On finite convergence of an explicit exchange method for convex semi-infinite programming problems with second-order cone constraints*

Shunsuke Hayashi[1], Liping Zhang[2], Soon-Yi Wu[3]

[1]Department of Applied Mathematics and Physics, Graduate School of Informatics, Kyoto University, Kyoto 606-8501, Japan (shunhaya@amp.i.kyoto-u.ac.jp)

[2]Department of Mathematical Sciences, Tsinghua University, Beijing 100084, China(lzhang@math.tsinghua.edu.cn)

[3]Department of Mathematics, National Cheng Kung University, Tainan, Taiwan; National Center for Theoretical Sciences, Taiwan (soonyi@mail.ncku.edu.tw)

Abstract. We consider the convex semi-infinite programming problem with second-order cone constraints (for short, SOCCSIP). We propose an explicit exchange method for solving SOCCSIP, and prove that the algorithm terminates in a finite number of iterations under some mild conditions. In the analysis, the complementarity slackness condition with respect to second-order cones plays an important role. To deal with such complementarity conditions, we utilize the spectral factorization techniques in Euclidean Jordan algebra. We also show that the obtained output is an approximate optimum of SOCCSIP. We also report some numerical results involving the application to the robust optimization in the classical convex semi-infinite programming.

Key words. semi-infinite programming, finite termination, explicit method, second-order cone

Mathematics subject Classification (2000): 90C30, 90C33

1 Introduction

In this paper, we focus on the following convex semi-infinite programming problem with second-order cone constraints (SOCCSIP):

$$\text{(SOCCSIP)} \qquad \begin{array}{ll} \text{Minimize} & f(x) \\ \text{subject to} & x \in \mathcal{K}, \ \ c(x,s) \le 0 \quad \forall s \in \Omega, \end{array} \tag{1.1}$$

where $\Omega \subseteq \mathbb{R}^m$ is a given compact set playing a role of index set, $f : \mathbb{R}^n \to \mathbb{R}$ is a continuously differentiable convex function, and $c : \mathbb{R}^{n+m} \to \mathbb{R}$ is a continuously differentiable function such that $c(\cdot, s)$ is convex for any index $s \in \Omega$. $\mathcal{K}$ is the Cartesian product of several second order cones (SOCs), i.e., $\mathcal{K} = \mathcal{K}^{n_1} \times \cdots \times \mathcal{K}^{n_m}$ where

$$\mathcal{K}^{n_j} := \begin{cases} \left\{ x = (x_1, \tilde{x}) \in \mathbb{R} \times \mathbb{R}^{n_j - 1} \,\middle|\, x_1 \ge \|\tilde{x}\| \right\} & (n_j \ge 2) \\ \{ x \in \mathbb{R} \,|\, x \ge 0 \} & (n_j = 1) \end{cases}$$

with $n = n_1 + \cdots + n_m$.

A semi-infinite programming (SIP) [9, 11, 16, 24, 27] problem, which contains finitely many variables on a feasible set described by infinitely many constraints, is written as follows:

$$\text{(SIP)} \qquad \begin{array}{ll} \text{Minimize} & f(x) \\ \text{subject to} & g(x,s) \le 0 \quad \forall s \in \Omega, \end{array}$$

*Technical report 2011-016, September 23, 2011. This research was supported in part by Grant-in-Aid for Young Scientists (B) from Japan Society for the Promotion of Science, and by National Center for Theoretical Sciences, Taiwan.

where $f : \mathbb{R}^n \to \mathbb{R}$ and $g : \mathbb{R}^{n+m} \to \mathbb{R}$ are continuously differentiable functions and $\Omega \subset \mathbb{R}^m$ is a given compact set. If Ω is finite, the SIP is a finite optimization problem usually called a nonlinear programming (NLP).

The SIP has strong practical backgrounds in approximation theory, optimal control and numerous engineering problems such as optimum filter design in signal processing, resource allocation in decentralized systems, and decision making under competition (see e.g. [12, 22] and references therein). The main difficulty for solving SIP is to deal with infinitely many constraints. During the last decade, many numerical algorithms have been developed for solving SIPs (see [7, 18, 21, 26, 29, 31] and references therein). A typical algorithm for solving SIP generally generates a sequence of finitely constrained auxiliary optimization problems that can be solved by standard algorithms for NLPs. Existing methods for SIPs can be roughly divided into three types: exchange methods [18, 19, 26, 32], discretization methods, and reduction-based methods (see, e.g. [26, 30] and references therein). Exchange and discretization methods are numerically expensive in general. The cost per iteration increases dramatically as the cardinality of the auxiliary problem grows. Globally convergent reduction based methods, on the other hand, require strong assumptions and are often conceptual methods which can be implemented in a rather simplified form merely. Exchange and discretization methods, therefore are often used only for the first stage of the solution process to generate an approximate solution of the SIP, whereas reduction-based methods are typically employed only in the final stage of the solution process in order to provide a higher accuracy of the solution and a better rate of convergence. Recently, Wu, Li, Qi and Zhou [33] proposed an iterative method for solving KKT system of the SIP in which they drop some redundant points at some certain iterations. Qi, Wu, and Zhou [25] and Li, Qi, Tam and Wu [20] presented semismooth Newton method and smoothing Newton method for solving SIP, in which the index set Ω is specified by $\Omega = \{s \,|\, c_j(s) \leq 0 \,(j = 1, \ldots, r)\}$ with twice continuously differentiable functions $c_j : \mathbb{R}^m \to \mathbb{R} \,(j = 1, \ldots, r)$. They proved that their algorithms have nice convergence properties, but these two methods cannot ensure the feasibility. For linear SIPs, Lai and Wu [19] proposed the explicit algorithm in which they solve a linear programming with a finite feasible set E_k. They drop out redundant points in E_k at each iteration and only keep active points, which ensures $|E_k| \leq n$ [28] for each k, and hence the algorithm is very efficient in saving computational time.

The main goal of this paper is to design an explicit exchange method for solving SOCCSIP (1.1) and study its convergence properties. In the algorithm we inherit Lai and Wu's explicit exchange technique [19], but we also introduce the relaxed scheme in which maximization problem with respect to $s \in \Omega$ need not be solved in each iteration. Our algorithm has only to find some $s \in \Omega$ such that a certain criterion with small $\eta > 0$ is satisfied. We also note that the spectral factorization associated with Euclidean Jordan algebra is exploited in the convergence analyses. In the classical studies on the cutting plane or exchange type methods (see [18, 21, 26, 32] and references therein), the convergence properties were analyzed in a componentwise manner. However, such analyses does not make sense for SOCCSIP any more since the SOC does not have componentwisely independent structure. To overcome these difficulties, we introduce the coordinate system based on the spectral factorization in Euclidean Jordan algebra associated with SOC. We then prove that the algorithm terminates in a finite number of iterations and the obtained output sufficiently approximates the optimum of SOCCSIP (1.1).

When f and $c(\cdot, s)$ are affine for any $s \in \Omega$, SOCCSIP (1.1) reduces to the SOCLSIP studied by Hayashi and Wu [13]. They proposed an explicit exchange method for solving SOCLSIP and provided its convergence theorems. However, their results for SOCLSIP cannot be applied to SOCCSIP (1.1) directly. Indeed, the difference from the SOCLSIP study [13] can be given as follows.

- In case of SOCCSIP, we have to evaluate the residual values $f(x^{k+1}) - f(x^k) - \nabla_x f(x^k)^\top (x^{k+1} - x^k)$ and $c(x^{k+1}, s) - c(x^k, s) - \nabla_x c(x^k, s)^\top (x^{k+1} - x^k)$ in each iteration. However, those values are absent when f and $c(\cdot, s)$ are affine.

- We introduce the finite index set $\Omega_0 \subset \Omega$, whose elements are never dropped throughout the iterations. Thanks to this scheme, the boundedness of the generated sequence is guaranteed. This technique was not introduced in [13].
- Some assumptions needed for the convergence analyses in [13] can be removed (especially in the strictly convex case).

This paper is organized as follows. In Section 2, we give some preliminaries needed for the later analyses. In Section 3, we develop an explicit exchange method for solving SOCCSIP (1.1). In Section 4, we establish the convergence analysis of the proposed algorithm. In Section 5, we give some numerical results.

2 Preliminaries

In this section, we give some fundamental knowledge on SOCCSIP (1.1) and the SOC which will be necessary in the subsequent sections.

Throughout the paper, we suppose that SOCCSIP (1.1) satisfies the following assumption.

Assumption A (i) *The solution set of* SOCCSIP (1.1) *is nonempty and compact.* (ii) *The nonnegative constraint in* SOCCSIP (1.1) *has an influence on the optimal value, that is,* $\inf\{f(x)\,|\,x \in \mathcal{K}\} < \inf\{f(x)\,|\,x \in \mathcal{K},\ c(x,s) \le 0\,(\forall s \in \Omega)\}$. (iii) *There exists* $\overline{x} \in \mathcal{K}$ *such that* $c(\overline{x}, s) < 0$ *for all* $s \in \Omega$.

Then, we have the following two theorems; the first one can be shown by using [3, Prop. 2.3.1], [1, Lem. 3.1], and the convergence theorem for the discretization method [26, Chap. 7], and the second one can be proved by an analogous argument in Theorems 5.97–5.99 and Proposition 5.104 in [5].

Theorem 2.1 *Suppose that Assumptions* A(i) *and* A(ii) *hold. Then, there exists* $\Omega_0 = \{s_1^0, \ldots, s_{m_0}^0\} \subset \Omega$ *such that the following statements hold:*

- f *is level-bounded on the set* $\mathcal{K} \cap \{x \,|\, c(x, s_i^0) \le 0\ (i = 1, 2, \ldots, m_0)\}$.
- $\inf\{f(x) \,|\, x \in \mathcal{K}\} < \inf\{f(x) \,|\, x \in \mathcal{K},\ c(x, s_i^0) \le 0\ (i = 1, 2, \ldots, m_0)\}$.

Theorem 2.2 *Suppose that Assumption* A(iii) *holds. Let* $M(\Omega)$ *be the space of all bounded regular Borel measures on* Ω*, and* $M^+(\Omega)$ *be the nonnegative cone in* $M(\Omega)$*. Then, there exist* $x \in \mathcal{K}$ *and* $\mu \in M^+(\Omega)$ *such that*

$$x^\top \left(\nabla f(x) + \int_\Omega \nabla_x c(x,s) d\mu(s) \right) = 0, \quad \int_\Omega c(x,s) d\mu(s) = 0,$$
$$\nabla f(x) + \int_\Omega \nabla_x c(x,s) d\mu(s) \in \mathcal{K}, \ \ c(x,s) \le 0 \ \ (\forall s \in \Omega),$$

where

$$\int_\Omega \nabla_x c(x,s) d\mu(s) := \left(\int_\Omega \frac{\partial}{\partial x_i} c(x,s) d\mu(s) \right)_{i=1,\ldots,n} \in \mathbb{R}^n.$$

Moreover, if x *is an optimum of* SOCCSIP (1.1)*, then there exist an nonnegative number* $q \le n$*, multipliers* $\{u_i\}_{i=1,\ldots,q}$ *and attainers* $\{s_i\}_{i=1,\ldots,q}$ *such that vectors* $\{\nabla_x c(x, s_i)\}_{i=1,\ldots,q}$ *are linearly independent, and*

$$\begin{aligned} &\mathcal{K} \ni x \perp \nabla f(x) + \sum_{i=1}^q \nabla_x c(x, s_i) u_i \in \mathcal{K}, \\ &u_i > 0, \ c(x, s_i) = 0 \ \ (i = 1, \ldots, q), \\ &c(x,s) \le 0 \ \ \forall s \in \Omega. \end{aligned} \tag{2.1}$$

We next give some properties on the SOC. We introduce the spectral factorization for a single SOC $\mathcal{K}^\ell$ in Euclidean Jordan algebra [8, 10]. For any vector $y := (y_1, \tilde{y}) \in \mathbb{R} \times \mathbb{R}^{\ell-1}$ with $\ell \geq 2$, its spectral factorization is defined by

$$y = \lambda_1(y)v_1(y) + \lambda_2(y)v_2(y),$$

where $\lambda_i(y) \in \mathbb{R}$ and $v_i(y) \in \mathbb{R}^\ell$ $(i = 1, 2)$ are the spectral values and vectors, respectively, defined by

$$\lambda_i(y) := y_1 + (-1)^i \|\tilde{y}\|, \tag{2.2}$$

$$v_i(y) := \begin{cases} \frac{1}{2}\left(1, (-1)^i \frac{\tilde{y}}{\|\tilde{y}\|}\right) & \text{if } \tilde{y} \neq 0, \\ \frac{1}{2}\left(1, (-1)^i w\right) & \text{if } \tilde{y} = 0. \end{cases}$$

Here, $w \in \mathbb{R}^{\ell-1}$ is an arbitrary vector with $\|w\| = 1$. It is obvious that

$$\lambda_1(y) \leq \lambda_2(y), \quad \lambda_1(y) \geq 0 \Leftrightarrow y \in \mathcal{K}^\ell,$$

$$\lambda_1(y) = 0 \Leftrightarrow y \in \operatorname{bd} \mathcal{K}^\ell, \ \lambda_1(y) > 0 \Leftrightarrow y \in \operatorname{int} \mathcal{K}^\ell$$

and

$$\|v_1(y)\| = \|v_2(y)\| = 1/\sqrt{2}, \quad v_1(y)^\top v_2(y) = 0.$$

Now, we study the relation between the complementarity on SOCs and the spectral factorization. The vectors $x \in \mathbb{R}^n$ and $z \in \mathbb{R}^n$ are said to satisfy the second-order cone complementarity if

$$x \in \mathcal{K}, \quad z \in \mathcal{K}, \quad x^\top z = 0. \tag{2.3}$$

It is easily seen that (2.3) holds if and only if

$$x_j \in \mathcal{K}^{n_j}, \ z_j \in \mathcal{K}^{n_j}, \text{ and } x_j^\top z_j = 0, \qquad (j = 1, \dots, m) \tag{2.4}$$

where x_j and z_j denote the Cartesian subvectors of x and z, respectively, i.e.,

$$x = (x_1, \dots, x_m), \ \ z = (z_1, \dots, z_m) \in \mathbb{R}^{n_1} \times \cdots \times \mathbb{R}^{n_m}. \tag{2.5}$$

Moreover, by [13, Prop. 2.3], we have

$$v_1(x_j) = v_2(z_j) \text{ and } v_2(x_j) = v_1(z_j).$$

3 The explicit exchange method for SOCCSIP

In this section, we propose an explicit exchange method for solving SOCCSIP (1.1) and give some fundamental properties. In each iteration of the algorithm, we solve a finitely constrained convex programming as a subproblem. For a finite set $E = \{s_1, \dots, s_m\} \subset \Omega$, let CSOCP$(E)$ denote the finitely constrained convex second-order cone programming problem defined as

$$\text{CSOCP}(E): \quad \begin{array}{rl} \text{Minimize} & f(x) \\ \text{subject to} & x \in \mathcal{K}, \ c(x, s_j) \leq 0 \ (j = 1, \dots, m). \end{array}$$

Then, the first-order optimality condition of CSOCP(E) is given by

$$\begin{aligned} & z := \nabla f(x) + \sum_{j=1}^m \nu(s_j) \nabla_x c(x, s_j) \in \mathcal{K}, \ x \in \mathcal{K}, \ x^\top z = 0, \\ & c(x, s_j)\nu(s_j) = 0, \ c(x, s_j) \leq 0, \ \nu(s_j) \geq 0 \quad (j = 1, \dots, m), \end{aligned} \tag{3.1}$$

where $\nu(s_j)$ $(j = 1, \ldots, m)$ denotes the Lagrange multiplier corresponding to the constraint $c(x, s_j) \leq 0$. For more details on the optimality conditions, see e.g., [4, 5]. Actually, (3.1) reduces to a monotone second-order cone complementarity problem (SOCCP), which can be solved by existing algorithms [6, 10, 14, 15, 23]. Nonlinear second-order cone programming such as CSOCP(E) is also studied by some researchers [17, 34], but they are still immature.

The concrete scheme of the algorithm is written as follows.

Algorithm 1

Step 0. *Find a finite index set* $\Omega_0 = \{s_1^0, \ldots, s_{m_0}^0\}$ *satisfying Theorem* 2.1, *and let* E_0 *be such that* $\Omega_0 \subset E_0 \subset \Omega$ *and* $|E_0| < \infty$. *Solve* CSOCP(E_0) *to obtain its optimum* x^0. *Choose a small number* $\eta > 0$ *and set* $k := 0$.

Step 1. *Find an* $s_{\rm new}^k \in \Omega$ *such that*

$$c(x^k, s_{\rm new}^k) > \eta. \tag{3.2}$$

If such an $s_{\rm new}^k$ *does not exist, i.e.,* $\max_{s \in \Omega} c(x^k, s) \leq \eta$, *then stop. Otherwise, let*

$$\overline{E}_{k+1} := E_k \cup \{s_{\rm new}^k\}.$$

Step 2. *Solve* CSOCP($\overline{E}_{k+1}$) *to obtain its optimum* x^{k+1} *and the Lagrange multipliers* $\{\nu_{k+1}(s) | s \in \overline{E}_{k+1}\}$.

Step 3. *Let*

$$E_{k+1} := \Omega_0 \cup \left\{s \in \overline{E}_{k+1} \setminus \Omega_0 \;\middle|\; \nu_{k+1}(s) > 0\right\}.$$

Let $k := k + 1$ *and go to Step* 1.

Steps 1 and 2 are the main differences from the algorithm in [19]. Note that, it is also possible to choose multiple elements satisfying (3.2) in Step 1. Although we merely deal with the single-point exchange scheme in the following analyses, they are also applicable to multiple exchange type algorithms. In Step 2, CSOCP($\overline{E}_{k+1}$) can be solved by using an existing method. Step 3 is to remove the constraints that are inactive at the optimum x^{k+1} and corresponding to $t \in \overline{E}_{k+1} \setminus \Omega_0$. Here, we note that x^{k+1} also solves CSOCP(E_{k+1}). Moreover, the sequence $\{x^k\}$ is bounded since we have $\Omega_0 \subset \overline{E}_{k+1}$ for all k. (See Proposition 4.1 given later.)

Next, we define some notations for convenience. For a finite set $E = \{s_1, \ldots, s_m\} \subset T$, we denote the feasible set and the optimal value of CSOCP(E) by $\mathcal{F}(E) \subset \mathbb{R}^n$ and $V(E) \in [-\infty, +\infty)$, respectively, i.e.,

$$\begin{aligned} \mathcal{F}(E) &:= \left\{x \;\middle|\; x \in \mathcal{K},\ c(x, s_j) \leq 0\ (j = 1, \ldots, m)\right\}, \\ V(E) &:= \inf\left\{f(x) \;\middle|\; x \in \mathcal{K},\ c(x, s_j) \leq 0\ (j = 1, \ldots, m)\right\}. \end{aligned}$$

Moreover, we denote the optimal value of SOCCSIP (1.1) by $V^* \in \mathbb{R}$, i.e.,

$$V^* := \inf\left\{f(x) \;\middle|\; x \in \mathcal{K},\ c(x, s) \leq 0\ (\forall s \in \Omega)\right\}.$$

Let $\{x^k\}$ and $\{\nu_k(s)\}$ be the sequences generated by Algorithm 1, and $\{z^k\}$ be defined as

$$z^k := \nabla f(x^k) + \sum_{s \in E_k} \nu_k(s) \nabla_x c(x^k, s). \tag{3.3}$$

Then, the complementarity slackness condition (3.1) yields

$$\begin{aligned} & x^k \in \mathcal{K}, \quad z^k \in \mathcal{K}, \quad (x^k)^\top z^k = 0, \\ & c(x^k, s) \leq 0, \quad \nu_k(s) \geq 0, \quad c(x^k, s)\nu_k(s) = 0, \quad \forall s \in E_k. \end{aligned} \tag{3.4}$$

Moreover, by using the spectral factorization for Cartesian subvectors x_j^k and z_j^k (see (2.5)), we define $\hat{x}_{ij}^k$, $\hat{z}_{ij}^k \in \mathbb{R}$ and $\hat{e}_{ij}^k \in \mathbb{R}^{n_j}$ for each k as follows:

- When $n_j = 1$,

$$\hat{x}_{1j}^k := \hat{x}_{2j}^k := \frac{1}{2}x_j^k, \;\; \hat{z}_{1j}^k := \hat{z}_{2j}^k := \frac{1}{2}z_j^k, \;\; \hat{e}_{1j}^k := \hat{e}_{2j}^k := 1. \tag{3.5}$$

- When $\Big[n_j \geq 2$ and $k = 1\Big]$ or $\Big[n_j \geq 2,\ k \geq 2$, and $\|\sqrt{2}v_1(x_j^k) - \hat{e}_{1j}^{k-1}\| \leq \|\sqrt{2}v_1(x_j^k) - \hat{e}_{2j}^{k-1}\|\Big]$,

$$\begin{aligned} &\hat{x}_{1j}^k := \lambda_1(x_j^k)/\sqrt{2},\ \ \hat{x}_{2j}^k := \lambda_2(x_j^k)/\sqrt{2},\ \ \hat{z}_{1j}^k := \lambda_2(z_j^k)/\sqrt{2},\ \ \hat{z}_{2j}^k := \lambda_1(z_j^k)/\sqrt{2}, \\ &\hat{e}_{1j}^k := \sqrt{2}v_1(x_j^k) = \sqrt{2}v_2(z_j^k),\ \ \hat{e}_{2j}^k := \sqrt{2}v_2(x_j^k) = \sqrt{2}v_1(z_j^k). \end{aligned} \tag{3.6}$$

- When $n_j \geq 2,\ k \geq 2$, and $\|\sqrt{2}v_1(x_j^k) - \hat{e}_{1j}^{k-1}\| > \|\sqrt{2}v_1(x_j^k) - \hat{e}_{2j}^{k-1}\|$,

$$\begin{aligned} &\hat{x}_{1j}^k := \lambda_2(x_j^k)/\sqrt{2},\ \ \hat{x}_{2j}^k := \lambda_1(x_j^k)/\sqrt{2},\ \ \hat{z}_{1j}^k := \lambda_1(z_j^k)/\sqrt{2},\ \ \hat{z}_{2j}^k := \lambda_2(z_j^k)/\sqrt{2}, \\ &\hat{e}_{1j}^k := \sqrt{2}v_2(x_j^k) = \sqrt{2}v_1(z_j^k),\ \ \hat{e}_{2j}^k := \sqrt{2}v_1(x_j^k) = \sqrt{2}v_2(z_j^k). \end{aligned} \tag{3.7}$$

Then, we have

$$x^k = (x_j^k)_{j=1}^m = \left(\hat{x}_{1j}^k\hat{e}_{1j}^k + \hat{x}_{2j}^k\hat{e}_{2j}^k\right)_{j=1}^m, \quad z^k = (z_j^k)_{j=1}^m = \left(\hat{z}_{1j}^k\hat{e}_{1j}^k + \hat{z}_{2j}^k\hat{e}_{2j}^k\right)_{j=1}^m, \tag{3.8}$$

for each k. For such factorizations, we have the following proposition

Proposition 3.1 [13, Prop. 3.4] *Let x_j^k and z_j^k be factorized as (3.8). Then, for each $i = 1, 2$, $j = 1, \ldots, m$, and $k \geq 1$, the following statements hold.*

(a) $\max(\hat{x}_{ij}^k, \hat{z}_{ij}^k) \geq 0$ *and* $\min(\hat{x}_{ij}^k, \hat{z}_{ij}^k) = 0$.

(b) (i) $\|\hat{e}_{ij}^k\| = 1$. (ii) $\hat{e}_{ij}^k \in \operatorname{bd}\mathcal{K}^{n_j}$ *and* $(\hat{e}_{1j}^k)^\top\hat{e}_{2j}^k = 0$ *if* $n_j \geq 2$

(c) $(\hat{e}_{ij}^k)^\top\hat{e}_{ij}^{k+1} \geq 1/2$.

We also define

$$\begin{aligned} d^k &:= x^{k+1} - x^k, \\ F_k &:= f(x^{k+1}) - f(x^k) - \nabla f(x^k)^\top d^k, \\ G_k &:= f(x^k) - f(x^{k+1}) + \nabla f(x^{k+1})^\top d^k \\ P_k(s) &:= c(x^{k+1}, s) - c(x^k, s) - \nabla_x c(x^k, s)^\top d^k, \\ Q_k(s) &:= c(x^k, s) - c(x^{k+1}, s) + \nabla_x c(x^{k+1}, s)^\top d^k. \end{aligned}$$

Since f and $c(\cdot, s)$ are continuously differentiable and convex, we have

$$\begin{aligned} &F_k = o(\|d^k\|),\ \ G_k = o(\|d^k\|),\ \ F_k \geq 0,\ \ G_k \geq 0 \\ &P_k(s) = o(\|d^k\|), \quad Q_k = o(\|d^k\|), \quad P_k(s) \geq 0, \quad Q_k(s) \geq 0. \end{aligned}$$

4 Convergence analysis

In this section, we show that the proposed algorithm terminates in a finite number of iterations under some mild conditions. Furthermore, we prove that the last output is sufficiently close to the optimal solution of SOCCSIP (1.1) if the criterion value is sufficiently close to zero.

4.1 Some technical propositions

In this subsection, we give some technical propositions that are important and convenient for analyzing the convergence of Algorithm 1.

We first show the boundedness of the generated sequence.

Proposition 4.1 *The sequence $\{x^k\}$ generated by Algorithm 1 is bounded, i.e., there exists $M > 0$ such that $\|x^k\| \le M$ for all k.*

Proof. Let $\mathcal{L}_*$ be the optimal level set of SOCCSIP (1.1), i.e., $\mathcal{L}_* := \{x \mid f(x) \le V^*\}$. Then, $\mathcal{L}_* \cap \mathcal{F}(\Omega_0)$ is bounded from Theorem 2.1. Moreover, we have $x^k \in \mathcal{L}_* \cap \mathcal{F}(\Omega_0)$ for all k since $f(x^k) \le V^*$ and $x^k \in \mathcal{F}(\overline{E}_k) \subset \mathcal{F}(\Omega_0)$. Hence $\{x^k\}$ is bounded. ∎

The following proposition evaluates the increment of the optimal value of CSOCP($\overline{E}_k$) in each iteration.

Proposition 4.2 *For all $k \ge 0$, we have*

$$\begin{aligned}
&f(x^{k+1}) - f(x^k)\\
&= (z^k)^\top x^{k+1} + F_k + \sum_{s\in E_k} \nu_k(s)(P_k(s) - c(x^{k+1}, s)) &(4.1)\\
&= -(z^{k+1})^\top x^k - G_k - \sum_{s\in E_k} \nu_{k+1}(s)\big(Q_k(s) - c(x^k, s)\big) + \nu_{k+1}(s^k_{\mathrm{new}})\big(c(x^k, s^k_{\mathrm{new}}) - Q_k(s^k_{\mathrm{new}})\big) &(4.2)\\
&= -(z^{k+1})^\top x^k - G_k - \sum_{s\in E_k} \nu_{k+1}(s)\big(Q_k(s) - c(x^k, s)\big) - \nu_{k+1}(s^k_{\mathrm{new}})\nabla_x c(x^{k+1}, s^k_{\mathrm{new}})^\top d^k &(4.3)
\end{aligned}$$

Proof. By the definitions of d^k, z^k, F_k and $P_k(s)$, we have

$$\begin{aligned}
(z^k)^\top x^{k+1} &= (z^k)^\top (x^{k+1} - x^k) + \textstyle\sum_{s\in E_k} \nu_k(s) c(x^k, s)\\
&= \big(\nabla f(x^k) + \textstyle\sum_{s\in E_k} \nu_k(s)\nabla_x c(x^k, s)\big)^\top d^k + \textstyle\sum_{s\in E_k} \nu_k(s) c(x^k, s)\\
&= \nabla f(x^k)^\top d^k + \textstyle\sum_{s\in E_k} \nu_k(s)\big(\nabla_x c(x^k, s)^\top d^k + c(x^k, s)\big)\\
&= f(x^{k+1}) - f(x^k) - F_k + \textstyle\sum_{s\in E_k} \nu_k(s)\big(c(x^{k+1}, s) - P_k(s)\big),
\end{aligned}$$

where the first equality follows from (3.4). Thus we have (4.1).

Next, we show the last two equalities. By the definitions of d^k, z^{k+1}, G_k and $Q_k(s)$, we have

$$\begin{aligned}
&-(z^{k+1})^\top x^k\\
&= (z^{k+1})^\top (x^{k+1} - x^k) + \textstyle\sum_{s\in E_{k+1}} \nu_{k+1}(s) c(x^{k+1}, s)\\
&= \big(\nabla f(x^{k+1}) + \textstyle\sum_{s\in E_{k+1}} \nu_{k+1}(s)\nabla_x c(x^{k+1}, s)\big)^\top d^k + \textstyle\sum_{s\in E_{k+1}} \nu_{k+1}(s) c(x^{k+1}, s)\\
&= \nabla f(x^{k+1})^\top d^k + \textstyle\sum_{s\in E_{k+1}} \nu_{k+1}(s)\big(\nabla_x c(x^{k+1}, s)^\top d^k + c(x^{k+1}, s)\big)\\
&= \nabla f(x^{k+1})^\top d^k + \textstyle\sum_{s\in E_{k+1}} \nu_{k+1}(s)\big(Q_k(s) - c(x^k, s)\big)\\
&= \nabla f(x^{k+1})^\top d^k + \textstyle\sum_{s\in E_k} \nu_{k+1}(s)\big(Q_k(s) - c(x^k, s)\big) + \nu_{k+1}(s^k_{\mathrm{new}})\big(Q_k(s^k_{\mathrm{new}}) - c(x^k, s^k_{\mathrm{new}})\big)\\
&= \nabla f(x^{k+1})^\top d^k + \textstyle\sum_{s\in E_k} \nu_{k+1}(s)\big(Q_k(s) - c(x^k, s)\big) + \nu_{k+1}(s^k_{\mathrm{new}})\nabla_x c(x^{k+1}, s^k_{\mathrm{new}})^\top d^k
\end{aligned}$$

where the first equality follows from (3.4) with $k := k+1$, the fifth equality follows from $\overline{E}_{k+1} = E_k \cup \{s^k_{\mathrm{new}}\}$ and $\nu_{k+1}(s) = 0$ for any $s \in \overline{E}_{k+1} \setminus E_{k+1}$, and the last equality holds since $\nu_{k+1}(s^k_{\mathrm{new}})$ $c(x^{k+1}, s^k_{\mathrm{new}}) = 0$. Thus we have (4.2) and (4.3). ∎

Note that we have $F_k \ge 0$, $\nu_k(s) \ge 0$, $c(x^{k+1}, s) \le 0$, and $P_k(s) \ge 0$ for any $s \in E_k$. Moreover, $z^k \in \mathcal{K}$ and $x^{k+1} \in \mathcal{K}$ entail $(z^k)^\top x^{k+1} \ge 0$. These inequalities and (4.1) lead us to the following corollary.

Corollary 4.1 *The sequence of optimal values* $\{f(x^k)\}$ *of* $\{\mathrm{CSOCP}(E_k)\}$ *is monotonically nondecreasing, i.e.,*

$$f(x^0) \le f(x^1) \le \cdots \le f(x^k) \le f(x^{k+1}) \le \cdots \le V^*.$$

The following proposition shows that the distance between x^k and x^{k+1} never converges to 0.

Proposition 4.3 *Let* $\{x^k\}$ *be the sequence generated by Algorithm* 1*. Then, there exists* $d_{\min} > 0$ *such that* $\|x^k - x^{k+1}\| \ge d_{\min}$ *for all* k*.*

Proof. Note that function $c(\cdot, s^k_{\mathrm{new}})$ is locally Lipschitzian since it is convex and $c(x, s^k_{\mathrm{new}}) < \infty$ for any $x \in \mathbb{R}^n$. Also, we have $c(x^k, s^k_{\mathrm{new}}) > \eta$ and $c(x^{k+1}, s^k_{\mathrm{new}}) \le 0$ for all k. Thus, noticing the boundedness of $\{x^k\}$ and Ω, we have

$$\begin{aligned} 0 < \eta &\le c(x^k, s^k_{\mathrm{new}}) - c(x^{k+1}, s^k_{\mathrm{new}}) \\ &\le L\|x^k - x^{k+1}\|, \end{aligned}$$

where $L > 0$ is the Lipschitzian constant. Hence, letting $d_{\min} := \eta/L > 0$, we obtain the result. ∎

The following two propositions show that the sequences $\{\sum_{s\in E_k} \nu_k(s)\}$ and $\{\|z^k\|\}$ are bounded, and $\sum_{s\in E_k} \nu_k(s)$ is positive away from 0.

Proposition 4.4 *Let* $\{x^k\}$ *and* $\{\nu_k\}$ *be the sequences generated by Algorithm* 1*, and* $\{z^k\}$ *be defined by* (3.3)*. Then, there exists* $M > 0$ *such that* $\|z^k\| \le M$ *and* $\sum_{s\in E_k} \nu_k(s) \le M$ *for all* k*.*

Proof. Let $\overline{x} \in \mathcal{K}$ and $\beta > 0$ be chosen so that $c(\overline{x}, s) \le -\beta$ for any $s \in \Omega$. (Such a vector and a positive number exist from Assumption A.) Then, we have

$$\begin{aligned} 0 &\le (z^k)^\top(\overline{x} - x^k) + \textstyle\sum_{s\in E_k} \nu_k(s) c(x^k, s) \\ &= \nabla f(x^k)^\top(\overline{x} - x^k) + \textstyle\sum_{s\in E_k} \nu_k(s)\big(\nabla_x c(x^k, s)^\top(\overline{x} - x^k) + c(x^k, s)\big) \\ &\le \nabla f(x^k)^\top(\overline{x} - x^k) + \textstyle\sum_{s\in E_k} \nu_k(s) c(\overline{x}, s) \\ &\le \nabla f(x^k)^\top(\overline{x} - x^k) - \beta \textstyle\sum_{s\in E_k} \nu_k(s), \end{aligned}$$

where the first inequality follows from $(z^k)^\top \overline{x} \ge 0$ and (3.4), the second inequality is due to the convexity of $c(\cdot, s)$, and the last inequality follows from $c(\overline{x}, s) \le -\beta$. Hence, we have $\sum_{s\in E_k} \nu_k(s) \le \nabla f(x^k)^\top(\overline{x} - x^k)/\beta$, which implies the boundedness of $\{\sum_{s\in E_k} \nu_k(s)\}$ since ∇f is continuous and $\{x^k\}$ is bounded. The boundedness of $\{\|z^k\|\}$ can be easily shown by the definition of z^k and the boundedness of $\{\sum_{s\in E_k} \nu_k(s)\}$. ∎

Proposition 4.5 *Let* $\{x^k\}$ *and* $\{\nu_k\}$ *be the sequences generated by Algorithm* 1*. Then, there exists* $\alpha > 0$ *such that* $\sum_{s\in E_k} \nu_k(s) \ge \alpha$ *for all* k*.*

Proof. Assume that the statement does not hold for contradiction. Then, Algorithm 1 does not terminate finitely, and $\liminf_{k\to\infty} \sum_{s\in E_k} \nu_k(s) = 0$. Since $\{x^k\}$ is bounded from Proposition 4.1, there exists $K \subseteq \{1, 2, \ldots\}$ such that $\overline{x} := \lim_{k\to\infty, k\in K} x^k$ and $\lim_{k\to\infty, k\in K} \sum_{s\in E_k} \nu_k(s) = 0$. Also, we have $\lim_{k\to\infty, k\in K} z^k = \nabla f(\overline{x})$ from (3.3) and the boundedness of $\{\nabla_x c(x^k, s)\}$. Thus, we have from (3.4) $\overline{x} \in \mathcal{K}$, $\nabla f(\overline{x}) \in \mathcal{K}$ and $\overline{x}^\top \nabla f(\overline{x}) = 0$, which imply $f(\overline{x}) = \min\{f(x) \mid x \in \mathcal{K}\}$ due to the convexity and the KKT conditions. Hence, by Theorem 2.1 and $\Omega_0 \subset \overline{E}_k$, we must have $f(\overline{x}) < V(\Omega_0) \le V(\overline{E}_k) = f(x^k)$. However, this contradicts $f(x^k) \le f(x^{k+1}) \le \cdots \le f(\overline{x})$. ∎

4.2 Finite termination for strictly convex case

We first prove that Algorithm 1 terminates in a finite number of iterations when f or $c(\cdot, s)$ is strictly convex.

Assumption B *At least one of the following statements holds*: (a) f *is strictly convex*; (b) $c(\cdot, s)$ *is strictly convex for any* $s \in \Omega$.

Theorem 4.1 *Suppose that Assumption* B *holds. Then, Algorithm* 1 *terminates in a finite number of iterations.*

Proof. Suppose to the contrary that Algorithm 1 does not finitely stop. Then, by Corollary 4.1, we have $V_\infty := \lim_{k\to\infty} f(x^k)$ with

$$f(x^1) \le f(x^2) \le \cdots \le f(x^k) \le f(x^{k+1}) \le \cdots \le V_\infty \le V^*. \tag{4.4}$$

Thus,

$$\lim_{k\to\infty} \left(f(x^{k+1}) - f(x^k) \right) = 0. \tag{4.5}$$

Let $s^k_{\min} := \operatorname{argmin}_{s\in E_k} P_k(s)$ for each k. (Notice that $E_k \neq \emptyset$ from Proposition 4.5.) Since $\{x^k\}$ and Ω are bounded, there exist $\overline{x}$, $\overline{d} \in \mathbb{R}^n$, $\overline{s}_{\min} \in \Omega$, and an index set $K \subseteq \{1, 2, \ldots\}$ such that $\lim_{k\to\infty, k\in K}(x^k, d^k, s^k_{\min}) = (\overline{x}, \overline{d}, \overline{s}_{\min})$. Hence, by (4.1) and its nonnegativity of each term, we have

$$0 = \lim_{k\to\infty, k\in K} F_k = f(\overline{x} + \overline{d}) - f(\overline{x}) - \nabla f(\overline{x})^\top \overline{d}, \tag{4.6}$$

$$0 = \lim_{k\to\infty, k\in K} \sum_{s\in E_k} \nu_k(s) P_k(s). \tag{4.7}$$

If (a) of Assumption B holds, then we have $\overline{d} = 0$ from (4.6) and the strict convexity of f. However, this contradicts Proposition 4.3. If (b) of Assumption B holds, then (4.7) together with $\sum_{s\in E_k} \nu_k(s) \ge \alpha > 0$ from Proposition 4.5 yields that

$$\begin{aligned} 0 = \lim_{\substack{k\to\infty \\ k\in K}} \sum_{s\in E_k} \nu_k(s) P_k(s) &\ge \lim_{\substack{k\to\infty \\ k\in K}} P_k(s^k_{\min}) \sum_{s\in E_k} \nu_k(s) \\ &\ge \left(c(\overline{x} + \overline{d}, \overline{s}_{\min}) - c(\overline{x}, \overline{s}_{\min}) - \nabla_x c(\overline{x}, \overline{s}_{\min})^\top \overline{d} \right) \alpha, \end{aligned}$$

which implies $c(\overline{x} + \overline{d}, \overline{s}_{\min}) - c(\overline{x}, \overline{s}_{\min}) - \nabla_x c(\overline{x}, \overline{s}_{\min})^\top \overline{d} = 0$. Since $c(\cdot, \overline{s}_{\min})$ is strictly convex, we have $\overline{d} = 0$. However, it also contradicts Proposition 4.3. ■

4.3 Finite termination without strict convexity

In this section, we show the finite termination of Algorithm 1 without assuming the strict convexity of f or $c(\cdot, s)$. First we give a lemma that plays a crucial role in later analyses.

Lemma 4.1 *Let* $\theta : \mathbb{R}^n \to \mathbb{R}$ *be an arbitrary continuously differentiable convex function, and* x, $y \in \mathbb{R}^n$ *be arbitrary vectors. Then we have*

$$\theta(y) - \theta(x) - \nabla\theta(x)^\top (y - x) = 0 \iff \theta(x) - \theta(y) + \nabla\theta(y)^\top (y - x) = 0.$$

Proof. The above formula holds evidently when $x = y$. Also, if we have $(\Rightarrow)$, then $(\Leftarrow)$ holds automatically by swapping x for y. Therefore, we only show $(\Rightarrow)$ with $x \neq y$.

Let x and y be arbitrary vectors such that $x \neq y$ and $\theta(y) - \theta(x) - \nabla\theta(x)^\top(y-x) = 0$. Choose $\alpha \in (0,1)$ arbitrarily. Then, we have

$$\begin{aligned} 0 &= \alpha\theta(y) - \alpha\theta(x) - \alpha\nabla\theta(x)^\top(y-x) \\ &= \Big[(1-\alpha)\theta(x) + \alpha\theta(y) - \theta\big((1-\alpha)x + \alpha y\big)\Big] \\ &\quad + \Big[\theta\big((1-\alpha)x + \alpha y\big) - \theta(x) - \nabla\theta(x)^\top\big((1-\alpha)x + \alpha y - x\big)\Big], \end{aligned}$$

which implies

$$(1-\alpha)\theta(x) + \alpha\theta(y) - \theta((1-\alpha)x + \alpha y) = 0 \tag{4.8}$$

and $\theta((1-\alpha)x + \alpha y) - \theta(x) - \nabla\theta(x)^\top((1-\alpha)x + \alpha y - x) = 0$ since θ is convex. Hence, we have

$$\begin{aligned} -\nabla\theta(y)^\top(y-x) &= \nabla\theta(y)^\top(x-y) \\ &= \lim_{t\downarrow 0}\big(\theta((1-t)y + tx)) - \theta(y)\big)/t \\ &= \lim_{t\downarrow 0}\big((1-t)\theta(y) + t\theta(x)) - \theta(y)\big)/t \\ &= \theta(x) - \theta(y), \end{aligned}$$

where the third equality follows from (4.8) with $\alpha := 1-t$. This completes the proof. ■

In order to show the finite iteration of the algorithm, we introduce the following assumption.

Assumption C *There exist $\hat{k} \geq 0$ and $\delta > 0$ such that the following statements holds.*

(i) *For all $k \geq \hat{k}$, it follows* $\max(\hat{x}_{ij}^k, \hat{z}_{ij}^k) \geq \delta$.

(ii) *For all $k \geq \hat{k}$ and $s \in E_k$, it follows* $\max(\nu_k(s), -c(x^k, s)) \geq \delta$.

(iii) *For all $k \geq \hat{k}$, it follows* $\nu_{k+1}(s_{\text{new}}^k) \geq \delta$.

(iv) *For all $k \geq \hat{k}$, it follows* $\nabla_x c(x^{k+1}, s_{\text{new}}^k)^\top(x^k - x^{k+1}) \geq \delta\|x^k - x^{k+1}\|$.

We note that this assumption can be checked in each iteration of the algorithm. Statements (i) and (ii) claim that the complementarity conditions in (3.4) should be satisfied sufficiently strictly. Due to the complementarity, we always have $\min(\nu_k(s), -c(x^k, s)) = \min(\hat{x}_{ij}^k, \hat{z}_{ij}^k) = 0$. Statement (ii) also implies $\nu_k(s) \geq \delta$ for all $s \in E_k \backslash \Omega_0$ since all inactive indices not belonging to Ω_0 are removed in Step 3. Statement (iv) requires that the angle between $\nabla_x c(x^{k+1}, s_{\text{new}}^k)$ and $x^k - x^{k+1}$ should be strictly less than $\pi/2$. (See Figures 1 and 2.) Although this statement may not be intuitively recognizable, it was satisfied for all test problems in our numerical experiment. Indeed, $\nabla_x c(x^{k+1}, s_{\text{new}}^k)^\top(x^k - x^{k+1})$ is guaranteed to be at least nonnegative when $\nu_k(s_{\text{new}}^k) > 0$.[*1] Moreover, as the following three propositions (Propositions 4.6−4.8) indicate, we have some mild and intuitive conditions under which Assumption C (iv) holds.

Proposition 4.6 *Suppose that $\mathcal{K}$ consists of a single SOC, i.e., $\mathcal{K} = \mathcal{K}^n$. Moreover, assume that there exist $\hat{k} \geq 0$ and $\delta > 0$ such that Assumption* C (i) *holds, $|\nabla f(x^k)^\top x^k| \geq \delta$, and $x^k \in \operatorname{bd}\mathcal{K}^n \setminus \{0\}$ for all $k \geq \hat{k}$. Then, Assumption* C (iv) *holds.*

[*1] By (4.3) together with the nonnegativity of $(x^k)^\top z^{k+1}$, G_k, ν_{k+1} and $Q_k(s)$, we have $0 \leq f(x^{k+1}) - f(x^k) \leq \nu_{k+1}(s_{\text{new}}^k)\nabla_x c(x^{k+1}, s_{\text{new}}^k)^\top(x^k - x^{k+1})$. Dividing both sides by $\nu_{k+1}(s_{\text{new}}^k) > 0$, we obtain the nonnegativity result.

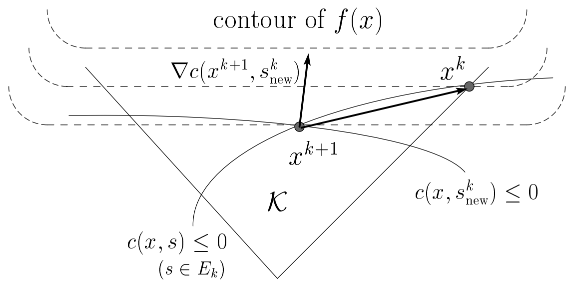

Figure 1: Assumption C (iv) holds.

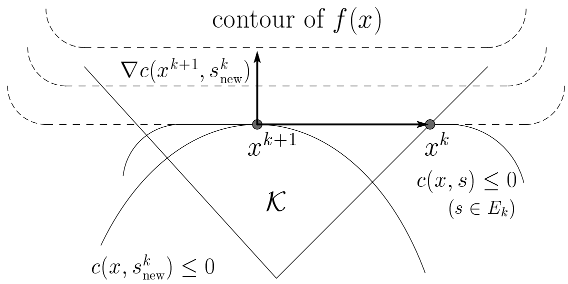

Figure 2: Assumption C (iv) does not hold.

Proof. The convexity of $c(\cdot, s)$ together with $0 \le \nu_{k+1}(s) \perp -c(x^{k+1}, s) \ge 0$ yields

$$\textstyle\sum_{s\in E_k} \nu_{k+1}(s)\nabla_x c(x^{k+1}, s)^\top (x^{k+1} - x^k) \ \ge\ \sum_{s\in E_k} \nu_{k+1}(s)\big(c(x^{k+1}, s) - c(x^k, s)\big) \ \ge\ 0. \qquad (4.9)$$

Moreover, noticing $z^{k+1} = \nabla f(x^{k+1}) + \nu_{k+1}(s^k_{\mathrm{new}})\nabla_x c(x^{k+1}, s^k_{\mathrm{new}}) + \sum_{s\in E_k} \nu_{k+1}(s)\nabla_x c(x^{k+1}, s)$, we have

$$\begin{aligned}
&-\nabla_x c(x^{k+1}, s^k_{\mathrm{new}})^\top (x^{k+1} - x^k) \\
&\quad = \big[(\nabla f(x^{k+1}) - z^{k+1})^\top (x^{k+1} - x^k) + \textstyle\sum_{s\in E_k} \nu_{k+1}(s)\nabla_x c(x^{k+1}, s)^\top (x^{k+1} - x^k)\big] \ /\ \nu_{k+1}(s^k_{\mathrm{new}}) \\
&\quad \ge (\nabla f(x^{k+1}) - z^{k+1})^\top (x^{k+1} - x^k) \ /\ \nu_{k+1}(s^k_{\mathrm{new}}) \\
&\quad \ge (\nabla f(x^{k+1}) - z^{k+1})^\top (x^{k+1} - x^k)\|x^{k+1} - x^k\| \ /\ (2M^2),
\end{aligned}$$

where the first inequality follows from (4.9), and the last inequality holds since $\|x^{k+1} - x^k\| \le \|x^{k+1}\| + \|x^k\| \le 2M$ and $\nu_{k+1}(s^k_{\mathrm{new}}) \le M$ by Propositions 4.1 and 4.4, respectively. Hence, it suffices to show the existence of a positive number $\delta' > 0$ such that $(\nabla f(x^{k+1}) - z^{k+1})^\top (x^{k+1} - x^k) \ge \delta'$ for all k sufficiently large. Suppose for contradiction that $\liminf_{k\to\infty}(\nabla f(x^{k+1}) - z^{k+1})^\top (x^{k+1} - x^k) \le 0$. Then we have $\liminf_{k\to\infty} \nabla f(x^{k+1})^\top (x^{k+1} - x^k) = \liminf_{k\to\infty} (z^{k+1})^\top (x^{k+1} - x^k) = 0$ since $\nabla f(x^{k+1})^\top (x^{k+1} - x^k) \ge 0$ and $-(z^{k+1})^\top (x^{k+1} - x^k) \ge 0$. Thus, due to the boundedness of $\{z^k\}$ and $\{x^k\}$, there exist $K \subset \{0, 1, 2, \ldots\}$ and vectors $\overline{x}$, $\overline{x}^+$ and $\overline{z}^+$ such that $\lim_{k\to\infty, k\in K}(x^k, x^{k+1}, z^{k+1}) = (\overline{x}, \overline{x}^+, \overline{z}^+)$ and

$$\nabla f(\overline{x}^+)^\top (\overline{x}^+ - \overline{x}) = (\overline{z}^+)^\top (\overline{x}^+ - \overline{x}) = 0. \qquad (4.10)$$

By $x^{k+1} \in \mathcal{K}^n \setminus \{0\}$ for all $k \ge \hat{k}$ and Assumption C (i), we must have $\overline{x}^+ \in \mathcal{K}^n \setminus \{0\}$, which together with $\mathcal{K}^n \ni \overline{z}^+ \perp \overline{x}^+ \in \mathcal{K}^n$ and Assumption C (i) yields $\overline{z}^+ \in \mathcal{K}^n \setminus \{0\}$. Moreover, since (4.10) implies $\mathcal{K}^n \ni \overline{z}^+ \perp \overline{x} \in \mathcal{K}^n$, there must exist a nonnegative scalar[*2] $\beta \ge 0$ such that $\overline{x} = \beta\overline{x}^+$ and $\beta \ne 1$. (Notice that $\overline{x}^+ \ne \overline{x}$.) Hence, by (4.10), we have $0 = \nabla f(\overline{x}^+)^\top (\overline{x}^+ - \overline{x}) = (1 - \beta)\nabla f(\overline{x}^+)^\top \overline{x}^+$, i.e., $\nabla f(\overline{x}^+)^\top \overline{x}^+ = 0$. However, this contradicts the assumption that $|\nabla f(x^k)^\top x^k| \ge \delta$ for all $k \ge \hat{k}$. Thus, Assumption C (iv) holds. ∎

Proposition 4.7 *If* $c(\cdot, s)$ *is affine for any* $s \in \Omega$*, then Assumption* C (iv) *holds.*

Proof. Since Proposition 4.1 implies $\|x^k - x^{k+1}\| \le \|x^k\| + \|x^{k+1}\| \le 2M$, we have

$$\nabla_x c(x^{k+1}, s^k_{\mathrm{new}})^\top (x^k - x^{k+1}) = c(x^k, s^k_{\mathrm{new}}) - c(x^{k+1}, s^k_{\mathrm{new}}) > \eta > \frac{\eta}{2M}\|x^{k+1} - x^k\|,$$

where the equality holds since $c(\cdot, s^k_{\mathrm{new}})$ is affine, and the first inequality follows from $c(x^k, s^k_{\mathrm{new}}) > \eta$ and $c(x^{k+1}, s^k_{\mathrm{new}}) \le 0$. Hence, letting $\delta := \eta/(2M)$, we obtain Assumption C (iv). ∎

[*2]Notice that, if $\mathcal{K}$ consists of multiple SOCs, then the three points $\overline{x}$, $\overline{x}^+$ and 0 may not be collinear.

Proposition 4.8 *Suppose that function c is quadratic with respect to x, i.e., it is given as*

$$c(x, s) := x^\top M(s)x + 2q(s)^\top x + r(s),$$

where $M : \Omega \to \mathbb{R}^{n\times n}$, $q : \Omega \to \mathbb{R}^n$ and $r : \Omega \to \mathbb{R}$ are continuous, and $M(s) \in \mathbb{R}^{n\times n}$ is symmetric and positive semidefinite for any $s \in \Omega$. Moreover, assume that $q(s)^\top \xi \neq 0$ for any $s \in \Omega$ and $\xi \in \mathbb{R}^n \setminus \{0\}$ with $\xi^\top M(s)\xi = 0$. Then, Assumption C(iv) *holds.*

Proof. Let $s^k \in E_k$ be an arbitrary element with $\nu_k(s^k) > 0$. Let $\xi^k \in \mathbb{R}^n$ be any vector such that $\|\xi^k\| = 1$ and $\nabla_x c(x^k, s^k)^\top \xi^k = 2(M(s^k)x^k + q(s^k))^\top \xi^k = 0$. Then we have

$$\begin{aligned} c(x^k + \xi^k, s^k) &= (x^k + \xi^k)^\top M(s^k)(x^k + \xi^k) + 2q(s^k)^\top (x^k + \xi^k) + r(s^k) \\ &= \left[(x^k)^\top M(s^k)x^k + 2q(s^k)^\top x^k + r(s^k)\right] + (\xi^k)^\top M(s^k)\xi^k + 2(M(s^k)x^k + q(s^k))^\top \xi^k \\ &= (\xi^k)^\top M(s^k)\xi^k, \end{aligned}$$

where the last equality follows since $\nabla_x c(x^k, s^k)^\top \xi^k = 0$ and $c(x^k, s^k) = 0$ from $\nu_k(s^k) > 0$.

We first show that there exists $\delta' > 0$ such that

$$(\xi^k)^\top M(s^k)\xi^k \geq \delta' \tag{4.11}$$

for all k. Suppose for contradiction that there does not exist such an $\delta' > 0$. Then, we must have $\liminf_{k\to\infty} (\xi^k)^\top M(s^k)\xi^k = 0$. Since $\{x^k\}$, $\{\xi^k\}$ and $\{s^k\}$ are bounded and $M(s)$ is continuous, there exists $K \subset \{0, 1, \ldots\}$ such that $\lim_{k\to\infty, k\in K} x^k = \overline{x}$, $\lim_{k\to\infty, k\in K} \xi^k = \overline{\xi}$, $\lim_{k\to\infty, k\in K} s^k = \overline{s}$, and $\overline{\xi}^\top M(\overline{s})\overline{\xi} = 0$. Notice that $\overline{\xi}^\top M(\overline{s})\overline{\xi} = 0$ implies $M(\overline{s})\overline{\xi} = 0$. Moreover, we have $\nabla_x c(\overline{x}, \overline{s})^\top \overline{\xi} = 0$ since $\nabla_x c(x^k, s^k)^\top \xi^k = 0$ for all k. We thus have $0 = \nabla_x c(\overline{x}, \overline{s})^\top \overline{\xi} = 2(M(\overline{s})\overline{x} + q(\overline{s}))^\top \overline{\xi} = q(\overline{s})^\top \overline{\xi}$. However, this contradicts the assumption.

Next, we show that there exists $\delta'' > 0$ such that

$$-\nabla_x c(x^k, s^k)^\top (x^{k+1} - x^k) \geq \delta'' \tag{4.12}$$

for all k. Let $\mathcal{P}_k := \left\{x \mid \nabla_x c(x^k, s^k)^\top (x - x^k)\right\} = 0$, $\tilde{x}^{k+1}$ be the Euclidean projection of x^{k+1} onto $\mathcal{P}_k$, and θ_k be the angle between $x^{k+1} - x^k$ and $\tilde{x}^{k+1} - x^k$. Notice that we always have $0 \leq \theta_k \leq \pi/2$ since $(x^{k+1} - x^k)^\top (\tilde{x}^{k+1} - x^k) = \left((\tilde{x}^{k+1} - x^k) - (\tilde{x}^{k+1} - x^{k+1})\right)^\top (\tilde{x}^{k+1} - x^k) = \|\tilde{x}^{k+1} - x^k\|^2$, where the last equality follows from $(\tilde{x}^{k+1} - x^{k+1}) \perp (\tilde{x}^{k+1} - x^k)$. If $\theta_k > \pi/4$, then we have $-\nabla_x c(x^k, s^k)^\top (x^{k+1} - x^k) = \|\nabla_x c(x^k, s^k)\| \|x^{k+1} - x^k\| \cos(\pi/2 - \theta_k) > \gamma d_{\min}/\sqrt{2}$, where $d_{\min} > 0$ is defined in Proposition 4.3 and $\gamma > 0$ is the positive number such that

$$\|\nabla_x c(x^k, s^k)\| \geq \gamma \tag{4.13}$$

for all k.[*3] If $\theta_k \leq \pi/4$, then we have $\|\tilde{x}^{k+1} - x^k\| = \|x^{k+1} - x^k\| \cos\theta_k \geq d_{\min}/\sqrt{2}$. Hence, letting $\xi^k := (\tilde{x}^{k+1} - x^k)/\|\tilde{x}^{k+1} - x^k\|$ and $\tau_k := \|\tilde{x}^{k+1} - x^k\|$, we have

$$\begin{aligned} c(\tilde{x}^{k+1}, s^k) &= (\tilde{x}^{k+1})^\top M(s^k)\tilde{x}^{k+1} + 2q(s^k)^\top \tilde{x}^{k+1} + r(s^k) \\ &= (x^k + \tau_k\xi^k)^\top M(s^k)(x^k + \tau_k\xi^k) + 2q(s^k)^\top (x^k + \tau_k\xi^k) + r(s^k) \\ &= \tau_k^2 (\xi^k)^\top M(s^k)\xi^k \\ &\geq d_{\min}^2 \delta'/2, \end{aligned}$$

where the last equality follows since $c(x^k, s^k) = 0$ and $\nabla_x c(x^k, s^k)^\top \xi^k = 0$, and the inequality is due to (4.11) and $\tau_k = \|\tilde{x}^{k+1} - x^k\| \geq d_{\min}/\sqrt{2}$. Since $c(x^{k+1}, s^k) \leq 0$ and c is locally Lipschitzian, there

[*3] Assumption A (iii) and Proposition 4.1 yield $0 < -c(\overline{x}, \overline{s}_{\min}) \leq c(x^k, s^k) - c(\overline{x}, s^k) \leq \nabla_x c(x^k, s^k)^\top (x^k - \overline{x}) \leq \|\nabla_x c(x^k, s^k)\|(M + \|\overline{x}\|)$, where $\overline{s}_{\min} := \operatorname{argmin}_{s\in\Omega} c(\overline{x}, s)$. Therefore, we can choose $\gamma := -c(\overline{x}, \overline{s}_{\min})/(M + \|\overline{x}\|) > 0$.

exists $L > 0$ such that $d_{\min}^2 \delta'/2 \leq c(\tilde{x}^{k+1}, s^k) - c(x^{k+1}, s^k) \leq L\|\tilde{x}^{k+1} - x^{k+1}\|$, that is, $\|\tilde{x}^{k+1} - x^{k+1}\| \geq d_{\min}^2 \delta'/(2L)$. So we have

$$-\nabla_x c(x^k, s^k)^\top (x^{k+1} - x^k) = \|\nabla_x c(x^k, s^k)\| \|\tilde{x}^{k+1} - x^{k+1}\| \geq \gamma d_{\min}^2 \delta'/(2L),$$

where $\gamma > 0$ is a positive number given by (4.13).

Finally, we show that Assumption C(iv) holds. Notice that

$$\begin{aligned} \nabla f(x^{k+1})^\top (x^{k+1} - x^k) &\geq \nabla f(x^k)^\top (x^{k+1} - x^k) \\ &= \left[z^k - \textstyle\sum_{s \in E_k} \nu_k(s) \nabla_x c(x^k, s)\right]^\top (x^{k+1} - x^k) \\ &\geq \textstyle\sum_{s \in E_k} \nu_k(s) \delta'' \geq \alpha \delta'', \end{aligned} \tag{4.14}$$

where the first inequality is due to the convexity of f, the second inequality follows from (4.12) and $(z^k)^\top x^k = 0 \leq (z^k)^\top x^{k+1}$, and the last inequality is due to Proposition 4.5. We thus have

$$\begin{aligned} &\nu_{k+1}(s_{\text{new}}^k) \nabla_x c(x^{k+1}, s_{\text{new}}^k)^\top (x^k - x^{k+1}) \\ &\quad = -\nu_{k+1}(s_{\text{new}}^k) \nabla_x c(x^{k+1}, s_{\text{new}}^k)^\top (x^{k+1} - x^k) \\ &\quad = \Big[- z^{k+1} + \nabla f(x^{k+1}) + \textstyle\sum_{s \in E_k} \nu_{k+1}(s) \nabla_x c(x^{k+1}, s) \Big]^\top (x^{k+1} - x^k) \\ &\quad \geq \alpha \delta'' + \textstyle\sum_{s \in E_k} \nu_{k+1}(s) \Big[Q_k(s) - c(x^k, s) + c(x^{k+1}, s) \Big] \\ &\quad \geq \alpha \delta'', \end{aligned} \tag{4.15}$$

where the first inequality is due to (4.14), $(z^{k+1})^\top x^{k+1} = 0 \leq (z^{k+1})^\top x^k$, and the definition of $Q_k(s)$, and the last inequality follows from $\nu_{k+1}(s) \geq 0$, $Q_k(s) \geq 0$, $c(x^k, s) \leq 0$ and $\nu_{k+1}(s) c(x^{k+1}, s) = 0$. By Propositions 4.1 and 4.4, we have $\nu_{k+1}(s_{\text{new}}^k) \leq M$ and $\|x^k - x^{k+1}\| \leq \|x^k\| + \|x^{k+1}\| \leq 2M$. Hence, dividing both sides of (4.15) by $\nu_{k+1}(s_{\text{new}}^k) > 0$, we obtain

$$\nabla_x c(x^{k+1}, s_{\text{new}}^k)^\top (x^k - x^{k+1}) \geq \frac{\alpha \delta''}{\nu_{k+1}(s_{\text{new}}^k)} \geq \frac{\alpha \delta''}{2M^2} \|x^k - x^{k+1}\|.$$

■

Now, by using the aforementioned assumption and lemma, we provide the theorem for the finite iteration of Algorithm 1.

Theorem 4.2 *Suppose that Assumption* C *holds. Then, Algorithm* 1 *terminates in a finite number of iterations.*

Proof. Suppose to the contrary that Algorithm 1 does not finitely terminate. Then, by Corollary 4.1 we have

$$f(x^1) \leq \cdots \leq f(x^k) \leq f(x^{k+1}) \leq \cdots \leq V^*,$$

which implies

$$\lim_{k \to \infty} (f(x^{k+1}) - f(x^k)) = 0. \tag{4.16}$$

Hence, each term in (4.1) also converges to 0 due to its nonnegativity. Let $E_k^+ := \{s \in E_k \mid \nu_k(s) > 0\} = \{s \in E_k \mid \nu_k(s) \geq \delta\}$, and $s_{\max}^k := \operatorname{argmax}_{s \in E_k^+} Q_k(s)$. Moreover, noticing the boundedness of $\{x^k\}$ and Ω, let $(\overline{x}, \overline{x}_+, \overline{s}_{\max}) \in \mathbb{R} \times \mathbb{R} \times \Omega$ be an arbitrary accumulation point of $\{(x^k, x^{k+1}, s_{\max}^k)\}$. Then, there exists an index set $K \subseteq \{0, 1, 2, \ldots\}$ such that $\lim_{k \to \infty, k \in K} (x^k, x^{k+1}, s_{\max}^k) = (\overline{x}, \overline{x}_+, \overline{s}_{\max})$. Since we have Proposition 4.3, it must hold $\overline{x} \neq \overline{x}_+$.

We first show that, for each $s \in \Omega_0$, there exists $\bar{k}$ such that either

$$\nu_k(s) \geq \delta \ (\forall k \geq \bar{k}) \ \text{ or } \ \nu_k(s) = 0 \ (\forall k \geq \bar{k}). \tag{4.17}$$

Fix $s \in \Omega_0$ arbitrarily. Then, by Assumption C(ii), we must have either $\limsup_{k\to\infty} \nu_k(s) = 0$ or $\limsup_{k\to\infty} \nu_k(s) \geq \delta$. If $\limsup_{k\to\infty} \nu_k(s) = 0$, then we obviously have $\nu_k(s) = 0$ for all k sufficiently large. If $\limsup_{k\to\infty} \nu_k(s) \geq \delta$, then there exists $K' \subset \{1, 2, \ldots\}$ such that $|K'| = \infty$ and $\nu_k(s) \geq \delta$ for all $k \in K'$. Since $\nu_k(s)c(x^{k+1}, s)$ converges to 0, we have an $\varepsilon \in (0, \delta^2)$ and $\bar{k}' \geq \hat{k}$ such that $0 \leq -\nu_k(s)c(x^{k+1}, s) < \varepsilon$ for all $k \geq \bar{k}'$. Now, choose an arbitrary $\bar{k} \geq \bar{k}'$ such that $\nu_{\bar{k}}(s) \geq \delta$. Then, we have $0 \leq -c(x^{\bar{k}+1}, s) < \varepsilon/\delta < \delta$, which implies $c(x^{\bar{k}+1}, s) = 0$ and $\nu_{\bar{k}+1}(s) \geq \delta$ from Assumption C(ii). We thus obtain (4.17) recursively.

We next show

$$\lim_{k\to\infty} \sum_{s\in E_k} \nu_{k+1}(s)c(x^k, s) = 0, \quad \lim_{k\to\infty} G_k = 0, \quad \lim_{k\to\infty} \sum_{s\in E_k} \nu_{k+1}(s)Q_k(s) = 0. \tag{4.18}$$

We readily have $\lim_{k\to\infty} \sum_{s\in E_k} \nu_{k+1}(s)c(x^k, s) = 0$ since (4.17) implies that either $\nu_{k+1}(s)$ or $c(x^k, s)$ is 0 for all k sufficiently large. Since $\lim_{k\to\infty} F_k = 0$, we have

$$\lim_{k\to\infty, k\in K} F_k = f(\overline{x}_+) - f(\overline{x}) - \nabla f(\overline{x})^\top(\overline{x}_+ - \overline{x}) = 0.$$

Hence, by Lemma 4.1, we have

$$\lim_{k\to\infty, k\in K} G_k = f(\overline{x}) - f(\overline{x}_+) + \nabla f(\overline{x}_+)^\top(\overline{x}_+ - \overline{x}) = 0.$$

Since $\overline{x}$ and $\overline{x}_+$ are arbitrary accumulation points, the above equality implies $\lim_{k\to\infty} G_k = 0$. Also, since $\lim_{k\to\infty} \sum_{s\in E_k} \nu_k(s)P_k(s) = 0$, $P_k(s) \geq 0$, $s^k_{\max} \in E_k^+$, and $\nu_k(s) \geq \delta$ for all $s \in E_k^+$, we have

$$0 = \lim_{k\to\infty, k\in K} P_k(s^k_{\max}) = c(\overline{x}_+, \overline{s}_{\max}) - c(\overline{x}, \overline{s}_{\max}) - \nabla_x c(\overline{x}, \overline{s}_{\max})^\top(\overline{x}_+ - \overline{x}).$$

Hence, by Lemma 4.1, we have $\lim_{k\to\infty, k\in K} Q_k(s^k_{\max}) = 0$. Now, by Proposition 4.4, there exists $M > 0$ such that $\sum_{s\in E_k^+} \nu_{k+1}(s) \leq \sum_{s\in \overline{E}_{k+1}} \nu_{k+1}(s) = \sum_{s\in E_{k+1}} \nu_{k+1}(s) \leq M$ for all k. Moreover, $\nu_{k+1}(s) = 0$ for all $s \in E_k \setminus E_k^+$ and sufficiently large k since we have (4.17) and $E_k \setminus E_k^+ \subset \Omega_0$. We thus have

$$0 \leq \sum_{s\in E_k} \nu_{k+1}(s)Q_k(s) = \sum_{s\in E_k^+} \nu_{k+1}(s)Q_k(s) \leq Q_k(s^k_{\max}) \sum_{s\in E_k^+} \nu_{k+1}(s) \leq MQ_k(s^k_{\max}),$$

which yields $\lim_{k\to\infty, k\in K} \sum_{s\in E_k} \nu_{k+1}(s)Q_k(s) = 0$. Since $\overline{x}$, $\overline{x}_+$ and $\overline{s}_{\max}$ are arbitrary accumulation points, we have $\lim_{k\to\infty} \sum_{s\in E_k} \nu_{k+1}(s)Q_k(s) = 0$.

Now, choose a sufficiently small number $\varepsilon > 0$ arbitrarily. By Assumption C (iv) and Proposition 4.3, we have

$$-\nabla_x c(x^{k+1}, s^k_{\text{new}})^\top d^k = \nabla_x c(x^{k+1}, s^k_{\text{new}})^\top(x^k - x^{k+1}) \geq \delta\|x^{k+1} - x^k\| = \delta d_{\min} > 0. \tag{4.19}$$

Hence, by (4.1) and (4.3) together with (4.16), (4.18) and (4.19), we have some positive integer $L = L(\varepsilon) \geq \hat{k}$ such that

$$0 \leq (z^k)^\top x^{k+1} < \varepsilon, \quad (x^k)^\top z^{k+1} > \frac{\delta d_{\min}}{2} =: \gamma \tag{4.20}$$

for all $k \geq L$. Choose $k \geq L$ arbitrarily, and let $\mathcal{I}_1^k$ and $\mathcal{I}_2^k$ be defined as

$$\mathcal{I}_1^k := \{(i,j) \mid \hat{x}^k_{ij} > 0\}, \quad \mathcal{I}_2^k := \{(i,j) \mid \hat{x}^k_{ij} = 0\}.$$

Then, we note that $\mathcal{I}_1^k \cup \mathcal{I}_2^k = \{1,2\} \times \{1,\dots,m\}$, and

$$\begin{aligned} (i,j) \in \mathcal{I}_1^k &\iff \hat{x}_{ij}^k \geq \delta \iff \hat{z}_{ij}^k = 0, \\ (i,j) \in \mathcal{I}_2^k &\iff \hat{x}_{ij}^k = 0 \iff \hat{z}_{ij}^k \geq \delta \end{aligned} \tag{4.21}$$

from Proposition 3.1(a) and Assumption C (i). Let $(i',j') \in \mathcal{I}_2^k$ be chosen arbitrarily[*4]. Then we have

$$\varepsilon > \sum_{i=1}^{2}\sum_{j=1}^{m} \hat{z}_{ij}^k \hat{x}_{ij}^{k+1} (\hat{e}_{ij}^k)^\top \hat{e}_{ij}^{k+1} \geq \hat{z}_{i'j'}^k \hat{x}_{i'j'}^{k+1} (\hat{e}_{i'j'}^k)^\top \hat{e}_{i'j'}^{k+1} \geq \frac{1}{2}\hat{x}_{i'j'}^{k+1}\delta,$$

where the second inequality is due to the nonnegativity of $\hat{x}_{ij}^{k+1}$, $\hat{z}_{ij}^k$ and $(\hat{e}_{ij}^{k+1})^\top \hat{e}_{ij}^k$, and the last inequality follows from Proposition 3.1(c) and $(i',j') \in \mathcal{I}_2^k$. Since $\varepsilon > 0$ can be chosen arbitrarily small and we have (4.21), the above inequality means $\hat{x}_{i'j'}^{k+1} = 0$. Hence, we have $\mathcal{I}_2^k \subset \mathcal{I}_2^{k+1}$.

Now, by (4.20), we have

$$\begin{aligned} \gamma &< \sum_{i=1}^{2}\sum_{j=1}^{m} \hat{z}_{ij}^{k+1} \hat{x}_{ij}^k (\hat{e}_{ij}^{k+1})^\top \hat{e}_{ij}^k \\ &= \sum_{(i,j)\in\mathcal{I}_1^{k+1}} \hat{z}_{ij}^{k+1} \hat{x}_{ij}^k (\hat{e}_{ij}^{k+1})^\top \hat{e}_{ij}^k + \sum_{(i,j)\in\mathcal{I}_2^{k}} \hat{z}_{ij}^{k+1} \hat{x}_{ij}^k (\hat{e}_{ij}^{k+1})^\top \hat{e}_{ij}^k + \sum_{(i,j)\in\mathcal{I}_2^{k+1}\setminus\mathcal{I}_2^k} \hat{z}_{ij}^{k+1} \hat{x}_{ij}^k (\hat{e}_{ij}^{k+1})^\top \hat{e}_{ij}^k \\ &= \sum_{(i,j)\in\mathcal{I}_2^{k+1}\setminus\mathcal{I}_2^k} \hat{z}_{ij}^{k+1} \hat{x}_{ij}^k (\hat{e}_{ij}^{k+1})^\top \hat{e}_{ij}^k, \end{aligned}$$

which implies $\mathcal{I}_2^{k+1} \setminus \mathcal{I}_2^k \neq \emptyset$. Since $k \geq L$ can be chosen arbitrarily, it must hold

$$|\mathcal{I}_2^L| < |\mathcal{I}_2^{L+1}| < |\mathcal{I}_2^{L+2}| < \cdots.$$

However, this contradicts the boundedness of $\{|\mathcal{I}_2^k|\}$. Thus, Algorithm 1 must terminate in a finite number of iterations. ■

Next, we derive another finite termination theorem for the single SOC case, i.e., $\mathcal{K} = \mathcal{K}^n$ and $n \geq 2$. In this case, we can show the finite termination result without Assumption C (i).

Theorem 4.3 *Suppose that $\mathcal{K} = \mathcal{K}^n$ with $n \geq 2$, and* (ii)–(iv) *of Assumption* C *hold. Then, Algorithm* 1 *terminates in a finite number of iterations.*

Proof. Assume that Algorithm 1 does not terminate in finitely many iterations for contradiction. Then, Corollary 4.1 implies the existence of $V_\infty := \lim_{k\to\infty} f(x^k) \leq V^*$. Also, by Propositions 4.1 and 4.4, there exists $M > 0$ such that $\|x^k\| < M$ and $\|z^k\| < M$ for all k.

Choose any small ε such that

$$0 < \varepsilon < \min\Big\{\gamma, \frac{\gamma^3}{M^4}\Big\}, \tag{4.22}$$

where $\gamma = \delta d_{\min}/2 > 0$. (See (4.20) and Proposition 4.3.) Then, by using a technique similar to the proof of Theorem 4.2, we obtain an integer $L = L(\varepsilon) \geq \bar{k}$ such that

$$0 \leq (z^k)^\top x^{k+1} < \varepsilon, \tag{4.23}$$

$$\gamma < (z^{k+1})^\top x^k \tag{4.24}$$

[*4] When $\mathcal{I}_2^k = \emptyset$, we immediately obtain the desired result $\mathcal{I}_2^k \subseteq \mathcal{I}_2^{k+1}$.

for all $k \geq L$. Now, notice that (4.24) implies $x^{k+1} \notin \operatorname{int} \mathcal{K}^n$ for all $k \geq L$ since it must hold $z^{k+1} = 0$ when $x^{k+1} \in \operatorname{int} \mathcal{K}^n$. Also we obviously have $x^k \neq 0$ from (4.24). Therefore, we have $x^k \in \operatorname{bd} \mathcal{K}^n \setminus \{0\}$ for all $k \geq L+1$. Similarly, we have $z^k \in \operatorname{bd} \mathcal{K}^n \setminus \{0\}$ for all $k \geq L+1$. Thus, by Proposition 3.1, we have

$$x^k = \hat{x}^k_{i_k} \hat{e}^k_{i_k} = \|x^k\| \hat{e}^k_{i_k} \text{ and } z^k = \hat{z}^k_{3-i_k} \hat{e}^k_{3-i_k} = \|z^k\| \hat{e}^k_{3-i_k} \tag{4.25}$$

for some $i_k \in \{1,2\}$, where $\hat{x}^k_{i_k}$, $\hat{z}^k_{i_k}$ and $\hat{e}^k_{i_k}$ are the scalars and the vector defined by (3.6)–(3.7). (Notice that the subscript j can be omitted without confusion since we have only one SOC.) Now, choose a positive integer $r \geq L+2$ arbitrarily. Then (4.23)–(4.25) hold for $k = r-1$, r and $r+1$.

Since (4.24) implies $\|z^r\|\|x^{r-1}\| > \gamma$ and $\|z^{r+2}\|\|x^{r+1}\| > \gamma$, we have

$$\|z^r\| > \frac{\gamma}{\|x^{r-1}\|} > \frac{\gamma}{M}, \quad \|x^{r+1}\| > \frac{\gamma}{\|z^{r+2}\|} > \frac{\gamma}{M}. \tag{4.26}$$

Moreover, (4.24) together with (4.25) yields

$$\frac{\gamma}{M^2} < \frac{(z^{r+1})^\top x^r}{\|z^{r+1}\|\,\|x^r\|} = (\hat{e}^{r+1}_{3-i_{r+1}})^\top \hat{e}^r_{i_r}. \tag{4.27}$$

Also, by the definitions of $\hat{e}^k_1$ and $\hat{e}^k_2$, we have

$$\begin{aligned}(\hat{e}^r_{3-i_r})^\top \hat{e}^{r+1}_{i_{r+1}} - (\hat{e}^{r+1}_{3-i_{r+1}})^\top \hat{e}^r_{i_r} &= \left(\hat{e}^r_{3-i_r} + \hat{e}^r_{i_r}\right)^\top \hat{e}^{r+1}_{i_{r+1}} - \left(\hat{e}^{r+1}_{3-i_{r+1}} + \hat{e}^{r+1}_{i_{r+1}}\right)^\top \hat{e}^r_{i_r} \\ &= \binom{\sqrt{2}}{0}^\top \hat{e}^{r+1}_{i_{r+1}} - \binom{\sqrt{2}}{0}^\top \hat{e}^r_{i_r} \\ &= 1 - 1 = 0. \end{aligned} \tag{4.28}$$

Thus, we have

$$\begin{aligned}(z^r)^\top x^{r+1} &= \|z^r\|\|x^{r+1}\|(\hat{e}^r_{3-i_r})^\top \hat{e}^{r+1}_{i_{r+1}} \\ &= \|z^r\|\|x^{r+1}\|(\hat{e}^{r+1}_{3-i_{r+1}})^\top \hat{e}^r_{i_r} > \frac{\gamma^3}{M^4},\end{aligned}$$

where the first equality follows from (4.25) with $k = r$, the second equality is due to (4.28), and the inequality follows from (4.26) and (4.27). However, this contradicts (4.22) and (4.23). Hence, Algorithm 1 must terminate in a finite number of iterations. ■

4.4 Approximation analysis for obtained solution

So far, we have shown the finite termination property of Algorithm 1 via the aforementioned theorems. Nevertheless, these theorems would be meaningless if the obtained solution is far from the optimum of SOCCSIP (1.1). The following theorem guarantees that if $\eta > 0$ is sufficiently close to 0, then the last output of Algorithm 1 is also close to the optimal solution of SOCCSIP (1.1).

Theorem 4.4 *Suppose that Algorithm* 1 *terminates in a finite number of iterations. Let $k^*(\eta)$ be the number of iterations in which Algorithm* 1 *terminates. Then,* $\lim_{\eta \to 0} \operatorname{dist}(x^{k^*(\eta)}, \mathcal{S}) = 0$.

Proof. Let $h : \mathbb{R}^n \to \mathbb{R}$, $X \subset \mathbb{R}^n$, and $\mathcal{S}_\eta \subset \mathbb{R}^n$ be defined by

$$h(x) := \max_{s \in \Omega} c(x,s), \quad X := \mathcal{K} \cap \left\{ x \;\middle|\; f(x) \leq V^* \right\}, \quad \mathcal{S}_\eta := X \cap \left\{ x \;\middle|\; h(x) \leq \eta \right\}.$$

Then, h is continuous and convex, X is closed and convex, and $\mathcal{S}_0$ coincides with the solution set of SOCCSIP (1.1). Moreover, since $f(k^*(\eta)) \leq V^*$, it follows $x^{k^*(\eta)} \in \mathcal{S}_\eta$ for any $\eta > 0$. We can show the remainder of the proof in a way analogous to [13, Theorem 3.2]. ■

Remark In Step 1 of Algorithm 1, we may also choose l (≥ 2) different points $\{s_1^k, \dots, s_l^k\}$ such that $c(x^k, s_i^k) > \eta$ for $i = 1, \dots, l$ with $\overline{E}_{k+1} := E_k \cup \{s_1^k, \dots, s_l^k\}$. For such a multiple explicit exchange method, Theorems 4.2–4.4 can be shown by using analogous techniques.

5 Numerical results

In this section, we report some numerical results. We implement Algorithm 1 by Matlab 7.10.0 (R2010a) and run the experiments on a computer with Pentium (R) CPUs 3.19GHz and 3.20GHz with 0.99MB RAM. Throughout the experiments, we set $\eta := 10^{-6}$ and $E_0 := \Omega_0$. In Step 1, we find an $s_{\rm new}^k \in \Omega$ with $c(x^k, s_{\rm new}^k) > \eta$ as follows. We first test $N\,(\approx 100)$ grid points[*5] $\tilde{s}_1, \dots, \tilde{s}_N \in \Omega$ to find $\overline{s}^k := \mathrm{argmax}_{i=1,2,\dots,N}\, c(x^k, \tilde{s}_i)$. If $c(x^k, \overline{s}^k) > \eta$, then we set $s_{\rm new}^k := \overline{s}^k$. Otherwise, we solve the constrained maximization problem: "Maximize $c(x^k, s)$ subject to $s \in \Omega$" by means of *fmincon* solver with the initial point $\overline{s}^k$. In Step 2, we solve CSOCP($\overline{E}_{k+1}$) by using the SOCCP reformulation technique together with the regularized smoothing Newton method [15]. In Step 3, we relax the criterion $\nu_{k+1}(s) > 0$ to $\nu_{k+1}(s) > 10^{-6}$. We stop the iteration of Algorithm 1 when $\max\{c(x^k, s) | s \in \Omega\} \leq \eta$.

Experiment 1 (Solving SOCCSIPs with various choices of parameters)

Let $p : \mathbb{R}^2 \to \mathbb{R}$, $q : \mathbb{R}^2 \to \mathbb{R}^n$ and $r : \mathbb{R}^2 \to \mathbb{R}$ be defined as

$$p(s) := 0.1\, s_1^2(1 + \sin s_2),\ \ q(s) := \big(\cos{(-1)^j(s_1 s_2 + 0.5\pi)}\big)_{j=1}^n + e,$$
$$r(s) := -(5 + \sin s_1 + \log(s_2 + 10)),$$

where $e \in \mathbb{R}^n$ denotes the identity element with respect to $\mathcal{K}$. (For example, $e = (1, 0, 0, 0, 1, 0, 0)^\top$ when $\mathcal{K} = \mathcal{K}^4 \times \mathcal{K}^3$.) Then, we solve the following SOCCSIP:

$$\begin{array}{rl} \text{Minimize} & f(x) := \log(1 + \exp(x^\top A x)) + b^\top x \\ \text{subject to} & c(x, s) := p(s)(x^\top M x)^{1.5} + q(s)^\top x + r(s) \leq 0 \quad \forall s \in \Omega := [-\beta\pi, \beta\pi]^2 \\ & x \in \mathcal{K}, \end{array} \tag{5.1}$$

where $\beta > 0$ is a given constant, and $A \in \mathbb{R}^{n\times n}$ and $M \in \mathbb{R}^{n\times n}$ are positive semidefinite symmetric matrices. Function f is convex, but is not strictly convex when $\mathrm{rank}(A) < n$. Also c is convex with respect to x, but is not strictly convex when $\mathrm{rank}(M) < n$. Matrices A and M are defined as $A := PP^\top$ and $M := QQ^\top$, where P and Q are respectively $(n \times n_a)$- and $(n \times n_m)$-dimensional matrices whose components are randomly chosen from $[-1, 1]$. Since P and Q are randomly generated, we almost always have $\mathrm{rank}(A) = \min(n, n_a)$ and $\mathrm{rank}(M) = \min(n, n_m)$. Also, each component of vector b is randomly chosen from $[-1, 1]$. In applying Algorithm 1, we set $\Omega_0 := \{0\}$ since $\mathcal{K} \cap \{x \,|\, c(x, 0) \leq 0\}$ is compact. SOCCSIP (5.1) has a Slater point since $c(0, s) = r(s) < 0$ for all $s \in \Omega$.

First, we solve SOCCSIP (5.1) with various choices of constant $\beta > 0$. We solve 12 problem instances, each of which has different values of A, M and b. For all instances, we set $\mathcal{K} = \mathcal{K}^{10}$, $r_a = 6$ and $r_m = 8$. Since A and M are rank-deficient, functions f and $c(\cdot, s)$ are not strictly convex. We show the obtained results in Table 1, in which λ_1 and λ_2 denote the spectral values defined by (2.2) of the obtained solutions x^*, ♯ite denotes the number of iterations, cpu(s) denotes the CPU time in seconds, and $E_k^{\rm final} \setminus \Omega_0$ denotes the final output of E_k except Ω_0. Notice that the number of ♯ite does not count

[*5] When $\Omega = [l_s, u_s] \subset \mathbb{R}$, we test the 101 points $l_s + i(u_s - l_s)/100$ with $i = 0, 1, \dots, 100$. When $\Omega = [l_s, u_s]^2 \subset \mathbb{R}^2$, we test the 121 points $(l_s + i(u_s - l_s)/10,\ l_s + j(u_s - l_s)/10)^\top$ with $i = 0, 1, \dots, 10$ and $j = 0, 1, \dots, 10$.

the first subproblem CSOCP(E_0). Therefore, Algorithm 1 actually solves $\sharp$ite + 1 CSOCPs for each instance. The rows with $\sharp$ite = 0 (Problems 4 and 9) imply that Algorithm 1 finds the SOCCSIP optimum $x^*(= x^0)$ in Step 0, and there exists no $s^0_{\rm new} \in \Omega$ with $c(x^0, s^0_{\rm new}) > \eta$. In other words, the convex constraint corresponding to $s = (0, 0)^\top$ is active at the solution x^*. We can also see that, when $\beta = 0.1$ (Problems 1–3), the algorithm finds the solution x^* with $k = 1$, and the final active index is $s = (-0.1\pi, 0.1\pi)^\top$. Notice that Ω is expressed as a two-dimensional square, and $(-0.1\pi, 0.1\pi)^\top$ is one of its *vertices*. Since Algorithm 1 checks the values of $c(x^k, s)$ at all the vertices in Step 1, we will find the SOCCSIP optimum very soon if the final active index is located at the vertex of Ω. On the contrary, for Problems 5–8 and 10–12, the final active indices are not located at the vertices of Ω.[*6] This may be the main reason why Problems 5–8 and 10–12 need more iterations and cpu time than Problems 1–3. This tendency seems to be more noticeable as β becomes larger.

Next, we solve SOCCSIP (5.1) with various choices of the Cartesian structure $\mathcal{K}$. We consider 9 different Cartesian structures, and solve 100 problems for each $\mathcal{K}$. We therefore solve 900 problems in total. For all problems, we set $r_a = r_m = 0.8n$ and $\beta = 1$. Since A and M are rank-deficient, functions f and $c(\cdot, s)$ are not strictly convex. We give the obtained results in Table 2, in which $\lambda_i^{\min}$ and $\lambda_i^{\max}$ $(i = 1, 2)$ denote the maximum and minimum of the spectral values (2.2) among all Cartesian subvectors $x_1^*, \dots, x_m^*$ in 100 problems for each $\mathcal{K}$.[*7] (In case of $\mathcal{K} = (\mathcal{K}^1)^{10} = \mathbb{R}^{10}_+$, we only give the values for λ_1.) Also, $\lambda_i^{\rm zero}$ denotes the frequency that the spectral values become 0, and $\sharp$ite and cpu(s) are the average values among the 100 trials for each $\mathcal{K}$. From the table, we can observe that the number of iterations does not change so much even when the dimension n of the variables or the number m of sub-SOCs increases. However, CPU time increases quite a bit as n becomes larger. This implies that the computational cost for solving each subproblem (CSOCPs solved in Steps 0 and 2) becomes more expensive as n increases. Also, we can see that the spectral value λ_1 often becomes 0, which implies that each subvector x_j^* is located on the boundary of the SOC $\mathcal{K}^{n_j}$. When $\mathcal{K} = (\mathcal{K}^1)^{10} = \mathbb{R}^{10}_+$, approximately 50% of obtained λ_1s are greater than 0, but it is not surprising since, in this case, each subvector x_j^* coincides with the j-th scalar component of vector x^*, and $\lambda_1(x_j^*) > 0$ implies $x_j^* > 0$. When $\mathcal{K} = (\mathcal{K}^2)^5$, we still have more than 20% positive λ_1s. It is also convincing since the dimension of each SOC is small (= 2) and the problems are generated randomly. In other seven cases with $\mathcal{K} = \mathcal{K}^{10}, (\mathcal{K}^{20})^5, \dots, \mathcal{K}^{200}$, we always have $\lambda_1(x_j^*) = 0$ and $\lambda_2(x_j^*) > 0$, which means that all the subvectors of x^* are located on the boundary of SOCs.

Finally, we solve SOCCSIP (5.1) with various degrees of rank deficiency of matrices A and M. For all problems, we set $\mathcal{K} = \mathcal{K}^{20} \times \mathcal{K}^{30}$, $\beta = 1$, and $r := r_a = r_m$, and choose 7 different values for r. We solve 100 problems for each r, and hence solve 700 problems in total. Note that we usually have $\mathrm{rank}(A) = \mathrm{rank}(M) = \min(50, r)$. Therefore, f and $c(\cdot, s)$ are not strictly convex when $r < 50$, and are (almost always) strictly convex when $r \geq 50$. We give the obtained results in Table 3, in which $\lambda_i^{\min}$, $\lambda_i^{\max}$ and $\lambda_i^{\rm zero}$ are defined analogously to the previous experiment, and $\sharp$ite and cpu(s) are the average of 100 problems for each r. As the table shows, we need a large number of iterations when r is small, i.e., matrices A and M have high rank-deficiency. On the other hand, when $r = 50$ and 100, i.e., f and $c(\cdot, s)$ are strictly convex, we obtain the solution in a very small number of iterations. Especially, when $r = 100$, we often obtain the SOCCSIP optimum in the initial step with $k = 0$. Also we can observe that, when r is small, the value of λ_2 sometimes becomes 0, which implies that the optimal solution of SOCCSIP (5.1) is $x^* = 0$. Actually, the above-mentioned features generally depend on the structure of each problem, but in many cases the ill-posedness of a problem seems to be relevant to the rank deficiency of certain matrices involved in the problem.

[*6] For Problem 7, the final active index is located in the interior of Ω, and for other six problems $(5, 6, 8, 10, 11, 12)$, they are located on the non-vertex boundary.

[*7] For example, when $\mathcal{K} = (\mathcal{K}^\ell)^m$, we have m subvectors $x_1^*, \dots, x_m^* \in \mathbb{R}^\ell$ for the optimum x^*. Therefore, if the obtained solutions of the 100 problems are $x^{*,1}, x^{*,2}, \dots, x^{*,100}$, then we have $\lambda_i^{\max} := \max\left\{\lambda_i(x_j^{*,p}) \;\middle|\; (j,p) \in \{1, 2, \dots, m\} \times \{1, 2, \dots, 100\}\right\}$ and $\lambda_i^{\min} := \min\left\{\lambda_i(x_j^{*,p}) \;\middle|\; (j,p) \in \{1, 2, \dots, m\} \times \{1, 2, \dots, 100\}\right\}$ for each $i = 1, 2$.

Table 1: Obtained results for SOCCSIP (5.1) with various choices of β

Problem	β	Output ($\mathcal{K}=\mathcal{K}^{10},\ r_m=8,\ r_a=6$)				
		λ_1	λ_2	♯ite	cpu(s)	$E_k^{\text{final}} \setminus \Omega_0$
Problem 1	0.1	0	5.97	1	0.39	$(-0.1\pi, 0.1\pi)^\top$
Problem 2	0.1	0	8.29	1	0.51	$(-0.1\pi, 0.1\pi)^\top$
Problem 3	0.1	0	8.13	1	0.89	$(-0.1\pi, 0.1\pi)^\top$
Problem 4	1	0	0.95	0	0.10	$\emptyset$
Problem 5	1	0	1.41	5	3.03	$(-\pi, 0.492\pi)^\top,\ (\pi, 0.492\pi)^\top$
Problem 6	1	0	1.63	2	0.57	$(\pi, 0.483\pi)^\top$
Problem 7	1.6	0	1.37	4	4.84	$(1.592\pi, -1.305\pi)^\top$
Problem 8	1.6	0	1.39	6	6.85	$(1.6\pi, -1.550\pi)^\top$
Problem 9	1.6	0	0.54	0	0.09	$\emptyset$
Problem 10	2	0	1.42	8	6.93	$(-2\pi, -1.567\pi)^\top$
Problem 11	2	0	1.08	5	6.21	$(2\pi, -1.636\pi)^\top$
Problem 12	2	0	2.06	4	1.35	$(2\pi, -1.652\pi)^\top$

Table 2: Obtained results for SOCCSIP (5.1) with various choices of $\mathcal{K}$

n	$\mathcal{K}$	Output ($r_m=r_a=0.8n,\ \beta=1$)							
		$\lambda_1^{\min}$	$\lambda_1^{\max}$	λ_1^{zero}(%)	$\lambda_2^{\min}$	$\lambda_2^{\max}$	λ_2^{zero}(%)	♯ite	cpu(s)
10	$(\mathcal{K}^1)^{10}$	0	0.98	51.2	—	—	—	1.70	1.39
10	$(\mathcal{K}^2)^5$	0	0.91	79.2	0	1.46	27.0	1.37	0.99
10	$\mathcal{K}^{10}$	0	0	100	0.12	2.36	0	1.44	0.84
100	$(\mathcal{K}^{20})^5$	0	0	100	0.06	0.32	0	2.12	2.66
100	$(\mathcal{K}^{50})^2$	0	0	100	0.17	0.45	0	2.12	2.63
100	$\mathcal{K}^{100}$	0	0	100	0.27	0.63	0	2.24	2.97
200	$(\mathcal{K}^{20})^{10}$	0	0	100	0.02	0.17	0	2.37	11.63
200	$(\mathcal{K}^{50})^4$	0	0	100	0.07	0.24	0	2.10	10.80
200	$\mathcal{K}^{200}$	0	0	100	0.27	0.38	0	2.05	13.86

Experiment 2 (Application to robust optimization)

The robust optimization [2] is one of distribution-free methodologies for handling problems with uncertain data. We usually assume that the uncertain data belong to some set, and try to solve another optimization problem called robust counterpart (RC), which is composed with taking the worst possible case into consideration.

Algorithm 1 is also applicable to the robust optimization for convex semi-infinite programs (CSIPs). Consider the following uncertain CSIP:

$$\begin{aligned} &\text{Minimize} && \hat{b}^\top x \\ &\text{subject to} && c(x,s) \le 0 \quad \forall s \in \Omega, \end{aligned} \tag{5.2}$$

where $\Omega := [-1,1]$ and $c(x,s) := x^\top \Xi(s) x + \eta(s)^\top x + \zeta(s)$ with

$$\Xi(s) := \begin{bmatrix} 19 & 3 & -6 & -7 & 5 \\ 3 & 18 & -5 & 2 & -4 \\ -6 & -5 & 15 & 4 & -5 \\ -7 & 2 & 4 & 10 & -2 \\ 5 & -4 & -5 & -2 & 16 \end{bmatrix} + \operatorname{diag}\big(\sin(\alpha s)\big)_{\alpha=1}^5$$

$$\eta(s) := (4,\ -3,\ 1,\ -2,\ 4)^\top + \big(\cos 1.3\alpha s\big)_{\alpha=1}^5, \quad \zeta(s) := -10 + (5+s)^{-1}.$$

Table 3: Obtained results for SOCCSIP (5.1) with various choices of $r := r_a = r_m$

r	Output ($\mathcal{K} = \mathcal{K}^{20} \times \mathcal{K}^{30},\ \beta = 1$)							
	$\lambda_1^{\min}$	$\lambda_1^{\max}$	$\lambda_1^{\text{zero}}(\%)$	$\lambda_2^{\min}$	$\lambda_2^{\max}$	$\lambda_2^{\text{zero}}(\%)$	♯ite	cpu(s)
1	0	0	100	0	12.03	30.0	12.57	12.17
2	0	0	100	0	12.03	19.0	13.01	11.67
5	0	0	100	0	11.99	2.0	13.20	10.26
10	0	0	100	0.859	8.91	0	8.80	7.07
20	0	0	100	0.181	4.50	0	2.88	2.69
50	0	0	100	0.085	0.56	0	1.16	0.94
100	0	0	100	0.031	0.27	0	0.03	0.22

Moreover, $\hat{b} \in \mathbb{R}^n$ is an uncertain vector such that $\hat{b} = b + \delta_b$, where $b = (-3, 4, 2, -4, 1)^\top$ is the nominal value of $\hat{b}$, and δ_b is the error term belonging to a certain set $\mathcal{B}$. Then, the RC of CSIP (5.2) is written as

$$\begin{aligned} &\text{Minimize} \quad \max_{\delta_b \in \mathcal{B}} (b + \delta_b)^\top x \\ &\text{subject to} \quad c(x, s) \le 0 \quad \forall s \in \Omega. \end{aligned} \tag{5.3}$$

Now, suppose that $\mathcal{B}$ is a closed sphere with radius ρ, that is, $\mathcal{B} := \{\delta_b \mid \|\delta_b\| \le \rho\}$. Then, the objective function of RC (5.3) can be calculated as $\max_{\delta_b \in \mathcal{B}} (b+\delta_b)^\top x = b^\top x + \max\{\delta_b^\top x \mid \|\delta_b\| \le \rho\} = b^\top x + \rho\|x\|$. Therefore, by introducing an auxiliary variable $u \in \mathbb{R}$, RC (5.3) can be rewritten equivalently as

$$\begin{aligned} &\underset{x,u}{\text{Minimize}} \quad b^\top x + \rho u \\ &\text{subject to} \quad \|x\| \le u,\ \ c(x, s) \le 0 \quad \forall s \in \Omega, \end{aligned} \tag{5.4}$$

which is of the form SOCCSIP (1.1) with $x := \binom{u}{x}$ and $\mathcal{K} := \mathcal{K}^{n+1}$.

Table 4 shows the obtained results on SOCCSIP (5.4) with various choices of ρ. In the table, $u^{*,\rho}$ and $x^{*,\rho}$ are the solutions of SOCCSIP (5.4), λ_1 and λ_2 are the spectral values for $\binom{u^{*,\rho}}{x^{*,\rho}}$, and ♯ite and cpu(s) denote the number of iterations and CPU time in seconds for Algorithm 1, respectively. As the table shows, the solution of RC (5.3) moves continuously as the radius ρ of $\mathcal{B}$ varies. Also, for all cases, we have $\lambda_1 = 0$, i.e., $u^{*,\rho} = \|x^{*,\rho}\|$, and the algorithm finds the solution in a small number of iterations and CPU time.

Next, we investigate the distribution of the functional values to observe the actual effect of the robust optimization. We generate 10,000 sample vectors $\delta_b^1, \delta_b^2, \ldots, \delta_b^{10000} \in \mathbb{R}^5$ for the error term δ_b. Each δ_b^i is defined as $\delta_b^i := \gamma_i v^i / \|v^i\|$, where $\gamma_i \in \mathbb{R}$ and each component of $v^i \in \mathbb{R}^5$ independently follow the normal distribution with mean 0 and deviation 1.5. Since the normal distribution contains 95% of the values within 2 standard deviations of the mean, we will have $\|\delta_b^i\| \le 3$ with 95% probability. To make the functional values more intuitive, we add a positive constant (= 18) to the objective function, that is, we check the value of $f_i(x^{*,\rho}) := (b + \delta_b^i)^\top x^{*,\rho} + 18$ for each i and ρ. The obtained results are summarized in Table 5, where the columns of 'best', 'mean' and 'worst' denote the minimum, average, and maximum of $f_i(x^{*,\rho})$ among $i = 1, 2, \ldots, 10,000$ for each ρ, respectively. The column of [6.5, 7) denotes the number of times that we had $6.5 \le f_i(x^{*,\rho}) < 7$ among $10,000$ sample vectors of δ_b^i. (Other columns are similar.) From the table, we can see that the values of 'worst' is smaller as ρ becomes larger, though it is opposite in the columns of 'best' and 'mean'. This means that the robust optimization may have disadvantage under average or lucky situations, but the serious damage can be avoided or reduced even when an unlucky situation occurs. Since we applied the normal distribution to generate the sample error vectors, we seldom encountered the unlucky situations such as $f_i(x^{*,\rho}) > 8.5$. However, if we apply another type of distribution, we may encounter such an undesirable situation more often, and the robust optimization can be more important.

Table 4: Obtained results for SOCCSIP (5.4) with various choices of ρ

ρ	Output $(b=(4,9,-6,8,-5)^\top,\ \mathcal{K}=\mathcal{K}^6)$					
	$u^{*,\rho}$	$x^{*,\rho}$	λ_1	λ_2	♯ite	cpu(s)
0	1.167	$(-0.567,\ 0.076, 0.394, -0.875, 0.336)^\top$	0	2.334	2	1.768
0.5	1.147	$(-0.555,\ 0.054, 0.391, -0.861, 0.331)^\top$	0	2.295	2	2.089
1	1.126	$(-0.542,\ 0.031, 0.388, -0.845, 0.326)^\top$	0	2.252	3	3.485
1.5	1.103	$(-0.529,\ 0.008, 0.384, -0.828, 0.321)^\top$	0	2.207	4	1.748
2	1.079	$(-0.514, -0.015, 0.380, -0.810, 0.315)^\top$	0	2.159	3	1.625
2.5	1.054	$(-0.499, -0.040, 0.376, -0.790, 0.310)^\top$	0	2.109	2	1.268
3	1.028	$(-0.483, -0.065, 0.371, -0.768, 0.303)^\top$	0	2.057	2	1.014

Table 5: Perturbed functional values of uncertain CSIP (5.2)

ρ	Output $(f_i(x^{*,\rho})=(b+\delta_b^i)^\top x^{*,\rho}+18)$								
	best	mean	worst	$[6.5,7)$	$[7,7.5)$	$[7.5,8)$	$[8,8.5)$	$[8.5,9)$	$[9,+\infty)$
0	0.464	5.358	9.967	342	162	87	45	12	4
0.5	0.581	5.363	9.864	331	162	86	39	10	4
1	0.717	5.379	9.763	337	160	83	37	9	4
1.5	0.770	5.408	9.666	348	155	87	34	8	4
2	0.813	5.450	9.574	357	163	86	36	6	4
2.5	0.877	5.506	9.488	382	173	89	34	7	3
3	0.961	5.578	9.410	438	182	91	38	7	2

References

[1] A. Auslender, M. A. Goberna, and M. A. López, *Penalty and smoothing methods for convex semi-infinite programming*, Mathematics of Operations Research, 34 (2009), pp. 303–319.

[2] A. Ben-Tal, L. El Ghaoui, and A. Nemirovski, *Robust Optimization*, Princeton University Press, Princeton, NJ, 2009.

[3] D. P. Bertsekas, A. Nedić, and A. E. Ozdaglar, *Convex Analysis and Optimization*, Athena Scientific, 2003.

[4] J. F. Bonnans and H. Ramírez, *Perturbation analysis of second-order cone programming problems*, Mathematical Programming, 104 (2005), pp. 205–227.

[5] J. F. Bonnans and A. Shapiro, *Perturbation analysis of optimization problems*, Springer-Verlag, New York-Berlin-Heidelberg, 2000.

[6] X. D. Chen, D. Sun, and J. Sun, *Complementarity functions and numerical experiments for second-order-cone complementarity problems*, Computational Optimization and Applications, 25 (2003), pp. 39–56.

[7] I. D. Coope and G. A. Watson, *A projected Lagrangian algorithm for semi-infinite programming*, Mathematical Programming, 32 (1985), pp. 337–356.

[8] J. Faraut and A. Korányi, *Analysis on Symmetric Cones*, Clarendon Press, New York, 1994.

[9] A. V. Fiacco and K. O. Kortanek, eds., *Semi-infinite Programming and Applications*, vol. 215 of Lecture notes in Economics and Mathematical Systems, Springer, 1983.

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

[11] M. A. Goberna and M. A. López, *Linear semi-infinite programming: An updated survey*, European Journal of Operational Research, 143 (2002), pp. 390–405.

[12] C. Gonzaga, E. Polak, and R. Trahan, *An improved algorithm for optimization problems with functional inequality constraints*, IEEE Transactions on Automatic Control, 25 (1980), pp. 49–54.

[13] S. Hayashi and S.-Y. Wu, *An explicit algorithm for linear semi-infinite programming problem with second-order cone constraints*, SIAM Journal on Optimization, 20 (2009), pp. 1527–1546.

[14] S. Hayashi, T. Yamaguchi, N. Yamashita, and M. Fukushima, *A matrix splitting method for symmetric affine second-order cone complementarity problems*, Journal of Computational and Applied Mathematics, 175 (2005), pp. 335–353.

[15] S. Hayashi, N. Yamashita, and M. Fukushima, *A combined smoothing and regularization method for monotone second-order cone complementarity problems*, SIAM Journal on Optimization, 15 (2005), pp. 593–615.

[16] R. Hettich and K. O. Kortanek, *Semi-infinite programming: theory, methods, and applications*, SIAM Review, 35 (1993), pp. 380–429.

[17] H. Kato and M. Fukushima, *An SQP-type algorithm for nonlinear second-order cone programs*, Optimization Letters, 1 (2007), pp. 129–144.

[18] K. Kortanek and H. No, *A central cutting plane algorithm for convex semi-infinite programming problems*, SIAM Journal on Optimization, 3 (1993), pp. 901–918.

[19] H. C. Lai and S.-Y. Wu, *On linear semi-infinite programming problems: an algorithm*, Numerical Functional Analysis and Optimization, 13 (1992), pp. 287–304.

[20] D. Li, L. Qi, J. Tam, and S.-Y. Wu, *A smoothing newton method for semi-infinite programming*, Journal of Global Optimization, 30 (2004), pp. 169–194.

[21] C.-J. Lin, S.-C. Fang, and S.-Y. Wu, *An unconstrained convex programming approach to linear semi-infinite programming*, SIAM Journal on Optimization, 8 (1998), pp. 443–456.

[22] H. Mine, M. Fukushima, and Y. Tanaka, *On the use of ε-most-active constraints in an exact penalty function method for nonlinear optimization*, IEEE Transactions on Automatic Control, 29 (1984), pp. 1040–1042.

[23] Y. Narushima, N. Sagara, and H. Ogasawara, *A smoothing Newton method with Fischer-Burmeister function for second-order cone complementarity problems*, Journal of Optimization Theory and Applications, 149 (2011), pp. 79–101.

[24] C. J. Price and I. D. Coope, *Numerical experiments in semi-infinite programming*, Computational Optimization and Applications, 6 (1996), pp. 169–189.

[25] L. Qi, S.-Y. Wu, and G. Zhou, *Semismooth newton for solving semi-infinite programming problems*, Journal of Global Optimization, 27 (2003), pp. 215–232.

[26] R. Reemtsen and S. Görner, *Numerical methods for semi-infinite programming: a survey*, in Semi-Infinite Programming, R. Reemtsen and J. Rückmann, eds., Kluwer Academic Publishers, Boston, 1998, pp. 195–275.

[27] R. Reemtsen and J. Rückmann, *Semi-Infinite Programming*, Kluwer Academic Publishers, Boston, MA, 1998.

[28] A. Shapiro, *First and second order optimality conditions and perturbation analysis of semi-infinite programming problems*, in Semi-Infinite Programming, R. Reemtsen and J. Rückmann, eds., Kluwer Academic Publishers, Boston, MA, 1998, pp. 103–133.

[29] O. Stein and G. Still, *Solving semi-infinite optimization problems with interior point techniques*, SIAM Journal on Control and Optimization, 42 (2003), pp. 769–788.

[30] G. Still, *Discretization in semi-infinite programming: The rate of convergence*, Mathematical Programming, 91 (2001), pp. 53–69.

[31] K. L. Teo and C. J. Goh, *A simple computational procedure for optimization problems with functional inequality constraints*, IEEE Transactions on Automatic Control, 1987 (32), pp. 940–941.

[32] S.-Y. Wu and S.-C. Fang, *Solving convex programs with infinitely linear constraints by a relaxed cutting plane method*, Computers and Mathematics with Applications, 38 (1999), pp. 23–33.

[33] S.-Y. Wu, D. H. Li, L. Qi, and G. Zhou, *An iterative method for solving KKT system of the semi-infinite programming*, Optimization Methods and Software, 20 (2005), pp. 629–643.

[34] H. Yamashita and H. Yabe, *A primal-dual interior point method for nonlinear optimization over second-order cones*, Optimization Methods and Software, 24 (2009), pp. 407–426.