A Matrix Splitting Method for Affine Second-Order Cone Complementarity Problems

Guidance

Professor	Masao FUKUSHIMA
Assistant Professor	Nobuo YAMASHITA

Takahiro YAMAGUCHI

2001 Graduate Course

in

Department of Applied Mathematics and Physics

Graduate School of Informatics

Kyoto University

February 2003

Abstract

The affine Second-Order Cone Complementarity Problem (SOCCP) is a wide class of problems that contains the Linear Complementarity Problem (LCP) as a secial case. The purpose of this paper is to propose an iterative method for the symmetric affine SOCCP based on the idea of matrix splitting. Originally, matrix splitting methods have been developed for the solution of the system of linear equations and have subsequently been extended to the linear complementarity problem (LCP) and the affine variational inequality problem. In this paper, we first give conditions under which the matrix splitting method converges to a solution of the affine SOCCP. We then present, as a particular realization of the matrix splitting method, the block successive overrelaxation (SOR) method for the affine SOCCP involving a positive definite matrix and establish its convergence. Finally, we report some numerical results with the proposed algorithm.

Contents

1 Introduction 1

2 Basic Algorithm 1

3 Convergence Theory 2

4 Block SOR Method 6

5 Solving Subproblems 9

6 Numerical Results 14

7 Conclusion 17

1 Introduction

The l-dimensional *second order cone* (SOC) is defined by

$$\mathcal{K}^l = \{(x_1, x_2) \in \mathbf{R} \times \mathbf{R}^{l-1} \mid x_1 \geq \|x_2\|\},$$

where $\|\cdot\|$ denotes the Euclidean norm. In particular, if $l = 1$, $\mathcal{K}^1$ is the set of nonnegative reals $\mathbf{R}_+$.

The complementarity problem over SOCs that involves an affine function is called the *second-order cone complementarity problem* (SOCCP):

$$\begin{aligned} &\text{Find} \quad z \in \mathbf{R}^n \\ &\text{such that} \quad z \in \mathcal{K}, \quad Mz + q \in \mathcal{K}, \quad z^{\mathrm{T}}(Mz + q) = 0, \end{aligned} \tag{1.1}$$

where $\mathcal{K} \subset \mathbf{R}^n$ is the Cartesian product of SOCs, i.e.,

$$\mathcal{K} = \mathcal{K}^{n_1} \times \mathcal{K}^{n_2} \times \cdots \times \mathcal{K}^{n_m}$$

with $n = n_1 + n_2 + \cdots + n_m$, $M \in \mathbf{R}^{n \times n}$ is a given matrix, $q \in \mathbf{R}^n$ is a given vector, and the superscript T denotes transpose. We denote the above SOCCP by SOCCP$(q, M, \mathcal{K})$. Throughout the paper, we consider the case where M is *symmetric*. The affine SOCCP contains the linear complementarity problem (LCP) as a special case where $n_1 = n_2 = \cdots = n_m = 1$, and $m = n$.

Recently, smoothing methods for the nonlinear complementarity problem [1, 2, 3] have been extended to the general SOCCP [7]. The purpose of this paper is to propose an iterative method for the affine SOCCP that uses the idea of matrix splitting. Originally, matrix splitting methods have been designed for the solution of the system of linear equations [9]. Several splitting schemes such as Jacobi, Gauss-Seidel, and the successive overrelaxation (SOR) have been suggested and various parallel algorithms have been developed on the basis of matrix splitting [14]. Subsequently, those methods have been extended to the linear complementarity problem (LCP) and the affine variational inequality problem [5, 11, 12, 13, 16].

The paper is organized as follows: In Section 2, we describe the basic algorithm of the matrix splitting method. In Section 3, we consider conditions for the algorithm to be convergent. In Section 4, we present the block SOR method as a particular realization of the basic splitting algorithm of Section 2. In Section 5 we give a concrete procedure to solve subproblems of the block SOR method. In Section 6, we report numerical results with the proposed method. Finally, we conclude the paper in Section 7.

2 Basic Algorithm

In this section, we describe the basic algorithm of the matrix splitting method for the affine SOCCP. The algorithm is a natural extension of the matrix splitting method for the LCP [5].

Let the matrix M be represented as the sum of two matrices $B \in \mathbf{R}^{n \times n}$ and $C \in \mathbf{R}^{n \times n}$, that is,

$$M = B + C. \tag{2.1}$$

The pair (B, C) is called a splitting of M. The basic algorithm of the matrix splitting method for the SOCCP is stated as follows:

Algorithm 2.1

Step 1. Choose a splitting (B, C) of M. Choose an initial point $z^0 \in \mathcal{K}$ and a sufficiently small constant $\varepsilon > 0$, and set $\nu := 0$.

Step 2. Solve the following affine SOCCP:

$$\begin{aligned} \text{Find} \quad & z \in \mathbf{R}^n \\ \text{such that} \quad & z \in \mathcal{K},\ Bz + q^\nu \in \mathcal{K},\ z^{\mathrm{T}}(Bz + q^\nu) = 0, \end{aligned} \tag{2.2}$$

where

$$q^\nu := q + Cz^\nu. \tag{2.3}$$

Let $z^{\nu+1}$ be a solution of problem (2.2).

Step 3. If $\|z^{\nu+1} - z^\nu\| \leq \varepsilon$, terminate. Otherwise, return to Step 2 with ν replaced by $\nu + 1$.

It is particularly important to choose a splitting (B, C) so that SOCCP (2.2) can be solved efficiently and the sequence $\{z^\nu\}$ generated by the algorithm converges to a solution of SOCCP (1.1). If SOCCP$(q, M, \mathcal{K})$ has a solution for any $q \in \mathbf{R}^n$, M is called a $\mathcal{K}$-Q-matrix analogously to a Q-matrix in the LCP theory [5]. If B is a $\mathcal{K}$-Q-matrix, then the splitting (B, C) is called a $\mathcal{K}$-Q-splitting. Therefore, SOCCP (2.2) always has a solution if (B, C) is a $\mathcal{K}$-Q-splitting. In addition, if $B - C$ is positive (semi-)definite, then the splitting (B, C) is said to be (weakly) regular. The regularity of the splitting (B, C) plays an important role in discussing the convergence property of Algorithm 2.1.

3 Convergence Theory

We consider conditions under which the algorithm described in the previous section converges to a solution of the affine SOCCP (1.1). Throughout this section, the parameter ε used to check the termination criterion of Algorithm 2.1 is assumed to be 0 .

Consider the following quadratic programming problem on SOCs:

$$\begin{aligned} \text{minimize} \quad & f(z) := \frac{1}{2} z^{\mathrm{T}} M z + q^{\mathrm{T}} z \\ \text{subject to} \quad & z \in \mathcal{K}. \end{aligned} \tag{3.1}$$

Since M is a symmetric matrix, SOCCP (1.1) serves as the KKT conditions of problem (3.1). Hence an optimal solution z^* of problem (3.1) is a solution of SOCCP (1.1), since problem (3.1) apparently satisfies Slater's constraint qualification. We will establish convergence of Algorithm 2.1 by means of the argument based on the relationship between SOCCP (1.1) and problem (3.1).

First, we give a lemma that relates the SOCCP to an equivalent *variational inequality problem* (VI).

Lemma 3.1 $z^* \in \mathcal{K}$ is a solution of SOCCP (1.1) if and only if z^* is a solution of the following VI:

$$\begin{aligned} &\text{Find} \quad z \in \mathcal{K} \\ &\text{such that} \quad (y-z)^{\mathrm{T}}(Mz+q) \geq 0 \ \ \forall y \in \mathcal{K}. \end{aligned}$$

Proof. For a closed convex cone $X \subseteq \mathbf{R}^n$ and a function $F : \mathbf{R}^n \to \mathbf{R}^n$, consider the generalized complementarity problem GCP(X, F):

$$\begin{aligned} &\text{Find} \quad x \in X \\ &\text{such that} \quad F(x) \in X^*, \quad F(x)^{\mathrm{T}}x = 0 \end{aligned}$$

and VI(X, F):

$$\begin{aligned} &\text{Find} \quad x \in X \\ &\text{such that} \quad (y-x)^{\mathrm{T}}F(x) \geq 0, \quad \forall y \in X, \end{aligned}$$

where X^* is the dual cone of X defined by

$$X^* = \{\xi \in \mathbf{R}^n \mid \xi^{\mathrm{T}}\eta \geq 0 \ \forall \eta \in X\}.$$

Then, $x^* \in X$ is a solution of GCP(X, F) if and only if x^* is a solution of VI(X, F) [10, Proposition 2.1]. Since $\mathcal{K}^{n_i} = (\mathcal{K}^{n_i})^*$ for the SOC $\mathcal{K}^{n_i}$[6], the following relation holds:

$$\begin{aligned} \mathcal{K}^* &= (\mathcal{K}^{n_1} \times \mathcal{K}^{n_2} \times \cdots \times \mathcal{K}^{n_m})^* \\ &= (\mathcal{K}^{n_1})^* \times (\mathcal{K}^{n_2})^* \times \cdots \times (\mathcal{K}^{n_m})^* = \mathcal{K}. \end{aligned}$$

By replacing x, X and $F(x)$ in the above problems with z, $\mathcal{K}$ and $Mz+q$, respectively, we obtain the desired result. □

The following lemma gives a sufficient condition for a sequence $\{f(z^\nu)\}$ generated by Algorithm 2.1 to be nonincreasing.

Lemma 3.2 Let (B, C) be a weakly regular $\mathcal{K}$-Q-splitting of the symmetric matrix M. Let $\{z^\nu\}$ be a sequence generated by Algorithm 2.1. Then, we have for each ν

$$f(z^\nu) - f(z^{\nu+1}) \geq \frac{1}{2}(z^\nu - z^{\nu+1})^{\mathrm{T}}(B-C)(z^\nu - z^{\nu+1}) \geq 0. \tag{3.2}$$

In particular, if (B, C) is a regular $\mathcal{K}$-Q-splitting, then the second inequality in (3.2) holds strictly whenever $z^\nu \neq z^{\nu+1}$, and moreover $f(z^\nu) = f(z^{\nu+1})$ if and only if $z^\nu = z^{\nu+1}$.

Proof. By (2.1) and (2.3), we have

$$\begin{aligned} f(z^\nu) - f(z^{\nu+1}) &= (z^\nu - z^{\nu+1})^{\mathrm{T}}(q + Mz^{\nu+1}) + \frac{1}{2}(z^\nu - z^{\nu+1})^{\mathrm{T}}M(z^\nu - z^{\nu+1}) \\ &= (z^\nu - z^{\nu+1})^{\mathrm{T}}(q + (B+C)z^{\nu+1}) \\ &\quad + \frac{1}{2}(z^\nu - z^{\nu+1})^{\mathrm{T}}(B-C)(z^\nu - z^{\nu+1}) + (z^\nu - z^{\nu+1})^{\mathrm{T}}C(z^\nu - z^{\nu+1}) \\ &= (z^\nu - z^{\nu+1})^{\mathrm{T}}(q^\nu + Bz^{\nu+1}) + \frac{1}{2}(z^\nu - z^{\nu+1})^T(B-C)(z^\nu - z^{\nu+1}). \end{aligned}$$

Since $z^{\nu+1}$ is a solution of SOCCP (2.2), by Lemma 3.1, we obtain

$$(z^{\nu} - z^{\nu+1})^{\mathrm{T}}(q^{\nu} + Bz^{\nu+1}) \geq 0.$$

Therefore, we have

$$f(z^{\nu}) - f(z^{\nu+1}) \geq \frac{1}{2}(z^{\nu} - z^{\nu+1})^{\mathrm{T}}(B - C)(z^{\nu} - z^{\nu+1}).$$

The second inequality in (3.2) holds from the positive semi-definiteness of $B - C$. The last assertion of the lemma is obtained by the fact that the regularity of (B, C) implies the positive definiteness of $B - C$. □

By the above lemmas, we obtain the following theorem concerning accumulation points of the sequence generated by Algorithm 2.1.

Theorem 3.1 Let (B, C) be a regular $\mathcal{K}$-Q-splitting of the symmetric matrix M. Then, any accumulation point of the sequence $\{z^{\nu}\}$ generated by Algorithm 2.1 is a solution of SOCCP (1.1).

Proof. Let $\tilde{z}$ be an arbitrary accumulation point of the sequence $\{z^{\nu}\}$ generated by Algorithm 2.1 and $\{z^{\nu_i}\}$ be a subsequence of $\{z^{\nu}\}$ converging to $\tilde{z}$. Then, by the continuity of f, the sequence $\{f(z^{\nu_i})\}$ converges to $f(\tilde{z})$. In addition, the entire sequence $\{f(z^{\nu})\}$ is bounded below, since $\{f(z^{\nu})\}$ is nonincreasing from Lemma 3.2 and the subsequence $\{f(z^{\nu_i})\}$ converges. Consequently, the sequence $\{f(z^{\nu})\}$ itself converges. This fact, along with the positive definiteness of $B - C$ and the inequality (3.2), yields that $\{z^{\nu} - z^{\nu+1}\}$ converges to 0 . Hence, the sequence $\{z^{\nu_i+1}\}$ also converges to $\tilde{z}$. Since z^{ν_i+1} satisfies

$$\begin{aligned}
&z^{\nu_i+1} \in \mathcal{K} \\
&Bz^{\nu_i+1} + Cz^{\nu_i} + q \in \mathcal{K} \\
&(z^{\nu_i+1})^{\mathrm{T}}(Bz^{\nu_i+1} + Cz^{\nu_i} + q) = 0,
\end{aligned}$$

passing to the limit $\nu_i \to \infty$ reveals that $\tilde{z}$ is a solution of SOCCP (1.1). □

This theorem indicates that, if a sequence $\{z^{\nu}\}$ generated by Algorithm 2.1 has an accumulation point, then it is a solution of SOCCP (1.1). However, the theorem says nothing about the existence of an accumulation point. In what follows, we consider conditions under which Algorithm 2.1 is actually convergent. The following definition gives a natural extension of the (strict) copositivity in the LCP theory.

Definition 3.1 A matrix $M \in \mathbf{R}^{n \times n}$ is said to be (strictly) $\mathcal{K}$-copositive if $z^{\mathrm{T}}Mz \geq (>)\ 0$ for all $z \in \mathcal{K}$.

Next, we give a lemma that relates the boundedness of the quadratic function f to the strict copositivity of matrix M.

Lemma 3.3 The matrix M is strictly $\mathcal{K}$-copositive if and only if the quadratic function f is bounded below on $\mathcal{K}$ for any $q \in \mathbf{R}^n$.

Proof. First, we show the 'if' part by contradiction. Suppose that M is not strictly $\mathcal{K}$-copositive. Then, there exists a $\bar{z} \in \mathcal{K}$ such that $\bar{z}^{\mathrm{T}} M \bar{z} < 0$. Moreover, we have $\alpha \bar{z} \in \mathcal{K}$ for any $\alpha > 0$ since $\mathcal{K}$ is a cone. Then, we can easily show that $f(\alpha \bar{z}) \to -\infty$ as $\alpha \to \infty$.

Next, we show the 'only if' part. Let M be strictly $\mathcal{K}$-copositive. Then we have

$$\sigma := \min_{\substack{\|e\|=1 \\ e \in \mathcal{K}}} e^{\mathrm{T}} M e > 0,$$

that is, we have for any $z \in \mathcal{K}$

$$z^{\mathrm{T}} M z \geq \sigma \|z\|^2. \tag{3.3}$$

Therefore, we have for any $z \in \mathcal{K}$

$$\begin{aligned} f(z) = \frac{1}{2} z^{\mathrm{T}} M z + q^{\mathrm{T}} z &\geq \frac{1}{2} \sigma \|z\|^2 - \|q\| \|z\| \\ &= \frac{1}{2} \sigma \left(\|z\| - \frac{1}{\sigma} \|q\| \right)^2 - \frac{1}{2\sigma} \|q\|^2, \end{aligned}$$

where the inequality follows from (3.3) and the Cauchy-Schwarz inequality. This implies that f is bounded below on $\mathcal{K}$. □

The following lemma refers to the boundedness of a sequence generated by Algorithm 2.1.

Lemma 3.4 Let (B, C) be a regular $\mathcal{K}$-Q-splitting of the symmetric matrix M. If M is strictly $\mathcal{K}$-copositive, then any sequence $\{z^\nu\}$ generated by Algorithm 2.1 is bounded.

Proof. Since the sequence $\{f(z^\nu)\}$ is bounded below and nonincreasing by Lemmas 3.2 and 3.3, $\{f(z^\nu)\}$ converges. Moreover, by (3.2), $\{z^\nu - z^{\nu+1}\}$ converges to 0. For the contradiction purpose, we assume that the sequence $\{z^\nu\}$ is not bounded. Without loss of generality, we may assume that $\|z^\nu\| \to \infty$. Since the sequence $\{z^\nu / \|z^\nu\|\}$ is bounded, it has an accumulation point $\tilde{z} \in \mathcal{K}$ such that $\|\tilde{z}\| = 1$. Let $\{z^{\nu_i} / \|z^{\nu_i}\|\}$ be a subsequence converging to $\tilde{z}$. Since z^{ν_i} is a solution of SOCCP (2.2) for $\nu = \nu_i - 1$, we have from (2.1) and (2.3) that

$$(z^{\nu_i})^{\mathrm{T}} (q + C(z^{\nu_i - 1} - z^{\nu_i}) + M z^{\nu_i}) = 0.$$

Dividing the above equality by $\|z^{\nu_i}\|^2$ and passing to the limit $\nu_i \to \infty$, we deduce that $\tilde{z}^T M \tilde{z} = 0$, which contradicts the assumption of strict $\mathcal{K}$-copositivity. Consequently, the sequence $\{z^\nu\}$ must be bounded. This completes the proof. □

By using the above lemmas, we establish the main theorem in this section.

Theorem 3.2 Let M be symmetric matrix, and q be an arbitrary vector. If M is strictly $\mathcal{K}$-copositive, then, for any initial point $z^0 \in \mathcal{K}$, the sequence $\{z^\nu\}$ generated by Algorithm 2.1 with regular $\mathcal{K}$-Q-splitting (B, C) is bounded, and its arbitrary accumulation point is a solution of SOCCP$(q, M, \mathcal{K})$. Conversely, if there exists a regular $\mathcal{K}$-Q-splitting of a symmetric matrix M, such that, for any $q \in \mathbf{R}^n$, the algorithm converges to a solution of SOCCP$(q, M, \mathcal{K})$ for any initial point $z^0 \in \mathcal{K}$, then M is strictly $\mathcal{K}$-copositive.

Proof. The first half of the theorem readily follows from Theorem 3.1 and Lemma 3.4.

We show the last half of the theorem. First, we show that SOCCP$(0, M, \mathcal{K})$ has the unique solution $z = 0$. Suppose to the contrary that there exists a nonzero solution $\tilde{z}$ of SOCCP$(0, M, \mathcal{K})$. Consider applying Algorithm 2.1 to SOCCP$(-B\tilde{z}, M, \mathcal{K})$ with regular $\mathcal{K}$-Q-splitting (B, C) and the initial point $z^0 = 0$. Then, it is not difficult to see that subproblem (2.2) has the solution $z^{\nu+1} = (\nu+1)\tilde{z}$ for each ν. This means that Algorithm 2.1 applied to SOCCP$(-B\tilde{z}, M, \mathcal{K})$ generates an unbounded sequence. However, this contradicts the fact shown in the first half of the theorem. Hence, SOCCP$(0, M, \mathcal{K})$ has the unique solution $z = 0$.

Now we assume that M is not strictly $\mathcal{K}$-copositive. Then, there exists a vector $\hat{z} \in \mathcal{K} \setminus \{0\}$ such that $\hat{z}^{\mathrm{T}} M \hat{z} \leq 0$. Since $\hat{z} \neq 0$, it is not a solution of SOCCP$(0, M, \mathcal{K})$ by the fact shown above. Consider the sequence $\{z^\nu\}$ generated by the algorithm applied to SOCCP$(0, M, \mathcal{K})$ with the initial point $z^0 = \hat{z}$. Since (B, C) is a regular splitting, by Lemma 3.2, we have

$$\tilde{f}(z^1) < \tilde{f}(z^0) \leq 0,$$

where $\tilde{f}(z) := \frac{1}{2} z^{\mathrm{T}} M z$. By assumption, the sequence $\{z^\nu\}$ is bounded and any accumulation point is a solution of SOCCP$(0, M, \mathcal{K})$. Since $z = 0$ is the unique solution of SOCCP$(0, M, \mathcal{K})$, it follows that $\{z^\nu\}$ converges to 0. Consequently, by Lemma 3.2 again, we deduce

$$0 = \lim_{\nu \to \infty} \tilde{f}(z^\nu) \leq \tilde{f}(z^1) < \tilde{f}(z^0) \leq 0,$$

which is a contradiction. This completes the proof. □

4 Block SOR Method

In the previous section, we have shown that Algorithm 2.1 with any regular $\mathcal{K}$-Q-splitting (B, C) converges under the assumption of strict $\mathcal{K}$-copositivity of M. In this section, we present the block successive overrelaxation (block SOR) method for solving affine SOCCPs by extending the corresponding method for LCPs [5]. In particular, we give conditions for a splitting (B, C) used in the block SOR method to be a regular $\mathcal{K}$-Q-splitting. In this and the next sections, we assume that M is a symmetric positive definite matrix. Evidently, any positive definite matrix is strictly $\mathcal{K}$-copositive.

First, we give an explicit expression of the splitting (B, C). To simplify the notation, for any symmetric matrix

$$A = \begin{pmatrix} a_1 & a_2^{\mathrm{T}} \\ a_2 & A_3 \end{pmatrix}, \tag{4.1}$$

we denote

$$\overline{A} := \begin{pmatrix} a_1 & 0^{\mathrm{T}} \\ 0 & A_3 \end{pmatrix} \tag{4.2}$$

where $a_1 \in \mathbf{R}$. Now let us partition the matrix M as follows:

$$M = \begin{pmatrix} M_{11} & M_{12} & \cdots & M_{1m} \\ M_{21} & M_{22} & & M_{2m} \\ \vdots & & \ddots & \vdots \\ M_{m1} & M_{m2} & \cdots & M_{mm} \end{pmatrix},$$

where $M_{ij} \in \mathbf{R}^{n_i \times n_j}$. For each block diagonal entry M_{ii}, we define $\overline{M}_{ii}$ as in (4.2). Using these notations, we let the splitting (B, C) take the form

$$B = \begin{pmatrix} B_{11} & & & & \text{\huge 0} \\ M_{21} & B_{22} & & & \\ M_{31} & M_{32} & \ddots & & \\ \vdots & & \ddots & \ddots & \\ M_{m1} & \cdots & \cdots & M_{m,m-1} & B_{mm} \end{pmatrix}, \quad C = M - B, \tag{4.3}$$

where B_{ii} is given by $B_{ii} := \omega_i^{-1} \overline{M}_{ii}$ $(i = 1, \ldots, m)$ with a positive scalar ω_i.

Since B is chosen to be a block lower triangular matrix, we can solve SOCCP (2.2) successively as follows: Let

$$z = \begin{pmatrix} z_1 \\ \vdots \\ z_m \end{pmatrix}, \quad q^\nu = \begin{pmatrix} q_1^\nu \\ \vdots \\ q_m^\nu \end{pmatrix}$$

with $z_i \in \mathbf{R}^{n_i}$ and $q_i^\nu \in \mathbf{R}^{n_i}$ $(i = 1, \ldots, m)$. Then, SOCCP (2.2) is equivalent to the problem of finding $z \in \mathbf{R}^n$ such that

$$\begin{aligned} & z_i \in \mathcal{K}^{n_i}, \quad B_{ii} z_i + r_i^\nu \in \mathcal{K}^{n_i}, \quad i = 1, \ldots, m, \\ & \sum_{i=1}^m z_i^{\mathrm{T}} (B_{ii} z_i + r_i^\nu) = 0, \end{aligned} \tag{4.4}$$

where

$$r_i^\nu := \begin{cases} q_1^\nu & (i = 1) \\ \displaystyle\sum_{j=1}^{i-1} M_{ij} z_j + q_i^\nu & (i \geq 2) \end{cases}.$$

Noticing that $z_i^{\mathrm{T}}(B_{ii} z_i + r_i^\nu) \geq 0$ holds from $z_i \in \mathcal{K}^{n_i}$ and $B_{ii} z_i + r_i^\nu \in \mathcal{K}^{n_i}$, we can decompose (4.4) into the following SOC complementarity conditions:

$$z_i^{\mathrm{T}}(B_{ii} z_i + r_i^\nu) = 0, \quad z_i \in \mathcal{K}^{n_i}, \quad B_{ii} z_i + r_i^\nu \in \mathcal{K}^{n_i}, \quad i = 1, \ldots, m. \tag{4.5}$$

We can solve (4.5) for z_i recursively from $i = 1$ to $i = m$, by regarding $z_1, \ldots, z_{i-1}$ and r_i^ν as known constants. Moreover, each SOCCP (4.5) can be solved efficiently by using the particular structure of $B_{ii} = \omega_i^{-1} \overline{M}_{ii}$ (as will be shown in the next section).

Next we consider conditions for the splitting (B, C) given by (4.3) to be a regular $\mathcal{K}$-Q-splitting. To this end, we give the following two lemmas.

Lemma 4.1 Let $A \in \mathbf{R}^{n\times n}$ be a symmetric positive definite matrix given by (4.1) and $\overline{A} \in \mathbf{R}^{n\times n}$ be given by (4.2). Let

$$G = \omega^{-1}\overline{A}, \quad H = A - \omega^{-1}\overline{A}, \tag{4.6}$$

where $0 < \omega \leq 1$. Then (G, H) is a regular splitting of A.

Proof. Note that

$$\begin{aligned} G - H &= 2\omega^{-1}\overline{A} - A \\ &= \begin{pmatrix} (2\omega^{-1} - 1)a_1 & -a_2^{\mathrm{T}} \\ -a_2 & (2\omega^{-1} - 1)A_3 \end{pmatrix}. \end{aligned} \tag{4.7}$$

Let $\gamma = 2\omega^{-1} - 1$. Then the Schur complement of $G - H$ with respect to γa_1 is given by

$$\begin{aligned} \gamma A_3 - \gamma^{-1}a_1^{-1}a_2a_2^{\mathrm{T}} &= \gamma^{-1}\{\gamma^2 A_3 - a_1^{-1}a_2a_2^{\mathrm{T}}\} \\ &= \gamma^{-1}\{(A_3 - a_1^{-1}a_2a_2^{\mathrm{T}}) + (\gamma^2 - 1)A_3\}. \end{aligned}$$

Note that the symmetric matrix A is positive definite if and only if $a_1 > 0$ and $A_3 - a_1^{-1}a_2a_2^{\mathrm{T}}$ is positive definite [15]. In addition, the positive definiteness of A implies that A_3 is also positive definite. Therefore, we can deduce that $G - H$ is positive definite if $\gamma > 0$ and $\gamma^2 - 1 \geq 0$. Since the last conditions are equivalent to $0 < \omega \leq 1$, the proof is complete. □

Lemma 4.2 Let $N \in \mathbf{R}^{n\times n}$ be represented as

$$N = D + L - L^{\mathrm{T}},$$

where the matrix D consists of the block diagonal submatrices of N and the matrix L consists of the block lower triangular submatrices of N. Then, N is positive definite if and only if D is positive definite.

Proof. For any vector $x \in \mathbf{R}^n$, we have

$$\begin{aligned} x^{\mathrm{T}}Nx &= x^{\mathrm{T}}(D - L + L^{\mathrm{T}})x \\ &= x^{\mathrm{T}}Dx, \end{aligned}$$

which yields the assertion. □

Using these lemmas, we can prove the main theorem in this section.

Theorem 4.1 Let the splitting (B, C) of M be given by (4.3) with $0 < \omega_i \leq 1$ $(i = 1, \ldots, m)$. Then the following statements are true:

(a) The splitting (B, C) is regular.

(b) B_{ii} is positive definite for each i.

(c) B is a $\mathcal{K}$-Q-matrix.

Proof. First, we show (a). Note that $B-C$ can be rewritten as

$$\begin{aligned} B-C &= 2B-M \\ &= \operatorname{diag}\{2B_{ii}-M_{ii}\}_{i=1}^{m}+L-L^{\mathrm{T}}, \end{aligned}$$

where $\operatorname{diag}\{2B_{ii}-M_{ii}\}_{i=1}^{m}$ denotes the block diagonal matrix whose block diagonal elements are $2B_{ii}-M_{ii}$, and $L\in\mathbf{R}^{n\times n}$ is the block lower triangular part of M. Since $B-C$ is represented as the sum of a block diagonal matrix and a block skew-symmetric matrix, by Lemma 4.2, $B-C$ is positive definite if and only if its block diagonal part $\operatorname{diag}\{2B_{ii}-M_{ii}\}_{i=1}^{m}$ is positive definite. By Lemma 4.1 and $0<\omega_i\le 1$ $(i=1,\ldots,m)$, $2B_{ii}-M_{ii}=2\omega_i\overline{M}_{ii}-M_{ii}$ is positive definite for each i, that is, $\operatorname{diag}\{2B_{ii}-M_{ii}\}_{i=1}^{m}$ is also positive definite. Hence (B,C) is a regular splitting.

Second, we show (b). Any block diagonal submatrx M_{ii} of M is positive definite because of the positive definiteness of M. In addition, this implies that $B_{ii}=\omega_i^{-1}\overline{M}_{ii}$ is also positive definite.

Finally, we show (c). Since SOCCP (2.2) is equivalent to SOCCP (4.5), it is sufficient to show that B_{ii} is a $\mathcal{K}^{n_i}$-Q-matrix for each i. For any vector $q\in\mathbf{R}^{n_i}$, consider the minimization problem

$$\begin{aligned} &\text{minimize} \quad f_i(z_i):=\frac{1}{2}z_i^{\mathrm{T}}B_{ii}z_i+q^{\mathrm{T}}z_i \\ &\text{subject to} \quad z_i\in\mathcal{K}^{n_i}. \end{aligned} \tag{4.8}$$

By the positive definiteness of B_{ii}, the minimization problem (4.8) has a unique optimal solution z_i^*. Since problem (4.8) is equivalent to $\mathrm{SOCCP}(q,B_{ii},\mathcal{K}^{n_i})$, z_i^* is also a solution of $\mathrm{SOCCP}(q,B_{ii},\mathcal{K}^{n_i})$. This proves (c). □

By this theorem and Theorem 3.2, the convergence of Algorithm 2.1 is guaranteed if the splitting (B,C) is given by (4.3) with $0<\omega_i\le 1$ for all i.

5 Solving Subproblems

In the previous section, we have shown that SOCCP (2.2) can be decomposed into m subproblems by choosing the splitting (B,C) as in (4.3). In this secton, we present a method of solving these subproblems efficiently. Throughout this section, we write z, B, r and n for z_i, B_{ii}, r_i^ν and n_i, respectively. This simplification will not cause any confusion. Then we can rewrite subproblem (4.5) as

$$z^{\mathrm{T}}(Bz+r)=0, \quad z\in\mathcal{K}^n, \quad Bz+r\in\mathcal{K}^n. \tag{5.1}$$

Note that problem (5.1) has a unique solution since B is positive definite by Theorem 4.1 (b).

The following three cases are possible for a solution z^* of problem (5.1):

(i) $z^*=0$,

(ii) $z^*\in\operatorname{int}\mathcal{K}^n$,

(iii) $z^*\in\operatorname{bd}\mathcal{K}^n\setminus\{0\}$, where int $\mathcal{K}^n$ and bd $\mathcal{K}^n$ denote the interior and the boundary of $\mathcal{K}^n$, respectively. Since $z^* \in \mathcal{K}^n$, it is clear that no case other than three is possible for a solution of (5.1). To solve problem (5.1) efficiently, it will be helpful to detect which case applies to the solution z^*. To this end, we provide the following two propositions.

Proposition 5.1 $z^* = 0$ solves problem (5.1) if and only if $r \in \mathcal{K}^n$.

Proof. If $z^* = 0$ solves problem (5.1), then we have $Bz^* + r = r \in \mathcal{K}^n$. Conversely, if $r \in \mathcal{K}^n$, it is easily seen that $z^* = 0$ solves problem (5.1). □

Proposition 5.2 If $-B^{-1}r \in \text{int}\, \mathcal{K}^n$, then $z^* = -B^{-1}r$ solves problem (5.1).

Proof. Let $z^* = -B^{-1}r$. Then, we have

$$Bz^* + r = B(-B^{-1}r) + r = 0 \in \mathcal{K}^n,$$

and hence $(z^*)^{\mathrm{T}}(Bz^* + r) = 0$. Thus, $z^* = -B^{-1}r \in \mathcal{K}^n$ implies that z^* solves problem (5.1). □

Recall that, in this and the previous sections, we assume B is positive definite. Propositions 5.1 and 5.2 imply that, if $r \in \mathcal{K}^n$ or $-B^{-1}r \in \text{int}\, \mathcal{K}^n$, then we can readily calculate the solution of problem (5.1). Moreover, from the uniqueness of the solution, we must have case (iii) when neither $r \in \mathcal{K}^n$ nor $-B^{-1}r \in \text{int}\, \mathcal{K}^n$ holds.

Now, we describe a method of finding the solution of problem (5.1) when case (iii) holds. Note that we have $Bz^* + r \in \text{bd}\, \mathcal{K}^n$, since the solution z^* belongs to bd $\mathcal{K}^n \setminus \{0\}$ and the inner product of z^* and $Bz^* + r$ is equal to 0 . Thus, we can write z^* and $Bz^* + r$ as

$$z^* = \lambda \begin{pmatrix} 1 \\ w \end{pmatrix}, \tag{5.2}$$

$$Bz^* + r = \mu \begin{pmatrix} 1 \\ -w \end{pmatrix}, \tag{5.3}$$

where $\lambda > 0$, $\mu \geq 0$ and w is an $(n-1)$-dimensional vector such that $\|w\| = 1$. Recall that B is chosen such that

$$B = \begin{pmatrix} b_1 & 0 \\ 0 & B_2 \end{pmatrix}, \tag{5.4}$$

where $b_1 \in \mathbf{R}$ and $B_2 \in \mathbf{R}^{(n-1)\times(n-1)}$. We have from the positive definiteness of B that $b_1 > 0$ and B_2 is positive definite. Substituting (5.4) and (5.2) into (5.3), we have

$$\begin{pmatrix} \lambda b_1 + r_1 \\ \lambda B_2 w + r_2 \end{pmatrix} = \begin{pmatrix} \mu \\ -\mu w \end{pmatrix}. \tag{5.5}$$

Eliminating μ in (5.5), we have

$$\{r_1 I + \lambda(b_1 I + B_2)\}w = -r_2.$$

Since $b_1 > 0$ and B_2 is positive definite, $b_1 I + B_2$ is also positive definite, and hence nonsingular. Therefore, we may rewrite the above equation as

$$\{r_1(b_1 I + B_2)^{-1} + \lambda I\}w = -(b_1 I + B_2)^{-1} r_2. \tag{5.6}$$

Moreover, (5.5) together with $\lambda > 0$ and $\mu \geq 0$ yields

$$\lambda \geq \lambda_{\mathrm{L}} := \max\{0, -r_1/b_1\}. \tag{5.7}$$

We then have the following proposition.

Proposition 5.3 $r_1(b_1 I + B_2)^{-1} + \lambda I$ is positive definite for all $\lambda > \lambda_{\mathrm{L}}$.

Proof. Note that $(b_1 I + B_2)^{-1}$ is positive definite since $b_1 I + B_2$ is positive definite.

If $r_1 \geq 0$, then $\lambda_{\mathrm{L}} = 0$. Thus, $r_1(b_1 I + B_2)^{-1} + \lambda I$ is positive definite for all $\lambda > \lambda_{\mathrm{L}} = 0$.

If $r_1 < 0$, then $\lambda_{\mathrm{L}} = -r_1/b_1 > 0$. Moreover, we have

$$\begin{aligned} r_1(b_1 I + B_2)^{-1} + \lambda I &= r_1(b_1 I + B_2)^{-1} + \lambda b_1 (b_1 I + B_2)^{-1} + \lambda \left(I - b_1(b_1 I + B_2)^{-1}\right) \\ &= (r_1 + \lambda b_1)(b_1 I + B_2)^{-1} + \lambda (b_1 B_2^{-1} + I)^{-1}. \end{aligned}$$

Note that $r_1 + \lambda b_1 > 0$ for all $\lambda > \lambda_{\mathrm{L}} = -r_1/b_1 > 0$, and $b_1 B_2^{-1} + I$ is positive definite since $b_1 > 0$ and B_2^{-1} is positive definite. Hence, the matrix $r_1(b_1 I + B_2)^{-1} + \lambda I$ is positive definite for all $\lambda > \lambda_{\mathrm{L}} = -r_1/b_1 > 0$. This proves the proposition. □

By Proposition 5.3, when $\lambda > \lambda_{\mathrm{L}}$, we may rewrite (5.6) to represent w as a function of λ such that

$$\begin{aligned} w(\lambda) &= -\{r_1(b_1 I + B_2)^{-1} + \lambda I\}^{-1}(b_1 I + B_2)^{-1} r_2 \\ &= -(H + \lambda I)^{-1} g, \end{aligned} \tag{5.8}$$

where $H = r_1(b_1 I + B_2)^{-1}$ and $g = (b_1 I + B_2)^{-1} r_2$. We define the function $\psi : \mathbf{R} \to \mathbf{R}$ as

$$\psi(\lambda) := \|w(\lambda)\|^2 = \|H(\lambda)^{-1} g\|^2,$$

where $H(\lambda) = H + \lambda I$. Since $\|w\| = 1$, if we find λ satisfying (5.7) and

$$\psi(\lambda) = 1, \tag{5.9}$$

we obtain the solution z^* of problem (5.1) through (5.8). Here, instead of applying Newton's method to (5.9) directly, we will use a more efficient method that is reminiscent of an approach well-known in the trust region literature [4, Chapter 7]. Consider the function

$$\phi(\lambda) = \frac{1}{\sqrt{\psi(\lambda)}} - 1.$$

Since $H(\lambda)$ is positive definite for any $\lambda > \lambda_{\mathrm{L}}$ by Proposition 5.3, the function ϕ is well-defined for $\lambda > \lambda_{\mathrm{L}}$. Though the nonlinear equation

$$\phi(\lambda) = 0 \tag{5.10}$$

is equivalent to (5.9), (5.10) has some advantages for practical calculation since it is almost linear for $\lambda > \lambda_{\mathrm{L}}$. For example, we illustrate the graphs of $\psi(\lambda)$ and $\phi(\lambda)$ in Figures 1 and 2, where H and g are given by

$$H = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 2 & 0 & 0 \\ 0 & 0 & 3 & 0 \\ 0 & 0 & 0 & 4 \end{pmatrix}, \quad g = \begin{pmatrix} 1 \\ 1 \\ 1 \\ 1 \end{pmatrix}.$$

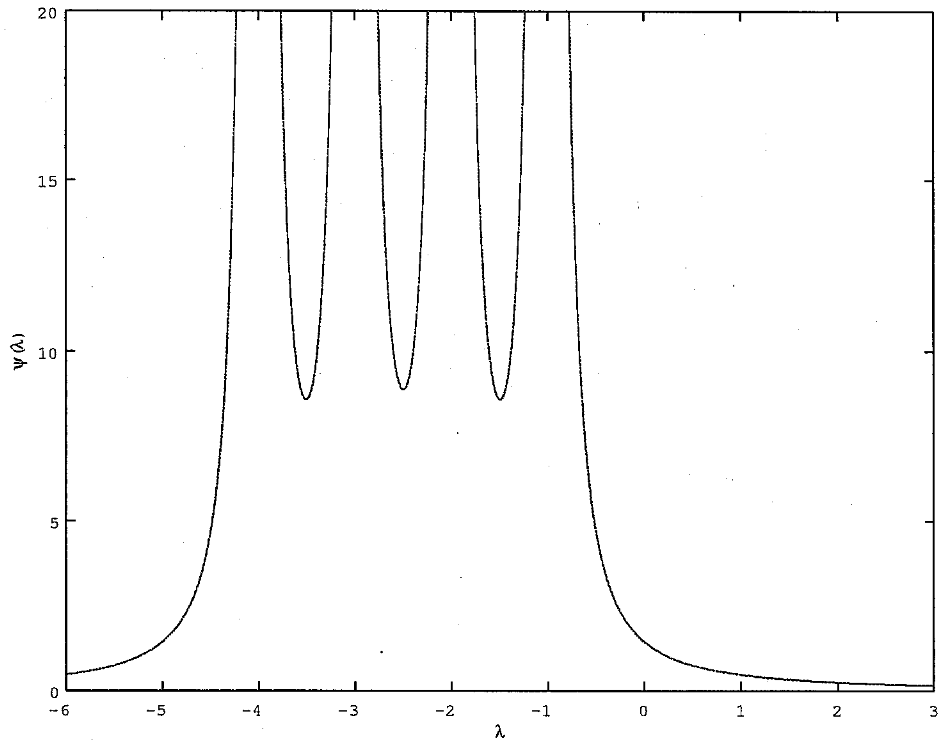

Figure 1: The graph of $\psi(\lambda)$

When applied to the nonlinear equation (5.10), Newton's method generates a sequence $\{\lambda^t\}$ using the formula

$$\lambda^{t+1} = \lambda^t - \frac{\phi(\lambda^t)}{\phi'(\lambda^t)}, \tag{5.11}$$

where the derivative ϕ' is obtained by

$$\begin{aligned} \phi'(\lambda) &= \left\{ \frac{1}{\|w(\lambda)\|} - 1 \right\}' \\ &= \frac{w(\lambda)^{\mathrm{T}} H(\lambda)^{-1} w(\lambda)}{\|w(\lambda)\|^3}. \end{aligned}$$

Since $H(\lambda)$ is positive definite for any $\lambda > \lambda_{\mathrm{L}}$, by using the Cholesky factorization $H(\lambda) = R(\lambda)R(\lambda)^{\mathrm{T}}$ with $R(\lambda)$ being upper triangular [8], we may rewrite the formula (5.11) as

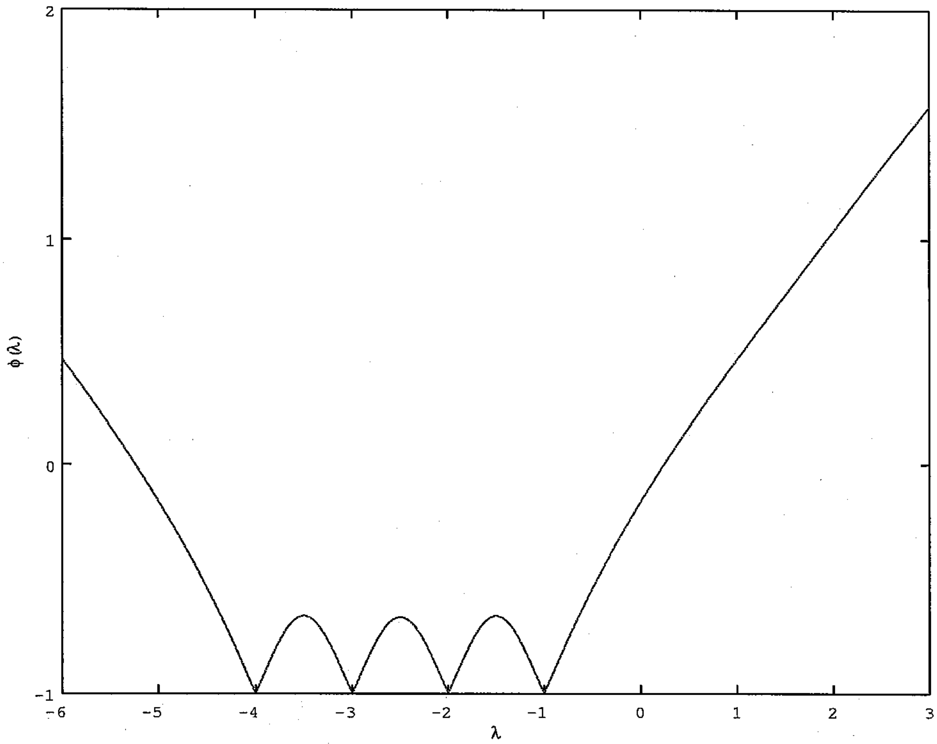

Figure 2: The graph of $\phi(\lambda)$

follows:

$$\lambda^{t+1} = \lambda^t - \frac{\phi(\lambda^t)}{\phi'(\lambda^t)} \tag{5.12}$$

$$= \lambda^t - \left(\frac{1}{\|w(\lambda^t)\|} - 1\right) \frac{\|w(\lambda^t)\|^3}{w(\lambda^t)^{\mathrm{T}}(R(\lambda^t)^{\mathrm{T}})^{-1}R(\lambda^t)^{-1}w(\lambda^t)} \tag{5.13}$$

$$= \lambda^t + (\|w(\lambda^t)\| - 1)\frac{\|w(\lambda^t)\|^2}{\|v(\lambda^t)\|^2}, \tag{5.14}$$

where $v(\lambda) = R(\lambda)^{-1}w(\lambda)$.

Newton's method is not generally guaranteed to be globally convergent. However, the soluton λ^* of (5.10) satisfies $\lambda^* \geq \lambda_{\mathrm{L}} \geq -\sigma_1$, where σ_1 is the smallest eigen value of H, and $\phi(\lambda)$ is strictly increasing and concave for $\lambda > -\sigma_1$ [4, Lemma 7.3.1]. Hence, if $\phi(\lambda_{\mathrm{L}}) = 0$, i.e., $\|w(\lambda_{\mathrm{L}})\| = 1$, then λ_{L} is the solution; and if $\phi(\lambda_{\mathrm{L}}) < 0$, then the Newton iteration (5.12) with initial point $\lambda^0 := \lambda_{\mathrm{L}}$ produces a sequence $\{\lambda^t\}$ such that $\phi(\lambda^t) < \phi(\lambda^{t+1}) \leq 0$, i.e., $\|w^t\| > \|w^{t+1}\| \geq 1$ is always satisfied and $\{\lambda^t\}$ converges to the solution λ^* of equation (5.10).

Summarizing the above arguments, we have the folowing algorithm for solving problem (5.1).

Algorithm 5.1

Step 1. If $r \in \mathcal{K}^n$ (case (i)), set $z^* := 0$ and terminate. If $-B^{-1}r \in \mathrm{int}\,\mathcal{K}^n$ (case(ii)), set $z^* := -B^{-1}r$ and terminate. Otherwise (case (iii)), go to Step 2.

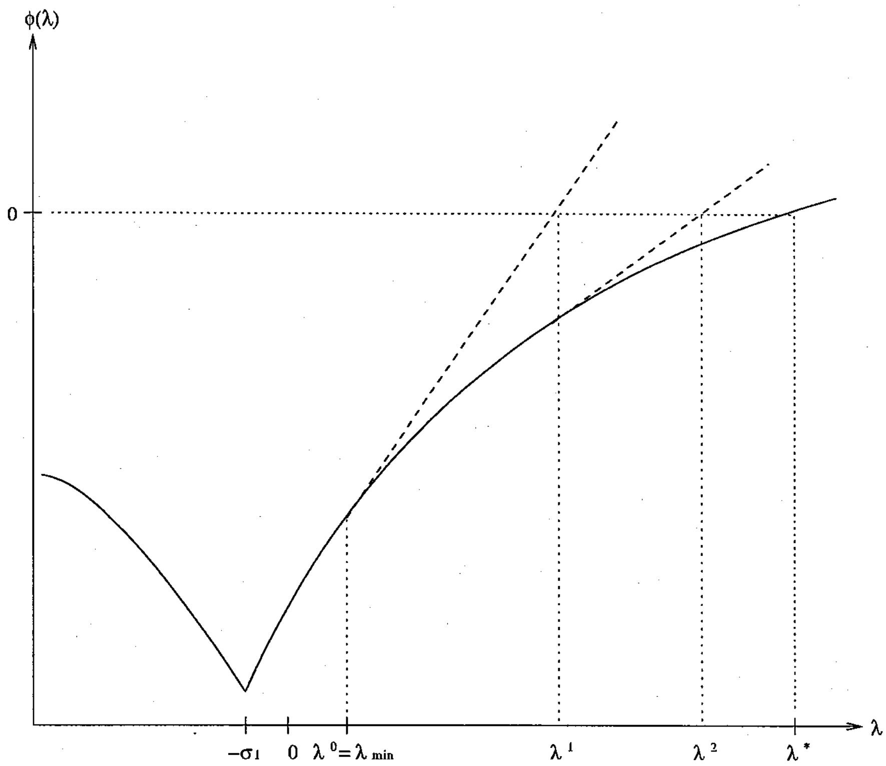

Figure 3: Newton's method for $\phi(\lambda) = 0$

Step 2. Let $\lambda^0 := \lambda_L$ and $w^0 := w(\lambda^0)$. Set $t := 0$.

Step 3. If $\|w^t\| \leq 1 + \delta$ holds for sufficiently small $\delta > 0$, set $z^* := \lambda^t(1, (w^t)^T)^T$ and terminate.

Step 4. Factorize $H(\lambda^t) = RR^T$.

Step 5. Calculate u^t such that $R^t u^t = -g$.

Step 6. Calculate w^t such that $(R^t)^T w^t = u^t$.

Step 7. Calculate v^t such that $R^t v^t = w^t$.

Step 8. Calculate $\lambda^{t+1} := \lambda^t + (\|w^t\| - 1)\dfrac{\|w^t\|^2}{\|v^t\|^2}$. Set $t := t + 1$ and go back to Step 3.

This procedure can be incorporated in Step 2 of Algorithm 2.1 to solve subproblem (4.5).

6 Numerical Results

In this section, we present some numerical results with the algorithms proposed in this paper. The program was coded in MATLAB 6.5.0 and run on a Windows PC with Pentium III of 500MHz and 320MB of memory. We have conducted the following experiments:

(A) Testing Algorithm 5.1 on SOCCP (5.1).

(B) Testing Algorithm 2.1 on SOCCP (1.1) with various Cartesian structures of $\mathcal{K}$.

(C) Testing Algorithm 2.1 with various values of ω_i.

In experiment (A), we solve SOCCP (5.1) of various sizes by Algorithm 5.1. We set $A = NN^{\mathrm{T}}+I$, where N is a square matrix generated randomly and I is the identity matrix, and then let $B = \overline{A}$, where $\overline{A}$ is determined from A as in (4.2). By suitably adjusting the sparsity of N, the matrix B is formed so that its nonzero density is approximately 1%. The nonzero elements of N are chosen randomly from the interval $[-5, 5]$. The elements of r are chosen randomly from the interval $[-100, 100]$. For each $n = 100, 200, \dots, 1000$, we generate ten problems and apply Algorithm 5.1. The parameter δ in the termination criterion of Step 3 is set to be 10^{-4}. We show the results in Table 1, where n denotes the number of variables. The number of iterations and the CPU time are averages of ten trials for each n. Interestingly, the number of iterations stays constant regardless of the problem size, although the CPU time grows rapidly as the problem size becomes large.

n	No. iterations	CPU time (s)
100	3.0	0.04
200	3.0	0.22
300	3.0	1.26
400	3.0	3.50
500	3.0	7.36
600	3.0	14.96
700	3.0	32.79
800	3.0	64.99
900	3.0	117.00
1000	3.0	176.98

Table 1: Results of experiment (A)

In experiment (B), we solve SOCCP (1.1) with $n = 400$ and determine M by $M = NN^{\mathrm{T}} + I$ with N being a randomly generated square matrix. By controlling the sparsity of N, matrix M is formed so that its nonzero density is approximately 1%. The nonzero elements of N are chosen randomly from the interval $[-5, 5]$. The elements of q are chosen randomly from in the interval $[-100, 100]$. To construct SOCs of various types, we choose n_i such that $n_1+\cdots+n_m = 400$ and $n_1 = \cdots = n_m$ for various values of m, where n_i is the dimension of every SOC composing $\mathcal{K}$, and m is the number of SOCs composing $\mathcal{K}$. For each $\mathcal{K}$, we apply Algorithm 2.1 to solve one hundred problems with randomly generated data (M, q). We set $\omega_1 = \cdots = \omega_m = 1$, the initial point z^0 to be 0, and the parameter ε in the termination criterion in Step 3 to be 10^{-4}. We show the results in Table 2, where the number of iterations and the CPU time are averages of one hundred trials for each $\mathcal{K}$. We find that the number of iterations generally decreases as the number m of SOCs in $\mathcal{K}$ decreases. However, the CPU time rapidly increases when n_i becomes large.

In experiment (C), we also solve SOCCP (1.1) with $n = 400$. We determine M and q in a way similar to experiment (B). Setting $\omega := \omega_1 = \cdots = \omega_m$, we apply Algorithm 2.1 to solve one hundred problems for each $\omega = 0.1, 0.2, \dots, 1.9$. Though the convergence of

n_i	m	No. iterations	CPU time (s)
1	400	17.22	3.30
2	200	15.70	5.91
5	80	14.77	4.06
10	40	13.71	2.50
20	20	13.50	2.00
40	10	13.16	2.32
80	5	13.28	3.29
200	2	12.81	10.33
400	1	7.93	30.21

Table 2: Results of experiment (B)

Algorithm 2.1 is not guaranteed for $\omega > 1$, it has often been observed that the algorithm still converges to a solution. We show the results in Table 3, where the number of iterations and the CPU time are averages that are taken over the successful trials among one hundred trials for each ω. Moreover, #failures denotes the number of trials where the algorithm fails to converge within 1000 iterations. We see that the algorithm converges most rapidly when $\omega = 1.1$.

ω	No. iterations	CPU time (s)	#failures
0.1	200.91	28.35	0
0.2	104.30	14.70	0
0.3	68.27	9.80	0
0.4	48.58	6.92	0
0.5	38.45	5.56	0
0.6	29.73	4.32	0
0.7	24.72	3.58	0
0.8	20.41	2.98	0
0.9	16.59	2.44	0
1.0	13.81	2.07	0
1.1	11.80	1.76	0
1.2	13.21	1.95	0
1.3	15.19	2.23	0
1.4	18.75	2.74	0
1.5	23.99	3.48	0
1.6	31.77	4.67	0
1.7	51.64	7.49	1
1.8	72.49	10.81	3
1.9	107.27	15.75	6

Table 3: Results of experiment (C)

7 Conclusion

In this paper, we have extended the matrix splitting method for LCPs to SOCCPs and showed that the algorithm converges under the assumption that the matrix M is strictly $\mathcal{K}$-copositive. Furthermore, we have proposed a block SOR method for solving SOCCPs with M being positive definite, and conducted some numerical experiments. From a practical view point, there is room for improvement in speeding up the algorithm. In particular, it would be highly effective if subproblems are solved in parallel. Moreover, it is desirable to have a regular splitting (B, C) that can be applied to SOCCPs with matrix M being $\mathcal{K}$-copositive.

Acknowledgements

I am grateful to Professor Masao Fukushima and Assistant Professor Nobuo Yamashita to their help. They looked through my poor draft carefully and give me much advice of value. I would also like to appreciate Mr. Shunsuke Hayashi's cooperation. He mentored me arduously and gave a helping hand to me. Moreover, I wish extend my gratitude to Associate Professor Tetsuya Takine. His elegant advice made my paper better. Finally, I would like to thank all members in Fukushima's Laboratory.

References

[1] Chen, C. and Mangasarian, O. L.: Smoothing methods for convex inequalities and linear complementarity problems, *Mathematical Programming*, Vol. 71 (1995), 51–69.

[2] Chen, C. and Mangasarian, O. L.: A class of smoothing functions for nonlinear and mixed complementarity problems, *Computational Optimization and applications*, Vol. 5 (1996), 97–138.

[3] Chen, X.: Smoothing methods for complementarity problems and their applications: a survey, *Journal of the Operations Research of Japan*, Vol. 43 (2000), 32–47.

[4] Conn, A. R., Gould, N. I. M. and Toint, P. L.: *Trust-Region Methods*, Society for Industrial & Applied Mathmatics (SIAM), 2000.

[5] Cottle, R. W., Pang, J.-S. and Stone, R. E.: *The Linear Complementarity Problem*, Academic Press, 1992.

[6] Faraut, J. and Korányi, A.: *Analisyis on Symmetric Cones*, Clarendon Press, 1994.

[7] Fukushima, M., Luo, Z.-Q. and Tseng, P.: Smoothing functions for second-order cone complementarity problems, *SIAM Journal on Optimization*, Vol. 12 (2001), 436–460.

[8] Golub, G. H. and Loan, C. F. V.: *Matrix Computations*, The Johns Hopkins University Press, 1996.

[9] Golub, G. and Ortega, J. M.: *Scientific Computing: An Introduction with Parallel Computing*, Academic Press, 1993.

[10] Harker, P. T. and Pang, J.-S.: Finite-dimensional vatiational inequality and nonlinear complementarity problems: A survey of theory, algorithms and applications, *Mathematical Programming*, Vol. 48 (1990), 161–220.

[11] Iusem, A. N.: On the convergence of iterative methods for symmetric linear complementarity problems, *Mathematical Programming*, Vol. 59 (1993), 33–48.

[12] Luo, Z.-Q. and Tseng, P.: Error bound and convergence analysis of matrix splitting algorithms for the affine variational inequality problem, *SIAM Journal of Optimization*, Vol. 2 (1992), 43–54.

[13] Machida, N., Fukushima, M. and Ibaraki, T.: A multisplitting method for symmetric linear complementarity problems, *Journal of Computational and Applied Mathematics*, Vol. 62 (1995), 217–227.

[14] Mangasarian, O. L. and Leone, R. D.: Parallel successive overrelaxation methods for symmetric linear complementarity problems, *Journal of Optimization Theory and Applications*, Vol. 54 (1987), 437–446.

[15] Wolkowicz, H., Saigal, R. and Vandenberghe, L. eds.: *Handbook of Semidefinite Programming: Theory, Algorithms and Applications*, Kluwer Academic Publishers, 2000.

[16] Zeng, J. P., Li, D. H. and Fukushima, M.: Weighted max-norm estimate of additive Schwarz iteration scheme for solving linear complementarity problems, *Journal of Computational and Applied Mathematics*, Vol. 131 (2001), 1–14.