*Applied Analysis and Optimization*
Volume 8, Number 2, 2024, 249–264

Yokohama Publishers
*ISSN 2189-1664 Online Journal*


# TWO NOVEL SMOOTH-TYPE ALGORITHMS FOR SECOND-ORDER CONE LINEAR COMPLEMENTARITY PROBLEMS

YULI SUN AND ZIJUN HAO*

Abstract. This paper presents two novel smooth-type functions of lower order penalty function algorithms for solving the second-order cone linear complementarity problem. These two new smooth functions are introduced by employing the method of generating smooth functions whose convolutional integration of kernel functions is the plus function and minus function. Numerical experiments are subsequently conducted using the smooth Newton method. The results demonstrate that as the penalty parameter tends to infinity and the smooth parameter monotonically decreases to zero, the solution sequence of the lower order penalty equations converges to the solution of the second-order cone linear complementarity problems under certain assumption. Furthermore, a comparison of the performance between the newly proposed smooth functions and the original smooth functions is carried out through numerical experiments. The findings indicate that one of the new smooth functions exhibits better numerical performance. Thereby, this generalizes the smooth functions of the projection function.

## 1. Introduction

In this paper, we consider the second-order cone linear complementarity problem (SOCLCP), which is to find $x \in \mathbb{R}^n$, such that

$$x \in \mathcal{K}, Ax - b \in \mathcal{K}, x^T(Ax-b) = 0, \tag{1.1}$$

where $A$ is an $n \times n$ matrix, $b$ is a vector in $\mathbb{R}^n$, and $\mathcal{K}$ is the Cartesian product of second-order cones (SOCs), also called Lorentz cones. In other words,

$$\mathcal{K} := \mathcal{K}^{n_1} \times \cdots \times \mathcal{K}^{n_r}, \tag{1.2}$$

with $r,\ n_1, \ldots, n_r \geq 1,\ n_1 + \cdots + n_r = n$, and $\mathcal{K}^{n_i} \subset \mathbb{R}^{n_i} (i = 1, \ldots, r)$ being the $n_i$-dimensional SOC, i.e.,

$$\mathcal{K}^{n_i} = \{(x_1, x_2) \in \mathbb{R} \times \mathbb{R}^{n_i - 1} \mid x_1 \geq \|x_2\|\}, i = 1, \ldots, r,$$

where $\|\cdot\|$ denotes the Euclidean norm and $(x_1, x_2) := (x_1, x_2^T)^T$. It is obvious that in (1.2), $\mathcal{K}$ is a closed, convex and self-dual cone in $\mathbb{R}^n_+$. If $n_i = 1$, $\mathcal{K}^1$ denotes the set of nonnegative set of real numbers $\mathbb{R}_+$. Therefore, the SOCLCP is an extension of the linear complementarity problem (LCP).

During the past several years, there are many methods proposed for solving the SOCLCPs (1.1)-(1.2), including the interior-point method [24], the smoothing

---

2020 *Mathematics Subject Classification.* 90C25, 90C30, 90C33.
*Key words and phrases.* Second-order cone, linear complementarity problem, lower order penalty approach, smooth-type algorithms.

*Corresponding author. The author was supported by the Natural Science Fund of Ningxia (No. 2022AAC03235).

Newton method [7,11,19], the semismooth Newton method [22,25], the smoothing-regularization method [18], the merit function method [4], and the matrix splitting method [17] etc. Although the effectiveness of some methods has been improved substantially in recent years, the fact remains that there still have many complementarity problem require efficient and accurate numerical methods. The penalty methods are well-known for solving constrained optimization problems which posses many nice properties. The $l_1$ penalty function [27] and lower order penalty functions [8,12,23] possess many nice properties and attract much attention. In [26], Wang and Yang proposed a power penalty approach for solving the LCPs. By this approach, LCP can be converted to asymptotic nonlinear equations. The merit of this approach shows that the solution sequence of the asymptotic nonlinear equations converge to the solution of the LCP at an exponential rate when the penalty parameter tends to positive infinity under some mild assumptions. In [20, 21], Huang and Wang extended the power penalty approach for solving the nonlinear complementarity problem (NCPs) and mixed nonlinear complementarity problem (MNCPs). In [14, 15], Hao and Wan extended the power penalty approach for solving the SOCLCPs and second-order cone nonlinear complementarity problems(SOCNCPs). In [16], the power penalty approach have been extended to the generalized lower-order penalty approach for solving the second-order cone mixed complementarity problems(SOCMCPs). Hao and Chen et al. [13] proposed an approximate lower order penalty approach for solving SOCLCPs, and four kinds of specific smoothing functions are considered. The four kinds of specific smoothing functions are based on the convolution integration of the kernel functions by Chen and Mangasarian [2]. However, in [13], when the power parameter is not equal to 1, the lower order penalty equations are not smooth in certain point situations. Thus, the discrete Newton method is used to solve the nonlinear equations in the calculating process. The inner loop is simultaneously added to realize the algorithm. Grounded on the study of [13], this paper proposes two new smooth functions, considering the smooth case that the power parameter is equal to 1. The smooth Newton method but not discrete Newton method is then used to solve the nonlinear equations, and a comparison between the proposed two functions and the past four smooth function in [13] is conducted.

This paper is organized as follows: In Sect. 2, we review some properties related to the single block SOC which is the basis for our subsequent analysis. In Sect. 3, based on the four smooth functions proposed by Hao in [13], two new smooth functions are proposed; In Sect. 4, the lower order penalty function algorithm for solving the SOCLCP and its convergence analysis are presented. In Sect. 5, the preliminary numerical experiments are presented, the numerical performance of the new smooth function and the original smooth function are compared. Finally, we draw the conclusion in Sect. 6. Throughout this paper, we use $\mathrm{int}(\mathcal{K}^n)$ and $\mathrm{bd}(\mathcal{K}^n)$ to denote the interior and the boundary of SOC $\mathcal{K}^n$ respectively. For any $x$, $y$ in $\mathbb{R}^n$, we write $x \succeq_{\mathcal{K}^n} y$ if $x - y \in \mathcal{K}^n$ and write $x \succ_{\mathcal{K}^n} y$ if $x - y \in \mathrm{int}(\mathcal{K}^n)$. In other words, we have $x \succeq_{\mathcal{K}^n} 0$ if and only if $x \in \mathcal{K}^n$, and $x \succ_{\mathcal{K}^n} 0$ if and only if $x \in \mathrm{int}(\mathcal{K}^n)$. The notation $\|\cdot\|_p$ denotes the usual $l_p$-norm on $\mathbb{R}^n$ for any $p \geq 1$. In particular, it is Euclidean norm $\|\cdot\|$ when $p = 2$.

## 2. Preliminary results

In this section, some preliminary results for a single block SOC $\mathcal{K} = \mathcal{K}^n$ are given, since the analysis can be easily extended to the general case (1.2). For any $x = (x_1, x_2) \in \mathbb{R} \times \mathbb{R}^{n-1}$, $y = (y_1, y_2) \in \mathbb{R} \times \mathbb{R}^{n-1}$, their *Jordan product* [1] is defined as

$$x \circ y = (x^T y,\ x_1 y_2 + y_1 x_2). \tag{2.1}$$

We write $x+y$ to mean the usual componentwise addition vectors and $x^2$ to mean $x \circ x$. It is easy to see that the Jordan product is *commutative* and $e = (1, 0, \ldots, 0)^T \in \mathbb{R}^n$ is the identity element. The Jordan product is *not associative* for $n > 2$ in general. However, we have $x \circ (x \circ x) = (x \circ x) \circ x$ for any $x \in \mathbb{R}^n$. Thus, for any positive integer $k$, the form $x^k$ is definite. We define $x^0 = e$ if $x \neq 0$ and recursively define the powers of element as $x^k = x \circ x^{k-1}$ for any integer $k \geq 1$. Therefore, the equality $x^{m+n} = x^m \circ x^n$ holds for any positive integer $m$ and $n$. Note that $\mathcal{K}^n$ is *not closed* under the Jordan product for $n > 2$ in general.

The Jordan product (2.1) associated with $\mathcal{K}^n$ results in some useful facts. For each $x = (x_1, x_2) \in \mathbb{R} \times \mathbb{R}^{n-1}$, the *trace* $\text{tr}(x) = 2x_1$ and the *determinate* $\det(x) = x_1^2 - \|x_2\|^2$. A vector $x$ is *invertible* if $\det(x) \neq 0$, the *inverse* of $x$ is denoted by $x^{-1}$, satisfying $x \circ x^{-1} = e$ and $x^{-1} = (x_1, -x_2)/\det(x)$. It is clear that $x \in \text{int}\mathcal{K}^n$ if and only if $x^{-1} \in \text{int}\mathcal{K}^n$. The unique square root of $x \in \mathcal{K}^n$ is denoted by $x^{\frac{1}{2}}$, satisfying $x^{\frac{1}{2}} \in \mathcal{K}^n$, $(x^{\frac{1}{2}})^2 = x$. Direct calculation yields $x^{\frac{1}{2}} = (s, x_2/(2s))$, where $s = ((x_1 + \sqrt{x_1^2 - \|x_2\|^2})/2)^{\frac{1}{2}}$ and the term $x_2/s$ is defined to be zero vector if $x_2 = 0$ and $s = 0$. For any $x \in \mathbb{R}^n$, the *absolute value vector* of $x$ is denote by $|x|$ satisfying $|x| = (x^2)^{\frac{1}{2}}$. For more details of these concepts, we can refer to references [1, 3, 10, 11].

Next, we introduce the spectral factorization of vectors in $\mathbb{R}^n$ associated with $\mathcal{K}^n$ [3, 11]. For any vector $x = (x_1, x_2) \in \mathbb{R} \times \mathbb{R}^{n-1}$, the vector can be decomposed as

$$x = \lambda_1 u^{(1)} + \lambda_2 u^{(2)}, \tag{2.2}$$

$$\lambda_i = x_1 + (-1)^i \|x_2\|,\ u^{(i)} = \begin{cases} \frac{1}{2}(1, (-1)^i \frac{x_2}{\|x_2\|}), & \text{if } x_2 \neq 0, \\ \frac{1}{2}(1, (-1)^i w), & \text{if } x_2 = 0, \end{cases} \quad i = 1, 2, \tag{2.3}$$

where $\lambda_1, \lambda_2$ are the spectral values and $u^{(1)}, u^{(2)}$ are the associated spectral vectors, $w$ is any vector in $\mathbb{R}^{n-1}$ satisfying $\|w\| = 1$. If $x_2 \neq 0$, the decomposition (2.2)-(2.3) is unique. It is obvious that $e = u^{(1)} + u^{(2)}$ and $\lambda_1 \leq \lambda_2$. Now we give some basic properties of spectral factorization.

**Property 2.1** ([3,11])**.** For any $x = (x_1, x_2) \in \mathbb{R} \times \mathbb{R}^{n-1}$, suppose that the spectral factorization of $x$ is given by (2.2)-(2.3). Then the following results hold.

(1) $u^{(1)}$ and $u^{(2)}$ are orthogonal under the Jordan product and have length $1/\sqrt{2}$, i.e., $u^{(1)} \circ u^{(2)} = 0, \|u^{(1)}\| = \|u^{(2)}\| = 1/\sqrt{2}$.
(2) $u^{(1)}$ and $u^{(2)}$ are idempotent under the Jordan product, i.e., $u^{(i)} \circ u^{(i)} = u^{(i)}, i = 1, 2$.
(3) $\lambda_1$ is nonnegative (positive) if and only if $x \in \mathcal{K}^n$ ($x \in \text{int}\mathcal{K}^n$).

The spectral factorization (2.2)-(2.3) and the Property 2.1 provide a very useful tool for evaluating the functions defined by the powers of Jordan product. For example, we have $x^2 = (\lambda_1 u^{(1)} + \lambda_2 u^{(2)}) \circ (\lambda_1 u^{(1)} + \lambda_2 u^{(2)}) = \lambda_1^2 u^{(1)} + \lambda_2^2 u^{(2)}$ for any $x \in \mathbb{R}^n$. Therefore, the spectral values of $x^2$ are nonnegative and $x^2 \in \mathcal{K}^n$. On the contrary, for any $x \in \mathcal{K}^n$ with spectral factorization (2.2)-(2.3), we have $0 \leq \lambda_1 \leq \lambda_2$. Let $w = \sqrt{\lambda_1} u^{(1)} + \sqrt{\lambda_2} u^{(2)}$, then $w^2 = x$. By the uniqueness of the square root, we have $x^{1/2} = \sqrt{\lambda_1} u^{(1)} + \sqrt{\lambda_2} u^{(2)}$. These show that squaring or taking square-root on a vector is the same as squaring or taking square-root on the spectral values, and the associated spectral vectors remain invariant.

By using the spectral factorization (2.2)-(2.3), a scalar function $\hat{f} : \mathbb{R} \to \mathbb{R}$ can be extended to an SOC vector-valued function associated with $\mathcal{K}^n (n \geq 1)$ [5, 11], which is given by

$$f(x) = \hat{f}(\lambda_1) u^{(1)} + \hat{f}(\lambda_2) u^{(2)}, \ \forall x = (x_1, x_2) \in \mathbb{R} \times \mathbb{R}^{n-1}, \tag{2.4}$$

where $\lambda_1, \lambda_2$ are the spectral values and $u^{(1)}, u^{(2)}$ are the associated spectral vectors.

For any $x \in \mathbb{R}^n$, the nearest-point (in the Euclidean norm) of $x$ onto $\mathcal{K}^n$, denoted by $[x]_+$, is called the *projection* of $x$, i.e., $[x]_+ \in \mathcal{K}^n$ and satisfying

$$\|x - [x]_+\| = \min\{\|x - y\| \mid y \in \mathcal{K}^n\}. \tag{2.5}$$

Clearly, the projection (2.5) reduces to $[t]_+ = \max\{0, t\}$ for $t \in \mathbb{R}$ when $n = 1$. The following lemma show that $|x|$ and $[x]_+$ have the form (2.4) (see [11, Proposition 3.3]).

**Lemma 2.2.** *For any $x = (x_1, x_2) \in \mathbb{R} \times \mathbb{R}^{n-1}$ with spectral factorization (2.2)-(2.3), then*

(1) $|x| = (x^2)^{1/2} = |\lambda_1| u^{(1)} + |\lambda_2| u^{(2)}$.

(2) *The projection of $x$ on $\mathcal{K}^n$ can be written as $[x]_+ = (x + |x|)/2 = [\lambda_1]_+ u^{(1)} + [\lambda_2]_+ u^{(2)}$, where for any scalar $\alpha \in \mathbb{R}, [\alpha]_+ = \max\{0, \alpha\}$.*

Similar to the concept of projection on $\mathcal{K}^n$, define [7]

$$[x]_- = [\lambda_1]_- u^{(1)} + [\lambda_2]_- u^{(2)}$$

for any vector $x$ with spectral factorization (2.2)-(2.3), where $[\lambda_i]_- = \max\{0, -\lambda_i\}$ for $i = 1, 2$. It is obvious that $[x]_-$ means the projection point of $-x$ onto $\mathcal{K}^n$, and $x = [x]_+ - [x]_-$, $[x]_+, [x]_- \in \mathcal{K}^n$ and $[x]_+ \circ [x]_- = 0$.

Putting these analyses for a single block SOC $\mathcal{K}^{n_i}, i = 1, \ldots, r$ into (1.2), we can extend them to the general case $\mathcal{K} = \mathcal{K}^{n_1} \times \cdots \times \mathcal{K}^{n_r}$. More specifically, for any $x = (x_1, \ldots, x_r) \in \mathbb{R}^{n_1} \times \ldots \times \mathbb{R}^{n_r}$, $y = (y_1, \ldots, y_r) \in \mathbb{R}^{n_1} \times \ldots \times \mathbb{R}^{n_r}$, their *Jordan product* is defined as

$$x \circ y := (x_1 \circ y_1, \ldots, x_r \circ y_r).$$

Let $[x]_+, [x]_-$ respectively denote the projection of $x$ onto $\mathcal{K}$ and the projection of $-x$ onto the dual cone $\mathcal{K}^* = \mathcal{K}$, then

$$[x]_+ := ([x_1]_+, \ldots, [x_r]_+), \ [x]_- := ([x_1]_-, \ldots, [x_r]_-). \tag{2.6}$$

where $[x_i]_+$, $[x_i]_-$, $i = 1, \ldots, r$ respectively denote the projection of $x_i$ onto the single block SOC $\mathcal{K}^{n_i}$ and the projection of $-x_i$ onto $(\mathcal{K}^{n_i})^*$.

## 3. Two new classes of smooth functions

This section is devoted to propose two new smooth functions. Based on the four smooth functions presented by Hao in [13], this study constructs the novel smoothing functions by referring to the method proposed by Chen and Managsarian [2]. Specifically, the kernel functions are convoluted to be either the plus function $[t]_+ = \max\{0, t\}$ or minus function $[t]_- = \max\{0, -t\}$. First, we consider the piecewise continuous function $d(t)$ with finite number of pieces, which is a density (kernel) function. In other words, it satisfies

$$d(t) \geq 0, \int_{-\infty}^{+\infty} d(t)dt = 1. \tag{3.1}$$

Next, we define $\hat{s}(\mu, t) := \frac{1}{\mu} d(\frac{t}{\mu})$, where $\mu$ is a positive parameter. If $\int_{-\infty}^{+\infty} |t|d(t)dt < +\infty$, then a smoothing approximation for $[t]_+$ is formed. In particular,

$$\phi^+(\mu, t) = \int_{-\infty}^{+\infty} (t - s)_+ \hat{s}(\mu, s)ds = \int_{-\infty}^{t} (t - s)\hat{s}(\mu, s)ds \approx [t]_+. \tag{3.2}$$

The following proposition states the properties of $\phi^+(\mu, t)$, whose proofs can be found in [2, Proposition 2.2].

**Proposition 3.1.** *Let $d(t)$ be a density function satisfying (3.1) and $\hat{s}(\mu, t) := \frac{1}{\mu} d(\frac{t}{\mu})$ with positive parameter $\mu$. If $d(t)$ is piecewise continuous with finite number of pieces and $\int_{-\infty}^{+\infty} |t|d(t)dt < +\infty$. Then, the function $\phi^+(\mu, t)$ defined by (3.2) possesses the following properties.*

(1) *$\phi^+(\mu, t)$ is continuously differentiable.*
(2) *$-D_2\mu \leq \phi^+(\mu, t) - [t]_+ \leq D_1\mu$, where*

$$D_1 = \int_{-\infty}^{0} |t|d(t)dt, \; D_2 = max\{\int_{-\infty}^{+\infty} td(t)dt, 0\}.$$

(3) *$\frac{\partial}{\partial t}\phi^+(\mu, t)$ is bounded satisfying $0 \leq \frac{\partial}{\partial t}\phi^+(\mu, t) \leq 1$.*

Applying the above way of generating smoothing function to approximate $[t]_- = \max\{0, -t\}$, we also achieve a smoothing approximation

$$\phi^-(\mu, t) = \int_{-\infty}^{-t} (-t - s)\hat{s}(\mu, -s)ds = \int_{t}^{+\infty} (s - t)\hat{s}(\mu, s)ds \approx [t]_-. \tag{3.3}$$

Similar to Proposition 3.1, we have the below properties for $\phi^-(\mu, t)$ (see [13, Proposition 3.2]).

**Proposition 3.2.** *Let $d(t)$ and $\hat{s}(\mu, t)$ be as in Proposition 3.1 with the same assumptions. Then, the function $\phi^-(\mu, t)$ defined by (3.3) possesses the following properties.*

(1) *$\phi^-(\mu, t)$ is continuously differentiable.*
(2) *$-D_2\mu \leq \phi^-(\mu, t) - [t]_+ \leq D_1\mu$, where*

$$D_1 = \int_{0}^{\infty} |t|d(t)dt, \; D_2 = max\{\int_{-\infty}^{+\infty} td(t)dt, 0\}.$$

(3) *$\frac{\partial}{\partial t}\phi^-(\mu, t)$ is bounded satisfying $-1 \leq \frac{\partial}{\partial t}\phi^-(\mu, t) \leq 0$.*

From (3.2),(3.3) and Proposition 3.1, Proposition 3.2, we know that $\phi^+(\mu,t)$ defined by (3.2) and $\phi^-(\mu,t)$ define by (3.3), are the smoothing functions of $[t]_+$ and $[t]_-$, respectively, i.e.,

$$\lim_{\mu\to 0^+}\phi^+(\mu,t)=[t]_+, \tag{3.4}$$

$$\lim_{\mu\to 0^+}\phi^-(\mu,t)=[t]_-. \tag{3.5}$$

The following are the four specific smoothing functions of $[t]_-$ proposed by Hao et al, [13]

$$\phi_1^-(\mu,t) = -t+\mu\ln\left(1+e^{\frac{t}{\mu}}\right). \tag{3.6}$$

$$\phi_2^-(\mu,t) = \begin{cases} 0 & \text{if} \quad \mathrm{t}\geq \frac{\mu}{2},\\ \frac{1}{2\mu}\left(-t+\frac{\mu}{2}\right)^2 & \text{if} -\frac{\mu}{2}<\mathrm{t}<\frac{\mu}{2},\\ -t & \text{if} \quad \mathrm{t}\leq -\frac{\mu}{2}. \end{cases} \tag{3.7}$$

$$\phi_3^-(\mu,t) = \frac{\sqrt{4\mu^2+t^2}-t}{2}. \tag{3.8}$$

$$\phi_4^-(\mu,t) = \begin{cases} 0 & \text{if} \quad \mathrm{t}>0,\\ \frac{t^2}{2\mu} & \text{if} \quad -\mu\leq \mathrm{t}\leq 0,\\ -t-\frac{\mu}{2} & \text{if} \quad \mathrm{t}<-\mu. \end{cases} \tag{3.9}$$

where the corresponding kernel function are

$$\begin{aligned} d_1(t) &= \frac{e^t}{(1+e^t)^2}.\\ d_2(t) &= \begin{cases} 1 & \text{if } -\frac{1}{2}\leq \mathrm{t}\leq \frac{1}{2},\\ 0 & \text{otherwise.}\end{cases}\\ d_3(t) &= \frac{2}{(t^2+4)^{\frac{3}{2}}}.\\ d_4(t) &= \begin{cases} 1 & \text{if } -1\leq \mathrm{t}\leq 0,\\ 0 & \text{otherwise.}\end{cases} \end{aligned}$$

We now give two new specific smoothing functions for $[t]_-$:

$$\phi_5^-(\mu,t) = \begin{cases} 0 & \text{if} \quad \mathrm{t}\geq\mu,\\ \frac{\mu}{2}[\ln(1+(\frac{t}{\mu})^2)+1-\ln 2]-\frac{t}{2} & \text{if} -\mu<\mathrm{t}<\mu,\\ -t & \text{if} \quad \mathrm{t}\leq -\mu, \end{cases} \tag{3.10}$$

$$\phi_6^-(\mu,t) = \frac{1}{2}(\mu^2+t^2)^{\frac{1}{2}}-\frac{\mu}{2}-\frac{t}{2}, \tag{3.11}$$

where the corresponding kernel function are

$$\begin{aligned} d_5(t) &= \begin{cases} \frac{1-t^2}{(1+t^2)^2} & \text{if } -1<\mathrm{t}<1,\\ 0 & \text{otherwise,}\end{cases}\\ d_6(t) &= \frac{1}{2(t^2+1)^{\frac{3}{2}}}. \end{aligned}$$

For those specific function (3.6)-(3.11), they certainly obey Proposition 3.2 and (3.5). The grapes of $[t]_-$ and $\phi_i^-(\mu,t)$, $i=1,\dots,6$ with $\mu=0.1$ are depicted in Fig. 1.

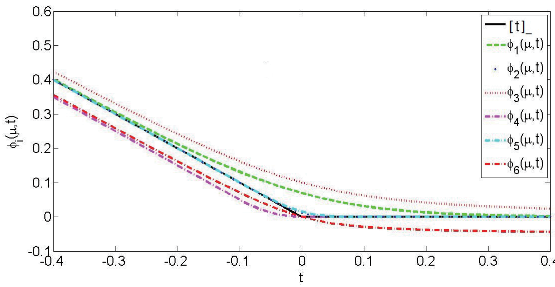


FIGURE 1. Graphs of $[t]_-$ and $\phi_i^-(\mu,t), i=1,\dots,6$ with $\mu=0.1$

From Fig. 1, we see that, for a fixed $\mu>0$, the function $\phi_2^-(\mu,t)$ seems the one which best approximate the function $[t]_-$ among all $\phi_i^-(\mu,t), i=1,\dots,6$. Indeed, for a fixed $\mu>0$ and all $t\in\mathbb{R}$, we have

$$\phi_3^-(\mu,t)\geq\phi_1^-(\mu,t)\geq\phi_5^-(\mu,t)\geq\phi_2^-(\mu,t)\geq[t]_-\geq\phi_4^-(\mu,t),\ \phi_6^-(\mu,t). \tag{3.12}$$

For any fixed $\mu>0$, it is clear that

$$\lim_{t\to\infty}|\phi_i^-(\mu,t)-[t]_-|=0,\ i=1,2,3,5.$$

The functions $\phi_i^-(\mu,t)-[t]_-, i=1,3$ have no stable point but unique non-differentiable point $t=0$, and $\phi_2^-(\mu,t)-[t]_-$ is non-zero only on the interval $(-\mu/2,\mu/2)$ with $\max_{t\in(-\mu/2,\mu/2)}\left|\phi_2^-(\mu,t)-[t]_-\right|=\phi_2^-(\mu,0)$, and $\phi_5^-(\mu,t)-[t]_-$ is non-zero only on the interval $(-\mu,\mu)$ with $\max_{t\in(-\mu,\mu)}\left|\phi_5^-(\mu,t)-[t]_-\right|=\phi_5^-(\mu,0)$. These imply that

$$\max_{t\in\mathbb{R}}|\phi_i^-(\mu,t)-[t]_-|=\phi_i^-(\mu,0),\ i=1,2,3,5.$$

Therefore, in the sense of the infinite norm, we have

$$\begin{aligned}\|\phi_1^-(\mu,t)-[t]_-\|_\infty &= (\ln 2)\mu\approx 0.7\mu,\\ \|\phi_2^-(\mu,t)-[t]_-\|_\infty &= \mu/8,\\ \|\phi_3^-(\mu,t)-[t]_-\|_\infty &= \mu,\\ \|\phi_5^-(\mu,t)-[t]_-\|_\infty &= \frac{\mu(1-\ln 2)}{2}\approx 0.3\mu.\end{aligned}$$

On the other hand, it is obvious that

$$\|\phi_4^-(\mu,t)-[t]_-\|_\infty=\max_{t\in\mathbb{R}}|\phi_4^-(\mu,t)-[t]_-|=\mu/2,$$

$$\|\phi_6^-(\mu,t)-[t]_-\|_\infty=\max_{t\in\mathbb{R}}|\phi_6^-(\mu,t)-[t]_-|=\mu/2.$$

In summary, we have

$$\begin{aligned}&\|\phi_3^-(\mu,t)-[t]_-\|_\infty>\|\phi_1^-(\mu,t)-[t]_-\|_\infty>\|\phi_4^-(\mu,t)-[t]_-\|_\infty\\ &=\|\phi_6^-(\mu,t)-[t]_-\|_\infty>\|\phi_5^-(\mu,t)-[t]_-\|_\infty>\|\phi_2^-(\mu,t)-[t]_-\|_\infty.\end{aligned} \tag{3.13}$$

The orderings of (3.12) and (3.13) indicate the behavior of $\phi_i^-(\mu,t), i = 1,\dots,6$ for fixed $\mu > 0$. When taking $\mu \to 0^+$, we know $\lim_{\mu\to 0^+} \phi_i^-(\mu,t) = [t]_-, i = 1,\dots,6$, which can be verified by geometric views depicted as in Fig. 2.

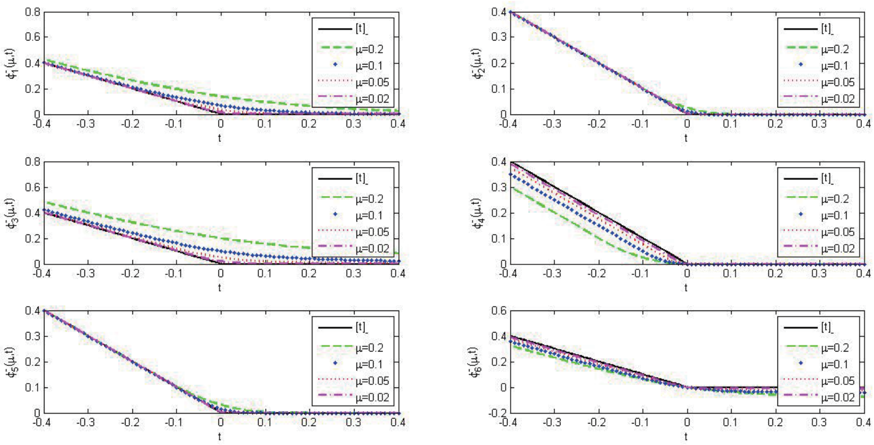

FIGURE 2. Graphs of $[t]_-$ and $\phi_i^-(\mu,t), i = 1,\dots,6$ with $\mu$

Through the above discussion, for any $x = (x_1,\dots,x_r) \in \mathbb{R}^{n_1}\times\cdots\times\mathbb{R}^{n_r}$, we are ready to show how to construct a smoothing function for vectors $[x]_+$, and $[x]_-$ associated with $\mathcal{K} = \mathcal{K}^{n_1}\times\cdots\times\mathcal{K}^{n_r}$. We start by constructing a smoothing function for vectors $[x_i]_+$, $[x_i]_-$ on a single block SOC $\mathcal{K}^{n_i}$, $i = 1\dots,r$ since $[x]_+$ and $[x]_-$ are show as (2.6) . First, given smoothing functions $\phi^+,\phi^-$ in (3.2), (3.3) and $x_i \in \mathbb{R}^{n_1}$, $i = 1\dots,r$, we define vector-value function $\Phi_i^+$, $\Phi_i^-$ : $\mathbb{R}_{++}\times\mathbb{R}^{n_i}\to\mathbb{R}^{n_i}$:

$$\Phi_i^+(\mu,x_i) := \phi^+(\mu,\lambda_1(x_i))u_{x_i}^{(1)} + \phi^+(\mu,\lambda_2(x_i))u_{x_i}^{(2)}, \tag{3.14}$$

$$\Phi_i^-(\mu,x_i) := \phi^-(\mu,\lambda_1(x_i))u_{x_i}^{(1)} + \phi^-(\mu,\lambda_2(x_i))u_{x_i}^{(2)}, \tag{3.15}$$

where $\mu \in \mathbb{R}_{++}$ is a parameter, $\lambda_1(x_i)$, $\lambda_2(x_i)$ are the spectral values, and $u_{x_i}^{(1)}$, $u_{x_i}^{(2)}$are the spectral vectors of $x_i$.

Consequently, $\Phi_i^+(\mu,x_i)$, $\Phi_i^-(\mu,x_i)$ are also smooth on $\mathbb{R}_{++}\times\mathbb{R}^{n_i}$ [5]. Moreover, it is easy to assert that

$$\lim_{\mu\to 0^+} \Phi_i^+(\mu,x_i) = [\lambda_1(x_i)]_+u_{x_i}^{(1)} + [\lambda_2(x_i)]_+u_{x_i}^{(2)} = [x_i]_+, \tag{3.16}$$

$$\lim_{\mu\to 0^+} \Phi_i^-(\mu,x_i) = [\lambda_1(x_i)]_-u_{x_i}^{(1)} + [\lambda_2(x_i)]_-u_{x_i}^{(2)} = [x_i]_-, \tag{3.17}$$

which means each function $\Phi_i^+(\mu,x_i)$, $\Phi_i^-(\mu,x_i)$ serves as a smoothing function of $[x_i]_+$, $[x_i]_-$ associated with single SOC $\mathcal{K}^{n_i}$, $i = 1\dots,r$, respectively.

Now we construct smoothing function for vector $[x]_+$ and $[x]_-$ associated with general cone (1.2). To this end, we define vector-valued function $\Phi^+$, $\Phi^-$: $\mathbb{R}_{++}\times$

$\mathbb{R}^n \to \mathbb{R}^n$:

$$\Phi^+(\mu, x) := (\Phi_1^+(\mu, x_1), \ldots, \Phi_r^+(\mu, x_r)), \tag{3.18}$$

$$\Phi^-(\mu, x) := (\Phi_1^-(\mu, x_1), \ldots, \Phi_r^-(\mu, x_r)), \tag{3.19}$$

where $\Phi_i^+(\mu, x_i)$, $\Phi_i^-(\mu, x_i), i = 1, \ldots, r$ are defined by (3.14), (3.15), respectively. Therefore, from (3.16), (3.17) and (2.6), $\Phi^+(\mu, x)$, $\Phi^-(\mu, x)$ serves as a smoothing function for $[x]_+$, $[x]_-$ associated with $\mathcal{K} = \mathcal{K}^{n_1} \times \cdots \times \mathcal{K}^{n_r}$, respectively.

**Lemma 3.3** (see [13])**.** *Suppose that $\Phi^+(\mu, x)$ and $\Phi^-(\mu, x)$ are defined by (3.18), (3.19), respectively. Then, $\Phi^+(\mu, x) \in \mathcal{K}, \Phi^-(\mu, x) \in \mathcal{K}$.*

## 4. Smooth-type algorithm and convergence analysis

In this section, using the six smooth functions of $[t]_-$ proposed in Section 3, we present the lower order penalty smooth-type algorithm for SOCLCPs and the corresponding convergence analysis.

We consider the lower order penalty smooth-type equations (LOPEs):

$$Ax - \alpha\Phi^-(\mu, x) = b, \tag{4.1}$$

where $\alpha \geq 1$ is a penalty parameter, $\mu > 0$ is a smoothing parameter and $\Phi^-(\mu, x)$ is defined as (3.19). The penalty term $\alpha\Phi^-(\mu, x)$ penalizes the 'negative part' of $x$ when $\alpha \to +\infty$ and $\mu \to 0^+$. By Lemma 3.3, it is easy to see that $Ax_{\mu,\alpha} - b \succeq_{\mathcal{K}} 0$ is always satisfied due to $\alpha\Phi^-(\mu, \alpha) \in \mathcal{K}$. Our goal is to show that the solution sequence $\{x_{\mu,\alpha}\}$ converges to the solution of SOCLCP (1.1) when $\alpha \to +\infty$ and $\mu \to 0^+$. Thus we convert SOCLCP (1.1) into LOPEs (4.1).

Equations (4.1) are different from $Ax - \alpha\Phi^-(\mu, x)^\sigma = b$, the lower order penalty equations described in Hao [13]. Superficially, (4.1) is a special form of $\sigma = 1$. However, for solving the nonlinear equations with $\sigma \neq 1$ in [13], it is still require to add an inner loop to transform them into $\sigma = 1$. By taking $\sigma = 1$ in (4.1), it is not necessary the inner loop. Additionally, the equations $Ax - \alpha\Phi^-(\mu, x)^\sigma = b$ in reference [13] are approximately smooth, and the discrete Newton method is used to solve the nonlinear equations. By contrast, the equations (4.1) in this paper are generally smooth, and the smooth Newton method is used to solve the equations.

Similar to reference [13], we make the assumption for matrix $A$ as below.

**Assumption 4.1.** The matrix $A$ is positive definite, but not necessarily symmetric, i.e., there exist a constant $a_0 > 0$, such that

$$y^T Ay \geq a_0\|y\|^2, \ \forall y \in \mathbb{R}^n. \tag{4.2}$$

Under Assumption 4.1, the SOCLCP (1.1) has a unique solution and the LOPEs (4.1) also has a unique solution, see for more details in [13]. Throughout this section, let $x_{\mu,\alpha}$ represent the unique solution of (4.1).

Since two new smooth functions of $[t]_-$ proposed in Section 3, $\phi_5^-(\mu, t)$, $\phi_6^-(\mu, t)$ and four smooth functions proposed by Hao in reference [13] are obtained by convolution integrals of their kernel functions, and they all satisfy Propositions 3.2 and (3.5), Therefore, the convergence analysis of the lower order penalty equations formed by $\phi_5^-(\mu, t)$, $\phi_6^-(\mu, t)$ is the same as that described in [13]. We only present the convergent result, while the detailed proof can be found in [13].

**Proposition 4.2.** *For any $\alpha \geq 1$, and sufficiently small $\mu$, the solution of LOPEs (4.1) is bounded, i.e., there exists a positive constant $M$, independent of $x_{\mu,\alpha}, \alpha$ and $\mu$, such that $\|x_{\mu,\alpha}\| \leq M$.*

**Proposition 4.3.** *For any $\alpha \geq 1$, and sufficiently small $\mu$, there exists a positive constant $C$, independent of $x_{\mu,\alpha}$, $\alpha$ and $\mu$, such that*

$$\|\Phi^-(\mu, x_{\mu,\alpha})\| \leq \frac{C}{\alpha}. \tag{4.3}$$

**Theorem 4.4.** *For any $\alpha \geq 1$ and sufficiently small $\mu$, let $x_{\mu,\alpha}$ and $x^*$ be the solution of LOPEs (4.1) and SOCLCP (1.1) respectively. Then, there exists a positive constant $C$, independent of $x^*, x_{\mu,\alpha}, \alpha$ and $\mu$, such that*

$$\|x^* - x_{\mu,\alpha}\| \leq \frac{C}{\alpha}. \tag{4.4}$$

According to Theorem 4.4, we given the algorithm as following.

**Algorithm 4.5.**

**Step 0** Given a vector $b \in \mathbb{R}^n$, and a matrix $A \in \mathbb{R}^n$ satisfying Assumption 4.1. Set $\tilde{x} = 0 \in \mathbb{R}^n$.
**Step 1** If $b \preceq_{\mathcal{K}} 0$, go to step 5; else, go to step 2.
**Step 2** Given the penalty parameter $\alpha \geq 1$, the smoothing parameter $0 < \mu < 1$, the error bound eps and the multiple parameter $c_1 > 1$ and $0 < c_2 < 1$, select an initial point $x^{(0)} = (x_1^{(0)}, \ldots, x_r^{(0)}) \in \mathbb{R}^{n_1} \times \cdots \times \mathbb{R}^{n_r}$ with $x_i^{(0)} = (x_{i1}^{(0)}, x_{i2}^{(0)}) \in \mathbb{R} \times \mathbb{R}^{n_i - 1}$ by taking $x_{i2}^{(0)} \neq 0$ while $x_{i1}^{(0)} \leq 0, i = 1, \ldots, r$.
**Step 3** For the parameters $\alpha, \mu$ and the initial point $x^{(0)}$, solve the nonlinear equations

$$Ax - \alpha\Phi^-(\mu, x) = b. \tag{4.5}$$

Suppose that $x_{\mu,\alpha}$ is the solution of (4.5) and let $\text{Tol} = |x_{\mu,\alpha}^T(Ax_{\mu,\alpha} - b)|$.
**Step 4** If Tol $\leq$ eps, set $\tilde{x} = x_{\mu,\alpha}$, go to step 5; else, let $x^{(0)} = x_{\mu,\alpha}, \alpha = c_1\alpha$ and $\mu = c_2\mu$, go to step 3.
**Step 5** The vector $\tilde{x}$ is the approximate optimal solution of SOCLCP (1.1), stop.

## 5. Numerical experiments

5.1. **Numerical examples.** We test some examples to show the efficiency of Algorithm 4.5. We use smooth Newton method to solve nonlinear equations for all examples. All numerical experiments are performed under the MATLAB 2012a running on PC with Intel(R) Core(TM)i5-2410M CPU 2.3GHz.

In the following numerical experiments, we mainly compare the two newly proposed novel functions $\phi_5^-(\mu, t), \phi_6^-(\mu, t)$ and the functions $\phi_2^-(\mu, t), [t]_-$. This is because numerical experiments in [13] have demonstrated that $\phi_2^-(\mu, t)$ is the function with the best numerical performance among all $\phi_i^-(\mu, t), i = 1, 2, 3, 4$, and they're all based on $[t]_-$.

**Example 5.1.** Consider the SOCLCP (1.1) on $\mathcal{K}^2$, where

$$A = \begin{pmatrix} 1 & 1 \\ 0 & 2 \end{pmatrix}, \; b = \begin{pmatrix} 0 \\ 4 \end{pmatrix}.$$

Nothing that the exact solution of this problem is $x^* = (1,1)^T$, we take an initial point $x^{(0)} = (0,2)^T$ and consider the tendency of numerical results while parameters $\mu, \alpha$ change. In the following two tables, "Err" denotes $\|\tilde{x} - x^*\|$, "Val" denotes $\tilde{x}^T(A\tilde{x} - b)$, where $\tilde{x}$ is the numerical solution.

We do the numerical experiment according to the following two steps:

1. First, we take $\mu = 1e-5$ and some $\alpha = 100, 400, 1600, 6400, 25600, 102400$, the numerical results are listed in Table 5.1;
2. Second, we take $\alpha = 1000$ and some $\mu = 0.5, 0.2, 0.1, 0.01, 0.001, 0.0001$, the numerical results are listed in Table 5.2.

According to Table 5.1, it can be seen that Err decreases as $\alpha$ increases and the values of Val are all negative, which indicates that the value of $\mu$ should decrease while the value of $\alpha$ increases. From Table 5.2, it can be seen that the overall situation is that as $\mu$ decreases, Val changes from positive to negative, and Err decreases but the trend is not obvious after Val becomes negative. For $\phi_5^-$ for example, Val is positive when $\mu = 0.2$, but negative when $\mu = 0.1$, which shows that $\mu = 0.1$ is sufficiently small when $\alpha = 1000$, and what is needed is to increase the value of $\alpha$ to obtain better numerical results. Therefore, it is necessary to increase the value of $\alpha$ while decreasing the value of $\mu$, which is consistent with the results of Algorithm 4.5.

Table 5.1 The numerical results for $\alpha$ change ($\mu = 1e-5$)

| $\alpha \to$ | | 100 | 400 | 1600 | 6400 | 25600 | 102400 |
|---|---|---|---|---|---|---|---|
| $\phi_2^-$ | Err | 0.031310 | 0.007886 | 0.001975 | 4.9403e-4 | 1.2352e-4 | 3.0881e-5 |
| | Val | -0.078424 | -0.019900 | -0.004994 | -0.001250 | -3.1248e-4 | -7.8123e-5 |
| $\phi_5^-$ | Err | 0.031310 | 0.007886 | 0.001975 | 4.9403e-4 | 1.2352e-4 | 3.0881e-5 |
| | Val | -0.078424 | -0.019900 | -0.004994 | -0.001250 | -3.1248e-4 | -7.8123e-5 |
| $\phi_6^-$ | Err | 0.031310 | 0.008230 | 0.001975 | 0.011545 | 0.045312 | 0.181034 |
| | Val | -0.078424 | -0.021904 | -0.004994 | -0.032998 | -0.124221 | -0.446544 |

Table 5.2 The numerical results for $\mu$ change ($\alpha = 1000$)

| $\mu \to$ | | 0.5 | 0.2 | 0.1 | 0.01 | 0.001 | 0.0001 |
|---|---|---|---|---|---|---|---|
| $\phi_2^-$ | Err | 0.146497 | 0.047199 | 0.017107 | 0.003115 | 0.003159 | 0.003159 |
| | Val | 0.387781 | 0.121187 | 0.043513 | -0.007872 | -0.007984 | -0.007984 |
| $\phi_5^-$ | Err | 0.228305 | 0.071374 | 0.026004 | 0.002897 | 0.003159 | 0.003159 |
| | Val | 0.619269 | 0.184638 | -0.007322 | -0.078424 | -0.078423 | -0.078423 |
| $\phi_6^-$ | Err | 0.321071 | 0.013340 | 0.028655 | 0.031310 | 0.031310 | 0.031310 |
| | Val | 0.894722 | 0.112603 | 0.047199 | 0.002091 | 0.003159 | 0.003159 |

The following two test examples are employed in [6], which will be solved by Algorithm 4.5. In our tests, we employ eps $= 1e-6$ as the termination criterion. In the following tables, IP($x^{(0)}$) denotes the initial points, Val dentes $|\tilde{x}^T(A\tilde{x} - b)|$, where $\tilde{x}$ is the numerical solution.

**Example 5.2.** Consider the SOCLCP (1.1) on $\mathcal{K}^5$, where

$$A = \begin{pmatrix} 15 & -5 & -1 & 4 & -5 \\ 0 & 5 & 0 & 0 & 1 \\ -1 & -3 & 8 & 2 & -3 \\ 2 & -4 & 2 & 9 & -4 \\ 0 & -5 & 0 & 0 & 10 \end{pmatrix}, \quad b = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 1 \end{pmatrix}.$$

In this example, the matrix $A$ is positive definite, but not symmetric, i.e., Assumption 4.1 holds. The exact solution $x^* \approx (0.049185,\ -0.0030997, 0.009$ $6024,\ 0.0031883,\ 0.048033)^T$ [6]. For different initial points, by taking $c_1 = 10$, $c_2 = 0.1$, initial $\alpha = 100$ and initial $\mu = 1e-5$, the test results are listed in Table 5.3.

Table 5.3 Numerical results for different initial points

| IP($x^{(0)}$) | Val($\phi_2^-$) | Val($\phi_5^-$) | Val($\phi_6^-$) | Val($[t]_-$) |
|---|---|---|---|---|
| $(1,1,1,1,1)$ | 5.3303e-7 | 5.3303e-7 | 7.7896e-7 | 5.3303e-7 |
| $(-1,\ldots,-1)$ | 5.3303e-7 | 5.3303e-7 | 7.7896e-7 | 5.3303e-7 |
| $(10,\ldots,10)$ | 5.3303e-7 | 5.3303e-7 | 7.7896e-7 | 5.3303e-7 |
| $(-10,\ldots,-10)$ | 5.3303e-7 | 5.3303e-7 | 7.7896e-7 | 5.3303e-7 |
| $(10^3,\ldots,10^3)$ | 5.3303e-7 | 5.3303e-7 | 7.7896e-7 | 5.3303e-7 |
| $(10^6,\ldots,10^6)$ | 5.3303e-7 | 5.3303e-7 | 7.7896e-7 | 5.3303e-7 |

**Example 5.3.** Consider the SOCLCP (1.1) on $\mathcal{K}^3$, where

$$A = \begin{pmatrix} 21 & -9 & 18 \\ -9 & 4 & -7 \\ 18 & -7 & 9 \end{pmatrix}, \quad b = \begin{pmatrix} -3 \\ -7 \\ -1 \end{pmatrix}.$$

In this example, the symmetric matric $A$ is positive semidefinite, but not positive definite. As indicated in [6], it has one solution $x^* \approx (0.183606,$ $-0.154346,\ -0.099440)^T$. For different initial points, we test this problem by taking $c_1 = 10$, $c_2 = 0.1$, initial $\alpha = 100$ and initial $\mu = 1e-6$, the results are listed in Table 5.4. This example indicates that, the Algorithm 4.5 is also applicable to those SOCLCPs, in which the matric $A$ is only positive semidefinite.

Table 5.4 Numerical results for different initial points

| IP($x^{(0)}$) | Val($\phi_2^-$) | Val($\phi_5^-$) | Val($\phi_6^-$) | Val($[t]_-$) |
|---|---|---|---|---|
| $(1,1,1)$ | 8.3325e-8 | 8.3325e-7 | 9.9999e-7 | 8.3325e-7 |
| $(-1,-1,-1)$ | 8.3325e-8 | 8.3325e-7 | 9.3721e-7 | 8.3325e-7 |
| $(10,10,10)$ | 8.3328e-8 | 8.3328e-7 | 9.1295e-7 | 8.3328e-7 |
| $(-10,-10,-10)$ | 8.3328e-8 | 8.3328e-7 | 9.1532e-7 | 8.3328e-7 |
| $(10^3,10^3,10^3)$ | 8.3328e-8 | 8.3328e-7 | 9.1295e-7 | 8.3328e-7 |
| $(10^6,10^6,10^6)$ | 8.3323e-8 | 8.3323e-7 | 9.9824e-7 | 8.3323e-7 |

The results in Table 5.3 and 5.4 show that Algorithm 4.5 does not have high requirements for initial points. Example 5.2 and 5.3 are two example of SOCLCP with SOC $\mathcal{K} = \mathcal{K}^n$. Next, we consider two examples of the SOCLCP (1.1) with multiple SOCs. In the following tables, IP($x^{(0)}$) denotes the initial points, Val dentes $|\tilde{x}^T(A\tilde{x} - b)|$, "-"denotes the Jacobian matrix closed to singular or badly scaled.

**Example 5.4.** Consider the SOCLCP (1.1) on $\mathcal{K}^3 \times \mathcal{K}^2$, where $A$ is show as in Example 5.2, and $b = (3, 0, 2, 2, 5)^T$.

The Assumption 4.1 also holds in this example. As indicated in [13], it has one solution $x^* \approx (0.255103,\ -0.053464,\ 0.249438,\ 0.367316,\ 0.367316)^T$. For different initial points, we test this problem by taking $c_1 = 10$, $c_2 = 0.1$, initial $\alpha = 100$ and initial $\mu = 1e-6$, the results are listed in Table 5.5.

Table 5.5 Numerical results for different initial points

| IP($x^{(0)}$) | Val($\phi_2^-$) | Val($\phi_5^-$) | Val($\phi_6^-$) | Val($[t]_-$) |
|---|---|---|---|---|
| $(1,1,1,1,1,1,1)$ | 2.6994e-7 | 2.7001e-7 | 5.8123e-7 | - |
| $(-1,\ldots,-1)$ | - | 2.6997e-7 | 5.8150e-7 | 2.6996e-7 |
| $(10,\ldots,10)$ | 2.7001e-7 | 2.7014e-7 | 5.8106e-7 | - |
| $(-10,\ldots,-10)$ | - | 2.6997e-7 | 5.8126e-7 | 2.6994e-7 |
| $(10^3,\ldots,10^3)$ | 2.7001e-7 | 2.7014e-7 | 5.8110e-7 | - |
| $(10^6,\ldots,10^6)$ | 2.7001e-7 | 2.7014e-7 | 5.8115e-7 | - |

**Example 5.5.** Consider the SOCLCP (1.1) on $\mathcal{K}^3 \times \mathcal{K}^4$, where

$$A = \begin{pmatrix} 2.9825 & -0.1495 & -1.1296 & 0.0953 & -0.9764 & 0.2920 & 0.2027 \\ -0.1495 & 4.1709 & -0.8850 & -0.8571 & 0.0150 & -0.1140 & -0.9578 \\ -1.1296 & -0.8850 & 3.6707 & -0.7818 & -0.2644 & -0.2435 & -0.2995 \\ 0.0923 & -0.8571 & -0.7818 & 3.6308 & 0.5886 & -1.8208 & -0.1765 \\ -0.9764 & 0.0150 & -0.2644 & 0.5886 & 3.3823 & -1.1758 & -0.9048 \\ 0.2920 & -0.1140 & -0.2435 & -1.8208 & -1.1758 & 5.2118 & -0.4727 \\ 0.2027 & -0.9578 & -0.2995 & -0.1765 & -0.9048 & -0.4722 & 5.0411 \end{pmatrix}$$

and $b = (2, -1, 3, -2, 4, -1, 3)^T$.

The Assumption 4.1 also holds since $A$ is symmetry positive definite. the exact solution is about $x^* \approx (1.723509,\ 0.465966,\ 1.659325,\ 0.996148,\ 0.831870,\ 0.212798,\ 0.504995)^T$. For different initial points, we test this problem by taking $c_1 = 10$, $c_2 = 0.1$, initial $\alpha = 100$ and initial $\mu = 1e-6$, the results are listed in Table 5.6.

Table 5.6 Numerical results for different initial points

| IP($x^{(0)}$) | Val($\phi_2^-$) | Val($\phi_5^-$) | Val($\phi_6^-$) | Val($[t]_-$) |
|---|---|---|---|---|
| $(1,1,1,1,1)$ | 3.5406e-7 | 3.5406e-7 | - | 3.5406e-7 |
| $(-1,\ldots,-1)$ | 3.5099e-7 | 3.5099e-7 | - | 3.5099e-7 |
| $(10,\ldots,10)$ | 3.5525e-7 | 3.5525e-7 | - | 3.5525e-7 |
| $(-10,\ldots,-10)$ | 3.4620e-7 | 3.4620e-7 | - | 3.4620e-7 |
| $(10^3,\ldots,10^3)$ | 3.2642e-7 | 3.2642e-7 | - | 3.2642e-7 |
| $(10^6,\ldots,10^6)$ | 3.4341e-7 | 3.4341e-7 | - | 3.4341e-7 |

Examples 5.4 and 5.5 illustrate that Algorithm 4.5 is applicable to the SOCLCP (1.1) with multiple SOCs.

5.2. **Performance profile of different $\phi_i^-(\mu,t)$.** According to [13], $\phi_2^-(\mu,t)$ is the function closest to $[t]_-$ among all $\phi_i^-(\mu,t)$ $i = 1,2,3,4$. Therefore, we only compare the performance of $\phi_2^-(\mu,t)$ with $\phi_5^-(\mu,t)$, $\phi_6^-(\mu,t)$ proposed in this paper. In order to compare the performance of function $\phi_i^-(\mu,t)$ $i = 2,5,6$, we consider the performance profile which is introduced in [9] as a means. Assume that there are $n_s$ solvers and $n_p$ test problems from the test set $\mathcal{P}$. We are interested in using computing time or iteration number as a performance measure. In the following, we

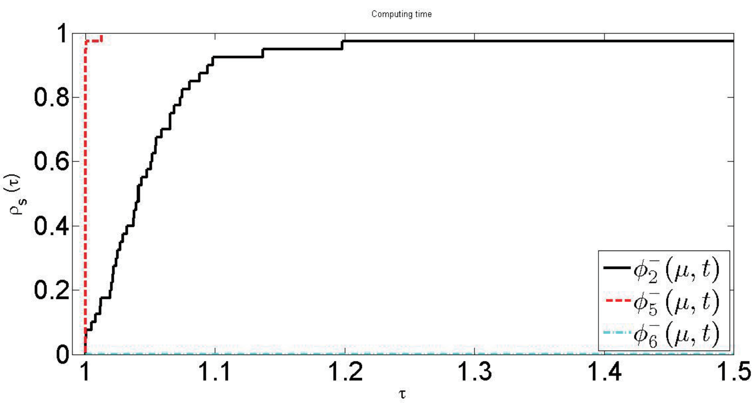


FIGURE 3. Performance profile of $\phi_i(\mu,t), i = 2,5,6$

take computing time as a performance measure. For each problem $p$ and solver $s$, we define

$$f_{p,s} = \text{computer time required to solved problem } p \text{ by solver } s.$$

We employ the performance ratio

$$r_{p,s} = \frac{f_{p,s}}{\min\{f_{p,s}|s \in \mathcal{S}\}},$$

where $\mathcal{S}$ is the solver set. We assume that a parameter $r_M$, such that $r_M \geq r_{p,s}$ for all $p$, $s$ is chosen, and $r_M = r_{p,s}$ if and only if solver $s$ does not solve problem $p$. The choice of $r_M$ does not affect the performance evaluation. In order to obtain an overall assessment for each solver, we define

$$\rho_s(\tau) = \frac{1}{n_p}\text{size}\{p \in \mathcal{P}|r_{p,s} \leq \tau\}.$$

The function $\rho_s(\tau)$ is the cumulative performance ratio, which is called the performance profile. In the performance profile, we use functions $\phi_i^-(\mu,t)$, $i = 2,5,6$ as three solvers, and take randomly generated 40 SOCLCPs with single SOC, in which the matrices are symmetric positive definite. The performance plot based on computing time is depicted in Fig. 3. By overall looking, from Fig. 3, we see that the function $\phi_5^-(\mu,t)$ has the best performance, then followed by $\phi_2^-(\mu,t)$. Note that the time efficiency of $\phi_6^-(\mu,t)$ is the worst. In other words, in view of computing time, there has

$$\phi_5^-(\mu,t) > \phi_2^-(\mu,t) > \phi_6^-(\mu,t),$$

where ">" means "better performance". In summary, for the SOCLCPs (1.1), when we use Algorithm 4.5 by applying functions $\phi_i^-(\mu,t)$ $i = 2,5,6$, the function $\phi_5^-(\mu,t)$ is the best choice.

## 6. CONCLUSIONS

Based on the four smooth functions in [13], two new smooth functions are proposed in this paper. The proposed functions are applied to the lower order penalty function algorithm for the SOCLCP (1.1). The general idea is to transform the SOCLCP (1.1) into a sequence of LOPEs (4.1) and to employ the smooth Newton method to solve the LOPEs. Under Assumption 4.1, the solution sequence of LOPEs (4.1) converges to the solution of SOCLCP (1.1). Numerical experiments indicates that the proposed two new smooth functions can be used in Algorithm 4.1 to solve the SOCLCP (1.1), and $\phi_5^-(\mu,t)$ has reached the best performance among $\phi_i^-(\mu,t),\ i=2,5,6$.

## References


[1] F. Alizadeh and D. Goldfarb, *Second-order cone programming*, J. Math. Program. **95** (2003), 3–51.

[2] C. Chen and O. L. Mangasarian, *A class of smoothing functions for nonlinear and mixed complementarity problems*, J. Comput. Optim. Appl. **5** (1996), 97–138.

[3] J. S. Chen, *The convex and monotone functions associated with second-order cone*, J. Optim. **55** (2006), 363–385.

[4] J. S. Chen, *Two classes of merit functions for the second-order cone complementarity problem*, J. Math. Meth. Oper. Res. **64** (2006), 495–519.

[5] J. S. Chen, X. Chen and P. Tseng, *Analysis of nonsmooth vector-valued functions associated with second- order cones*, J. Math. Program. **101** (2004), 95–117.

[6] L. Chen and C. Ma, *A modified smoothing and regularized Newton method for monotone second-order cone complementarity problems*, J. Comput. Math. Appl. **61** (2011), 1407–1418.

[7] X. D. Chen, D. Sun and J. Sun, *Complementarity functions and numerical experiments on some smoothing Newton methods for second-order-cone complementarity problems*, J. Comput. Optim. Appl. **25** (2003), 39–56.

[8] G. Di. Pillo and L. Grippo, *An exact penalty function method with global convergence properties for nonlinear programming problems*, J. Math. Program. **36** (1986), 1–18.

[9] E. D. Dolan and J. J. Moré, *Benchmarking optimization software with performance profiles*, J. Math. Program. **91** (2002), 201–213.

[10] J. Faraut and A. Korányi, *Analysis on Symmetric Sones*, Oxford University Press, 1994.

[11] M. Fukushima, Z. Q. Luo and P. Tseng, *Smoothing functions for second-order-cone complementarity problems*, SIAM J. Optim. **12** (2002), 436–460.

[12] S. P. Han and O. L. Mangasarian, *Exact penalty function in nonlinear programming*, J. Math. Program. **17** (1979), 251–269.

[13] Z. Hao, C. T. Nguyen and J. S. Chen, *An approximate lower order penalty approach for solving second-order cone linear complementarity problems*, J. Glob. Optim. **83** (2022), 671–697.

[14] Z. Hao, Z. Wan and X. Chi, *A power penalty method for second-order cone linear complementarity problems*, J. Oper. Res. Lett. **43** (2015), 137–142.

[15] Z. Hao, Z. Wan, X. Chi and Z. F. Jin, *A power penalty method for second-order cone nonlinear complementarity problems*, J. Comput. Appl. Math. **290** (2015), 136–149.

[16] Z. Hao, Z. Wan, X. Chi and Z. F. Jin, *Generalized lower-order penalty algorithm for solving second-order cone mixed complementarity problems*, J. Comput. Appl. Math. **385** (2021): 113168.

[17] S. Hayashi, T. Yamaguchi and N. Yamashita, *A matrix-splitting method for symmetric affine second-order cone complementarity problems*, J. SIAM J. Optim. **175** (2005), 335–353.

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

[19] Z. H. Huang and T. Ni, *Smoothing algorithms for complementarity problems over symmetric cones*, J. Comput. Optim. Appl. **45** (2010), 557–579.

[20] C. Huang and S. Wang, *A power penalty approach to a nonlinear complementarity problem*, J. Oper. Res. Lett. **38** (2010),72–76.

[21] C. Huang and S. Wang, *A penalty method for a mixed nonlinear complementarity problem*, J. Nonl. Anal. The. Meth. Appl. **75** (2012), 588–597.
[22] C. Kanzow, I. Ferenczi and M. Fukushima, *On the local convergence of semismooth Newton methods for linear and nonlinear second-order cone programs without strict complementarity*, J. SIAM J. Optim. **20** (2009), 297–320.
[23] Z. Q. Luo, J. S. Pang and D. Ralph, *Mathematical Programs with Equilibrium Constraints*, Cambridge University Press, 1996.
[24] R. D. C. Monteiro and T. Tsuchiya, *Polynomial convergence of primal-dual algorithms for the second-order cone program based on the MZ-family of directions*, J. Math. Program. **88** (2000), 61–83.
[25] S. Pan and J. S. Chen, *A damped Gauss-Newton method for the second-order cone complementarity problem*, J. Appl. Math. Optim. **59** (2009), 293–318.
[26] S. Wang and X. Yang, *A power penalty method for linear complementarity problems*, J. Oper. Res. Lett. **36** (2008), 211–214.
[27] W. I. Zangwill, *Non-linear programming via penalty functions*, J. Manag. Sci. **13** (1967), 344–358.

*Manuscript received February 19 2024*
*revised April 18 2024*

Y. Sun
School of Mathematics and Information Science, North Minzu University, Yinchuan 750021, China
*E-mail address*: `758203155@qq.com`

Z. Hao
School of Mathematics and Information Science, North Minzu University, Yinchuan 750021, China
*E-mail address*: `zijunhao@126.com`