ISSN 1746-7659, England, UK
Journal of Information and Computing Science
Vol. 3, No. 2, 2008, *pp*. 139-145

A Smoothing Method with a Smoothing Variable for Second-order Cone Programming

Xiaoni Chi [1+] and Wenlve Chen[1]

[1] College of Mathematics and Information Science, Huanggang Normal University, Huangzhou 438000, Hubei, P.R. China

(*Received March 15, 2005, accepted April 28, 2005*)

Abstract. Based on the Chen-Harker-Kanzow-Smale smoothing function, a smoothing method with a smoothing variable is proposed for solving the second-order cone programming. Our algorithm needs to solve only one linear system of equations and to perform only one line search at each iteration. Without restrictions regarding its starting point, the algorithm is shown to be globally convergent.

Keywords: second-order cone programming, smoothing method, smoothing variable, global convergence.

1. Introduction

The second-order cone programming (SOCP) problem is to minimize or maximize a linear function over the intersection of an affine space with the Cartesian product of a finite number of second-order cones. Consider the following SOCP problem

$$(P) \quad \min\left\{\sum_{i=1}^{n} c_i^T x_i : \sum_{i=1}^{n} A_i x_i = b,\, x_i \in K_i,\, i=1,\ldots,n\right\}, \tag{1}$$

or its dual problem [1]

$$(D) \quad \max\left\{b^T y : A_i^T y + s_i = c_i,\, s_i \in K_i,\, i=1,\ldots,n\right\}. \tag{2}$$

Here $A_i \in R^{m\times k_i}$, $c_i \in R^{k_i}$, $i=1,\ldots,n, b\in R^m$ are the data, and $x_i \in K_i$, $s_i \in K_i$, $i=1,\ldots,n,\ y\in R^m$ are the variables. The set $K_i\ (i=1,\ldots,n)$ is the second-order cone (SOC) of dimension k_i defined by

$$K_i := \left\{x_i = (x_{i0}, x_{i1}) \in R\times R^{k_i-1} : x_{i0} - \| x_{i1}\| \ge 0\right\},$$

Where $\|\cdot\|$ stands for the Euclidean norm. It is easy to see that the SOC K_i is self-dual and its interior is

$$K_i^0 = \left\{x_i = (x_{i0}, x_{i1}) \in R\times R^{k_i-1} : x_{i0} - \| x_{i1}\| > 0\right\}.$$

Let

$$k = k_1 + \cdots + k_n, \quad K = K_1 \times \cdots \times K_n,$$
$$A = (A_1, \cdots, A_n) \in R^{m\times k}, \quad c = (c_1, \cdots, c_n) \in R^k,$$
$$x = (x_1, \cdots, x_n) \in K, \quad s = (s_1, \cdots, s_n) \in K,$$

where we use $x = (x_1, \cdots, x_n)$ for the column vector $x = (x_1^T, \cdots, x_n^T)^T$. Thus, problems (1) and (2) can be simply written as

$$(P) \quad \min\{c^T x : Ax = b, x\in K\},$$
$$(D) \quad \max\{b^T y : A^T y + s = c, s\in K\}.$$

The set of strictly feasible solutions of (1) and (2) are:

+ Corresponding author. *E-mail address*: chixiaoni@126.com.

$$F^0(P) = \{x : Ax = b, x \in K^0\},$$

$$F^0(D) = \{(y,s) : A^T y + s = c, s \in K^0\}$$

respectively, where

$$K^0 = K_1^0 \times K_2^0 \times \cdots \times K_n^0.$$

Throughout this paper, we assume that $F^0(P) \times F^0(D) \neq \emptyset.$ Therefore it can be shown that both (1) and (2) have optimal solutions and their optimal values are coincident [1].

Recently the SOCP problems have received considerable attention, since they have a wide range of engineering applications [6] and include a large class of problems as special cases [5], such as linear programs and convex quadratic programs. Without loss of generality, we focus on the SOCP problems (1) and (2) with $n = 1$ and $k = k_1$ in the following analysis.

As novel algorithms for solving optimization problems, smoothing methods (non-interior continuation methods) [2, 7, 8] perform very well in both theory and practice. However, many existing algorithms [2, 8] need to solve two linear systems of equations and to perform two or three line searches at each iteration. Moreover, the smoothing methods available are mostly for solving complementarity problems [2, 8] and variational inequality problems [7], whereas there is little work on smoothing methods for the SOCP.

In this paper, we present a smoothing method for the SOCP based on the Chen-Harker-Kanzow-Smale (CHKS) smoothing function [4]. By introducing a smoothing variable, our algorithm reformulates the SOCP as a nonlinear system of equations. Then we solve the system by using Newton's method. It is shown that our algorithm has the following good features:

(i) unlike interior point methods, our algorithm does not have restrictions regarding its starting point;

(ii) the algorithm solves only one linear system of equations and performs only one line search at each iteration;

(iii) if A has full row rank, then any accumulation point of the iteration sequence generated by our algorithm is a solution of the SOCP.

This paper is organized as follows. In Section 2, we give some preliminary results for the SOC and propose a smoothing method with a smoothing variable for the SOCP. In Section 3, we analyze the convergence properties of our algorithm.

The following notations are used throughout this paper. The space of k-dimensional real column vectors (respectively, real numbers) is denoted by R^k (respectively, R). The set of all $m \times k$ matrices with real entries is denoted by $R^{m \times k}$. The superscript T denotes the transpose. For convenience, we often write $x = (x_0, x_1)$ instead of the column vector $x = (x_0, x_1^T)^T \in R \times R^{k-1}$. $\|\cdot\|$ denotes the Euclidean norm. For any $\alpha, \beta > 0$, $\alpha = o(\beta)$ means that α / β tends to zero as $\beta \to 0$.

2. Preliminaries and Algorithm

In this section, we briefly review the Euclidean Jordan algebra associated with the SOC [1, 3], which will play an important role in the design and analysis of our algorithm. Then we propose a smoothing method with a smoothing variable for the SOCP based on the CHKS smoothing function. We show the well-definedness of our algorithm and investigate some properties of the iteration sequence generated by our algorithm.

The Euclidean Jordan algebra for the SOC K is the algebra defined by

$$x \circ s = (x^T s, x_0 s_1 + s_0 x_1), \quad \forall x, s \in R^k,$$

with $e = (1, 0, \cdots, 0) \in R^k$ being its unit element. Given an element $x \in R^k$, we define the symmetric matrix

$$L_x = \begin{pmatrix} x_0 & x_1^T \\ x_1 & x_0 I \end{pmatrix}.$$

It is easy to verify that

$$x \circ s = L_x s = L_s x, \quad \forall s \in R^k.$$

Moreover, L_x is positive semidefinite (positive definite) if and only if $x \in K$ ($x \in K^0$).

Spectral factorization is one of the basic concepts in the Euclidean Jordan algebra. For any $x = (x_0, x_1) \in R \times R^{k-1}$, its spectral factorization with respect to the SOC K is defined as

$$x = \lambda_1 u^{(1)} + \lambda_2 u^{(2)},$$

where λ_1, λ_2 and $u^{(1)}, u^{(2)}$ are the spectral values and the associated spectral vectors given by

$$\lambda_i = x_0 + (-1)^i \|x_1\|,$$

$$u^{(i)} = \begin{cases} \frac{1}{2}(1, (-1)^i \frac{x_1}{\| x_1 \|}), & \text{if } x_1 \neq 0 \\ \frac{1}{2}(1, (-1)^i \omega), & \text{otherwise} \end{cases}$$

for $i = 1, 2,$ with $\omega \in R^{k-1}$ such that $\|\omega\| = 1.$ For any $x \in K,$ it is obvious that both spectral values of x are nonnegative. By using the spectral factorization, we may define the functions on R^k associated with the SOC K by

$$x^2 = \lambda_1^2 u^{(1)} + \lambda_2^2 u^{(2)}, \quad \forall x \in R^k,$$

$$\sqrt{x} = \sqrt{\lambda_1} u^{(1)} + \sqrt{\lambda_2} u^{(2)}, \quad \forall x \in K.$$

Under the assumption that $F^0(P) \times F^0(D) \neq \varnothing,$ solving the SOCP problems (1) and (2) is equivalent to [1] finding $(x, y, s) \in R^k \times R^m \times R^k$ such that

$$\begin{cases} Ax = b, & x \in K, \\ A^T y + s = c, & s \in K, \\ x \circ s = 0. \end{cases} \tag{3}$$

Our smoothing method aims to reformulate the optimality conditions (3) as a nonlinear system of equations, which does not contain any explicit inequality constraints like $x \in K, s \in K$ or $x \in K^0, \ s \in K^0.$ By applying Newton's method to the system of equations, one can expect to find a solution of (1) and (2).

Our algorithm is based on the Chen-Harker-Kanzow-Smale (CHKS) smoothing function [4] $\phi : R^k \times R^k \times R \to R^k$ defined by

$$\phi(x, s, \mu) = x + s - \sqrt{(x - s)^2 + 4\mu^2 e}.$$

By Proposition 4.1 in [4], $\phi(x, s, 0)$ satisfies

$$\phi(x, s, 0) = 0 \Leftrightarrow x \circ s = 0, x \in K, s \in K. \tag{4}$$

Since $\phi(x, s, 0)$ is typically nonsmooth and $\phi(x, s, \mu)$ is continuously differentiable for any $\mu > 0$ [4], the variable μ is referred to as the smoothing variable. Let $z := (x, y, s) \in R^k \times R^m \times R^k$ and define

$$H_\sigma(z, \mu) := \begin{pmatrix} \Phi(z, \mu) \\ \sigma\mu \end{pmatrix}, \tag{5}$$

Where $\sigma \in (0,1]$ is the parameter and

$$\Phi(z,\mu):=\begin{pmatrix} Ax-b \\ A^T y+s-c \\ \phi(x,s,\mu) \end{pmatrix}.$$

Let $H(z,\mu):=H_1(z,\mu)$. Since the system of equations $H(z,\mu)=0$ automatically implies $\mu=0$, it follows from (3), (4) and (5) that

$$z^*:=(x^*,y^*,s^*) \text{ solves (3)} \Leftrightarrow (z^*,0) \text{ solves } H(z,\mu)=0.$$

In this way, we obtain a reformulation of the optimality conditions (3) where μ is viewed as an independent variable.

Algorithm 2.1

Step 0 Choose $\sigma,\delta\in(0,1)$ and $\mu_0\in(0,\infty)$. Let $(x_0,y_0,s_0)\in R^k\times R^m\times R^k$ be an arbitrary point and $z_0:=(x_0,y_0,s_0)$. Choose $\beta>0$ such that $\|\Phi(z_0,\mu_0)\|\le\beta\mu_0$. Set $k:=0$.

Step 1 If $\Phi(z_k,0)=0$, stop.

Step 2 Compute a solution $(\Delta z_k,\Delta\mu_k):=(\Delta x_k,\Delta y_k,\Delta s_k,\Delta\mu_k)\in R^k\times R^m\times R^k\times R$ of the linear system

$$H'(z_k,\mu_k)\begin{pmatrix}\Delta z\\ \Delta\mu\end{pmatrix}=-H_\sigma(z_k,\mu_k). \tag{6}$$

Step 3 Let $\lambda_k=\max\{\delta^l \mid l=0,1,2,\ldots\}$ such that

$$\|\Phi(z_k+\lambda_k\Delta z_k,(1-\sigma\lambda_k)\mu_k)\|\le\beta(1-\sigma\lambda_k)\mu_k. \tag{7}$$

Step 4 Set $z_{k+1}:=z_k+\lambda_k\Delta z_k$ and $\mu_{k+1}:=(1-\sigma\lambda_k)\mu_k$. Set $k:=k+1$ and go to Step 1.

Remark Many smoothing methods available [2, 8] view μ as a smoothing parameter. Thus they [2, 8] have to perform at least two line searches at each iteration to make both $\|\Phi(z,\mu)\|$ and μ decrease gradually. However, in Algorithm 2.1 we view μ as a smoothing variable. Therefore we need to perform only one line search at each iteration.

To analyze Algorithm 2.1, we study the Lipschitzian and differential properties of the function $H(z,\mu)$ and derive the computable formula for its Jacobian. By using Corollary 5.3 in [4] and following the proof of Proposition 6.2 in [4], it is not difficult to obtain the following properties of $H(z,\mu)$.

Lemma 2.1 (i) The function $H(z,\mu)$ is globally Lipschitz continuous in R^{m+2k+1}. For any $\mu>0$, $H(z,\mu)$ is continuously differentiable with its Jacobian

$$H'(z,\mu)=\begin{pmatrix} A & 0 & 0 & 0 \\ 0 & A^T & I & 0 \\ M(z,\mu) & 0 & N(z,\mu) & P(z,\mu) \\ 0 & 0 & 0 & 1 \end{pmatrix}, \tag{8}$$

where

$$M(z,\mu)=I-L_w^{-1}L_v,\quad N(z,\mu)=I+L_w^{-1}L_v,\quad P(z,\mu)=-4\mu L_w^{-1}e,$$

$$v:=x-s,\ \ w:=\sqrt{v^2+4\mu^2 e}.$$

(ii) If A has full row rank, $H'(z,\mu)$ is nonsingular for any $\mu>0$.

Theorem 2.2 If A has full row rank, Algorithm 2.1 is well-defined.

Proof Since $\mu_k>0$ by the algorithm and A has full row rank, it follows from Lemma 2.1 (ii) that $H'(z_k,\mu_k)$ is nonsingular. This demonstrates the well-definedness of Step 2.

Now we show that Step 3 is well-defined by induction.

Due to (5), (6) and (8), we have

$$\Phi'(z_0,\mu_0)\begin{pmatrix}\Delta z_0\\ \Delta\mu_0\end{pmatrix} = -\Phi(z_0,\mu_0), \tag{9}$$

$$\Delta\mu_0 = -\sigma\mu_0. \tag{10}$$

For any $\alpha \in (0,1]$, let

$$r(\alpha) := \Phi(z_0+\alpha\Delta z_0, \mu_0+\alpha\Delta\mu_0) - \Phi(z_0,\mu_0) - \alpha\Phi'(z_0,\mu_0)\begin{pmatrix}\Delta z_0\\ \Delta\mu_0\end{pmatrix}. \tag{11}$$

By Lemma 2.1 (i), Φ is continuously differentiable around (z_0,μ_0). Then it follows from (11) that

$$\| r(\alpha)\| = o(\alpha). \tag{12}$$

Taking into account the fact $\| \Phi(z_0,\mu_0)\| \le \beta\mu_0$ and using (9), (10), (11) and (12), we have

$$\begin{aligned}\| \Phi(z_0+\alpha\Delta z_0, (1-\sigma\alpha)\mu_0)\| &= \| \Phi(z_0+\alpha\Delta z_0, \mu_0+\alpha\Delta\mu_0)\| \\ &\le (1-\alpha)\| \Phi(z_0,\mu_0)\| + \| r(\alpha)\| \\ &\le \beta(1-\alpha)\mu_0 + o(\alpha).\end{aligned} \tag{13}$$

(13) implies that there exists a constant $\bar{\alpha} \in (0,1]$ such that

$$\| \Phi(z_0+\alpha\Delta z_0, (1-\sigma\alpha)\mu_0)\| \le \beta(1-\sigma\alpha)\mu_0$$

holds for any $\alpha \in (0,\bar{\alpha}]$ and hence $\| \Phi(z_1,\mu_1)\| \le \beta\mu_1$ holds by Step 4.

Assume that (7) holds at the k-th iteration, i.e., there exists λ_k such that

$$\| \Phi(z_{k+1},\mu_{k+1})\| \le \beta\mu_{k+1}, \tag{14}$$

where $z_{k+1} = z_k + \lambda_k\Delta z_k$ and $\mu_{k+1} = (1-\sigma\lambda_k)\mu_k$.

Then we consider the $(k+1)$-st iteration. Using the same argument as above and (14), we obtain that there exists λ_{k+1} such that

$$\| \Phi(z_{k+1}+\lambda_{k+1}\Delta z_{k+1}, (1-\sigma\lambda_{k+1})\mu_{k+1})\| \le \beta(1-\sigma\lambda_{k+1})\mu_{k+1}.$$

Therefore by Step 4 we have

$$\| \Phi(z_{k+2},\mu_{k+2})\| \le \beta\mu_{k+2},$$

where $z_{k+2} = z_{k+1} + \lambda_{k+1}\Delta z_{k+1}$ and $\mu_{k+2} = (1-\sigma\lambda_{k+1})\mu_{k+1}$. Hence Step 3 is well-defined.

This completes the proof.

From Theorem 2.2, Algorithm 2.1 generates an infinite sequence $\{(z_k,\mu_k)\}$. A simple induction argument allows us to state the following properties of $\{(z_k,\mu_k)\}$, which are useful in analyzing the convergence of our algorithm.

Lemma 2.3 Suppose that A has full row rank. Then we have

(i) $Ax_k - b = (1-\lambda_{k-1})\cdots(1-\lambda_0)(Ax_0-b) \to 0$ as $k\to\infty$;

(ii) $A^T y_k + s_k - c = (1-\lambda_{k-1})\cdots(1-\lambda_0)(A^T y_0 + s_0 - c) \to 0$ as $k\to\infty$;

(iii) $\mu_k = (1-\sigma\lambda_{k-1})\cdots(1-\sigma\lambda_0)\mu_0 > 0$ for any $k \ge 0$;

(iv) $\| \Phi(z_k,\mu_k)\| \le \beta\mu_k$ for any $k \ge 0$.

3. Convergence Analysis

In this section, we establish the global convergence of Algorithm 2.1. We show that the sequence $\{\mu_k\}$ converges to zero and that any accumulation point of the sequence $\{z_k\}$ is a solution of the SOCP problems (1) and (2).

Theorem 3.1 Suppose that A has full row rank and that the iteration sequence $\{z_k\}$ generated by Algorithm 2.1 has at least one accumulation point. Then the sequence $\{\mu_k\}$ converges to 0.

Proof Since the sequence $\{\mu_k\}$ is monotonically decreasing and bounded from below by zero by Lemma 2.3 (iii), it converges to a number $\mu^* \geq 0$. If $\mu^* = 0$, we obtain the desired result.

Assume that $\mu^* > 0$. Let z^* be an accumulation point of the sequence $\{z_k\}$. Without loss of generality, we may assume that

$$\lim_{k\to\infty}(z_k, \mu_k) = (z^*, \mu^*).$$

By Lemma 2.1 (i), Φ is continuously differentiable around (z^*, μ^*). Then

$$\lim_{k\to\infty}\Phi(z_k, \mu_k) = \Phi(z^*, \mu^*), \quad \lim_{k\to\infty}\Phi'(z_k, \mu_k) = \Phi'(z^*, \mu^*).$$

Since $\mu_k \to \mu^* > 0$ by assumption, it follows from Lemma 2.3 (iii) that $\lim_{k\to\infty}\lambda_k = 0$. By Step 3, the steplength $\bar{\lambda}_k := \lambda_k/\delta$ does not satisfy the line search criterion (7), i.e.,

$$\| \Phi(z_k + \bar{\lambda}_k\Delta z_k, (1-\sigma\bar{\lambda}_k)\mu_k)\| > \beta(1-\sigma\bar{\lambda}_k)\mu_k. \tag{15}$$

On the one hand, we get from (15) and Lemma 2.3 (iv) that

$$\begin{aligned}\| \Phi(z_k + \bar{\lambda}_k\Delta z_k, (1-\sigma\bar{\lambda}_k)\mu_k)\| &> (1-\sigma\bar{\lambda}_k)\beta\mu_k \\ &\geq (1-\sigma\bar{\lambda}_k)\| \Phi(z_k, \mu_k)\|.\end{aligned} \tag{16}$$

Since $\Delta\mu_k = -\sigma\mu_k$ by (6), (16) implies that

$$\frac{\| \Phi(z_k + \bar{\lambda}_k\Delta z_k, \mu_k + \bar{\lambda}_k\Delta\mu_k)\| - \| \Phi(z_k, \mu_k)\|}{\bar{\lambda}_k} \geq -\sigma\| \Phi(z_k, \mu_k)\|.$$

Taking the limit $k \to \infty$ in the last inequality and using $\bar{\lambda}_k \to 0$, we obtain

$$\Phi(z^*, \mu^*)^T\Phi'(z^*, \mu^*)\begin{pmatrix}\Delta z^* \\ \Delta\mu^*\end{pmatrix} \geq -\sigma\| \Phi(z^*, \mu^*)\|^2. \tag{17}$$

By (6), we have

$$\Phi'(z^*, \mu^*)\begin{pmatrix}\Delta z^* \\ \Delta\mu^*\end{pmatrix} = -\Phi(z^*, \mu^*).$$

Substituting this relation into (17) and using the fact $\sigma \in (0,1)$, we have

$$\| \Phi(z^*, \mu^*)\| = 0. \tag{18}$$

On the other hand, taking the limit $k \to \infty$ in (15) and using $\bar{\lambda}_k \to 0$ yield

$$\| \Phi(z^*, \mu^*)\| \geq \beta\mu^* > 0,$$

which contradicts (18).

This completes the proof.

As a consequence of Theorem 3.1, we have the following main global convergence result for Algorithm 2.1.

Theorem 3.2 Suppose that A has full row rank. Then any accumulation point of the iteration sequence $\{(z_k, \mu_k)\}$ generated by Algorithm 2.1 is a solution of $H(z, \mu) = 0$.

Proof Let (z^*, μ^*) be an accumulation point of the sequence $\{(z_k, \mu_k)\}$. Without loss of generality, we may assume that

$$\lim_{k\to\infty}(z_k, \mu_k) = (z^*, \mu^*).$$

On account of Lemma 2.3 (iv), we obtain

$$\| \Phi(z_k, \mu_k)\| \le \beta\mu_k. \tag{19}$$

It follows from Lemma 2.1 that Φ is a continuous function in both z and μ. Then taking the limit $k \to \infty$ in (19) and using Theorem 3.1, we have

$$\| \Phi(z^*, 0)\| = \| \Phi(z^*, \mu^*)\| \le \beta\mu^* = 0.$$

This completes the proof.

4. Acknowledgements

This research was partially supported by the National Natural Science Foundation of China (Grant No. 60574075, 60674108).

5. References

[1] Alizadeh F. and Goldfarb D. Second-order cone programming。 Math. Program 2003; 95: 3-51.

[2] Chen B. and Xiu N. A global linear and local quadratic non-interior continuation method for nonlinear complementarity problems based on Chen-Mangasarian smoothing functions. SIAM J. Optim. 1999; 9: 605-623.

[3] Faraut J. and Korányi A. Analysis on Symmetric Cones. Oxford University Press, 1994.

[4] Fukushima M., Luo Z. Q. and Tseng P. Smoothing functions for second-order-cone complementarity problems. SIAM J. Optim. 2001; 12: 436-460.

[5] Liu F. and Huang T. The preconditioned simultaneous displacement method for linear complementarity problems. Journal of Information and Computing Science 2006; 3: 125-128.

[6] Lobo M. S., Vandenberghe L., Boyd S. and Lebret H. Applications of second-order cone programming. Linear Algebra Appl. 1998; 284: 193-228.

[7] Qi L., Sun D. and Zhou G. A new look at smoothing Newton methods for nonlinear complementarity problems and box constrained variational inequalities. Math. Program 2000; 87: 1-35.

[8] Tseng P. Analysis of a non-interior continuation method based on Chen-Mangasarian smoothing functions for complementarity problems, in Reformulation-Nonsmooth, Piecewise Smooth, Semismooth and Smoothing Methods, edited by Fukushima L.and Qi L. Kluwer Academic Publishers, Boston, MA, 1999; 381-404.