率测度. 于是, 问题(1) 的约束条件可以改写为

$$P(\mathbf{A}_i \mathbf{x} + \mathbf{B}_i \mathbf{y} \leq \mathbf{b}_i) \geq \alpha_i \quad i = 1, 2, \dots, m \tag{2}$$

假设 $F_i(\tau)$ 表示随机变量 $\mathbf{b}_i$ 的分布函数, 于是方程(2) 可表示为

$$P(\mathbf{A}_i \mathbf{x} + \mathbf{B}_i \mathbf{y} \leq \mathbf{b}_i) = 1 - F(\mathbf{A}_i \mathbf{x} + \mathbf{B}_i \mathbf{y}) \tag{3}$$

由式(2) 和式(3) 可得

$$F(\mathbf{A}_i \mathbf{x} + \mathbf{B}_i \mathbf{y}) \leq 1 - \alpha_i \tag{4}$$

假设 $\tau = F^{-1}(1 - \alpha_i)$, $K_{1-\alpha_i} = \max \tau$. 根据分布函数 $F_i(\tau)$ 的单调性, 式(2) 可以表示为:

$$\mathbf{A}_i \mathbf{x} + \mathbf{B}_i \mathbf{y} \leq \mathbf{K}_{1-\alpha} \quad i = 1, 2, \dots, m \tag{5}$$

即

$$\mathbf{Ax} + \mathbf{By} \leq \mathbf{K}_{1-\alpha} \tag{6}$$

其中 $\mathbf{K}_{1-\alpha} = (K_{1-\alpha_1}, K_{1-\alpha_2}, \dots, K_{1-\alpha_m})$. 于是, 基于方程(6), 问题(1) 可以改写为带有确定约束的随机线性二层规划问题, 具体形式如下:

$$\begin{aligned}
\min_{\mathbf{x}, \mathbf{y}} F(\mathbf{x}, \mathbf{y}) &= \mathbf{c}_1^T \mathbf{x} + \mathbf{d}_1^T \mathbf{y} \\
\min_{\mathbf{y}} f(\mathbf{x}, \mathbf{y}) &= \mathbf{c}_2^T \mathbf{x} + \mathbf{d}_2^T \mathbf{y} \\
\text{s. t. } \mathbf{Ax} + \mathbf{By} &\leq \mathbf{K}_{1-\alpha} \\
\mathbf{x} &\geq \mathbf{0}, \mathbf{y} \geq \mathbf{0}
\end{aligned} \tag{7}$$

为了处理目标函数中带有随机变量的二层线性规划问题, 本文将采用上、下层目标函数值最小方差模型, 从而将模型(7) 改写为如下确定性问题:

$$\begin{aligned}
\min_{\mathbf{x}, \mathbf{y}} \text{Var}(F(\mathbf{x}, \mathbf{y})) &= (\mathbf{x}^T, \mathbf{y}^T) \mathbf{V}_1 \begin{pmatrix} \mathbf{x} \\ \mathbf{y} \end{pmatrix} \\
\min_{\mathbf{y}} \text{Var}(f(\mathbf{x}, \mathbf{y})) &= (\mathbf{x}^T, \mathbf{y}^T) \mathbf{V}_2 \begin{pmatrix} \mathbf{x} \\ \mathbf{y} \end{pmatrix} \\
\text{s. t. } \mathbf{Ax} + \mathbf{By} &\leq \mathbf{K}_{1-\alpha} \\
\mathbf{x} &\geq \mathbf{0}, \mathbf{y} \geq \mathbf{0}
\end{aligned} \tag{8}$$

其中 $\text{Var}(\cdot)$ 表示目标函数值的方差, $\mathbf{V}_1$ 与 $\mathbf{V}_2$ 分别表示 $(\mathbf{c}_1, \mathbf{d}_1)$ 以及 $(\mathbf{c}_2, \mathbf{d}_2)$ 的协方差. 在本文中, 我们假设 $\mathbf{V}_1$ 与 $\mathbf{V}_2$ 均正定.

尽管模型(8) 能够保证上、下层目标函数值分布均匀, 具有较小的落差, 但上层决策者和下层决策者往往对上层目标函数值和下层目标函数值具有一定的数学期望. 因此, 假设上层决策者和下层决策者的预设期望水平分别为 β_1 和 β_2 , 则模型(8) 可改写成如下形式:

$$\begin{aligned}
\min_{\mathbf{x}, \mathbf{y}} \text{Var}(F(\mathbf{x}, \mathbf{y})) &= (\mathbf{x}^T, \mathbf{y}^T) \mathbf{V}_1 \begin{pmatrix} \mathbf{x} \\ \mathbf{y} \end{pmatrix} \\
\min_{\mathbf{y}} \text{Var}(f(\mathbf{x}, \mathbf{y})) &= (\mathbf{x}^T, \mathbf{y}^T) \mathbf{V}_2 \begin{pmatrix} \mathbf{x} \\ \mathbf{y} \end{pmatrix} \\
\text{s. t. } \mathbf{Ax} + \mathbf{By} &\leq \mathbf{K}_{1-\alpha} \\
\mathbf{m}_1^{cT} \mathbf{x} + \mathbf{m}_1^{dT} \mathbf{y} &\leq \beta_1 \\
\mathbf{m}_2^{cT} \mathbf{x} + \mathbf{m}_2^{dT} \mathbf{y} &\leq \beta_2
\end{aligned}$$

$$\mathbf{x} \geq \mathbf{0}, \mathbf{y} \geq \mathbf{0} \quad (9)$$

其中： $\mathbf{m}_1^c$ 与 $\mathbf{m}_1^d$ 分别为 $\mathbf{c}_1$ 和 $\mathbf{d}_1$ 的均值， $\mathbf{m}_2^c$ 与 $\mathbf{m}_2^d$ 分别为 $\mathbf{c}_2$ 和 $\mathbf{d}_2$ 的均值。

2 分支定界—粒子群混合算法

为了给出模型(9)最优解，我们首先要给出该二层规划问题的合理反映集。给定上层决策变量 $\mathbf{x}$ ，则问题(9)的合理反应集 $\mathbb{R}(\mathbf{x})$ 可通过求解下列规划获得：

$$\begin{aligned} \min_{\mathbf{y}} \text{Var}(f(\mathbf{x}, \mathbf{y})) &= (\bar{\mathbf{x}}^T, \mathbf{y}^T) \mathbf{V}_2 \begin{pmatrix} \bar{\mathbf{x}} \\ \mathbf{y} \end{pmatrix} \\ \text{s. t. } \mathbf{A}\bar{\mathbf{x}} + \mathbf{B}\mathbf{y} &\leq \mathbf{K}_{1-a} \\ \mathbf{m}_1^{cT} \bar{\mathbf{x}} + \mathbf{m}_1^{dT} \mathbf{y} &\leq \beta_1 \\ \mathbf{m}_2^{cT} \bar{\mathbf{x}} + \mathbf{m}_2^{dT} \mathbf{y} &\leq \beta_2 \\ \mathbf{y} &\geq \mathbf{0} \end{aligned} \quad (10)$$

记 $\mathbf{A}'_1 = (\mathbf{A}, \mathbf{m}_1^c, \mathbf{m}_2^c)^T$, $\mathbf{A}'_2 = (\mathbf{B}, \mathbf{m}_1^d, \mathbf{m}_2^d)^T$, $\mathbf{b}' = (\mathbf{K}_{1-a}, \beta_1, \beta_2)^T$ ，于是式(9)可简记为：

$$\begin{aligned} \min_{\mathbf{x}, \mathbf{y}} \text{Var}(F(\mathbf{x}, \mathbf{y})) &= (\bar{\mathbf{x}}^T, \mathbf{y}^T) \mathbf{V}_1 \begin{pmatrix} \bar{\mathbf{x}} \\ \mathbf{y} \end{pmatrix} \\ \min_{\mathbf{y}} \text{Var}(f(\mathbf{x}, \mathbf{y})) &= (\bar{\mathbf{x}}^T, \mathbf{y}^T) \mathbf{V}_2 \begin{pmatrix} \bar{\mathbf{x}} \\ \mathbf{y} \end{pmatrix} \\ \text{s. t. } \mathbf{A}'_1 \bar{\mathbf{x}} + \mathbf{A}'_2 \mathbf{y} &\leq \mathbf{b}' \\ \mathbf{x} &\geq \mathbf{0}, \mathbf{y} \geq \mathbf{0} \end{aligned} \quad (11)$$

假设 $\mathbf{x}^*$ 为给定的上层决策变量，模型(11)下层问题的拉格朗日函数为：

$$L(\mathbf{y}, \boldsymbol{\lambda}^T, \boldsymbol{\omega}^T, \mathbf{x}^*) = (\mathbf{x}^{*T}, \mathbf{y}^T) \mathbf{V}_2 \begin{pmatrix} \mathbf{x}^* \\ \mathbf{y} \end{pmatrix} + \boldsymbol{\lambda}^T (\mathbf{A}'_1 \mathbf{x}^* + \mathbf{A}'_2 \mathbf{y} - \mathbf{b}') - \boldsymbol{\omega}^T \mathbf{y} \quad (12)$$

其中 $\boldsymbol{\lambda}, \boldsymbol{\omega}$ 为拉格朗日乘子。于是，模型(11)的下层问题的 KKT 条件为：

$$\begin{aligned} 2 \sum_{j=1}^{n_1} \mathbf{V}_2(n_1 + i, j) x_j^* + 2 \sum_{j=n_1+1}^{n_1+n_2} \mathbf{V}_2(n_1 + i, j) y_{j-n_1} + \lambda_i \mathbf{A}'_2(i) - \omega_i &= 0 \quad i = 1, 2, \dots, n_2 \\ \mathbf{A}'_1 \mathbf{x}^* + \mathbf{A}'_2 \mathbf{y} - \mathbf{b}' &\leq \mathbf{0} \\ \boldsymbol{\lambda}^T (\mathbf{A}'_1 \mathbf{x}^* + \mathbf{A}'_2 \mathbf{y} - \mathbf{b}') &= \mathbf{0}, \boldsymbol{\omega}^T \mathbf{y} = \mathbf{0} \\ \mathbf{y} &\geq \mathbf{0}, \boldsymbol{\lambda} \geq \mathbf{0}, \boldsymbol{\omega} \geq \mathbf{0} \end{aligned} \quad (13)$$

基于方程(13)，二层规划模型(11)可以转化为如下单层规划问题：

$$\begin{aligned} \min_{\mathbf{x}, \mathbf{y}} \text{Var}(F(\mathbf{x}, \mathbf{y})) &= (\mathbf{x}^T, \mathbf{y}^T) \mathbf{V}_1 \begin{pmatrix} \mathbf{x} \\ \mathbf{y} \end{pmatrix} \\ \text{s. t. } 2 \sum_{j=1}^{n_1} \mathbf{V}_2(n_1 + i, j) x_j^* + 2 \sum_{j=n_1+1}^{n_1+n_2} \mathbf{V}_2(n_1 + i, j) y_{j-n_1} + \lambda_i \mathbf{A}'_2(i) - \omega_i &= 0 \quad i = 1, 2, \dots, n_2 \\ \mathbf{A}'_1 \mathbf{x}^* + \mathbf{A}'_2 \mathbf{y} - \mathbf{b}' &\leq \mathbf{0} \\ \boldsymbol{\lambda}^T (\mathbf{A}'_1 \mathbf{x}^* + \mathbf{A}'_2 \mathbf{y} - \mathbf{b}') &= \mathbf{0}, \boldsymbol{\omega}^T \mathbf{y} = \mathbf{0} \\ \mathbf{y} &\geq \mathbf{0}, \boldsymbol{\lambda} \geq \mathbf{0}, \boldsymbol{\omega} \geq \mathbf{0} \end{aligned} \quad (14)$$

2.1 算法设计

为了算法说明的方便，先将模型(14)中线性互补约束简写为 $\sum_{i=1}^{m+n_2+2} u_i g_i = 0$ 。模型(14)为带互补约束的线性二次规划问题。设计该类模型算法的关键在于处理模型中的互补约束条件。本文将结合分支定界法^[23]和粒子群算法，设计求解模型(14)的混合算法。

算法 1

Step 1 初始化上层种群 P , 种群规模为 N . 初始化上层循环变量 $t = 0$, 最大迭代次数为 G .

Step 2 对于每一个粒子 P_i , 计算适应度值. 确定粒子 P_i 的个体最优值 p_{best_i} 以及粒子群全局最优值 g_{best} .

Step 3 假设 $z = (x, y)$, 更新粒子的速度和位置:

$$v^{t+1} = \omega v^t + c_1 r_1 (p_{best} - z^t) + c_2 r_2 (g_{best} - z^t)$$

$$z^{t+1} = z^t + v^{t+1}$$

Step 4 $t_l = t_l + 1$, 若 $t \leq G$, 则转 Step 5. 否则, 输出问题的最优解.

Step 5 个体粒子最优粒子与全局最优粒子按 Step 5.1—Step 5.2 更新. 然后, 转 Step 3.

Step 5.1 对于每一个粒子 i , 若 $f(z^{t+1}) < f(p_{best}^t)$, 则 $p_{best}^{t+1} = z^{t+1}$; 否则, $p_{best}^{t+1} = p_{best}^t$.

Step 5.2 $g_{best}^{t+1} = \operatorname{argmin}\{f(p_{best_i}^{t+1}), f(g_{best}^t)\}$, $i = 1, 2, \dots, N$.

在上述 PSO 算法中, 本文设计了如下适应度函数来处理约束问题:

$$F(x) = \begin{cases} f(x) & \text{若 } g_j(x) \leq 0, j = 1, 2, \dots, m \\ f_{\max} + \sum_{j=1}^m \langle g_j(x) \rangle & \text{否则} \end{cases}$$

其中: $\langle g_j(x) \rangle$ 表示当 $g_j(x) > 0$ 时 $g_j(x)$ 的值. $f_{\max}$ 是群体中最差可行个体的目标函数值.

算法 2

假设指标集 $W = \{1, 2, \dots, m + n_2 + 2\}$, $W_k \subseteq W$, $S_k^+ = \{i \mid u_i = 0, i \in W_k\}$, $S_k^- = \{i \mid g_i = 0, i \in W_k\}$, $S_k^0 = \{i \mid i \notin W_k\}$, $P_k = \{i \mid u_i = 0 \text{ 或 } g_i = 0, i \in W_k\}$.

Step 1 令 $k = 0$, $S_k^+ = \Phi$, $S_k^- = \Phi$, $S_k^0 = W$, $V = -\infty$.

Step 2 当 $u_i = 0 (i \in S_k^+)$ 及 $g_i = 0 (i \in S_k^-)$ 时, 利用算法 1 求解模型(14). 如果此解不可行, 则转 Step 6. 否则, 令 $k = k + 1$, $(x^k, y^k, \lambda^k, \omega^k)$ 为当前所求解.

Step 3 如果 $\operatorname{Var}(F_k) \geq V$, 转 Step 6.

Step 4 如果 $u_i g_i = 0, i = 1, 2, \dots, m + n_2 + 2$, 则转 Step 5. 否则, 找出满足条件 $u_i^* g_i^* = \max_{1 \leq i \leq m + n_2 + 2} \{u_i g_i\}$ 的 i^* , 令 $S_k^+ = S_k^+ \cup \{i^*\}$, $S_k^0 = S_k^0 \setminus \{i^*\}$, $S_k^- = S_k^-, i^* \in P_k$. 转 Step 2.

Step 5 更新 V . 令 $\operatorname{Var}(F_k) = V$.

Step 6 如果现存节点探索完毕, 则转 Step 7. 否则, 分支到最新的节点, 更新 S_k^+ , S_k^- , S_k^0 及 P_k , 转 Step 2.

Step 7 终止条件检测. 若 $V = -\infty$, 则原问题无可行解. 否则, 与 V 对应的 (x^*, y^*) 为原问题的最优解.

2.2 算法时间复杂度分析

在本文中, 求解随机线性二层规划问题的分支定界-粒子群混合算法的计算主要包括 2 部分: PSO 算法对子二次规划的寻优计算及互补松弛条件的验证计算. 假设粒子群的规模为 N , 迭代次数为 G , 单个粒子更新位置及计算适应度值的时间为 t_1 . 假设计算 $u_i g_i(x^k, y^k)$ 的时间为 t_2 , 假设顶点个数为 K .

系列子规划的寻优时间为 $T_1 = N \times G \times t_1$. 在调用算法 1 之前, 总要检查互补松弛条件的满足情况. 在算法 2 中, 对于维数为 $(m + n_2 + 2)$ 的向量, 其每一个分量有 2 种状态: 为 0 或不为 0, 因此, 该向量共有 2^{m+n_2+2} 种形式. 这里考察计算代价最大的情形, 即直到最后一种组合才满足互补松弛条件的情形, 则完成一次检查共需时间为 $T_2 = 2^{m+n_2+2} \times t_2$, 故算法的时间复杂度为 $T = (T_1 + T_2) \times K$.

3 数值仿真与结果分析

在算法 1 中, 若 $u_i g_i = 1 \times 10^{-6}$ 则认为其满足模型(14) 中的互不松弛条件. 在算法 2 中, 粒子群优化算法的相关参数设置为: $r_1, r_2 \in \operatorname{random}(0, 1)$, 惯性权重 $\omega = 0.7298$; 社会系数和认知系数 $c_1 = c_2 =$

1.496 18；粒子规模 $N = 20$ 。计算机硬件配置为：（CPU 为 AMD Phenon（tm）II X6 1055T 2.80 GHz；内存为 3.25 GB RAM）。利用 C# 编程进行求解。

算例 1 假设 $x \in \mathbb{R}, y \in \mathbb{R}$ ，考虑如下问题^[24]：

$$\begin{aligned} \min_{x,y} F(x,y) &= c_1x + d_1y \\ \min_y f(x,y) &= c_2x + d_2y \\ \text{s. t. } &-x + 3y \leqslant b_1, 10x - y \leqslant b_2 \\ &3x + y \geqslant b_3, x + 2y \geqslant b_4 \\ &3x + 2y \geqslant b_5, x \geqslant 0, y \geqslant 0 \end{aligned} \tag{15}$$

其中： $m_1^c = -2.0, m_1^d = -3.0, m_2^c = 2.0, m_2^d = 1.0, \mathbf{V}_1 = \begin{pmatrix} 2 & 1 \\ 1 & 3 \end{pmatrix}, \mathbf{V}_2 = \begin{pmatrix} 1 & -1 \\ -1 & 6 \end{pmatrix}$ 。假设约束的右端随机变量 $\mathbf{b} = (b_1, b_2, \cdots, b_5)$ 服从正态分布，其均值、方差以及允许约束违反的最大概率见表 1。

表 1 约束条件右端随机变量的均值、方差和可接受概率

	b_1	b_2	b_3	b_4	b_5
平均值	50.11	113.15	15.16	13.16	25.63
方差	9.0	36	9.0	4.0	16
可接受概率	0.85	0.7	0.9	0.7	0.8

根据上述条件，问题(15) 可表述为：

$$\begin{aligned} \min_{x,y} F(x,y) &= (x,y) \begin{pmatrix} 2 & 1 \\ 1 & 3 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} \\ \min_y f(x,y) &= (x,y) \begin{pmatrix} 1 & -1 \\ -1 & 6 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} \\ \text{s. t. } &-x + 3y \leqslant 47, 10x - y \leqslant 110 \\ &3x + y \geqslant -19, x + 2y \geqslant -15 \\ &3x + 2y \geqslant -29, -2x - 3y \leqslant -31 \\ &2x + y \leqslant 33, x \geqslant 0, y \geqslant 0 \end{aligned} \tag{16}$$

根据方程(14) 将二层规划转化为单层规划：

$$\begin{aligned} \min_{x,y} F(x,y) &= (x,y) \begin{pmatrix} 2 & 1 \\ 1 & 3 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} \\ \text{s. t. } &-x + 3y \leqslant 47, 10x - y \leqslant 110 \\ &3x + y \geqslant -19, x + 2y \geqslant -15 \\ &3x + 2y \geqslant -29, -2x - 3y \leqslant -31 \\ &2x + y \leqslant 33, x \geqslant 0, y \geqslant 0 \\ &2x + 12y + 3\lambda_1 - \lambda_2 - \lambda_3 - 2\lambda_4 - 2\lambda_5 - 3\lambda_6 + \lambda_7 - \omega = 0 \\ &\lambda_1(x - 3y + 47) + \lambda_2(-10x + y + 110) + \lambda_3(3x + y - 19) + \\ &\lambda_4(x + 2y - 15) + \lambda_5(3x + 2y - 29) + \lambda_6(2x + 3y - 31) + \\ &\lambda_7(-2x - y + 33) + \omega y = 0 \\ &x \geqslant 0, y \geqslant 0, \lambda_i \geqslant 0, \omega \geqslant 0, i = 1, 2, \cdots, 7 \end{aligned} \tag{17}$$

针对算例(17)，我们对带均值约束情况和无均值约束 2 种情况进行了求解，并将求解结果与文献[24] 作对比(表 2)。从比较结果来看，在上层函数均值相同的情况下，利用本文算法所获得上层函数方差要略优于文献[24] 的算法。此外，本文所设计的算法的计算效率也有明显提高。

表 2 问题(17) 运行 11 次的最优解比较

	算法	最优解	上层平均值	下层平均值	上层方差	下层方差	运行时间 /s
有均值约束	本文算法	(5.15, 6.90)	− 31.00	17.20	266.95	241.11	0.033 5
	文献[24] 算法	(5.17, 6.89)	− 31.00	17.23	267.12	240.32	0.087 9
无均值约束	本文算法	(6.99, 4.00)	− 25.98	17.99	201.64	88.94	0.017 3
	文献[24] 算法	(7.00, 4.00)	− 26.00	18.00	202.00	89.00	0.020 8

算例 2 假设 $x \in \mathbb{R}^5$, $y \in \mathbb{R}^3$, $c_1, c_2 \in \mathbb{R}^5$, $d_1, d_2 \in \mathbb{R}^3$, $A \in \mathbb{R}^{3 \times 5}$, $B \in \mathbb{R}^{3 \times 3}$, $b \in \mathbb{R}^3$, 考虑如下问题^[24]:

$$\min_{x, y} F(x, y) = c_1^T x + d_1^T y$$
$$\min_y f(x, y) = c_2^T x + d_2^T y$$
$$\text{s. t. } Ax + By \leqslant b, x \geqslant 0, y \geqslant 0$$

(18)

其中: $b_1 \sim N(220, 4^2)$, $b_2 \sim N(145, 3^2)$, $b_3 \sim N(-18, 5^2)$; $\alpha_1 = 0.85$, $\alpha_2 = 0.95$, $\alpha_3 = 0.90$, 与之相对应的 $K_{1-\alpha_1} = 215.85$, $K_{1-\alpha_2} = 140.07$, $K_{1-\alpha_3} = 24.41$. 上层决策者和下层决策者的预设期望水平分别为 $\beta_1 = 60.0$, $\beta_2 = 35.0$. 约束条件的系数矩阵如下:

$$A = \begin{bmatrix} 7 & 2 & 6 & 9 & 11 \\ 5 & 6 & -4 & 3 & 3 \\ -4 & -7 & -2 & -6 & -8 \end{bmatrix} \quad B = \begin{bmatrix} 4 & 3 & 8 \\ -7 & -1 & -3 \\ -3 & -5 & -6 \end{bmatrix}$$

c_1, c_2, d_1, d_2 的数学期望分别是: $m_{c_1} = (5, 2, 1, 2, 1)$, $m_{c_2} = (10, -7, 1, -2, -5)$, $m_{d_1} = (6, 1, 3)$, $m_{d_2} = (3, -4, 6)$.

$$V_1 = \begin{bmatrix} 16.0 & -1.6 & 1.8 & -3.5 & 1.3 & -2.0 & 4.0 & -1.4 \\ -1.6 & 25.0 & -2.2 & 1.6 & -0.7 & 0.5 & -1.3 & 2.0 \\ 1.8 & -2.2 & 25.0 & -2.0 & 5.0 & -2.4 & 1.2 & -2.1 \\ -3.5 & 1.6 & -2.0 & 16.0 & -2.0 & 3.0 & 2.2 & 2.8 \\ 1.3 & -0.7 & 5.0 & -2.0 & 4.0 & -1.0 & 0.8 & -2.0 \\ -2.0 & 0.5 & -2.4 & 3.0 & -1.0 & 1.0 & -1.5 & 0.6 \\ 4.0 & -1.3 & 1.2 & 2.2 & 0.8 & -1.5 & 4.0 & -2.3 \\ -1.4 & 2.0 & -2.1 & 2.8 & -2.0 & 0.6 & -2.3 & 4.0 \end{bmatrix}$$

$$V_2 = \begin{bmatrix} 4.0 & -1.4 & 0.8 & 0.2 & 1.6 & 1.0 & 1.2 & 2.0 \\ -1.4 & 4.0 & 0.2 & -1.0 & -2.2 & 0.8 & 0.9 & 1.8 \\ 0.8 & 0.2 & 9.0 & 0.2 & -1.5 & 1.5 & 1.0 & 0.6 \\ 0.2 & -1.0 & 0.2 & 36.0 & 0.8 & 0.4 & -1.5 & 0.7 \\ 1.6 & -2.2 & -1.5 & 0.8 & 25.0 & 1.2 & -0.2 & 2.0 \\ 1.0 & 0.8 & 1.5 & 0.4 & 1.2 & 25.0 & 0.5 & 1.4 \\ 1.2 & 0.9 & 1.0 & -1.5 & -0.2 & 0.5 & 9.0 & 0.8 \\ 2.0 & 1.8 & 0.6 & 0.7 & 2.0 & 1.4 & 0.8 & 16.0 \end{bmatrix}$$

根据模型(8), 问题(18) 可表述为:

$$\min_{x, y} F(x, y) = (x^T, y^T)V_1 \begin{pmatrix} x \\ y \end{pmatrix}$$
$$\min_y f(x, y) = (x^T, y^T)V_2 \begin{pmatrix} x \\ y \end{pmatrix}$$
$$\text{s. t. } Ax + By \leqslant (215.85, 140.07, 24.41)^T$$
$$m_{c_1}^T x + m_{d_1}^T y \leqslant 60.0$$

$$\boldsymbol{m}_{d_2}^{\text{T}} \boldsymbol{x} + \boldsymbol{m}_{d_2}^{\text{T}} \boldsymbol{y} \leqslant 35.0$$

(19)

针对问题(19)，我们对带均值约束情况 and 无均值约束 2 种情况进行了求解，并将求解结果与文献[24]作了对比分析，如表 3 所示。从比较结果来看，在上层函数均值相同的情况下，利用本文算法所获得上层函数方差要略优于文献[24] 的算法。此外，随着问题的维数的增加，本文所设计的算法表现出较好的计算效率。

表 3 问题(19) 运行 11 次的最优解比较结果

算法		最优解	上层平均值	下层平均值	上层方差	下层方差	运行时间 /s
有均值约束	本文算法	(0.052 9, 0.250 9, 0, 0.007 2, 0.950 1; 0.598 7, 1.232 1, 1.139 9)	9.976 9	− 2.282 2	5.968 9	77.725 3	4.389 8
	文献[24]	(0.053 0, 0.251 0, 0, 0.007 0, 0.950 0; 0.599 0, 1.232 0, 1.140 0)	9.977 0	− 2.282 0	5.970 0	77.725 0	21.217 2
无均值约束	本文算法	(0.410 1, 0.632 0, 0.067 1, 0.270 2, 0.801 2; 0.275 7, 0.973 1, 0.748 8)	9.597 9	− 3.372 7	20.653 6	48.355 2	0.965 4
	文献[24]	(0.410 0, 0.632 0, 0.067 0, 0.270 0, 0.801 0, 0.276 0, 0.973 0, 0.749 0)	9.598 0	− 3.372 0,	20.654 0	48.355 0	7.183 6

上述算例 1 和算例 2 均为求解具有一般性的随机线性二层规划问题，即上、下层目标函数系数以及约束函数的系数均为随机变量。为了进一步说明该算法对具有特殊结构的随机线性二层规划问题同样有效，我们将以带补偿的随机线性二层规划问题为例进行求解，该模型的主要特点表现为约束函数的系数为随机变量，目标函数系数为确定值。

算例 3 假设 $x \in \mathbb{R}, y \in \mathbb{R}$ ，考虑如下问题^[19]：

$$\begin{aligned} \min_{x, y} F(x, y) &= -x - 2y \\ \min_y f(x, y) &= -x + 11y \\ \text{s. t. } P \left[\begin{aligned} A_1x - B_1y &\leqslant 4 \\ 2A_2x - B_2y &\leqslant 24 \\ 3A_3 + 4B_3y &\leqslant 96 \\ A_4x + 7B_4y &\leqslant 126 \end{aligned} \right] &\geqslant 0.8, x \geqslant 0, y \geqslant 0 \end{aligned}$$

(20)

其中： $A_1, A_2, A_3, A_4, B_1, B_2, B_3, B_4 \sim \text{U}(0, 1)$ ，每个条件约束可接受的概率 $p = 0.8$ 。本文将分别采用 10 个子样和 30 个子样对问题(20) 进行建模，从而得到确定性线性二层规划问题(21) 和(22)。

10 子样建模：

$$\begin{aligned} \min_{x, y} F(x, y) &= -x - 2y \\ \min_y f(x, y) &= -x + 11y \\ \text{s. t. } 0.516x - 1.6y &\leqslant 3.2 \\ 1.133x - 0.5441y &\leqslant 19.2 \\ 1.608x + 2.139y &\leqslant 76.8 \\ 0.528x + 3.516y &\leqslant 100.8 \\ x \geqslant 0, y &\geqslant 0 \end{aligned}$$

(21)

30 子样建模：

$$\begin{aligned} \min_{x, y} F(x, y) &= -x - 2y \\ \min_y f(x, y) &= -x + 11y \\ \text{s. t. } 0.4325x - 1.556y &\leqslant 3.2 \end{aligned}$$

$$\begin{aligned} 0.793x - 0.4260y &\leq 19.2 \\ 1.2186x + 1.6556y &\leq 76.8 \\ 0.4865x + 3.4041y &\leq 100.8 \\ x \geq 0, y \geq 0 \end{aligned} \tag{22}$$

利用本文的算法与文献[19]中的算法分别对模型(21)和(22)进行求解,其比较结果如表4所示.从比较结果来看,两种算法在最优解几乎一样的情况下,本文所设计的算法具有较高的计算效率.

表 4 问题(21), (22) 运行 11 次的最优解比较结果

	算法	最优解	上层最优值	下层最优值	运行时间 /s
10 子样模型	本文算法	(64.883 9, 18.924 8)	− 102.733 35	78.407 0	0.001 79
	文献[19]	(64.884 0, 18.925 0)	− 102.734 0	78.407 0	0.037 9
30 子样模型	本文算法	(75.243 1, 18.856 9)	− 112.956 9	56.943 3	0.002 01
	文献[19]	(75.243 2, 18.857 0)	− 112.957 2	56.939 5	0.046 3

4 总结

针对一类随机线性二层规划模型,本文提出了基于分支定界的粒子群优化算法,并利用数值算例进行了带期望约束和不带期望约束的仿真实验,实验结果表明本文算法有效改善了上层问题的协方差结果且计算效率得到了较显著提高.此外,该算法也适合较为特殊的带补偿的随机线性二层规划问题.

在未来的工作中,主要有下列问题需要考虑:①在算法2的step2中,能否针对子问题设计对应的并行粒子群算法,从而提高计算效率;②针对随机非线性规划问题,如何设计其求解算法;③针对随机二层多目标规划问题,又该如何设计其求解算法.

参考文献:

[1] BARD J F, FALK J E. An Explicit Solution to the Multilevel Programming Problem [J]. Computers and Operations Research, 1982, 9(1): 77–100.

[2] DEMPLE S. Foundation of Bilevel Programming [M]. London: Kluwer Academic, 2002.

[3] 藤春贤,李智慧. 二层规划理论与应用 [M]. 北京: 科学出版社, 2002.

[4] BARD J. An Algorithm for Solving the General Bilevel Programming Problem [J]. Mathematics of Operations Research, 1983, 8(2): 260–272.

[5] EDMUNDS T, BARD J. Algorithm for Nonlinear Bilevel Mathematical Programs [J]. IEEE Transactions on Systems, Men, and Cybernetics, 1991, 21(1): 83–89.

[6] AMOUZEGAR M. A Global Optimization Method for Nonlinear Bilevel Programming Problems [J]. IEEE Transactions on Systems, Men, and Cybernetics, 1999, 29(6): 771–777.

[7] ETOA J. Solving Quadratic Convex Bilevel Programming Problems Using a Smoothing Method [J]. Applied Mathematics and Computation, 2011, 217(15): 680–6690.

[8] BARD J, FALK J. An Explicit Solution to the Multi-Level Programming Problem [J]. Computers & Operations Research, 1982, 9(1): 77–100.

[9] ISHIZUKA Y, AIYOSHI E. A New Computational Method for Stackelberg and Min-Max Problems by Use of a Penalty Method [J]. IEEE Transactions on Automatic Control, 1981, 26(2): 460–466.

[10] AIYOSHI E, SHIMUZU K. A Solution Method for the Static Constrained Stackelberg Problem via Penalty Method [J]. IEEE Transactions on Automatic Control, 1984, 29(12): 1112–1114.

[11] ISHIZUKA Y, AIYOSHI E. Double Penalty Method for Bilevel Optimization Problems [J]. Annals of Operations Research, 1992, 34(1): 73–88.

[12] LV Y, HU T, WANG G, WAN Z. A Penalty Function Method Based on Kuhn-Tucker Condition for Solving Linear Bilevel Programming [J]. Applied Mathematics and Computation, 2007, 188(1): 808–813.

[13] SAVARD G, GAUVIN J. The Steepest Descent Direction for the Nonlinear Bilevel Programming Problem [J]. Opera-

- tions Research Letters, 1994, 15(5): 265–272.
- [14] FALK J, LIU J. On Bilevel Programming, Part I: General Nonlinear Cases [J]. Mathematical Programming, 1995, 70(1–3): 47–72.
- [15] PATRIKSSON M, WYNTER L. Stochastic Mathematical Programs with Equilibrium Constraints [J]. Operations Research Letters, 1999, 25(4): 159–167.
- [16] GAO J, LIU B, GEN M. A Hybrid Intelligent Algorithm for Stochastic Multilevel Programming [J]. IEEE Transactions on Electronics, Information and Systems, 2004, 24(10): 1991–1998.
- [17] 万仲平, 肖昌育, 王先甲, 等. 不确定市场下的一种二层规划最优竞价模型[J]. 电力系统自动化, 2004, 28(19): 12–16.
- [18] 万仲平, 樊 恒, 王广民, 等. 基于随机二层规划的不确定电力市场中交互式模糊竞价决策方法[J]. 武汉大学学报(工学版), 2006, 39(5): 85–90.
- [19] 何 云, 冯春强. 带补偿的二层随机规划逼近算法研究[J]. 纺织高校基础科学学报, 2013, 26(1): 110–113.
- [20] 周婉娜, 霍永亮. 二层随机规划逼近最优解集的稳定性分析[J]. 重庆工商大学学报(自然科学版), 2013, 30(7): 19–23.
- [21] 周婉娜, 霍永亮, 胡之英. 二层随机规划逼近 ϵ -最优解集的 Hausdorff 收敛性[J]. 湖南师范大学自然科学学报, 2016, 39(3): 80–83.
- [22] 刘 勇, 王 慧, 徐裕生, 等. 二层随机规划逼近解的收敛性[J]. 纯粹数学与应用数学, 2008, 24(4): 768–773.
- [23] BARD J F, MOORE J T. A Branch and Bound Algorithm for the Bilevel Programming Problem [J]. SIAM Journal on Scientific and Statistical Computing, 1990, 11(2): 81–292.
- [24] MASATOSHI S, ICHIRO N. Cooperative and Non-cooperative Multi-Level Programming [M]. New York: Springer Science, 2009.

A Particle Swarm Optimization-Branch and Bound Algorithm for the Linear Stochastic Bilevel Programming Problem

ZHANG Tao

School of Information and Mathematics, Yangtze University, Jingzhou Hubei 434023, China

Abstract: First, a class of stochastic linear bilevel programming models is transformed into a deterministic linear bilevel covariance programming model with expected constraints. Then, using the KKT condition, the deterministic bilevel covariance programming problem is transformed into a deterministic single level programming problem. After that, the single level programming problem is solved by the particle swarm optimization-branch and bound algorithm. The results of a comparative experiment show that the proposed algorithm can improve the variance results of the upper level problem and significantly increase the computational efficiency.

Key words: linear stochastic bilevel programming, bilevel variance model with expectation constraints, particle swarm optimization algorithm, branch and bound method

责任编辑 张 构