文章编号:1000－2367(2018)01－0009－07　　DOI:10.16366/j.cnki.1000－2367.2018.01.002

# 线性两比式和的全局优化新算法

夏勇，王龙飞

(北京航空航天大学 数学与系统科学学院，北京 100191)

**摘　要:**对线性两比式和这一非凸 NP-困难的优化问题提出新的全局优化算法.首先把原问题等价地转化为一维参数优化问题.设计了巧妙的下界估计方法,在此基础上提出相应的分支定界算法,该算法最坏情况下可需要 $O(1/\varepsilon)$ 迭代步以求得 $\varepsilon$ -近似全局最优解.数值结果表明,提出的新算法优于商业软件包 BARON.此外,针对线性两比式和问题的一个具有隐凸性(等价于一个二阶锥规划)的应用特例,分支定界算法比基于 CVX 平台调用 SDPT3 求解相应的二阶锥规划等价模型效率更高.

**关键词:**线性比式和;线性规划;分支定界;对偶;二阶锥规划

**中图分类号:**O221.2　　**文献标志码:**A

一般的线性比式和问题模型为:

$$\begin{aligned} \min \quad & \sum_{i=1}^{p} \frac{a_i^{\mathrm{T}} y + a_{0i}}{b_i^{\mathrm{T}} y + b_{0i}}, \\ \text{s.t.} \quad & By = c, 0 \leq y \leq d, \end{aligned} \tag{1}$$

其中,$B \in \mathbf{R}^{m \times n}, a_i, b_i \in \mathbf{R}^n (i = 1, \cdots, p), c \in \mathbf{R}^m, d \in \mathbf{R}^n, a_{0i}, b_{0i} \in \mathbf{R} (i = 1, 2, \cdots, p)$.

线性比式和问题在叠层制造[1－2]、材料布局[3－5]和经济学[6－7]等领域中有许多重要应用.其算法为众多学者所关注[8－18].例如,基于文献[11],通过引进界提升和锥压缩等策略,文献[12] 提出了新的分支缩减定界算法.文献[13]把原问题等价地转化为双线性规划问题,然后利用双变量乘积函数的凸包络与凹包络的线性化技巧,松弛构造原问题的下界,再应用分支定界算法求解.文献[14]利用线性化技巧构造原问题的线性松弛下界,再利用剪枝策略提出了一个新的算法且给出了此算法的收敛性.文献[15]提出近似求解算法,当 $p$ 是固定数时复杂度分析表明其是完全多项式时间近似算法.文献[16]将该结论推广到带线性矩阵不等式的线性比式和问题.

本文考虑一个最简单的线性比式和问题,即 $p = 2$.通过 Charnes-Cooper 变换[19],线性两比式和问题可以进一步等价简化为如下优化问题:

$$\begin{aligned} (\mathrm{P}) \quad \min \quad & \frac{a_1^{\mathrm{T}} x + b_1}{a_2^{\mathrm{T}} x + b_2} + a_3^{\mathrm{T}} x, \\ \text{s.t.} \quad & x \in D = \{ x \mid \widetilde{B} x = \tilde{c}, 0 \leq x \leq \tilde{d} \}. \end{aligned}$$

为避免分母为零造成问题没有定义,假定 $D \subseteq \{ x \mid a_2^{\mathrm{T}} x + b_2 > 0 \}$.

问题(P)在经济学[20]和债券投资组合[21]等领域中依然有诸多应用.问题(P)的目标函数既不是拟凸也不是拟凹函数,即使对于二维变量,文献[22] 给出了一个三维图形表明该目标函数既不是凹函数也不是凸函数,因此问题(P)的求解仍有很大的挑战性[23],事实上,文献[24－25]已经证明问题(P)为 NP-难问题.文献[26]基于最优水平解的概念,从一个初始的最优水平解出发,利用带参数的序列算法求得一系列局部最优解,从而得到问题的最优解.文献[27]通过引入辅助变量,把原问题等价转化为($n$＋2)-维优化问题,调用外逼

---

**收稿日期:**2017-12-20;**修回日期:**2017-12-30.

**基金项目:**国家自然科学基金(11571029;11471325;11771056)

**作者简介(通信作者):**夏勇(1980－),男,江苏海安县人,北京航空航天大学副教授,博士生导师,研究方向为全局最优化,E-mail:yxia@buaa.edu.cn.

近算法求解.

本文首先将(P)提升为一个等价的一维参数优化问题:

$$(\mathrm{PM})\quad \min_{r\in[r_{\min},r_{\max}]}\left\{\mathcal{G}(r):=\min_{x\in D,a_2^{\mathrm{T}}x+b_2=r}\left\{\frac{a_1^{\mathrm{T}}x+b_1}{r}+a_3^{\mathrm{T}}x\right\}\right\}. \tag{2}$$

其中 $r_{\min}=\min_{x\in D}a_2^{\mathrm{T}}x+b_2$,$r_{\max}=\max_{x\in D}a_2^{\mathrm{T}}x+b_2$,对应着两个线性规划问题.(PM)的特点是对于任意给定的参数$r$,通过求解一个线性规划问题可以获取目标函数值$\mathcal{G}(r)$,然而却无法获知$\mathcal{G}(r)$关于$r$的显式函数表达式.

本文提出了一个巧妙的估界方法,通过任意两个端点处$\mathcal{G}(r)$值的信息,可以显式给出整个区间上的下界函数,基于该下界函数,进一步设计了高效分支定界算法来求解(PM),且可以在至多$O(1/\varepsilon)$步迭代后找到$\varepsilon$-近似全局最优解.

作为对比,先将(P)等价改写为如下非凸二次约束二次规划,然后调用商业软件包 BARON[28] 进行求解:

$$\begin{aligned}(\mathrm{QP})\quad &\min_{x,t}\quad t+a_3^{\mathrm{T}}x,\\ &\mathrm{s.t.}\quad t(a_2^{\mathrm{T}}x+b_2)=a_1^{\mathrm{T}}x+b_1,x\in D.\end{aligned}$$

此外,还计算了(P)的一个特例[29]:$a_2=a_3$,它衡量了某种意义上的公平性,具体见文献[29].该特例中的目标函数可以证明是伪凸函数,且可以证明该特例进一步等价于下列的二阶锥规划问题[29]:

$$\begin{aligned}(\mathrm{SOCP_{LP}})\quad \min_{s,t,y,z}\quad & t-b_2,\\ \mathrm{s.t.}\quad & Ay-sb=0,\\ & a_2^{\mathrm{T}}y+b_2s=1,\\ & z=2b_2a_2^{\mathrm{T}}y+b_2^2+a_1^{\mathrm{T}}y+(b_2^2+b_1)s,\\ & \|(2a_2^{\mathrm{T}}y,s-t+z)^T\|\le s+t-z,\\ & y\ge 0,\end{aligned}$$

其中范数为欧氏范数,其所在的约束为一个三维的二阶锥约束.基于 CVX 平台[30]的 SDPT3 软件包求解.如果$(\bar{s},\bar{t},\bar{y},\bar{z})$是$(\mathrm{SOCP_{LP}})$的最优解,则$\bar{x}=\bar{y}/\bar{s}$是问题(P)的最优解.

## 1 新的估计方法

考虑调用分支定界算法求解(PM),一维区间上的分支极为简单,即将$[r_{\min},r_{\max}]$剖分为$\bigcup_{i=1}^{k}[r_i,r_{i+1}]$,算法效率取决于在每个剖分区间$[r_i,r_{i+1}]$上高效计算

$$(\mathrm{PM1})\quad \min_{r\in[r_i,r_{i+1}]}\mathcal{G}(r)$$

的一个高质量的下界,前提是只有在两个区间端点$r_i$和$r_{i+1}$处求解相应目标函数$\mathcal{G}(r)$而获得的解的信息.

首先注意到在两端点处的函数值计算$\mathcal{G}(r_i)$和$\mathcal{G}(r_{i+1})$对应于两个线性规划问题.求解得到最优的$x$的解和对应于约束$a_2^{\mathrm{T}}x+b_2=r$的最优乘子,分别记为$(x(r_i),\lambda(r_i))$和$(x(r_{i+1}),\lambda(r_{i+1}))$.下面利用两端点处的这些解信息来巧妙估计(PM1)的下界.

简便起见,令$p(x)=a_1^{\mathrm{T}}x+b_1$,$h(x)=a_3^{\mathrm{T}}x$,$g(x)=a_2^{\mathrm{T}}x+b_2$,则对于任意的$r\in[r_i,r_{i+1}]$,目标函数值$\mathcal{G}(r)$对应求解未知量为$x$的线性规划问题,根据线性规划的强对偶性,有,

$$\mathcal{G}(r)=\max_{\lambda\in\mathbf{R}}\min_{x\in D}\frac{p(x)}{r}+h(x)-\lambda(g(x)-r)= \tag{3}$$

$$\max_{\lambda\in\mathbf{R}}\frac{p(x(\lambda,r))}{r}+h(x(\lambda,r))-\lambda\, g(x(\lambda,r))+r\lambda, \tag{4}$$

其中,$x(\lambda,r)$是(3)式的内层优化问题的最优解.

令$(x(r),\lambda(r))$是线性规划$\mathcal{G}(r)$的解,则由强对偶性质,$\mathcal{G}(r)$等价于在最优乘子处拉格朗日函数的最优值,即

$$\mathcal{G}(r)=\min_{x\in D}\frac{p(x)}{r}+h(x)-\lambda(r)(g(x)-r). \tag{5}$$

在(5)式中分别令 $r=r_i,r_{i+1}$，由其定义，目标函数(5) 在 $x(\lambda,r)$ 处的值则要不小于其值，从而有：

$$\frac{p(x(\lambda,r))}{r_i}+h(x(\lambda,r))-\lambda(r_i)\,g(x(\lambda,r))+r_i\lambda(r_i)\geq \mathcal{G}(r_i), \tag{6}$$

$$\frac{p(x(\lambda,r))}{r_{i+1}}+h(x(\lambda,r))-\lambda(r_{i+1})\,g(x(\lambda,r))+r_{i+1}\lambda(r_{i+1})\geq \mathcal{G}(r_{i+1}), \tag{7}$$

在(4)式中，未知量 $p(x(\lambda,r))$，$h(x(\lambda,r))$ 和 $g(x(\lambda,r))$ 满足(6)～(7)式，于是分别将其松弛为 3 个满足(6)～(7)式的一维变量 $y_p,y_h,y_g$，从而得到 $\mathcal{G}(r)$ 关于 $r$ 在$[r_i,r_{i+1}]$上的下界函数为：

$$\underline{\mathcal{G}}(\alpha):=\max_{\lambda\in\mathbf{R}}\ \min_{y_p,y_h,y_g}\quad \frac{y_p}{r}+y_h-\lambda\,y_g+r\lambda, \tag{8}$$

$$\text{s.t.}\qquad \frac{y_p}{r_i}+y_h-\lambda(r_i)\,y_g+r_i\lambda(r_i)\geq \mathcal{G}(r_i), \tag{9}$$

$$\frac{y_p}{r_{i+1}}+y_h-\lambda(r_{i+1})\,y_g+r_{i+1}\lambda(r_{i+1})\geq \mathcal{G}(r_{i+1}), \tag{10}$$

(8)～(10)式的内层优化问题是关于未知量 $(y_p,y_h,y_g)$ 的三维线性规划，由线性规划的强对偶性等价于其极大化型的对偶问题.这样，(8)～(10)式等价于下列的极大优化问题：

$$\max_{\lambda\in\mathbf{R}}\quad \max_{\mu_1,\mu_2}\quad \mu_1(\mathcal{G}(r_i)-r_i\lambda(r_i))+\mu_2(\mathcal{G}(r_{i+1})-r_{i+1}\lambda(r_{i+1}))+r\lambda, \tag{11}$$

$$\text{s.t}\qquad \frac{1}{r_i}\mu_1+\frac{1}{r_{i+1}}\mu_2=\frac{1}{r}, \tag{12}$$

$$\mu_1+\mu_2=1, \tag{13}$$

$$\lambda(r_i)\mu_1+\lambda(r_{i+1})\mu_2=\lambda, \tag{14}$$

$$\mu_1,\mu_2\geq 0. \tag{15}$$

因为 $r_i\neq r_{i+1}$，由(12)～(13)式可以唯一地求出 $\mu_1,\mu_2$：

$$\mu_1=\frac{r_i(r_{i+1}-r)}{r(r_{i+1}-r_i)},\mu_2=\frac{r_{i+1}(r-r_i)}{r(r_{i+1}-r_i)}. \tag{16}$$

进一步，因为 $0<r_i\leq r\leq r_{i+1}$ 且后两个不等式中的等式不能同时取到，上述 $\mu_1$ 和 $\mu_2$ 的表达式(16) 式自动满足(15) 式.此外将解 $\mu_1$、$\mu_2$ 和(16) 式代入(14) 式，可知未知量 $\lambda$ 可行解唯一：

$$\lambda=\frac{r_i\,r_{i+1}}{r_{i+1}-r_i}(\lambda(r_i)-\lambda(r_{i+1}))\,\frac{1}{r}+\frac{1}{r_{i+1}-r_i}\,(\lambda(r_{i+1})\,r_{i+1}-\lambda(r_i)\,r_i). \tag{17}$$

将(16)式和(17)式代入(11)式，可得下界函数 $\underline{\mathcal{G}}(r)$ 的解析表达式：

$$\underline{\mathcal{G}}(r)=c_1\,r+\frac{c_2}{r}+c_3, \tag{18}$$

其中各系数表达式如下：

$$c_1=\frac{r_{i+1}\lambda(r_{i+1})-r_i\lambda(r_i)}{r_{i+1}-r_i}, \tag{19}$$

$$c_2=r_i\,r_{i+1}\left(c_1-\frac{\mathcal{G}(r_{i+1})-\mathcal{G}(r_i)}{r_{i+1}-r_i}\right), \tag{20}$$

$$c_3=\frac{r_{i+1}\mathcal{G}(r_{i+1})-r_i\mathcal{G}(r_i)}{r_{i+1}-r_i}-c_1(r_{i+1}+r_i). \tag{21}$$

进一步，不难直接验证知该下界函数在两个端点 $r_i,r_{i+1}$ 处是紧的：

$$\underline{\mathcal{G}}(r_{i+1})=\mathcal{G}(r_{i+1}),\underline{\mathcal{G}}(r_i)=\mathcal{G}(r_i). \tag{22}$$

那么，在区间 $[r_i,r_{i+1}]$ 上，$\underline{\mathcal{G}}(r)$ 的最小值给出了(PM1) 的下界：

$$\min_{\alpha\in[r_i,r_{i+1}]}\underline{\mathcal{G}}(r)=\begin{cases}2\sqrt{c_1c_2}+c_3, & c_1>0,c_2>0,r_i<\dfrac{\sqrt{c_2}}{\sqrt{c_1}}<r_{i+1},\\ \min\{\underline{\mathcal{G}}(r_{i+1}),\underline{\mathcal{G}}(r_i)\},\text{其他}.\end{cases}$$

结合(22)式，容易证明下面的定理.

**定理1** 令 $c_1,c_2$ 和 $c_3$ 分别如(19)～(21)式所定义.如果,

$$c_1>0,c_2>0,\bar{r}:=\sqrt{\frac{c_2}{c_1}}\in(r_i,r_{i+1}),\tag{23}$$

那么 $\min\limits_{r\in[r_i,r_{i+1}]}\mathscr{G}(r)\geq\min\limits_{r\in[r_i,r_{i+1}]}\underline{\mathscr{G}}(r)=2\sqrt{c_1c_2}+c_3$,且 $\bar{r}$ 是 $\underline{\mathscr{G}}(r)$ 在区间 $[r_i,r_{i+1}]$ 上的唯一极小点.否则, $\min\limits_{r\in[r_i,r_{i+1}]}\mathscr{G}(r)=\min\{\mathscr{G}(r_i),\mathscr{G}(r_{i+1})\}$.

## 2 算法及其复杂度

本节用分支定界算法求解(PM)并分析其收敛性以及复杂度.采用的是 $\omega$ 分支方法,即选取下界函数 $\underline{\mathscr{G}}(r)$ 的极小点 $\bar{r}$ (23)式作为剖分点来剖分当前区间 $[r_i,r_{i+1}]$.本文的终止准则是寻找到一个全局近似解,定义如下.

**定义1** 可行解 $r^*$ 称为是(PM)的 $\varepsilon$-近似全局最优解,如果它满足:

$$\mathscr{G}(r^*)\leq \mathrm{OPT}(PM)+\varepsilon,\tag{24}$$

其中 $\varepsilon>0$ 且 OPT(PM)是(PM)的全局最小值.

本文的具体算法如下:

分支定界算法

步骤1 初始化:$\varepsilon>0,r_1=r_{\min},r_2=r_{\max}$.

步骤2 通过求解线性规划计算 $\mathscr{G}(r_i)(i=1,2)$ 及对应 $a_2^{\mathrm{T}}x+b_2=r$ 的拉格朗日乘子 $\lambda_i(i=1,2)$.

令 $UB=\mathscr{G}(r_1),r^*=r_1$ 和 $T=\varnothing$.若 $\mathscr{G}(r_2)<UB$,更新 $UB=\mathscr{G}(r_2)$ 并置 $r^*=r_2$.

令 $k=2$(计算函数值的次数(即迭代次数)).

步骤3 若在区间 $[r_i,r_{i+1}]=:[r_1,r_2]$ 上(23)式不成立,则执行步骤7.

否则,选取 $\bar{r}$(23)式,并计算 $\mathscr{G}(\bar{r})$ 和对应 $a_2^{\mathrm{T}}x+b_2=r$ 的拉格朗日乘子 $\tilde{\lambda}$.更新 $k:=k+1$.

如果 $\mathscr{G}(\bar{r})<UB$,更新 $UB=\mathscr{G}(\bar{r})$ 并置 $r^*=\bar{r}$.

步骤4 利用定理1分别计算在区间 $[r_i,r_{i+1}]=:[r_1,\bar{r}]$ 与 $[r_i,r_{i+1}]=:[\bar{r},r_2]$ 上的下界 $LB_1,LB_2$.

如果 $LB_1<UB-\varepsilon$,更新 $T:=T\cup\{(LB_1,r_1,\bar{r})\}$.如果 $LB_2<UB-\varepsilon$,更新 $T:=T\cup\{(LB_2,\bar{r},r_2)\}$.

步骤5 如果 $T=\varnothing$,执行步骤6.否则,计算 $(LB^*,r_1,r_2):=\arg\min\limits_{(t,*,*)\in T}t$.

如果 $LB^*\geq UB-\varepsilon$,执行步骤6,否则,更新 $T:=T\backslash\{(LB^*,r_1,r_2)\}$ 并返回步骤3.

步骤6 输出 $r^*$:(PM)的 $\varepsilon$-近似全局最优解.

步骤7 输出 $r^*$:(PM)的全局最优解.

注意到(P)的定义中,$D$ 是有界的.从而对任给 $r\in[r_{\min},r_{\max}]$,线性规划 $\mathscr{G}(r)$ 与其对偶问题都有解,对应 $a_2^{\mathrm{T}}x+b_2=r$ 的拉格朗日乘子也是对偶问题的最优解的分量(当对偶最优解集无穷多个取值时选取最小的),设 $U$ 为在区间 $\in[r_{\min},r_{\max}]$ 上 $\mathscr{G}(r)$ 拉格朗日乘子的绝对值上界.

**定理2** 任给 $\varepsilon>0$,本文提出的新算法最多迭代

$$\left[\frac{4Ur_{\max}{}^2(r_{\max}-r_{\min})}{r_{\min}{}^2\varepsilon}\right]\tag{25}$$

步即可找到问题(PM)的 $\varepsilon$-近似全局最优解.

**证明** 假设 $(LB,r_i,r_{i+1})\in T$ 是选取的当前迭代点,则 $LB=LB^*$.可以假定(23)式在区间 $[r_i,r_{i+1}]$ 上成立.否则,从定理1中可知 $LB=UB$,算法终止得到问题的全局最优解.

若(23)式成立,表明下界函数 $\underline{\mathscr{G}}(r)$ 是一个凸函数.因此,对于任意 $r\in[r_i,r_{i+1}]$,有

$$\underline{\mathscr{G}}(r)\geq\underline{\mathscr{G}}(r_i)+\underline{\mathscr{G}}'(r_i)(r-r_i)=\mathscr{G}(r_i)+\left(c_1-\frac{c_2}{r_i^2}\right)(r-r_i)\geq\tag{26}$$

$$\mathscr{G}(r_i)+\left(c_1-\frac{c_1r_{i+1}^2}{r_i^2}\right)(r-r_i)\geq\tag{27}$$

$$\mathscr{G}(r_i)-\frac{r_{i+1}^2-r_i{}^2}{r_i{}^2}c_1(r_{i+1}-r_i),\tag{28}$$

其中，第一个不等式由于凸函数的性质，等式(26)式由(22)式可得，不等式(27)由(23)式的第 3 个不等式得到.此外，由 $c_1$ 和 $U$ 的定义，有

$$c_1(r_{i+1}-r_i)=r_{i+1}\lambda(r_{i+1})-r_i\lambda(r_i)\leq(r_{i+1}+r_i)U. \tag{29}$$

把(29)式代入(28)式有 $LB^*=\min\limits_{r\in[r_i,r_{i+1}]}\mathcal{G}(r)\geq\mathcal{G}(r_i)-\dfrac{(r_{i+1}+r_i)^2(r_{i+1}-r_i)}{r_i{}^2}U\geq UB-\dfrac{4r_{\max}^2(r_{i+1}-r_i)}{r_{\min}^2}U.$ 因此，终止准则 $LB^*>UB-\varepsilon$ 必然满足的一个充分条件是剖分区间足够小，具体而言是 $r_{i+1}-r_i<\dfrac{r_{\min}^2}{4Ur_{\max}^2}\cdot\varepsilon$. 注意到整个区间长度为 $r_{\max}-r_{\min}$，从而得到证明.

# 3　数值算例

为了验证新算法的可行性与高效性，通过随机生成的算例进行测试.除了 $\tilde{d}$ 设成分量全是 2 的固定的向量，所有问题(P)的数据都是在区间 $[-\delta,\delta]$ 上(参数 $\delta=1$ 或 10)通过均匀分布产生.

将本文的全局优化算法与软件包进行对比.如前言末尾部分所述，针对一般的问题(P)，对比对象为调用商业软件包 BARON[28] 求解等价问题(QP).针对 $a_2=a_3$ 的特殊应用问题[29]，对比对象为基于 CVX 平台[30] 调用 SDPT3 软件包求解等价问题($\mathrm{SOCP_{LP}}$)，注意此时该非凸优化问题等价于这一凸规划问题.

数值实验是用 MATLAB R2014a 在配置为 2.6 G 的双核处理器和 32 G 内存的服务器上运行的.在表 1 中，当 $\delta=1$ 且 $n\geq500$ 时，利用 SDPT3 求解时，计算结果为"NaN"，报错原因为"Steps are too short"，该错误在换用配置为 2.5 G 的单核处理器和 8 G 内存的笔记本上运行时并不发生，此种情况下切换为后者的计算环境.同样地，当 $\delta=10$ 且 $n\geq100$ 时，亦因同样原因作如此切换.本文的算法以及 BARON 精度均设置为 $\varepsilon=1\mathrm{e}\text{-}6$.因为与 SDPT3 的终止准则不一致，为公平起见，调整 SDPT3 的精度使其给出的解与本文方法给出的解具有同样的精度.

在所有的数据实验中，都是随机独立地产生 10 个算例分别进行计算，在表 1、表 2 中汇报 10 次计算的平均结果.其中"time"指运行时间的平均值，"iter."指迭代次数的平均值，当运行时间超过 3 600 s 时，用"—"表示.

**表 1　新算法与 SDPT3 求解特殊问题(P)( $\delta=1,10$ )的对比结果**

| $\delta$ | 1 | | | | 10 | | | |
|---|---|---|---|---|---|---|---|---|
| $n$ | SDPT3 | | Ours | | SDPT3 | | Ours | |
| | time | iter. | time | iter. | time | iter. | time | iter. |
| 50 | 0.39 | 29.3 | 0.01 | 17.7 | 0.63 | 37.6 | 0.01 | 9.3 |
| 100 | 0.48 | 34.1 | 0.03 | 17.6 | 0.62 | 45.5 | 0.02 | 14.6 |
| 200 | 0.89 | 40.3 | 0.10 | 17.9 | 1.10 | 50.9 | 0.12 | 21.3 |
| 300 | 1.82 | 43.4 | 0.34 | 23.6 | 2.21 | 53.4 | 0.32 | 22.7 |
| 500 | 9.29 | 49.9 | 2.07 | 26.1 | 9.01 | 57.1 | 1.23 | 20.0 |
| 1 000 | 65.57 | 56.2 | 17.14 | 27.5 | 68.72 | 61.6 | 13.52 | 21.6 |
| 1 500 | 267.16 | 55.9 | 69.97 | 28.7 | 262.56 | 70.8 | 66.01 | 25.3 |
| 2 000 | 476.14 | 58.7 | 181.40 | 29.3 | 551.15 | 64.8 | 191.97 | 27.3 |
| 2 500 | 712.74 | 45.0 | 372.15 | 28.8 | 833.81 | 50.6 | 389.69 | 27.6 |
| 3 000 | 1 316.13 | 46.1 | 734.58 | 29.7 | 1 598.06 | 47.5 | 754.34 | 27.5 |
| 3 500 | 2 605.54 | 46.7 | 1 202.59 | 31.6 | 3 149.32 | 54.7 | 1 411.54 | 30.8 |
| 4 000 | — | — | 2 065.11 | 32.0 | — | — | 2 322.86 | 33.3 |

对一般问题(P)，当 $\delta=1$ 时，从表 2 中可以看出，新算法的运行时间要远远少于商业软件包 BARON.随着维度的增加，BARON 的计算时间迅速增长，本文算法时间的增幅则比较缓慢.当 $n=1\ 200$ 时，BARON 的运行时间已经超过 3 600 s，而本文的算法运行时间依然还不足 30 s.即使到 $n=8\ 000$，本文算法运行时间也不足 3 600 s.类似地，当 $\delta=10$ 时，从表 2 中可以观察到同样的对比结果，此外，$\delta=10$ 时生成的例子略难于

$\delta=1$ 时的算例.

**表 2 新算法与 BARON 求解一般问题(P)( $\delta=1,10$ )的对比结果**

| $\delta$ | 1 | | | | 10 | | | |
|---|---|---|---|---|---|---|---|---|
| $n$ | BARON | | Ours | | BARON | | Ours | |
| | time | iter. | time | iter. | time | iter. | time | iter. |
| 50 | 0.49 | 1.2 | 0.03 | 21.6 | 0.55 | 1.8 | 0.03 | 24.6 |
| 100 | 1.60 | 1.4 | 0.05 | 24.0 | 1.63 | 1.4 | 0.05 | 25.5 |
| 200 | 9.19 | 2.2 | 0.15 | 29.1 | 9.31 | 1.8 | 0.13 | 28.4 |
| 300 | 19.86 | 1.6 | 0.38 | 28.6 | 27.00 | 1.8 | 0.38 | 30.2 |
| 400 | 60.58 | 2.0 | 0.89 | 31.6 | 63.25 | 1.0 | 0.78 | 29.4 |
| 500 | 107.76 | 1.8 | 1.78 | 32.6 | 187.00 | 2.8 | 1.61 | 30.3 |
| 600 | 215.83 | 1.2 | 3.18 | 33.8 | 202.87 | 1.2 | 3.11 | 33.8 |
| 700 | 460.35 | 1.8 | 5.05 | 33.7 | 412.63 | 1.1 | 5.23 | 34.7 |
| 800 | 637.42 | 1.4 | 7.70 | 33.3 | 616.24 | 1.1 | 6.76 | 30.1 |
| 900 | 1 050.73 | 2.0 | 11.11 | 32.4 | — | — | 11.57 | 34.3 |
| 1 000 | 1 423.45 | 1.2 | 15.57 | 33.5 | — | — | 15.06 | 32.2 |
| 1 100 | 2 009.90 | 1.7 | 21.34 | 33.7 | — | — | 20.64 | 33.0 |
| 1 200 | — | — | 29.84 | 35.0 | — | — | 29.09 | 35.8 |
| 5 000 | — | — | 663.46 | 39.9 | — | — | 766.89 | 43.1 |
| 8 000 | — | — | 3 341.81 | 44.4 | — | — | 3 072.22 | 41.6 |
| 10 000 | — | — | 7 539.28 | 47.0 | — | — | 6 302.52 | 44.6 |

对特殊问题 $a_2=a_3$,由于其等价于二阶锥规划,基于 CVX 平台调用软件包 SDPT3 求解($\mathrm{SOCP_{LP}}$)来与本文的全局优化算法做对比.数值结果如表 1 所示.当 $\delta=1$ 时,从表 1 中可以看出,本文算法的运行时间也远远少于 SDPT3.特别地,当 $n=50$ 时,SDPT3 的运行时间是本文的 39 倍.$\delta=10$ 的例子也是类似.本文算法的迭代次数都不足 35 次.从运行时间来看,本文算法明显优于 SDPT3.注意到从最坏复杂度来看,本文算法并不是多项式时间算法的,而 SDPT3 却是.

## 4 结 论

线性两比式和是非凸 NP-困难的优化问题,本文基于一个巧妙的估界技术提出新的全局优化算法.与商业软件包 BARON 数值对比,本文的算法具有极大的优势.对一个特殊结构的线性两比式和问题(具体而言等价于一个二阶锥规划问题),本文针对原问题设计的全局优化算法比基于 CVX 平台调用 SDPT3 来求解其等价凸规划问题还要更有效.本文的算法方法有望被推广到求解更广泛的非凸优化问题.

## 参 考 文 献


[1] Majhi J,Janardan R,Schwerdt J,et al.Minimizing support structures and trapped area in two-dimensional layered manufacturing[J].Computational Geometry,1999,12(3/4):241-267.

[2] Majhi J,Janardan R,Smid M,et al.On some geometric optimization problems in layered manufacturing[C]//Proc Fifth Workshop on Algorithms and Data Structures.Halifax:Springer,1997:136-149.

[3] Arkin E M,Chinang Y J,Held M,et al.On minimum-area hulls[J].Algorithmica,1998,21:119-136.

[4] Daniels K.The restrict/evaluate/subdivide paradigm for translational containment[C].Proc Fifth MSI Stony Bronk Workshop on Computational Geometry,New York,1995.

[5] Milenkovic V,Daniels K,Li Z.Placement and compaction of nonconvex polygons for clothing manufacture[C]//Pro Fourth Canad Conf on Comput Geom.Newfoundland:The CCCG Library,1992:236-243.

[6] Falk J E,Palocsay S W.Optimizing the sum of linear fractional functions[M].Collection:Recent Advances in Global Optimization,1992:221-258.

[7] Schaible S.Fractional programming[M].Collection:Handbook of Global Optimization,1995:495-608.

[8] Wang Y J,Shen P P,Liang Z A.A branch-and-bound algorithm to globally solve the sum of several linear ratios[J].Appl Math Comput,2005,168:89-101.

[9] Shen P P,Wang C F.Global optimization for sum of linear ratios problem with coefficients[J].Appl Math Comput,2006,176:219-229.

[10] Jin L,Wang R,Shen P P.Global optimization for the sum of linear ratios problem over convex feasible region[C].International Conference on Advances in Swarm Intelligence,Shenzhen,2012.

[11] Kuno T.A revision of the trapezoidal branch-and-bound algorithm for linear sum-of-ratios problems[J].Journal of Global Optimization,2005,33(2):215-234.

[12] Shen P P,Li W M,Liang Y C.Branch-reduction-bound algorithm for linear sum-of-ratios fractional programs[J].Pacific Journal of Optimization,2015,11(1):79-99.

[13] Wang C F,Shen P P.A deterministic global optimization algorithm for sum of linear ratios problem[J].Journal of Inner Mongolia Normal University,2013,3:785-790.

[14] Jiao H W,Feng Q G,Shen P P,et al.Global optimization for sum of linear ratios problem using new pruning technique[J].Mathematical Problems in Engineering,2008,4:267-290.

[15] Depetrini D,Locatelli M.Approximation of linear fractional-multiplicative problems[J].Math Program,2011,128:437-443.

[16] Xia Y,Wang L F,Wang S.Minimizing the sum of linear fractional functions over the cone of positive semidefinite matrices: approximation and applications[J].Operations Research Letters,2017.DOI:10.1016/j.orl.2017.11.010.

[17] Dur M,Horst R,Thoai N V.Solving sum-of-ratios fractional programs using efficient points[J].Optimization,2001,49:447-466.

[18] Carosi L,Martein L.A sequential method for a class of pseudoconcave fractional problems[J].Cent Eur J Oper Res,2008,16(2):153-164.

[19] Charnes A,Cooper W W.Programming with linear fractional functionals[J].Naval Research Logistics Quarterly,1962,9:181-186.

[20] Cameron A C,Trivedi P K.Microeconometrics: methods and applications[J].Chian Machine Press,2005,116:F161-F162.

[21] Konno H,Inori M.Bond portfolio optimization by bilinear fractional programming[J].J Oper Res Sot of Japan,1988,32(2):143-158.

[22] Konno H,Yajima Y,Matsui T.Parametric simplex algorithms for solving a special class of nonconvex minimization problems[J].J Glob Optim,1991,1:65-81.

[23] Schaible S.A note on the sum of a linear and linear fractional function[J].Naval Research Logistics Quarterly,1977,24:691-693.

[24] Gao L,Mishra S K.An extension of branch-and-bound algorithm for solving sum-of-nonlinear-ratios problem[J].Optim Lett,2012,6:221-230.

[25] Matsui T.NP-hardness of linear multiplicative programming and related problems[J].J Global Optim,1996,9:113-119.

[26] Cambini A,Martein L,Schaible S.On maximizing a sum of ratios[J].J Inform Optim Sci,1989,10:65-79.

[27] Konno H,Yamashita H.Minimizing sums and products of linear fractional functions over a polytope[J].Naval Res Logist,1999,46:583-596.

[28] Sahinidis N.BARON user manual v. 17.8.9[EB/OL].[2017-12-10].http://www.minlp.com.

[29] Fakhri A,Ghatee M.Minimizing the sum of a linear and a linear fractional function applying conic quadratic representation: continuous and discrete problems[J].Optimization,2016,65(5):1023-1038.

[30] Grant M,Boyd S.CVX:Matlab software for disciplined convex programming[EB/OL].[2017-12-10].http://cvxr.com/cvx.

# A new global optimization algorithm for minimizing the sum of two linear ratios

Xia Yong, Wang Longfei

(School of Mathematics and Systems Science,Beihang University,Beijing 100191,China)

**Abstract**:We propose a new global optimization algorithm for minimizing the sum of two linear ratios over a polytope (P),which is NP-hard.We first reformulate (P) as a univariate minimization problem.Based on a newly developed lower bounding approach, we propose an efficient branch-and-bound algorithm, which can find a global $\varepsilon$-approximation solution in at most $O(1/\varepsilon)$ iterations.Numerical results show that our new algorithm highly outperforms the software BARON.Moreover, for a special case of (P) which has a second-order cone programming reformulation (SOCP), our branch-and-bound algorithm even works much faster than calling SDPT3 for solving (SOCP).

**Keywords**:sum-of-linear-ratios; global optimization; branch and bound; duality; second-order cone programming

[责任编校　陈留院]