

文章编号:1000-0887(2011)04-0497-12

二阶锥规划两个新的预估-校正算法*

曾友芳[1,2], 白延琴[1], 简金宝[2], 唐春明[2]

(1. 上海大学 数学系, 上海 200444;
2. 广西大学 数学与信息科学学院, 南宁 530004)

摘要: 基于不可行内点法和预估-校正算法的思想,提出两个新的求解二阶锥规划的内点预估-校正算法.其预估方向分别是 Newton 方向和 Euler 方向,校正方向属于 Alizadeh-Haeberly-Overton (AHO)方向的范畴.算法对于迭代点可行或不可行的情形都适用.主要构造了一个更简单的中心路径的邻域,这是有别于其它内点预估-校正算法的关键.在一些假设条件下,算法具有全局收敛性、线性和二次收敛速度,并获得了 $O(r\ln(\varepsilon_0/\varepsilon))$ 的迭代复杂性界,其中 r 表示二阶锥规划问题所包含的二阶锥约束的个数.数值实验结果表明提出的两个算法是有效的.

关 键 词: 二阶锥规划; 不可行内点算法; 预估-校正算法; 全局收敛性; 复杂性分析

中图分类号: O221.2 **文献标志码**: A

DOI: 10.3879/j.issn.1000-0887.2011.04.012

引 言

考虑如下二阶锥规划问题[1]:

$$
(\mathrm{P}_1)\quad \begin{cases} \min & \displaystyle\sum_{i=1}^{r} \boldsymbol{c}_i^{\mathrm{T}}\boldsymbol{x}_i, \\ \text{s. t.} & \displaystyle\sum_{i=1}^{r} \boldsymbol{A}_i\boldsymbol{x}_i = \boldsymbol{b}, \\ & \boldsymbol{x}_i \succcurlyeq_{\boldsymbol{K}^{n_i}} \boldsymbol{0}, \qquad i \in \boldsymbol{I} = \{1,2,\cdots,r\}, \end{cases}
$$

其中, $\boldsymbol{b} \in R^m$, $\boldsymbol{c}_i \in R^{n_i}$ 且 $\boldsymbol{A}_i \in R^{m\times n_i}$. $\boldsymbol{x}_i \succcurlyeq_{\boldsymbol{K}^{n_i}} \boldsymbol{0}$ 表示 $\boldsymbol{x}_i \in \boldsymbol{K}^{n_i}$,

$$
\boldsymbol{K}^{n_i} = \{\boldsymbol{x}_i = (x_{i0}, x_{i1}, \cdots, x_{i(n_i-1)})^{\mathrm{T}} = (x_{i0}; \bar{\boldsymbol{x}}_i) \in R^{n_i} \mid x_{i0} \geqslant \|\bar{\boldsymbol{x}}_i\|\} \qquad (i \in \boldsymbol{I})
$$

是一个二阶锥, 其中 $\|\cdot\|$ 表示 2-模.

二阶锥规划因其具有广泛的应用性和计算的易处理性而使人们对它展开了独立研究.很早以前人们就开始对二阶锥规划的一些特例进行研究,比如 Fermat-Weber 问题[2].二阶锥规划的进一步应用可参考文献[3].近 10 年来,二阶锥规划的应用日益广泛,比如在力学方面的应用[4-5].线性规划(LP)可以转化为二阶锥规划问题,而二阶锥规划又是半定规划的特例.所以二阶锥规划介于线性规划和半定规划之间.与线性规划和半定规划一样,二阶锥规划问题可

* **收稿日期**: 2010-12-20; **修订日期**: 2011-03-03

基金项目: 国家自然科学基金资助项目(71061002;11071158);广西自然科学基金资助项目(0832052;2010GXNSFB013047)

作者简介: 曾友芳(1974—), 女,广西靖西人,讲师,博士生(联系人. E-mail: zengyf@gxu.edu.cn).

以用内点法在多项式时间内求解. Nemirovskii 和 Scheinberg[6] 证明了线性规划的原始或对偶内点法可以逐字逐句地推广到二阶锥规划的情形. 随后, Nesterov 和 Todd (NT)[7-8] 开始研究二阶锥规划的原始-对偶内点算法. 他们建立了一个特殊的原始-对偶方法,称为 NT 方法并提出了 NT 方向. 进而,他们证明了基于 NT 方向的短步路径跟踪算法具有 $O(\sqrt{r}\ln\varepsilon^{-1})$ 迭代复杂性界, 其中 r 表示问题(P_1)中二阶锥的个数. Adler 和 Alizadeh[9] 研究了求解半定规划和二阶锥规划统一的原始-对偶方法,提出了适用于二阶锥规划的搜索方向. Alizadeh, Haeberly 和 Overton[10] 阐述的关于半定规划的 AHO 方向与这个方向类似.

此后出现了一些使用 AHO 方向的预估-校正算法[11-12]. Monteiro 和 Tsuchiya[11] 介绍了确定二阶锥规划 AHO 搜索方向 $(\Delta\boldsymbol{x},\Delta\boldsymbol{y},\Delta\boldsymbol{s})$ 的 Newton 方程组:

$$\begin{cases}\boldsymbol{A}\Delta\boldsymbol{x}=\boldsymbol{b}-\boldsymbol{A}\boldsymbol{x},\\ \boldsymbol{A}^{\mathrm{T}}\Delta\boldsymbol{y}+\Delta\boldsymbol{s}=\boldsymbol{c}-\boldsymbol{A}^{\mathrm{T}}\boldsymbol{y}-\boldsymbol{s},\\ \boldsymbol{S}\Delta\boldsymbol{x}+\boldsymbol{X}\Delta\boldsymbol{s}=\sigma\boldsymbol{\mu}\boldsymbol{e}-\boldsymbol{X}\boldsymbol{s},\end{cases}\tag{1}$$

其中, $\sigma\in\mathbf{R}$, $\boldsymbol{X},\boldsymbol{S},\boldsymbol{e}$ 的定义见预备知识部分. 通过推广 Mizuno, Todd 和 Ye[13] 关于线性规划的预估-校正算法,他们给出了沿 AHO 方向的二阶锥规划预估-校正算法并分析了算法的多项式收敛性. 令 $\sigma=0$, $(\boldsymbol{x},\boldsymbol{y},\boldsymbol{s})=(\boldsymbol{x}^k,\boldsymbol{y}^k,\boldsymbol{s}^k)$, 计算方程组(1), 解 $(\Delta\boldsymbol{x}^{\mathrm{p}},\Delta\boldsymbol{y}^{\mathrm{p}},\Delta\boldsymbol{s}^{\mathrm{p}})$ 表示预估方向. 然后令 $\sigma=1$, $(\boldsymbol{x},\boldsymbol{y},\boldsymbol{s})=(\boldsymbol{x}^k,\boldsymbol{y}^k,\boldsymbol{s}^k)+\alpha_k(\Delta\boldsymbol{x}^{\mathrm{p}},\Delta\boldsymbol{y}^{\mathrm{p}},\Delta\boldsymbol{s}^{\mathrm{p}})$, α_k 是沿着预估方向的步长. 再次求解方程组(1),解 $(\Delta\boldsymbol{x}^{\mathrm{c}},\Delta\boldsymbol{y}^{\mathrm{c}},\Delta\boldsymbol{s}^{\mathrm{c}})$ 表示校正方向. 在 α_k 仅与 τ,r 有关的条件下,算法的迭代复杂性界是 $O(\sqrt{r}\ln\varepsilon^{-1})$. 但算法要求初始点严格可行. Chi 和 Liu[12] 将其推广到初始点、迭代点和近似最优解都是不可行的情形. 文献[11-12]中心路径的邻域都是

$$N(\gamma,\tau)=\{(\boldsymbol{x},\boldsymbol{y},\boldsymbol{s})\in\boldsymbol{K}^0\times R^m\times\in\boldsymbol{K}^0\mid d_2(\boldsymbol{x},\boldsymbol{s})=\sqrt{2}\,\|\boldsymbol{w}_{\boldsymbol{xs}}-\tau\boldsymbol{e}\|\leqslant\gamma\tau\},\tag{2}$$

其中,参数 $\gamma\in(0,1)$, $\tau>0$, $\boldsymbol{K}^0=\{\boldsymbol{x}\in\boldsymbol{K}\mid x_{i0}>\|\bar{\boldsymbol{x}}_i\|,i\in\boldsymbol{I}\}$ 表示二阶锥 $\boldsymbol{K}$ 的内部, $\boldsymbol{w}_{\boldsymbol{xs}}=\boldsymbol{T}_{\boldsymbol{x}}\boldsymbol{s}$, $\boldsymbol{T}_{\boldsymbol{x}}=\operatorname{diag}(\boldsymbol{T}_{\boldsymbol{x}_1},\boldsymbol{T}_{\boldsymbol{x}_2},\cdots,\boldsymbol{T}_{\boldsymbol{x}_r})$,

$$\boldsymbol{T}_{\boldsymbol{x}_i}=\begin{pmatrix}x_{i0} & \bar{\boldsymbol{x}}_i^{\mathrm{T}}\\ \bar{\boldsymbol{x}}_i & \beta_{\boldsymbol{x}_i}\boldsymbol{E}+\dfrac{\bar{\boldsymbol{x}}_i\bar{\boldsymbol{x}}_i^{\mathrm{T}}}{\beta_{\boldsymbol{x}_i}+x_{i0}}\end{pmatrix},\quad \beta_{\boldsymbol{x}_i}=\sqrt{x_{i0}^2-\|\bar{\boldsymbol{x}}_i\|^2}.$$

相关记号的定义可查阅预备知识部分.

因为这个邻域较复杂,与最优性条件的对应关系不明显,为此考虑构造一个新的更简单的邻域. 注意到 Miao[14] 提出的关于线性规划的两个不可行内点预估-校正算法,预估方向分别采用了 Newton 方向和 Euler 方向. 每次迭代算法需求解两个线性方程组, 算法是全局收敛的且有 $O(n\ln(1/\varepsilon))$ 迭代复杂性界,其中 n 表示线性规划问题中自变量的维数. 而且,预估方向是 Newton 方向的算法在最优解存在的假设下还具有 Q-二次收敛性.

因此, 基于文献[14]中预估-校正方向的思想和邻域,提出两个新的求解二阶锥规划的(不)可行内点预估-校正算法. 预估方向分别采用 Newton 方向和 Euler 方向,而校正方向类似 AHO 方向. 在一些假设条件下,算法具有全局收敛性、Q-(超)线性收敛速度、Q-二次收敛速度且有 $O(r\ln(\varepsilon_0/\varepsilon))$ 复杂性界. 数值实验结果表明算法是有效的.

文中结构如下: 第 1 节是预备知识;第 2 节将给出一个新的不可行内点预估-校正算法;第 3 节和第 4 节分别讨论算法的收敛性和迭代复杂性界;第 5 节给出另外一个新的不可行内点预估-校正算法并简单讨论了它的性质;最后在第 6 节给出两个算法的数值实验结果.

1 预 备 知 识

为便于说明，使用如下记号：

$$\begin{pmatrix} \boldsymbol{x} \\ \boldsymbol{y} \end{pmatrix} = (\boldsymbol{x}^{\mathrm{T}}, \boldsymbol{y}^{\mathrm{T}})^{\mathrm{T}} = (\boldsymbol{x}, \boldsymbol{y}),\ n \stackrel{\Delta}{=} \sum_{i=1}^{r} n_i,$$

$$\boldsymbol{c} = (\boldsymbol{c}_1, \boldsymbol{c}_2, \cdots, \boldsymbol{c}_r) \in R^n,\ \boldsymbol{x} = (\boldsymbol{x}_1, \boldsymbol{x}_2, \cdots, \boldsymbol{x}_r) \in \boldsymbol{K},$$

$$\boldsymbol{K} = \boldsymbol{K}^{n_1} \times \boldsymbol{K}^{n_2} \times \cdots \times \boldsymbol{K}^{n_r},$$

$$\boldsymbol{A} = (\boldsymbol{A}_1, \boldsymbol{A}_2, \cdots, \boldsymbol{A}_r) = (\boldsymbol{a}_1, \cdots, \boldsymbol{a}_l, \cdots, \boldsymbol{a}_m)^{\mathrm{T}} \in R^{m \times n},$$

其中 $\boldsymbol{a}_l \in R^n$ 表示矩阵 $\boldsymbol{A}$ 的第 l 行，$l = 1, 2, \cdots, m$. 文中总假设秩$(\boldsymbol{A}) = m$. 这样，问题(P_1)可表示成

$$(\mathrm{P})\ \min\{\boldsymbol{c}^{\mathrm{T}}\boldsymbol{x} \mid \boldsymbol{A}\boldsymbol{x} = \boldsymbol{b},\ \boldsymbol{x} \in \boldsymbol{K}\}.$$

其对偶问题是

$$(\mathrm{D})\ \max\{\boldsymbol{b}^{\mathrm{T}}\boldsymbol{y} \mid \boldsymbol{A}^{\mathrm{T}}\boldsymbol{y} + \boldsymbol{s} = \boldsymbol{c},\ \boldsymbol{s} \in \boldsymbol{K}, \boldsymbol{y} \in R^m\},$$

其中, $\boldsymbol{s} = (\boldsymbol{s}_1; \boldsymbol{s}_2; \cdots; \boldsymbol{s}_r)$，$\boldsymbol{s}_i \in \boldsymbol{K}^{n_i}$，$i \in \boldsymbol{I}$.

以下介绍一些基本知识，深入的学习可参考文献[1]. 在 Euclidean Jordan 代数中,对于任意两个向量 $\boldsymbol{x}_i, \boldsymbol{s}_i \in R^{n_i}$，定义

$$\boldsymbol{x}_i \circ \boldsymbol{s}_i \stackrel{\Delta}{=} \begin{pmatrix} \boldsymbol{x}_i^{\mathrm{T}}\boldsymbol{s}_i \\ \boldsymbol{x}_{i0}\bar{\boldsymbol{s}}_i + \boldsymbol{s}_{i0}\bar{\boldsymbol{x}}_i \end{pmatrix} = \mathrm{Arw}(\boldsymbol{x}_i)\boldsymbol{s}_i = \mathrm{Arw}(\boldsymbol{x}_i)\mathrm{Arw}(\boldsymbol{s}_i)\boldsymbol{e}_i,$$

其中矩阵

$$\boldsymbol{X}_i \stackrel{\Delta}{=} \mathrm{Arw}(\boldsymbol{x}_i) = \begin{pmatrix} x_{i0} & \bar{\boldsymbol{x}}_i^{\mathrm{T}} \\ \bar{\boldsymbol{x}}_i & x_{i0}\boldsymbol{E}_i \end{pmatrix},$$

$\boldsymbol{e}_i = (1; \boldsymbol{0}) \in R^{n_i}$ 且 $\boldsymbol{E}_i$ 是一个$(n_i - 1)$维单位矩阵. 那么

$$\boldsymbol{X} \stackrel{\Delta}{=} \mathrm{Arw}(\boldsymbol{x}) = \mathrm{diag}(\mathrm{Arw}(\boldsymbol{x}_1), \cdots, \mathrm{Arw}(\boldsymbol{x}_r)) = \mathrm{diag}(\boldsymbol{X}_i, i \in \boldsymbol{I}),$$

$$\boldsymbol{x} \circ \boldsymbol{s} = (\boldsymbol{x}_1 \circ \boldsymbol{s}_1; \cdots; \boldsymbol{x}_r \circ \boldsymbol{s}_r) = \boldsymbol{X}\boldsymbol{s} = \boldsymbol{S}\boldsymbol{x},\ \boldsymbol{e} = (\boldsymbol{e}_1; \cdots; \boldsymbol{e}_r).$$

由强对偶定理[1]可知如下的假设 1 是必需的,文中都用此假设.

假设 1 问题(P)和(D)都有严格可行解.

根据最优性条件[1]定义向量值函数 $\boldsymbol{F}: \boldsymbol{K} \times R^m \times \boldsymbol{K} \to R^{2n+m}$ 如下：

$$\boldsymbol{F}(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{s}) = \begin{pmatrix} \boldsymbol{A}\boldsymbol{x} - \boldsymbol{b} \\ \boldsymbol{A}^{\mathrm{T}}\boldsymbol{y} + \boldsymbol{s} - \boldsymbol{c} \\ \boldsymbol{x} \circ \boldsymbol{s} \end{pmatrix} = \begin{pmatrix} \boldsymbol{A}\boldsymbol{x} - \boldsymbol{b} \\ \boldsymbol{A}^{\mathrm{T}}\boldsymbol{y} + \boldsymbol{s} - \boldsymbol{c} \\ \boldsymbol{X}\boldsymbol{s} \end{pmatrix}. \tag{3}$$

显然，最优性条件等价于

$$\boldsymbol{F}(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{s}) = \boldsymbol{0}, \qquad \boldsymbol{x}, \boldsymbol{s} \in \boldsymbol{K}.$$

进而, $\boldsymbol{F}$ 关于$(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{s})$ 的导数是

$$\boldsymbol{F}'(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{s}) = \begin{pmatrix} \boldsymbol{A} & \boldsymbol{0} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{A}^{\mathrm{T}} & \boldsymbol{E} \\ \boldsymbol{S} & \boldsymbol{0} & \boldsymbol{X} \end{pmatrix}. \tag{4}$$

为便于陈述，引入记号：

$$\boldsymbol{r}^{\mathrm{p}} = \boldsymbol{b} - \boldsymbol{A}\boldsymbol{x},\ \boldsymbol{r}^{\mathrm{d}} = \boldsymbol{c} - \boldsymbol{A}^{\mathrm{T}}\boldsymbol{y} - \boldsymbol{s}$$

其中 $\boldsymbol{r}_k^{\mathrm{p}} = \boldsymbol{b} - \boldsymbol{A}\boldsymbol{x}^k$, $\boldsymbol{r}_k^{\mathrm{d}} = \boldsymbol{c} - \boldsymbol{A}^{\mathrm{T}}\boldsymbol{y}^k - \boldsymbol{s}^k$, $\quad k = 0,1,2,\cdots$

及如下集合：

$$\boldsymbol{S} = \{(\boldsymbol{x},\boldsymbol{y},\boldsymbol{s}) \in \boldsymbol{K} \times R^m \times \boldsymbol{K} \mid \boldsymbol{F}(\boldsymbol{x},\boldsymbol{y},\boldsymbol{s}) = \boldsymbol{0}\}, \tag{5}$$

$$\boldsymbol{S}_\varepsilon = \{(\boldsymbol{x},\boldsymbol{y},\boldsymbol{s}) \in \boldsymbol{K}^0 \times R^m \times \boldsymbol{K}^0 \mid \boldsymbol{x}^{\mathrm{T}}\boldsymbol{s} \leqslant \varepsilon,\ \|\boldsymbol{r}^{\mathrm{p}}\| \leqslant \varepsilon,\ \|\boldsymbol{r}^{\mathrm{d}}\| \leqslant \varepsilon\}, \tag{6}$$

$$\boldsymbol{C} = \{(\boldsymbol{x},\boldsymbol{y},\boldsymbol{s}) \in \boldsymbol{K}^0 \times R^m \times \boldsymbol{K}^0 \mid \boldsymbol{r}^{\mathrm{p}} = (\mu/\mu_0)\boldsymbol{r}_0^{\mathrm{p}},\ \boldsymbol{r}^{\mathrm{d}} = (\mu/\mu_0)\boldsymbol{r}_0^{\mathrm{d}}, \boldsymbol{x} \circ \boldsymbol{s} = \mu \boldsymbol{e}\}. \tag{7}$$

对于任意给定的 $\varepsilon > 0$, $\boldsymbol{S}_\varepsilon$ 中的点是问题(P)和(D)的 ε-近似解，称 $\boldsymbol{C}$ 是起始点为$(\boldsymbol{x}^0,\boldsymbol{y}^0,\boldsymbol{s}^0) \in \boldsymbol{K}^0 \times R^m \times \boldsymbol{K}^0$ 的(不)可行中心路径，其中

$$\mu_0 = \frac{(\boldsymbol{x}^0)^{\mathrm{T}}\boldsymbol{s}^0}{r} > 0,\ \mu = \frac{\boldsymbol{x}^{\mathrm{T}}\boldsymbol{s}}{r} > 0.$$

2 预估-校正算法 I

本节提出二阶锥规划的一个预估-校正算法，预估方向采用 Newton 方向，校正方向属于 AHO 方向的范畴. 设常数 $\alpha \in (0,1/4]$. 定义中心路径 $\boldsymbol{C}$ 的邻域如下：

$$N(\alpha,\mu) = \{(\boldsymbol{x},\boldsymbol{y},\boldsymbol{s}) \in \boldsymbol{K}^0 \times R^m \times \boldsymbol{K}^0 \mid \|\boldsymbol{X}\boldsymbol{s} - \mu\boldsymbol{e}\| \leqslant \alpha\mu\}. \tag{8}$$

算法 I

初始步 设 $\varepsilon > 0$, $\alpha \in (0,1/4]$, Γ 是一个与 α 有关的常数. 取$(\boldsymbol{x}^0,\boldsymbol{y}^0,\boldsymbol{s}^0) \in N(\alpha,\mu_0)$，其中，$\mu_0 = (\boldsymbol{x}^0)^{\mathrm{T}}\boldsymbol{s}^0/r$. 令 $k := 0$.

步骤 1 如果 $(\boldsymbol{x}^k,\boldsymbol{y}^k,\boldsymbol{s}^k) \in \boldsymbol{S}_\varepsilon$，停，$(\boldsymbol{x}^k,(\boldsymbol{y}^k,\boldsymbol{s}^k))$ 是(P)和(D)的最优解对；否则，转到步骤 2.

步骤 2(确定预估方向) 求解如下的 Newton 方程组得到预估方向 $(\Delta\boldsymbol{x}^{\mathrm{p}},\Delta\boldsymbol{y}^{\mathrm{p}},\Delta\boldsymbol{s}^{\mathrm{p}})$：

$$\boldsymbol{F}'(\boldsymbol{x}^k,\boldsymbol{y}^k,\boldsymbol{s}^k)\begin{pmatrix}\Delta\boldsymbol{x}^{\mathrm{p}}\\ \Delta\boldsymbol{y}^{\mathrm{p}}\\ \Delta\boldsymbol{s}^{\mathrm{p}}\end{pmatrix} = -\boldsymbol{F}(\boldsymbol{x}^k,\boldsymbol{y}^k,\boldsymbol{s}^k),$$

即

$$\begin{cases}\boldsymbol{A}\Delta\boldsymbol{x}^{\mathrm{p}} = \boldsymbol{r}_k^{\mathrm{p}},\\ \boldsymbol{A}^{\mathrm{T}}\Delta\boldsymbol{y}^{\mathrm{p}} + \Delta\boldsymbol{s}^{\mathrm{p}} = \boldsymbol{r}_k^{\mathrm{d}},\\ \boldsymbol{S}^k\Delta\boldsymbol{x}^{\mathrm{p}} + \boldsymbol{X}^k\Delta\boldsymbol{s}^{\mathrm{p}} = -\boldsymbol{X}^k\boldsymbol{s}^k.\end{cases} \tag{9}$$

步骤 3(确定步长) 令

$$(\boldsymbol{x}(\theta),\boldsymbol{y}(\theta),\boldsymbol{s}(\theta)) = (\boldsymbol{x}^k,\boldsymbol{y}^k,\boldsymbol{s}^k) + \theta(\Delta\boldsymbol{x}^{\mathrm{p}},\Delta\boldsymbol{y}^{\mathrm{p}},\Delta\boldsymbol{s}^{\mathrm{p}}),\qquad \theta \in [0,1]. \tag{10}$$

选取

$$\theta_k = \max\{\theta \in [0,1] \mid \|\boldsymbol{X}(\theta)\boldsymbol{s}(\theta) - (1-\theta)\mu_k\boldsymbol{e}\| \leqslant \Gamma\alpha(1-\theta)\mu_k,\ (\boldsymbol{x}(\theta),\boldsymbol{s}(\theta)) \in \boldsymbol{K}\times\boldsymbol{K}\}.$$

令

$$(\hat{\boldsymbol{x}}^k,\hat{\boldsymbol{y}}^k,\hat{\boldsymbol{s}}^k) = (\boldsymbol{x}^k,\boldsymbol{y}^k,\boldsymbol{s}^k) + \theta_k(\Delta\boldsymbol{x}^{\mathrm{p}},\ \Delta\boldsymbol{y}^{\mathrm{p}},\ \Delta\boldsymbol{s}^{\mathrm{p}}). \tag{11}$$

步骤 4(确定搜索方向) 如果 $\theta_k = 1$，停，$(\hat{\boldsymbol{x}}^k,(\hat{\boldsymbol{y}}^k,\hat{\boldsymbol{s}}^k))$ 是(P)和(D)的最优解对；否则，求解如下方程组得到校正方向 $(\Delta\boldsymbol{x}^{\mathrm{c}},\Delta\boldsymbol{y}^{\mathrm{c}},\Delta\boldsymbol{s}^{\mathrm{c}})$：

$$\begin{cases}\boldsymbol{A}\Delta\boldsymbol{x}^{\mathrm{c}} = 0,\\ \boldsymbol{A}^{\mathrm{T}}\Delta\boldsymbol{y}^{\mathrm{c}} + \Delta\boldsymbol{s}^{\mathrm{c}} = 0,\\ \hat{\boldsymbol{S}}^k\Delta\boldsymbol{x}^{\mathrm{c}} + \hat{\boldsymbol{X}}^k\Delta\boldsymbol{s}^{\mathrm{c}} = (1-\theta_k)\mu_k\boldsymbol{e} - \hat{\boldsymbol{X}}^k\hat{\boldsymbol{s}}^k.\end{cases} \tag{12}$$

步骤 5 取

$$\begin{cases}(\boldsymbol{x}^{k+1},\boldsymbol{y}^{k+1},\boldsymbol{s}^{k+1}) = (\hat{\boldsymbol{x}}^k,\hat{\boldsymbol{y}}^k,\hat{\boldsymbol{s}}^k) + (\Delta\boldsymbol{x}^{\mathrm{c}},\ \Delta\boldsymbol{y}^{\mathrm{c}},\Delta\boldsymbol{s}^{\mathrm{c}}),\\ \mu_{k+1} = (\boldsymbol{x}^{k+1})^{\mathrm{T}}\boldsymbol{s}^{k+1}/r.\end{cases} \tag{13}$$

令 $k := k+1$，转到步骤 1.

注 1 算法 Ⅰ 的预估方向和校正方向属于 AHO 方向[11]的范畴，但算法 Ⅰ 的邻域和步长 θ_k 与文献[11]中的预估-校正算法不同.

记

$$v_{k+1} = \prod_{j=0}^{k}(1-\theta_j),\qquad k=1,2,\cdots. \tag{14}$$

由算法 Ⅰ 有以下结论.

引理 1 设 $\{(\boldsymbol{x}^k,\boldsymbol{y}^k,\boldsymbol{s}^k)\}$ 是算法 Ⅰ 产生的迭代点列，则

$$\begin{cases}\boldsymbol{r}_{k+1}^{\mathrm{p}} = (1-\theta_k)\boldsymbol{r}_k^{\mathrm{p}} = v_{k+1}\boldsymbol{r}_0^{\mathrm{p}},\ \boldsymbol{r}_{k+1}^{\mathrm{d}} = (1-\theta_k)\boldsymbol{r}_k^{\mathrm{d}} = v_{k+1}\boldsymbol{r}_0^{\mathrm{d}},\\ \mu_{k+1} = (1-\theta_k)\mu_k = v_{k+1}\mu_0,\end{cases} \tag{15}$$

$$(\boldsymbol{x}^{k+1})^{\mathrm{T}}\boldsymbol{s}^{k+1} = (1-\theta_k)(\boldsymbol{x}^k)^{\mathrm{T}}\boldsymbol{s}^k = v_{k+1}(\boldsymbol{x}^0)^{\mathrm{T}}\boldsymbol{s}^0. \tag{16}$$

由引理 1，当 k 充分大时，迭代点列 $\{(\boldsymbol{x}^k,\boldsymbol{y}^k,\boldsymbol{s}^k)\}\in\boldsymbol{S}_\varepsilon$，于是算法可终止. 在随后的讨论中，总是假设 $\{(\boldsymbol{x}^k,\boldsymbol{y}^k,\boldsymbol{s}^k)\}$ 是无穷序列.

注 2 步骤 3 中步长 θ_k 的选取使得迭代点 $(\hat{\boldsymbol{x}}^k,\hat{\boldsymbol{y}}^k,\hat{\boldsymbol{s}}^k)\in\boldsymbol{N}(\Gamma\alpha,(1-\theta_k)\mu_k)$. 此处 Γ 与 α 的取值有关. 比如，当 $\alpha = 1/8$ 时，$\Gamma = 2$. 文中都用这一取值.

θ_k 的选取依据源自下面的引理 2 至引理 4.

引理 2 如果 $(\boldsymbol{x}^k,\boldsymbol{y}^k,\boldsymbol{s}^k)\in\boldsymbol{N}(1/8,\mu_k)$ 且存在一个 $\theta\in(0,1)$ 使得

$$\|\boldsymbol{X}(\theta)\boldsymbol{s}(\theta)-(1-\theta)\mu_k\boldsymbol{e}\|\leqslant(1-\theta)\mu_k/4, \tag{17}$$

那么

$$(\boldsymbol{x}(\theta),\boldsymbol{s}(\theta))\in\boldsymbol{K}^0\times\boldsymbol{K}^0.$$

引理 3 设 $\{(\boldsymbol{x}^k,\boldsymbol{y}^k,\boldsymbol{s}^k)\}$ 由算法 Ⅰ 产生，那么

$$\|\Delta\boldsymbol{X}^{\mathrm{c}}\Delta\boldsymbol{s}^{\mathrm{c}}\|\leqslant\frac{\sqrt{2}}{2}\|(\hat{\boldsymbol{X}}^k\hat{\boldsymbol{S}}^k)^{-1/2}(\hat{\boldsymbol{X}}^k\hat{\boldsymbol{s}}^k-\mu_{k+1}\boldsymbol{e})\|^2. \tag{18}$$

引理 4 设 $\{(\boldsymbol{x}^{k+1},\boldsymbol{y}^{k+1},\boldsymbol{s}^{k+1})\}$ 由算法 Ⅰ 产生，那么对于 $k\geqslant 0$：

(i) $(\hat{\boldsymbol{x}}^k,\hat{\boldsymbol{y}}^k,\hat{\boldsymbol{s}}^k)\in\boldsymbol{N}(1/4,(1-\theta_k)\mu_k)$；

(ii) $(\boldsymbol{x}^{k+1},\boldsymbol{y}^{k+1},\boldsymbol{s}^{k+1})\in\boldsymbol{N}(1/8,\mu_{k+1})$.

引理 1 至引理 4 的证明类似文献[14]中引理 1 至引理 4 的证明.

注 3 若 $(\boldsymbol{x}^k,\boldsymbol{y}^k,\boldsymbol{s}^k)$ 可行，则随后的迭代点都可行. 进而，若 $(\boldsymbol{x}^0,\boldsymbol{y}^0,\boldsymbol{s}^0)\in\boldsymbol{N}(\alpha,\mu_0)$，则算法 Ⅰ 是一个可行内点法.

3 收敛性分析

首先给出收敛性证明中需要的一些有关技巧的引理. 设

$$\boldsymbol{z}^k \triangleq (\boldsymbol{x}^k,\boldsymbol{y}^k,\boldsymbol{s}^k),\qquad k=0,1,2,\cdots,$$

$\{\boldsymbol{z}^k\}$ 是由算法 Ⅰ 产生的迭代点列.

引理 5

$$\theta_k \geqslant \delta_1 \equiv \min\left\{\frac{1}{2}, \sqrt{\frac{\alpha\mu_k}{2\|\Delta \boldsymbol{X}^{\mathrm{p}}\Delta \boldsymbol{s}^{\mathrm{p}}\|}}\right\}. \tag{19}$$

证明 对于 $\theta \in [0,1]$, $\boldsymbol{s}(\theta) = \boldsymbol{s}^k + \theta\,\Delta \boldsymbol{s}^{\mathrm{p}}$, $\boldsymbol{X}(\theta) = \boldsymbol{X}^k + \theta\Delta \boldsymbol{X}^{\mathrm{p}}$, 有

$$\begin{aligned}&\|\boldsymbol{X}(\theta)\boldsymbol{s}(\theta) - (1-\theta)\mu_k \boldsymbol{e}\| = \\&\quad \|\boldsymbol{X}^k\boldsymbol{s}^k + \theta(-\boldsymbol{X}^k\boldsymbol{s}^k) + \theta^2\,\Delta \boldsymbol{X}^{\mathrm{p}}\Delta \boldsymbol{s}^{\mathrm{p}} - (1-\theta)\,\mu_k\boldsymbol{e}\| \leqslant \\&\quad (1-\theta)\,\|\boldsymbol{X}^k\boldsymbol{s}^k - \mu_k\boldsymbol{e}\| + \theta^2\,\|\Delta \boldsymbol{X}^{\mathrm{p}}\Delta \boldsymbol{s}^{\mathrm{p}}\| \leqslant \\&\quad (1-\theta)\alpha\mu_k + \theta^2\,\|\Delta \boldsymbol{X}^{\mathrm{p}}\Delta \boldsymbol{s}^{\mathrm{p}}\|.\end{aligned}$$

对于所有的 $\theta \in [0,\delta_1]$, 有 $\theta \leqslant 1/2$ 和 $\theta^2\,\|\Delta \boldsymbol{X}^{\mathrm{p}}\Delta \boldsymbol{s}^{\mathrm{p}}\| \leqslant \alpha\mu_k/2 \leqslant (1-\theta)\alpha\mu_k$. 则上面的不等式说明

$$\|\boldsymbol{X}(\theta)\boldsymbol{s}(\theta) - (1-\theta)\mu_k\boldsymbol{e}\| \leqslant 2\alpha(1-\theta)\mu_k.$$

故由 θ_k 的定义可知 $\theta_k \geqslant \delta_1$. □

易证如下引理 6.

引理 6 对于任意的可行点 $(\boldsymbol{x},\boldsymbol{y},\boldsymbol{s})$ 和 $(\tilde{\boldsymbol{x}},\tilde{\boldsymbol{y}},\tilde{\boldsymbol{s}})$, 有 $(\boldsymbol{x}-\tilde{\boldsymbol{x}})^{\mathrm{T}}(\boldsymbol{s}-\tilde{\boldsymbol{s}}) = \boldsymbol{0}$.

下面的引理 7 与文献[15]中关于线性规划问题的引理 6.1 类似.

引理 7 序列 $\{\boldsymbol{z}^k\}$ 有界, 即存在一个常数 $\gamma_1 > 0$(与问题的数据和 r 有关)使得

$$\|\boldsymbol{z}^k\| \leqslant \gamma_1. \tag{20}$$

引理 8 序列 $\{\boldsymbol{z}^k\}$ 的任意聚点都属于解集 $\boldsymbol{S}$.

证明 由引理 7 知 $\{\boldsymbol{z}^k\}$ 有界, 故有收敛子列. 设 $\boldsymbol{z}^* = (\boldsymbol{x}^*, \boldsymbol{y}^*, \boldsymbol{s}^*)$ 是 $\{\boldsymbol{z}^k\}$ 的任一聚点. 因为 $(\boldsymbol{x}_i^k, \boldsymbol{s}_i^k) \in (\boldsymbol{K}^{n_i})^0 \times (\boldsymbol{K}^{n_i})^0 (i \in \boldsymbol{I})$, 故 $(\boldsymbol{x}_i^*, \boldsymbol{s}_i^*) \in \boldsymbol{K}^{n_i} \times \boldsymbol{K}^{n_i} (i \in \boldsymbol{I})$, 从而 $(\boldsymbol{x}^*, \boldsymbol{s}^*) \in \boldsymbol{K} \times \boldsymbol{K}$. 结合引理 1 知 $\lim_{k\to\infty} \boldsymbol{r}_k^{\mathrm{p}} = 0$, $\lim_{k\to\infty} \boldsymbol{r}_k^{\mathrm{d}} = 0$, 所以 $\boldsymbol{z}^*$ 可行. 又因为 $\lim_{k\to\infty}(\boldsymbol{x}^k)^{\mathrm{T}}\boldsymbol{s}^k = \boldsymbol{0}$, 所以 $(\boldsymbol{x}^*)^{\mathrm{T}}\boldsymbol{s}^* = \boldsymbol{0}$. 这表明 $(\boldsymbol{x}^*, \boldsymbol{s}^*)$ 是问题(P)和(D)的互补解. 于是 $\boldsymbol{z}^* \in \boldsymbol{S}$. 也就是说, 算法 Ⅰ 是全局收敛的. □

下面的引理 9 类似文献[14]中引理 3.3.

引理 9 存在一个常数 $\gamma_2 > 0$(与问题的数据及 r 有关), 使得

$$\|\Delta \boldsymbol{X}^{\mathrm{p}}\Delta \boldsymbol{s}^{\mathrm{p}}\| \leqslant \gamma_2[(\boldsymbol{x}^k)^{\mathrm{T}}\boldsymbol{s}^k]^2. \tag{21}$$

以下考虑算法 Ⅰ 的收敛速度.

定理 1 序列 $\{(\boldsymbol{x}^k)^{\mathrm{T}}\boldsymbol{s}^k\}$, $\{\boldsymbol{r}_k^{\mathrm{p}}\}$ 和 $\{\boldsymbol{r}_k^{\mathrm{d}}\}$ 线性收敛到 0.

证明 要证算法 Ⅰ 是线性收敛的, 由引理 1, 就是要证明 $\theta_k \geqslant \theta_0 > 0$. 由引理 5 和引理 10 可得

$$\theta_k \geqslant \min\left\{\frac{1}{2}, \sqrt{\frac{\alpha}{2r\gamma_2(\boldsymbol{x}^k)^{\mathrm{T}}\boldsymbol{s}^k}}\right\} \geqslant \min\left\{\frac{1}{2}, \sqrt{\frac{\alpha}{2r\gamma_2(\boldsymbol{x}^0)^{\mathrm{T}}\boldsymbol{s}^0}}\right\} \equiv \theta_0 > 0. \tag{22}$$

这样, $1-\theta_k \leqslant 1-\theta_0 < 1$, 序列 $\{(\boldsymbol{x}^k)^{\mathrm{T}}\boldsymbol{s}^k\}$, $\{\boldsymbol{r}_k^{\mathrm{p}}\}$ 和 $\{\boldsymbol{r}_k^{\mathrm{d}}\}$ 线性收敛到 0. □

若 $(1-\theta_k) \to 0$, 则 $\{(\boldsymbol{x}^k)^{\mathrm{T}}\boldsymbol{s}^k\}$, $\{\boldsymbol{r}_k^{\mathrm{p}}\}$ 和 $\{\boldsymbol{r}_k^{\mathrm{d}}\}$ 是超线性收敛的. 而且, 如果对于某一常数 $\gamma_0 > 0$, 有

$$1-\theta_k \leqslant \gamma_0(\boldsymbol{x}^k)^{\mathrm{T}}\boldsymbol{s}^k, \tag{23}$$

则 $\{(\boldsymbol{x}^k)^{\mathrm{T}}\boldsymbol{s}^k\}$, $\{\boldsymbol{r}_k^{\mathrm{p}}\}$ 和 $\{\boldsymbol{r}_k^{\mathrm{d}}\}$ 是二次收敛的. 以下引理给出满足式(23)的 θ_k 更大的下界.

引理 10

$$\theta_k \geqslant \frac{2}{1+\sqrt{1+(4/\alpha)\|\Delta X^{\mathrm{p}}\Delta s^{\mathrm{p}}/\mu_k\|}}. \tag{24}$$

证明 由引理5的证明可知

$$\|X(\theta)s(\theta)-(1-\theta)\mu_k e\| \leqslant \alpha(1-\theta)\mu_k+\theta^2\|\Delta X^{\mathrm{p}}\Delta s^{\mathrm{p}}\|.$$

那么

$$\alpha(1-\theta)\mu_k+\theta^2\|\Delta X^{\mathrm{p}}\Delta s^{\mathrm{p}}\| \leqslant 2\alpha(1-\theta)\mu_k, \tag{25}$$

确保式(17)成立. 显然, 对于所有的 $\theta \leqslant \theta_+$, 式(25)成立:

$$\theta_+ = \frac{2}{1+\sqrt{1+(4/\alpha)\|\Delta X^{\mathrm{p}}\Delta s^{\mathrm{p}}/\mu_k\|}}. \tag{26}$$

θ_+ 是对应式(25)的如下二次方程的正根:

$$(1/\alpha)\|\Delta X^{\mathrm{p}}\Delta s^{\mathrm{p}}/\mu_k\|\theta^2+\theta-1=0.$$

因此 $\theta_k \geqslant \theta_+$. □

类似文献[14]中引理3.5, 可得如下引理11.

引理11 存在一个常数 $\gamma_3>0$ 使得

$$1-\theta_k \leqslant \gamma_3 (x^k)^{\mathrm{T}} s^k. \tag{27}$$

定理2 序列 $\{(x^k)^{\mathrm{T}}s^k\}$, $\{r_k^{\mathrm{p}}\}$ 和 $\{r_k^{\mathrm{d}}\}$ 是二次收敛到0的.

证明 由引理1和引理11,有

$$(x^{k+1})^{\mathrm{T}}s^{k+1} \leqslant \gamma_3[(x^k)^{\mathrm{T}}s^k]^2$$

和

$$\|r_{k+1}^{\mathrm{p}}\| \leqslant r\gamma_3\frac{\mu_0}{\|r_0^{\mathrm{p}}\|}\|r_k^{\mathrm{p}}\|^2,\quad \|r_{k+1}^{\mathrm{d}}\| \leqslant r\gamma_3\frac{\mu_0}{\|r_0^{\mathrm{d}}\|}\|r_k^{\mathrm{d}}\|^2.$$ □

4 复杂性分析

由第3节已知 $(x^k)^{\mathrm{T}}y^k$ 有线性收敛速度$(1-\theta_0)$. 然而,式(22)中 θ_k 的下界 θ_0 包含常数 γ_2,这个和问题的数据及 r 有关.

为便于分析算法的复杂性, 假设对于初始内点 $(x^0, y^0, s^0) \in N(\alpha, \mu_0)$,存在某一$(x^*, y^*, s^*) \in S$ 和 $\rho \geqslant 1$ 满足

$$x^* \leqslant \rho x^0,\ s^* \leqslant \rho s^0. \tag{28}$$

如文献[16]定义衡量初始点可行性和最优性的指标 ω_{f} 和 ω_{o}:

$$\omega_{\mathrm{f}}=\omega_{\mathrm{f}}(x^0,y^0,s^0)= \min\left\{\max\left\{\frac{\|x-x^0\|_\infty}{\|x^0\|},\frac{\|s-s^0\|_\infty}{\|s^0\|}\right\}\middle| Ax=b,\ A^{\mathrm{T}}y+s=c\right\}, \tag{29}$$

$$\omega_{\mathrm{o}}=\omega_{\mathrm{o}}(x^0,y^0,s^0)= \min\left\{\max\left\{\frac{(x^{**})^{\mathrm{T}}s^0}{(x^0)^{\mathrm{T}}s^0},\frac{(x^0)^{\mathrm{T}}s^{**}}{(x^0)^{\mathrm{T}}s^0},1\right\}\middle| (x^{**},y^{**},s^{**}) \in S\right\}, \tag{30}$$

其中, $\|x\|_\infty = \max\{|x_i|,\ i \in I\}$.

下面的引理12给出了 ω_{f} 和 ω_{o} 的上界, 这与文献[16]引理3.4关于线性规划的结果类似.

引理12 设 $\{(x^0,y^0,s^0)\}$ 由式(28)确定. 那么

$$\omega_{\mathrm{f}} \leqslant \rho+1,\ \omega_{\mathrm{o}} \leqslant \rho. \tag{31}$$

证明 由式(28)和(29)有

$$\omega_{\mathrm{f}}=\min\left\{\max\left\{\frac{\|\boldsymbol{x}-\boldsymbol{x}^0\|_\infty}{\|\boldsymbol{x}^0\|},\frac{\|\boldsymbol{s}-\boldsymbol{s}^0\|_\infty}{\|\boldsymbol{s}^0\|}\right\}\,\middle|\,\boldsymbol{Ax}=\boldsymbol{b},\ \boldsymbol{A}^{\mathrm{T}}\boldsymbol{y}+\boldsymbol{s}=\boldsymbol{c}\right\}\leqslant$$
$$\max\left\{\frac{|\boldsymbol{x}_{ij}^*-\boldsymbol{x}_{ij}^0|}{\|\boldsymbol{x}^0\|},\frac{|\boldsymbol{s}_{ij}^*-\boldsymbol{s}_{ij}^0|}{\|\boldsymbol{s}^0\|},\ i\in\boldsymbol{I},\ j=1,2,\cdots,\ n_i-1\right\}\leqslant$$
$$\max\left\{\frac{(\rho+1)|\boldsymbol{x}_{ij}^0|}{\|\boldsymbol{x}^0\|},\ \frac{(\rho+1)|\boldsymbol{s}_{ij}^0|}{\|\boldsymbol{s}^0\|},\ i\in\boldsymbol{I},\ j=1,2,\cdots,\ n_i-1\right\}\leqslant$$
$$\rho+1,$$
$$\omega_{\mathrm{o}}=\min\left\{\max\left\{\frac{(\boldsymbol{x}^*)^{\mathrm{T}}\boldsymbol{s}^0}{(\boldsymbol{x}^0)^{\mathrm{T}}\boldsymbol{s}^0},\ \frac{(\boldsymbol{x}^0)^{\mathrm{T}}\boldsymbol{s}^*}{(\boldsymbol{x}^0)^{\mathrm{T}}\boldsymbol{s}^0},1\right\}\,\middle|\,(\boldsymbol{x}^*,\boldsymbol{y}^*,\boldsymbol{s}^*)\in\boldsymbol{S}\right\}\leqslant$$
$$\max\left\{\frac{(\boldsymbol{x}^*)^{\mathrm{T}}\boldsymbol{s}^0}{(\boldsymbol{x}^0)^{\mathrm{T}}\boldsymbol{s}^0},\frac{(\boldsymbol{x}^0)^{\mathrm{T}}\boldsymbol{s}^*}{(\boldsymbol{x}^0)^{\mathrm{T}}\boldsymbol{s}^0},1\right\}\leqslant\rho.$$ □

引理 13 设 $\{(\boldsymbol{x}^k,\boldsymbol{y}^k,\boldsymbol{s}^k)\}$ 由算法 Ⅰ 产生, $\{(\boldsymbol{x}^0,\boldsymbol{y}^0,\boldsymbol{s}^0)\}$ 由式(28)给出,则

$$\max\{\|(\boldsymbol{D}^k)^{-1}\Delta\boldsymbol{x}^{\mathrm{p}}\|,\|\boldsymbol{D}^k\Delta\boldsymbol{s}^{\mathrm{p}}\|\}\leqslant\zeta\sqrt{r(\boldsymbol{x}^k)^{\mathrm{T}}\boldsymbol{s}^k},\tag{32}$$

其中

$$\boldsymbol{D}^k=(\boldsymbol{X}^k)^{1/2}(\boldsymbol{S}^k)^{1/2},\tag{33}$$

$$\zeta=\frac{1}{\sqrt{r}}+\omega_{\mathrm{f}}(1+2\omega_{\mathrm{o}})\frac{1}{\sqrt{1-\alpha}}.\tag{34}$$

引理 14 设 $\{(\boldsymbol{x}^k,\boldsymbol{y}^k,\boldsymbol{s}^k)\}$ 由算法 Ⅰ 产生, $\{(\boldsymbol{x}^0,\boldsymbol{y}^0,\boldsymbol{s}^0)\}$ 由式(28) 给出. 那么存在与 r 无关的常数 $\gamma_4\in(0,1]$, 使得

$$\theta_k\geqslant\frac{\gamma_4}{r}.\tag{35}$$

证明 由引理 12 和引理 13, 可得

$$\|\Delta\boldsymbol{X}^{\mathrm{p}}\Delta\boldsymbol{s}^{\mathrm{p}}\|=\|\Delta\boldsymbol{X}^{\mathrm{p}}(\boldsymbol{D}^k)^{-1}\boldsymbol{D}^k\Delta\boldsymbol{s}^{\mathrm{p}}\|\leqslant\|(\boldsymbol{D}^k)^{-1}\Delta\boldsymbol{x}^{\mathrm{p}}\|\|\boldsymbol{D}^k\Delta\boldsymbol{s}^{\mathrm{p}}\|\leqslant$$
$$r\zeta^2(\boldsymbol{x}^k)^{\mathrm{T}}\boldsymbol{s}^k\leqslant r(\zeta^*)^2(\boldsymbol{x}^k)^{\mathrm{T}}\boldsymbol{s}^k,\tag{36}$$

其中

$$\zeta^*=1+(\rho+1)(1+2\rho)\frac{1}{\sqrt{1-\alpha}}.\tag{37}$$

所以,由前面的不等式,结合引理 5 得:

$$\theta_k\geqslant\min\left\{\frac{1}{2},\sqrt{\frac{\alpha}{2(\zeta^*)^2r^2}}\right\}.$$

考虑到 $\alpha=1/8$, 可得:

$$\theta_k\geqslant\frac{\gamma_4}{r},$$

其中 $\gamma_4=1/(4\zeta^*)$. 那么 $\theta_k=O(1/r)$. □

于是, 由引理 1 和引理 14 可得到算法 Ⅰ 的复杂性界.

定理 3 设 $\{(\boldsymbol{x}^k,\boldsymbol{y}^k,\boldsymbol{s}^k)\}$ 由算法 Ⅰ 产生, $\{(\boldsymbol{x}^0,\boldsymbol{y}^0,\boldsymbol{s}^0)\}$ 由式(28) 给出. 对于任意的 $\varepsilon>0$, 算法最多在 $O(rL)$ 次迭代后终止于点 $(\boldsymbol{x}^k,\boldsymbol{y}^k,\boldsymbol{s}^k)\in\boldsymbol{S}_\varepsilon$, 其中 $L=\ln(\varepsilon_0/\varepsilon)$, $\varepsilon_0=\max\{(\boldsymbol{x}^0)^{\mathrm{T}}\boldsymbol{s}^0,\|\boldsymbol{r}_0^{\mathrm{p}}\|,\|\boldsymbol{r}_0^{\mathrm{d}}\|\}$.

特别地, 如果初始点 $(\boldsymbol{x}^0,\boldsymbol{y}^0,\boldsymbol{s}^0)$ 可行,那么 $\omega_{\mathrm{f}}=0$, $\zeta=1/\sqrt{r}$. 此时,

$$\theta_k\geqslant\min\left\{\frac{1}{2},\sqrt{\frac{1}{8r}}\right\}=\sqrt{\frac{1}{8r}}.\tag{38}$$

所以, 算法 I 最多在 $O(\sqrt{r}L)$ 次迭代后终止,$(\boldsymbol{x}^k)^{\mathrm{T}}\boldsymbol{s}^k \leqslant \varepsilon$, 其中

$$L = \ln((\boldsymbol{x}^0)^{\mathrm{T}}\boldsymbol{s}^0/\varepsilon). \tag{39}$$

5 预估-校正算法 II

采用 Euler 方向作为预估方向, 本节给出另一个预估-校正算法. 算法 II 与算法 I 除了步骤 2 的预估方向的选取不同之外都是相同的. 显然,两个算法产生的迭代点列具有相似的性质. 下面仅给出算法 II 与算法 I 一个不同的步骤(步骤 2)和两个相似的结论.

算法 II

步骤 2(确定预估方向) 求解如下的方程组得到预估方向 $(\Delta\boldsymbol{x}^{\mathrm{p}},\Delta\boldsymbol{y}^{\mathrm{p}},\Delta\boldsymbol{s}^{\mathrm{p}})$:

$$\boldsymbol{F}'(\boldsymbol{x}^k,\boldsymbol{y}^k,\boldsymbol{s}^k)\begin{pmatrix}\Delta\boldsymbol{x}^{\mathrm{p}}\\ \Delta\boldsymbol{y}^{\mathrm{p}}\\ \Delta\boldsymbol{s}^{\mathrm{p}}\end{pmatrix} = \begin{pmatrix}\boldsymbol{r}_k^{\mathrm{p}}\\ \boldsymbol{r}_k^{\mathrm{d}}\\ -\mu_k\boldsymbol{e}\end{pmatrix},$$

即

$$\begin{cases}\boldsymbol{A}\Delta\boldsymbol{x}^{\mathrm{p}} = \boldsymbol{r}_k^{\mathrm{p}},\\ \boldsymbol{A}^{\mathrm{T}}\Delta\boldsymbol{y}^{\mathrm{p}} + \Delta\boldsymbol{s}^{\mathrm{p}} = \boldsymbol{r}_k^{\mathrm{d}},\\ \boldsymbol{S}^k\Delta\boldsymbol{x}^{\mathrm{p}} + \boldsymbol{X}^k\Delta\boldsymbol{s}^{\mathrm{p}} = -\mu_k\boldsymbol{e}.\end{cases} \tag{40}$$

引理 15 设 $\{(\boldsymbol{x}^k,\boldsymbol{y}^k,\boldsymbol{s}^k)\}$ 由算法 II 产生. 则

$$\theta_k \geqslant \delta_2 \equiv \min\left\{\frac{1}{4},\sqrt{\frac{\alpha\mu_k}{2\|\Delta\boldsymbol{X}^{\mathrm{p}}\Delta\boldsymbol{s}^{\mathrm{p}}\|}}\right\}. \tag{41}$$

证明 对于 $\theta \in (0,1]$, 有

$$\begin{aligned}&\|\boldsymbol{X}(\theta)\boldsymbol{s}(\theta) - (1-\theta)\mu_k\boldsymbol{e}\| = \\ &\quad \|\boldsymbol{X}^k\boldsymbol{s}^k + \theta(-\mu_k\boldsymbol{e}) + \theta^2\Delta\boldsymbol{X}^{\mathrm{p}}\Delta\boldsymbol{s}^{\mathrm{p}} - (1-\theta)\mu_k\boldsymbol{e}\| \leqslant \\ &\quad \|\boldsymbol{X}^k\boldsymbol{s}^k - \mu_k\boldsymbol{e}\| + \theta^2\|\Delta\boldsymbol{X}^{\mathrm{p}}\Delta\boldsymbol{s}^{\mathrm{p}}\| \leqslant \\ &\quad \alpha\mu_k + \theta^2\|\Delta\boldsymbol{X}^{\mathrm{p}}\Delta\boldsymbol{s}^{\mathrm{p}}\|.\end{aligned}$$

对于所有的 $\theta \in (0,\delta_2]$, 有 $\theta \leqslant 1/4$ 和 $\theta^2\|\Delta\boldsymbol{X}^{\mathrm{p}}\Delta\boldsymbol{s}^{\mathrm{p}}\| \leqslant \alpha\mu_k/2$. 那么,

$$\|\boldsymbol{X}(\theta)\boldsymbol{s}(\theta) - (1-\theta)\mu_k\boldsymbol{e}\| \leqslant (3/2)\alpha\mu_k \leqslant 2\alpha(1-\theta)\mu_k.$$

因此,由 θ_k 的定义可知 $\theta_k \geqslant \delta_2$. □

由

$$\|\boldsymbol{X}(\theta)\boldsymbol{s}(\theta) - (1-\theta)\mu_k\boldsymbol{e}\| \leqslant \alpha\mu_k + \theta^2\|\Delta\boldsymbol{X}^{\mathrm{p}}\Delta\boldsymbol{s}^{\mathrm{p}}\| \leqslant 2\alpha(1-\theta)\mu_k, \tag{42}$$

可得一个对应式(42)的二次不等式:

$$\frac{1}{\alpha}\|\Delta\boldsymbol{X}^{\mathrm{p}}\Delta\boldsymbol{s}^{\mathrm{p}}/\mu_k\|\theta^2 + 2\theta - 1 \leqslant 0.$$

这样, 存在 1 个正根

$$\theta_{++} = \frac{1}{1+\sqrt{1+(1/\alpha)\|\Delta\boldsymbol{X}^{\mathrm{p}}\Delta\boldsymbol{s}^{\mathrm{p}}/\mu_k\|}}, \tag{43}$$

使得当 $\theta \leqslant \theta_{++}$ 时,式(42) 成立. 这表明在算法 II 中 $\theta_k \geqslant \theta_{++}$.

定理 4 对于任意的 $\varepsilon > 0$, 算法 II 最多在 $O(rL)$ 次迭代后终止于点$(\boldsymbol{x}^k,\boldsymbol{y}^k,\boldsymbol{s}^k) \in \boldsymbol{S}_\varepsilon$,其中,$L = \ln(\varepsilon_0/\varepsilon)$,$\varepsilon_0 = \max\{(\boldsymbol{x}^0)^{\mathrm{T}}\boldsymbol{s}^0, \|\boldsymbol{r}_0^{\mathrm{p}}\|, \|\boldsymbol{r}_0^{\mathrm{d}}\|\}$.

6 数值实验结果

本节对提出的两个算法进行了一些数值实验,并比较了算法的数值效果. 数值实验是在 Windows XP 系统环境下,使用 MATLAB7.5 编程运行的.

问题 1 问题取自文献[17]. 原问题(P)和对偶问题(D)的数据详见文献[17]. 可行初始点的选取如下:

$$\boldsymbol{x}^0=(\boldsymbol{x}_1;\boldsymbol{x}_2;\boldsymbol{x}_3;\boldsymbol{x}_4)\in\boldsymbol{K},\ \boldsymbol{s}^0=(\boldsymbol{s}_1;\boldsymbol{s}_2;\boldsymbol{s}_3;\boldsymbol{s}_4)\in\boldsymbol{K},\ \boldsymbol{y}^0=(0;0;0;0)\in R^4,$$

其中, $\boldsymbol{x}_i=\boldsymbol{s}_i=(2;1;0;0)\in\boldsymbol{K}^4,\ i=1,2,3,4$. 记 $\boldsymbol{u}^0\triangleq(\boldsymbol{x}^0,\boldsymbol{y}^0,\boldsymbol{s}^0)$.

在算法Ⅰ和算法Ⅱ中选取不可行初始内点 $\boldsymbol{z}^0$:

$$\boldsymbol{z}^0\triangleq(\boldsymbol{x}^0,\boldsymbol{y}^0,\boldsymbol{s}^0)=(\gamma\boldsymbol{e},\boldsymbol{0},\gamma\boldsymbol{e}),\qquad \gamma>0,\ \boldsymbol{e}\in R^{16}.$$

比较文献[17], SeDuMi 1.3 和算法Ⅰ、Ⅱ的迭代次数 k 及计算机运行时间(单位:s),结果列在表 1 中. 取 $\alpha=0.125$, $\Gamma=2$, $\varepsilon=10^{-8}$.

表 1 测试问题 1 的数值结果比较

Table 1 Numerical results of algorithms Ⅰ and Ⅱ for test problem 1

algorithms	feasible initial point $\boldsymbol{u}^0$		infeasible initial point $\boldsymbol{z}^0$							
			$\gamma=0.5$		$\gamma=1$		$\gamma=3$			
	k	t/s	k	t/s	k	t/s	k	t/s	k	t/s
in[17]	39	/								
SeDuMi 1.3									10	0.4
algorithm Ⅰ	9	0.078 1	15	0.109 4	12	0.109 4	10	0.078 1		
algorithm Ⅱ	/	/	16	0.125 0	13	0.093 8	11	0.109 4		

表 2 4 个测试问题

Table 2 Four test problems

problems' name	n	m	r	structure of SOCs
nb	2 383	123	797	[4×1, 793×3]
nb-L1	3 176	915	1 590	[797×1, 793×3]
nb-L2	4 195	123	843	[4×1,1×1 677, 838×3]
nb-L2-bessel	2 641	123	843	[4×1,1×123, 838×3]

表 3 算法Ⅰ和算法Ⅱ求解表 2 测试问题的数值结果

Table 3 Numerical results of algorithms Ⅰ and Ⅱ for four test problems

test problems	starting points	algorithms Ⅰ		algorithms Ⅱ		in [19]	
		k	t/s	k	t/s	k	t/s
nb	$\boldsymbol{x}=\boldsymbol{s}=0.05\boldsymbol{e},\ \boldsymbol{y}=\boldsymbol{0}$	52	90.453 1	52	91.078 1		
						29	311.2
nb-L1	$\boldsymbol{x}=2\boldsymbol{e},\ \boldsymbol{y}=\boldsymbol{0},\ \boldsymbol{s}=0.1\boldsymbol{e}$	55	109.234 4	55	108.484 4		
						87	158 7.4
nb-L2	$\boldsymbol{x}=\boldsymbol{s}=0.45\boldsymbol{e},\ \boldsymbol{y}=\boldsymbol{0}$	28	103.625 0	28	103.125 0		
						/	/
nb-L2-bessel	$\boldsymbol{x}=\boldsymbol{s}=0.17\boldsymbol{e},\ \boldsymbol{y}=\boldsymbol{0}$	22	37.093 8	22	37.437 5		
						9	113.7

在表 1 中,算法Ⅰ和算法Ⅱ的最优值是 $p^*=d^*=10.426\,186\,788$. 这些结果以及最优解和

文献[17]上的一致. 另外, 因为 $\boldsymbol{u}^0 \notin N(\alpha,5)$,当初始点为 $\boldsymbol{u}^0$ 时,算法Ⅱ不能求解. 从表1可以看到文中提出的两个算法是有效的. 它们不仅能够求解初始点可行的情形, 而且能够求解初始点不可行的情形. 对于可行初始点 $\boldsymbol{u}^0$,算法 Ⅰ 的迭代次数 k 比文献[17]的少. 对于不可行初始点 z^0,当 $\gamma = 1$ 或3时, 这两个算法的数值效果也很好. 虽然它们的迭代次数比用 SeDduMi 1.3 计算的要多,但是前者的运行时间比后者要少.

随后, 从文献[18]中选出4个关于天线问题的例子,相关数据列在表2中. 记号[4×1, 793×3]表示 $\boldsymbol{K}$ 包含4个 $\boldsymbol{K}^1$ 和793个 $\boldsymbol{K}^3$. 表3中给出了算法 Ⅰ 和算法 Ⅱ 对这4个测试问题的数值实验结果,其中 $\boldsymbol{e}$ 在不同的问题里表示相应维数的向量. 取 $\alpha = 0.125$,$\Gamma = 2$,$\varepsilon = 10^{-4}$. 测试结果与文献[19]的相比较,可以看出文中提出的两个算法是有效的.

参考文献:

[1] Alizadeh F, Goldfarb D. Second-order cone programming[J]. *Mathematical Programming Ser B*, 2003,**95**(1):3-51.

[2] Witzgall C. Optimal location of a central facility, mathematical models and concepts[R]. Technical Report 8388, National Bureau of Standards,Washington, DC, 1964.

[3] Lobo M S, Vandenberghe L, Boyd S, Lebret H. Applications of second-order cone programming[J]. *Linear Algebra and Its Applications*, 1998, **284**(1/3): 193-228.

[4] Kanno Y, Ohsaki M, Ito J. Large-deformation and friction analysis of non-linear elastic cable networks by second-order cone programming[J]. *International Journal for Numerical Methods in Engineering*,2002, **55**(9):1079-1114.

[5] Makrodimopoulos A, Martin C M. Lower bound limit analysis of cohesive-frictional materials using second-order cone programming[J]. *International Journal for Numerical Methods in Engineering*,2006, **66**(4):604-634.

[6] Nemirovskii A, Scheinberg K. Extension of Karmarkar's algorithm onto convex quadratically constrained quadratic problems[J]. *Mathematical Programming*, 1996,**72**(3):273-289.

[7] Nesterov Y E, Todd M J. Self-scaled barriers and interior-point methods for convex programming[J]. *Mathematics of Operations Research*, 1997, **22**(1):1-42.

[8] Nesterov Y E, Todd M J. Primal-dual interior-point methods for self-scaled cones[J]. *SIAM Journal on Optimization*, 1998, **8**(2): 324-364.

[9] Adler I, Alizadeh F. Primal-dual interior-point algorithms for convex quadratically constrained and semidefinite optimization problems[R]. Technical Report RRR 46-95, RUTCOR, Rutgers University, 1995.

[10] Alizadeh F, Haeberly J P A, Overton M L. Primal-dual interior-point methods for semidefinite programming: convergence rates, stability and numerical results[J]. *SIAM Journal on Optimization*, 1998, **8**(3): 746-768.

[11] Monteiro R D C, Tsuchiya T. Polynomial convergence of primal-dual algorithms for the second-order cone program based on the MZ-family of directions[J]. *Mathematical Programming Ser A*, 2000, **88**(1):61-83.

[12] Chi X N, Liu S Y. An infeasible-interior-point predictor-corrector algorithm for the second-order cone program[J]. *Acta Mathematica Scientia B*, 2008, **28**(3): 551-559.

[13] Mizuno S, Todd M J, Ye Y. On adaptive-step primal-dual interior-point algorithms for linear programming[J]. *Mathematics of Operations Research*, 1993, **18**(4): 964-981.

[14] Miao J M. Two infeasible interior-point predictor-corrector algorithms for linear programming

[J]. *SIAM Journal on Optimization*, 1996, **6**(3): 587-599.

[15] Zhang Y. On the convergence of a class of infeasible interior-point methods for the horizontal linear complementarity problem[J]. *SIAM Journal on Optimization*, 1994, **4**(1): 208-227.

[16] Kojima M. Basic lemmas in polynomial-time infeasible-interior point methods for linear programs[J]. *Annals of Operations Research*, 1996, **62**(1):1-28.

[17] Bai Y Q, Wang G Q, Roos C. Primal-dual interior-point algorithms for second-order cone optimization based on kernel functions[J]. *Nonlinear Analysis*, 2009, **70**(10): 3584-3602.

[18] Pataki G, Schmieta S. The DIMACS library of semidefinite-quadratic-linear programs[R]. Technical report. Computational Optimization Research Center, Columbia University, 2002.

[19] Pan Sh H, Chen J Sh. A semismooth Newton method for SOCCPs based on a one-parametric class of SOC complementarity functions[J]. *Computational Optimization and Applications*, 2010, **45**(1):59-88.

Two New Predictor-Corrector Algorithms for Second-Order Cone Programming

ZENG You-fang[1,2], BAI Yan-qin[1], JIAN Jin-bao[2], TANG Chun-ming[2]

(1. *Department of Mathematics*, *Shanghai University*, *Shanghai* 200444, *P. R. China*;
2. *College of Mathematics and Information Science*, *Guangxi University*, *Nanning* 530004, *P. R. China*)

Abstract: Based on the ideas of infeasible interior-point methods and predictor-corrector algorithms, two interior-point predictor-corrector algorithms for second-order cone programming (SOCP) were presented. They use the Newton direction and the Euler direction as the predictor directions, respectively. The corrector directions belong to the category of the Alizadeh-Haeberly-Overton (AHO) directions. The two new algorithms were suitable to cases of feasible and infeasible interior iterative points. A simpler neighborhood of central path for the SOCP was proposed, which was the pivotal difference from other interior-point predictor-corrector algorithms. Under some assumptions, the algorithms possess global, linear and quadratic convergence. The complexity bound $O(r\ln(\varepsilon_0/\varepsilon))$ was obtained, where r denotes the number of second-order cones in SOCP problem. The numerical results show that the proposed algorithms are effective.

Key words: second-order cone programming; infeasible interior-point algorithm; predictor-corrector algorithm; global convergence; complexity analysis