SIAM J. OPTIM.
Vol. 22, No. 4, pp. 1607–1633

DIFFERENTIABLE EXACT PENALTY FUNCTIONS FOR NONLINEAR SECOND-ORDER CONE PROGRAMS*

ELLEN H. FUKUDA†, PAULO J. S. SILVA†, AND MASAO FUKUSHIMA‡

Abstract. We propose a method for solving nonlinear second-order cone programs (SOCPs), based on a continuously differentiable exact penalty function. The construction of the penalty function is given by incorporating a multipliers estimate in the augmented Lagrangian for SOCPs. Under the nondegeneracy assumption and the strong second-order sufficient condition, we show that a generalized Newton method has global and superlinear convergence. We also present some preliminary numerical experiments.

Key words. nonlinear second-order cone program, exact penalty function, semi-smooth reformulation, generalized Newton method

AMS subject classifications. 49M15, 65K05

DOI. 10.1137/110852401

1. Introduction.

We consider the following *nonlinear second-order cone program* (nonlinear SOCP):

$$\text{(SOCP)} \qquad \begin{array}{ll} \min & f(x) \\ \text{subject to} & g(x) \in \mathcal{K}, \\ & h(x) = 0, \end{array}$$

where $f\colon \mathbb{R}^n \to \mathbb{R}$, $g\colon \mathbb{R}^n \to \mathbb{R}^m$, and $h\colon \mathbb{R}^n \to \mathbb{R}^p$ are twice continuously differentiable functions, and $\mathcal{K} \doteq \mathcal{K}_1 \times \cdots \times \mathcal{K}_r$ is a Cartesian product of *second-order cones* (or *Lorentz cones*),

$$\mathcal{K}_i \doteq \left\{(y_0, \bar{y}) \in \mathbb{R} \times \mathbb{R}^{m_i - 1}\colon y_0 \geq \|\bar{y}\|\right\} \subset \mathbb{R}^{m_i}, \quad i = 1, \ldots, r,$$

with $\|\cdot\|$ denoting the Euclidean norm and $m_1 + \cdots + m_r = m$.

There are many problems that can be formulated as SOCPs in such diverse fields as engineering, finance, robust optimization, and combinatorial optimization [1, 28]. Although the SOCP can be viewed as a special case of the semidefinite programming problem, from the computational point of view, it is desirable to treat it directly, because of its particular structure [1]. If the functions f, g, and h are linear, the above problem is the *linear SOCP*, which has many efficient methods in the literature [1, 9, 28]. The treatment of *nonlinear SOCPs*, where the involved functions (especially the objective function f) are nonlinear, is more recent. Some works deal with theoretical properties or associated reformulations [7, 10, 18], but there have not been many efficient methods developed until now.

*Received by the editors October 20, 2011; accepted for publication (in revised form) September 28, 2012; published electronically December 12, 2012. This work was supported by FAPESP (grants 2010/20572-0, 2011/23638-5), PRONEX-Optimization (PRONEX-CNPq/FAPERJ E-26/171.510/2006-APQ1), CNPq grant 305740/2010-5, and a Grant-in-Aid for Scientific Research from Japan Society for the Promotion of Science.

http://www.siam.org/journals/siopt/22-4/85240.html

†Department of Applied Mathematics, Institute of Mathematics, Statistics and Computer Science, State University of Campinas, Campinas 13083-859, Brazil (ellen@ime.usp.br, pjssilva@ime.unicamp.br).

‡Department of Applied Mathematics and Physics, Graduate School of Informatics, Kyoto University, Kyoto 606-8501, Japan (fuku@i.kyoto-u.ac.jp).

Among the existing methods, we mention a sequential quadratic programming–type method from Kato and Fukushima [25], a primal-dual interior point method from Yamashita and Yabe [39], and an augmented Lagrangian method from Liu and Zhang [26, 27]. Each of these methods has some drawbacks and open questions. For example, there is no proof for the convergence rate in [39], the convergence is slow in [26, 27], the strict complementarity assumption for the convergence results is required in [26], and the treatment of the nonlinear SOCP is not direct in [25], because it replaces the original nonlinear SOCP with a sequence of linear SOCPs. Recently, Kanzow, Ferenczi, and Fukushima [24] proposed a method that deals with the nonlinear SOCP directly, with a fast convergence rate result, without using the strict complementarity condition. Basically, they construct a reformulation of the KKT system associated to the nonlinear SOCP, using a characterization of the projection mapping onto second-order cones [18]. Their reformulation is semismooth and a generalized Newton method applied to it converges locally with a superlinear rate. However, their analysis was done for a particular case of (SOCP), with $g(x) \doteq x$ and h being a linear mapping. Furthermore, a way to globalize the method was also not suggested.

In this work, we propose another method for solving general nonlinear SOCPs. More precisely, we construct a continuously differentiable *exact penalty* function for the nonlinear SOCP, which is actually an extension of differentiable exact penalty functions for nonlinear programming. The gradient mapping of this penalty function is semismooth, which makes it an SC^1 function (i.e., a continuously differentiable function whose gradient mapping is semismooth). This allows the use of a generalized Newton method. As in [24], we can prove that the convergence rate is superlinear (or quadratic) without the strict complementarity condition. Here we also suggest an implementable algorithm with global and superlinear convergence.

The literature of exact penalty functions for nonlinear programming is vast, but it is still an ongoing topic of research. Roughly speaking, a function $w_c \colon \mathbb{R}^n \to \mathbb{R}$, that depends on a positive parameter $c \in \mathbb{R}$, is an exact penalty function for the original problem if there is an appropriate choice of the penalty coefficient c such that a single minimization of w_c recovers a solution of the original problem. For nonlinear programming problems, Zangwill [40] proposed a nondifferentiable exact penalty function. Since it demands special methods to solve the unconstrained problem, many authors have developed continuously differentiable exact penalty functions afterward [5, 17, 19, 30]. In particular, in [11, 12], Di Pillo and Grippo proposed a differentiable exact penalty function based on a Lagrange multipliers estimate presented by Glad and Polak [19]. Their idea is to incorporate such an estimate in to the classical augmented Lagrangian function for nonlinear programming [22, 33, 36]. More recently, André and Silva [3] extended their idea for solving variational inequalities, incorporating the same multipliers estimate in to the augmented Lagrangian for this kind of problems [4].

The idea of the latter works [3, 11, 12] was further used by Andreani, Fukuda, and Silva [2] to solve nonlinear programming problems with both equality and inequality constraints. One advantage of the exact penalty method developed in [2] is that it does not deal with third-order derivatives, which is clearly important from the numerical point of view. Other ways to avoid third-order derivatives were also proposed in [13, 14]. Here, we take the ideas from the aforementioned works [3, 2, 11, 12, 19] to construct a continuously differentiable exact penalty function. We also extend Lucidi's idea [29] to construct a multipliers estimate that does not require a strong regularity assumption.

Throughout the paper, the following notation will be used. We denote by $\|\cdot\|$ and $\langle\cdot,\cdot\rangle$ the Euclidean norm and inner product, respectively. The notation $\mathbb{R}_{++}$ is used to represent the positive real numbers. The identity matrix with dimension ℓ is denoted by I_ℓ, and for any matrix Z, its transpose is denoted by $Z^\top$. For functions $p\colon \mathbb{R}^s \to \mathbb{R}$ and $q\colon \mathbb{R}^s \to \mathbb{R}^\ell$, the gradient of p and the Jacobian matrix of q at $x \in \mathbb{R}^s$ are given by $\nabla p(x) \in \mathbb{R}^s$ and $Jq(x) \in \mathbb{R}^{\ell \times s}$, respectively. For a second-order cone $\mathcal{K}_i \subset \mathbb{R}^{m_i}$, its interior, boundary, and boundary excluding the origin are denoted by $\mathrm{int}(\mathcal{K}_i)$, $\mathrm{bd}(\mathcal{K}_i)$, and $\mathrm{bd}^+(\mathcal{K}_i)$, respectively.

The paper is organized as follows. In section 2, we begin with some necessary notation and results associated to second-order cones and semismooth functions. The construction of the Lagrange multipliers estimate and the continuously differentiable exact penalty function are given in section 3. In section 4, we present the exactness results. In section 5, we show that a generalized Newton method converges superlinearly, under the strong second-order sufficient condition. In section 6, we present an implementable method that has global and superlinear convergence. Finally, in section 7, we show some preliminary numerical experiments.

2. Preliminaries. In this section we introduce some notation and results that will be used in this work. For any vector $y \in \mathbb{R}^\ell$, we consider the block notation $y \doteq (y_0, \bar{y}) \in \mathbb{R} \times \mathbb{R}^{\ell-1}$. Moreover, the *Jordan product* of $y, z \in \mathbb{R}^\ell$ is defined by

$$y \circ z \doteq \begin{bmatrix} \langle y, z\rangle \\ y_0 \bar{z} + z_0 \bar{y} \end{bmatrix}.$$

For the sake of completeness, we state some relations involving the Jordan product [1, section 4]. For any vectors $w, y, z \in \mathbb{R}^\ell$, the following properties hold:

(a) $y \circ z = z \circ y$; (commutativity 1)
(b) $y \circ (y^2 \circ z) = y^2 \circ (y \circ z)$, where $y^2 = y \circ y$; (commutativity 2)
(c) $\mathbf{e} \circ y = y \circ \mathbf{e} = y$, where $\mathbf{e} \doteq (1, 0, \ldots, 0) \in \mathbb{R}^\ell$; (identity)
(d) $(w+y) \circ z = (w \circ z) + (y \circ z)$. (distributivity)

We recall that the Jordan product is not associative in general. Besides, for any $y \in \mathbb{R}^\ell$, if we define the symmetric matrix

$$\mathrm{Arw}(y) \doteq \begin{bmatrix} y_0 & \bar{y}^\top \\ \bar{y} & y_0 I_{\ell-1} \end{bmatrix},$$

then $y \circ z = \mathrm{Arw}(y) z$ for all $z \in \mathbb{R}^\ell$. The matrix $\mathrm{Arw}(y)$ is called the *arrow matrix* of y. It is positive definite if and only if $y \in \mathrm{int}(\mathcal{K})$. Also, if $y \in \mathcal{K}$, then $\mathrm{Arw}(y)$ is singular if and only if $y \in \mathrm{bd}(\mathcal{K})$. Let us now give a result that will be used later.

LEMMA 2.1. *Let $p, q\colon \mathbb{R}^s \to \mathbb{R}^\ell$ be differentiable mappings. Then the Jacobian of the mapping $(p \circ q)\colon \mathbb{R}^s \to \mathbb{R}^\ell$ at $x \in \mathbb{R}^s$ is given by*

$$J(p \circ q)(x) = \mathrm{Arw}(p(x)) Jq(x) + \mathrm{Arw}(q(x)) Jp(x).$$

Proof. We omit the proof since it follows easily from the product rule associated with the Jordan product. ∎

For simplicity, let us now consider the ℓ-dimensional second-order cone $\mathcal{K} \doteq \{(y_0, \bar{y}) \in \mathbb{R} \times \mathbb{R}^{\ell-1} \colon y_0 \geq \|\bar{y}\|\}$. This cone is self-dual, which means that $\mathcal{K}$ is equal to its dual cone $\mathcal{K}^* \doteq \{z \in \mathbb{R}^\ell \colon \langle z, y\rangle \geq 0 \text{ for all } y \in \mathcal{K}\}$.

LEMMA 2.2. *Let $y, z \in \mathbb{R}^\ell$, and let $\mathcal{K} \subset \mathbb{R}^\ell$ be a second-order cone. Then, the following are equivalent:* (a) *$y, z \in \mathcal{K}$ and $y \circ z = 0$* (b) *$y, z \in \mathcal{K}$ and $\langle y, z\rangle = 0$ and* (c) *$y - P_\mathcal{K}(y - z) = 0$, where $P_\mathcal{K}$ denotes the orthogonal projection onto $\mathcal{K}$.*

Proof. See [18, Propositions 2.1 and 4.1]. ∎

Concerning the projection operator $P_{\mathcal{K}}$, we also have the following useful results.

LEMMA 2.3. *Let $y, z \in \mathbb{R}^\ell$, and let $\mathcal{K} \subset \mathbb{R}^\ell$ be a second-order cone. Then, the following statements hold:*

(a) $P_{\mathcal{K}}(z) = P_{\mathcal{K}}(-z) + z$;

(b) $\langle P_{\mathcal{K}}(z), P_{\mathcal{K}}(-z)\rangle = 0$;

(c) $\|P_{\mathcal{K}}(y - z) - y\| \geq \|P_{\mathcal{K}}(-z)\|$.

Proof. (a) It is straightforward from [23, section 3.2].

(b) The inequality $\langle P_{\mathcal{K}}(z), P_{\mathcal{K}}(-z)\rangle \geq 0$ holds because $\mathcal{K}$ is self-dual. Also, from a variational inequality characterization of projections, we have $\langle z - P_{\mathcal{K}}(z), y - P_{\mathcal{K}}(z)\rangle \leq 0$ for all $y \in \mathcal{K}$, which along with (a) implies $\langle P_{\mathcal{K}}(-z), y - P_{\mathcal{K}}(z)\rangle \geq 0$. Taking $y = 0$, we obtain $\langle P_{\mathcal{K}}(z), P_{\mathcal{K}}(-z)\rangle \leq 0$ and the proof is complete.

(c) From (a), $P_{\mathcal{K}}(y - z) = P_{\mathcal{K}}(z - y) - z + y$ and $P_{\mathcal{K}}(-z) = P_{\mathcal{K}}(z) - z$. Thus, the inequality (c) is equivalent to $\|P_{\mathcal{K}}(z - y) - z\| \geq \|P_{\mathcal{K}}(z) - z\|$. But this inequality holds because, from the definition of projection, $\|P_{\mathcal{K}}(z) - z\| = \min_{u\in\mathcal{K}} \|u - z\|$. ◻

Now, let $W : \mathbb{R}^s \to \mathbb{R}^t$ be a locally Lipschitz continuous function, and let $D_W \subseteq \mathbb{R}^s$ be the set of points where W is differentiable. Then, the set

$$\partial_B W(x) \doteq \left\{ V \in \mathbb{R}^{t\times s} : V = \lim_{D_W \ni x^k \to x} JW(x^k) \right\}$$

is nonempty and called the *B-subdifferential* of W at $x \in \mathbb{R}^s$. Its convex hull $\partial W(x) \doteq \mathrm{conv}\partial_B W(x)$ is called the *generalized Jacobian* (of Clarke). These definitions show that $\partial W(x) = \partial_B W(x) = \{JW(x)\}$ when W is continuously differentiable at x. Moreover, W is said to be *semismooth* at $x \in \mathbb{R}^s$ if W is directionally differentiable at x and $W(x + d) - W(x) - Vd = o(\|d\|)$ for any $V \in \partial W(x + d)$ with $d \to 0$. If W is semismooth at x and $W(x + d) - W(x) - Vd = O(\|d\|^2)$ for any $V \in \partial W(x + d)$ with $d \to 0$, then W is said to be *strongly semismooth*. We refer to [15, 34] for details about (strongly) semismooth functions.

3. Construction of exact penalty function. The construction of an exact penalty function for the SOCP is based on those for nonlinear programming [2, 11, 12] and variational inequalities [3]. It consists in incorporating a Lagrange multipliers estimate in an augmented Lagrangian function. Let us recall that a triple $(x, \lambda, \mu) \in \mathbb{R}^n \times \mathbb{R}^m \times \mathbb{R}^p$ satisfies the *Karush–Kuhn–Tucker* (KKT) *conditions* associated to (SOCP) if

$$\begin{aligned} \nabla_x L(x, \lambda, \mu) &= 0, \\ h(x) &= 0, \\ \langle g_i(x), \lambda_i\rangle &= 0, \qquad i = 1, \ldots, r, \\ g_i(x), \lambda_i &\in \mathcal{K}_i, \qquad i = 1, \ldots, r, \end{aligned} \tag{3.1}$$

where $\lambda \doteq (\lambda_1, \ldots, \lambda_r)$ and $g \doteq (g_1, \ldots, g_r)$, with $\lambda_i \in \mathbb{R}^{m_i}$ and $g_i : \mathbb{R}^n \to \mathbb{R}^{m_i}$ for all $i = 1, \ldots, r$,

$$L(x, \lambda, \mu) \doteq f(x) - \langle g(x), \lambda\rangle + \langle h(x), \mu\rangle$$

is the *Lagrangian* function associated to (SOCP), and $\nabla_x L(x, \lambda, \mu)$ denotes its gradient with respect to x. Note that, from Lemma 2.2, the third condition in (3.1), which is called the *complementarity condition*, can be replaced by $g_i(x) \circ \lambda_i = 0$, $i = 1, \ldots, r$.

For a given $x \in \mathbb{R}^n$, we consider the following unconstrained problem in order to estimate the value of the multipliers:

$$\min_{\lambda,\mu} \|\nabla_x L(x,\lambda,\mu)\|^2 + \zeta_1^2 \sum_{i=1}^{r} \big\|\mathrm{Arw}(g_i(x))\lambda_i\big\|^2 + \zeta_2^2\alpha(x)\big(\|\lambda\|^2+\|\mu\|^2\big), \tag{3.2}$$

where $\zeta_1, \zeta_2 > 0$ and

$$\alpha(x) \doteq \frac{1}{2}\left(\|h(x)\|^2 + \sum_{i=1}^{r}\|P_{\mathcal{K}_i}(-g_i(x))\|^2\right). \tag{3.3}$$

The idea underlying problem (3.2) is to force the KKT conditions (3.1) to hold, in particular, $\nabla_x L(x,\lambda,\mu)=0$ and the complementarity condition $g_i(x)\circ\lambda_i = 0$, $i=1,\ldots,r$. Moreover, note that α is continuously differentiable and, for each x, $\alpha(x)$ is a nonnegative number that represents a measure for the feasibility of x. In fact, $\alpha(x)=0$ if and only if x is feasible for (SOCP).

We point out that problem (3.2) is an extension of the multipliers estimate for nonlinear programming introduced in [19]. The employment of a feasibility measure–type function was also given in [29] in order to weaken the conditions for a unique solution of (3.2). In practice, the value of ζ_2 should be small. Otherwise, unfavorable dependence among the multipliers could appear [2, section 6]. Furthermore, note that problem (3.2) can be rewritten as

$$\min_{\lambda,\mu}\left\|\begin{bmatrix} -Jg(x)^\top & Jh(x)^\top \\ \zeta_1\mathrm{Arw}(g(x)) & 0 \\ \zeta_2\alpha(x)^{1/2}I_m & 0 \\ 0 & \zeta_2\alpha(x)^{1/2}I_p\end{bmatrix}\begin{bmatrix}\lambda\\ \mu\end{bmatrix} - \begin{bmatrix}-\nabla f(x)\\ 0\\ 0\\ 0\end{bmatrix}\right\|^2, \tag{3.4}$$

where $\mathrm{Arw}(g(x)) \doteq \mathrm{diag}(\mathrm{Arw}(g_i(x)))$ is a block diagonal matrix with $\mathrm{Arw}(g_i(x))$ as its entries. Thus, (3.2) is actually a linear least squares problem. Let us present now some properties associated to this estimate, under the following assumption [7].

Assumption 3.1. Every $x\in\mathbb{R}^n$ feasible for (SOCP) is *nondegenerate*, that is,

$$\begin{bmatrix} Jh(x)\\ Jg(x)\end{bmatrix}\mathbb{R}^n + \begin{bmatrix}\{0\}\\ \mathrm{lin}T_{\mathcal{K}}(g(x))\end{bmatrix} = \begin{bmatrix}\mathbb{R}^p\\ \mathbb{R}^m\end{bmatrix}, \tag{3.5}$$

where $T_{\mathcal{K}}(g(x))$ is the tangent cone of $\mathcal{K}$ at $g(x)$ and lin stands for the linearity space.

The condition (3.5) extends the well-known *linear independence constraint qualification* (LICQ) in nonlinear programming. We observe that, in the literature of exact penalty functions [12, 29], the assumption that LICQ holds in the feasible set (or in the whole space) is usually required. Therefore, Assumption 3.1 should also be reasonable in the context of SOCP. The next lemma shows that the nondegeneracy condition (3.5) can be rewritten as a "linear independence-type" condition. We define the following sets of indices, for each $x\in\mathbb{R}^n$ feasible for (SOCP):

$$\begin{aligned} I_I(x) &\doteq \big\{i\in\{1,\ldots,r\}\colon g_i(x)\in\mathrm{int}(\mathcal{K}_i)\big\},\\ I_B(x) &\doteq \big\{i\in\{1,\ldots,r\}\colon g_i(x)\in\mathrm{bd}^+(\mathcal{K}_i)\big\},\\ I_0(x) &\doteq \big\{i\in\{1,\ldots,r\}\colon g_i(x)=0\big\}.\end{aligned} \tag{3.6}$$

To clarify this notation, we point out that the subscripts I, B, and 0 mean, respectively, the interior, the boundary excluding the origin, and the origin itself. Besides, observe that x is feasible if and only if $I_I(x)\cup I_B(x)\cup I_0(x) = \{1,\ldots,r\}$.

LEMMA 3.2. *Let $x \in \mathbb{R}^n$ be feasible for* (SOCP). *Then, x is nondegenerate if and only if $\nabla h_j(x), j = 1, \ldots, p$, $\nabla g_{i,j}(x), i \in I_0(x), j = 1, \ldots, m_i$, and*

$$Jg_i(x)^\top \begin{bmatrix} 1 & 0^\top \\ 0 & -I_{m_i-1} \end{bmatrix} g_i(x), \quad i \in I_B(x),$$

are linearly independent, where $\nabla g_{i,j}(x)$ is the gradient of the jth entry of g_i at x.

Proof. See [38, Lemma 3.1] and [8, section 4.6]. ∎

The next proposition shows that, under the nondegeneracy condition (3.5), problem (3.2) has a unique solution. Also, it recovers the Lagrange multipliers associated to a KKT point, which is crucial for the definition of multipliers estimate.

PROPOSITION 3.3. *Suppose that Assumption* 3.1 *holds, and define the matrix $N(x) \in \mathbb{R}^{(m+p)\times(m+p)}$ by*

$$N(x) \doteq \begin{bmatrix} Jg(x)Jg(x)^\top + \zeta_1^2 \mathrm{Arw}(g(x))^2 & -Jg(x)Jh(x)^\top \\ -Jh(x)Jg(x)^\top & Jh(x)Jh(x)^\top \end{bmatrix} + \zeta_2^2 \alpha(x) I_{m+p}. \tag{3.7}$$

Then, the following statements are true:

(a) *$N(\cdot)$ is continuously differentiable and $N(x)$ is positive definite for all $x \in \mathbb{R}^n$.*

(b) *For a given $x \in \mathbb{R}^n$, the solution of* (3.2) *(or, equivalently,* (3.4)*) is unique and is given by*

$$\begin{bmatrix} \lambda(x) \\ \mu(x) \end{bmatrix} = N(x)^{-1} \begin{bmatrix} Jg(x) \\ -Jh(x) \end{bmatrix} \nabla f(x). \tag{3.8}$$

(c) *If $(x^*, \lambda^*, \mu^*) \in \mathbb{R}^n \times \mathbb{R}^m \times \mathbb{R}^p$ satisfies the KKT conditions* (3.1), *then $\lambda(x^*) = \lambda^*$ and $\mu(x^*) = \mu^*$.*

(d) *The mappings $\lambda(\cdot)$ and $\mu(\cdot)$ are continuously differentiable. Moreover, the Jacobian matrices of $\lambda(\cdot)$ and $\mu(\cdot)$ at x are given by*

$$\begin{bmatrix} J\lambda(x) \\ J\mu(x) \end{bmatrix} = N(x)^{-1} \begin{bmatrix} R_1(x) \\ -R_2(x) \end{bmatrix},$$

with

$$\begin{aligned} R_1(x) \doteq{}& \left[\sum_{j=1}^{m_i} e_j^{m_i} \nabla_x L(x, \lambda(x), \mu(x))^\top \nabla^2 g_{i,j}(x) \right]_{i=1}^r + Jg(x) \nabla_{xx}^2 L(x, \lambda(x), \mu(x)) \\ &- \zeta_1^2 \mathrm{diag}\Big[\mathrm{Arw}(g_i(x)) \mathrm{Arw}(\lambda_i(x)) + \mathrm{Arw}\big(g_i(x) \circ \lambda_i(x)\big) \Big] Jg(x) \\ &- \zeta_2^2 \lambda(x) \nabla \alpha(x)^\top, \\ R_2(x) \doteq{}& \sum_{j=1}^{p} e_j^p \nabla_x L(x, \lambda(x), \mu(x))^\top \nabla^2 h_j(x) + Jh(x) \nabla_{xx}^2 L(x, \lambda(x), \mu(x)) \\ &+ \zeta_2^2 \mu(x) \nabla \alpha(x)^\top, \end{aligned}$$

where $e_j^{m_i}$ and e_j^p are the jth elements of the canonical bases of $\mathbb{R}^{m_i}$ and $\mathbb{R}^p$, respectively, $\nabla^2 g_{i,j}(x)$ and $\nabla^2 h_j(x)$ represent the Hessian of the jth component of g_i and the Hessian of h_j at x, respectively, and

$$\begin{aligned} \nabla_x L(x, \lambda(x), \mu(x)) &\doteq \nabla_x L(x, \lambda, \mu)\big|_{\lambda=\lambda(x), \mu=\mu(x)}, \\ \nabla_{xx}^2 L(x, \lambda(x), \mu(x)) &\doteq \nabla_{xx}^2 L(x, \lambda, \mu)\big|_{\lambda=\lambda(x), \mu=\mu(x)}. \end{aligned}$$

Proof. (a) Observe that $N(\cdot)$ is continuously differentiable because all the functions involved in the formula (3.7) are continuously differentiable. Now, let $A(x) \in \mathbb{R}^{(n+2m+p)\times(m+p)}$ be the matrix associated to the linear least squares problem (3.4), that is,

$$A(x) \doteq \begin{bmatrix} -Jg(x)^\top & Jh(x)^\top \\ \zeta_1 \operatorname{Arw}(g(x)) & 0 \\ \zeta_2 \alpha(x)^{1/2} I_m & 0 \\ 0 & \zeta_2 \alpha(x)^{1/2} I_p \end{bmatrix}. \tag{3.9}$$

If x is infeasible, then $A(x)$ has full column rank because $\alpha(x) \neq 0$. Now, suppose that x is feasible, i.e., $\alpha(x) = 0$. Using the notation given in (3.6), we can write, without loss of generality, $Jg(x)^\top = \left[Jg_{I_I}(x)^\top, Jg_{I_B}(x)^\top, Jg_{I_0}(x)^\top\right]$, where $Jg_{I_I}(x)$ corresponds to the submatrix of $Jg(x)$ consisting of $Jg_i(x)$ with $i \in I_I(x)$. In a similar way, we define $Jg_{I_B}(x)$, $Jg_{I_0}(x)$, $\operatorname{Arw}(g_{I_I}(x))$, $\operatorname{Arw}(g_{I_B}(x))$, and $\operatorname{Arw}(g_{I_0}(x))$. Thus, we have

$$A(x) = \begin{bmatrix} -Jg_{I_I}(x)^\top & -Jg_{I_B}(x)^\top & -Jg_{I_0}(x)^\top & Jh(x)^\top \\ \zeta_1 \operatorname{Arw}(g_{I_I}(x)) & 0 & 0 & 0 \\ 0 & \zeta_1 \operatorname{Arw}(g_{I_B}(x)) & 0 & 0 \\ 0 & 0 & \zeta_1 \operatorname{Arw}(g_{I_0}(x)) & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}.$$

Note that $\operatorname{Arw}(g_{I_0}(x)) = 0$ and that $\operatorname{Arw}(g_{I_I}(x))$ is nonsingular because $\operatorname{Arw}(g_i(x))$ is positive definite for all $i \in I_I(x)$. To conclude that $A(x)$ has full column rank, we have only to show that the submatrix

$$\begin{bmatrix} -Jg_{I_B}(x)^\top & -Jg_{I_0}(x)^\top & Jh(x)^\top \\ \zeta_1 \operatorname{Arw}(g_{I_B}(x)) & 0 & 0 \end{bmatrix} \tag{3.10}$$

has full column rank. For any $\ell \in I_B(x)$, notice that[1]

$$g_\ell(x) = \left[\operatorname{Arw}(g_\ell(x))\right]_1 = \sum_{j=2}^{m_\ell} \frac{g_{\ell,j}(x)}{g_{\ell 0}(x)} \left[\operatorname{Arw}(g_\ell(x))\right]_j, \tag{3.11}$$

where $[\operatorname{Arw}(g_\ell(x))]_j$ is the jth column of $\operatorname{Arw}(g_\ell(x))$ and $g_{\ell,j}(x)$ is the jth entry of $g_\ell(x)$. It means that the first column of $\operatorname{Arw}(g_\ell(x))$ is a linear combination of the other columns (which are linearly independent) using coefficients $g_{\ell,j}(x)/g_{\ell 0}(x)$, $j = 2, \ldots, m_\ell$. Now, let us assume, for the purpose of contradiction, that (3.10) does not have full column rank. Then, there exist $\psi_{\ell,j} \in \mathbb{R}$, $\ell \in I_B(x)$, $j = 1, \ldots, m_\ell$, $\theta_{i,j} \in \mathbb{R}$, $i \in I_0(x)$, $j = 1, \ldots, m_i$, and $\upsilon_j \in \mathbb{R}$, $j = 1, \ldots, p$, not all zero, such that

$$-\sum_{\ell \in I_B(x)} \left(\psi_{\ell,1} \nabla g_{\ell 0}(x) + \sum_{j=2}^{m_\ell} \psi_{\ell,j} \nabla g_{\ell,j}(x) \right) - \sum_{i \in I_0(x)} \sum_{j=1}^{m_i} \theta_{i,j} \nabla g_{i,j}(x) + \sum_{j=1}^{p} \upsilon_j \nabla h_j(x) = 0 \tag{3.12}$$

and $\zeta_1 \psi_{\ell,1} g_\ell(x) + \zeta_1 \sum_{j=2}^{m_\ell} \psi_{\ell,j} \left[\operatorname{Arw}(g_\ell(x))\right]_j = 0$ for all $\ell \in I_B(x)$. Recalling equality (3.11), we can write $\psi_{\ell,j} = -\left(\psi_{\ell,1} g_{\ell,j}(x)\right)/g_{\ell 0}(x)$ for all $\ell \in I_B(x)$, $j = 2, \ldots, m_\ell$. Using these coefficients in (3.12) yields

$$-\sum_{\ell \in I_B(x)} \frac{\psi_{\ell,1}}{g_{\ell 0}(x)} Jg_\ell(x)^\top \begin{bmatrix} 1 & 0^\top \\ 0 & -I_{m_\ell - 1} \end{bmatrix} g_\ell(x) - \sum_{i \in I_0(x)} \sum_{j=1}^{m_i} \theta_{i,j} \nabla g_{i,j}(x) + \sum_{j=1}^{p} \upsilon_j \nabla h_j(x) = 0.$$

[1]Note that $g_\ell(x) = \left(g_{\ell 0}(x), \overline{g_\ell(x)}\right) = \left(g_{\ell,1}(x), \ldots, g_{\ell,m_\ell}(x)\right)$, i.e., $g_{\ell 0}(x) = g_{\ell,1}(x)$.

From Lemma 3.2, this yields a contradiction, and hence the matrix (3.10) has full column rank. We can then conclude that $A(x)$ has full column rank. The result follows because, for any $x \in \mathbb{R}^n$, $N(x) = A(x)^\top A(x)$.

(b) Differentiating the objective function of (3.4) and setting the result to zero give

$$A(x)^\top A(x) \begin{bmatrix} \lambda(x) \\ \mu(x) \end{bmatrix} = A(x)^\top \begin{bmatrix} -\nabla f(x) \\ 0 \\ 0 \\ 0 \end{bmatrix},$$

where $A(x)$ is defined as in (3.9). The result then holds because $N(x) = A(x)^\top A(x)$ is nonsingular from (a).

(c) From the complementarity condition, $\operatorname{Arw}(g_i(x^*))\lambda_i^* = 0$ for all $i = 1, \ldots, r$. Also, since $\nabla_x L(x^*, \lambda^*, \mu^*) = 0$ and $\alpha(x^*) = 0$, the objective function of problem (3.2) at (λ^*, μ^*) is equal to zero. Since this objective function is always greater than or equal to zero, (λ^*, μ^*) is a solution of the problem, and the result follows.

(d) The mappings $\lambda(\cdot)$ and $\mu(\cdot)$ are continuously differentiable from (a) and equality (3.8). Thus, once again from (3.8), we obtain

$$0 = -Jg(x)\nabla_x L(x, \lambda(x), \mu(x)) + \zeta_1^2 \operatorname{Arw}(g(x))^2 \lambda(x) + \zeta_2^2 \alpha(x)\lambda(x), \tag{3.13}$$

$$0 = Jh(x)\nabla_x L(x, \lambda(x), \mu(x)) + \zeta_2^2 \alpha(x)\mu(x). \tag{3.14}$$

First, observe that $\operatorname{Arw}(g(x))^2\lambda(x) = [\operatorname{Arw}(g_i(x))^2\lambda_i(x)]_{i=1}^r$, that is, it is a block vector with $\operatorname{Arw}(g_i(x))^2\lambda_i(x)$ as its entries, and $\operatorname{Arw}(g_i(x))^2\lambda_i(x) = g_i(x) \circ (g_i(x) \circ \lambda_i(x))$ for each $i = 1, \ldots, r$. Differentiating this expression using Lemma 2.1, we get

$$\begin{aligned} &J(g_i \circ (g_i \circ \lambda_i))(x) \\ &\quad = \operatorname{Arw}(g_i(x))^2 J\lambda_i(x) + \operatorname{Arw}(g_i(x))\operatorname{Arw}(\lambda_i(x))Jg_i(x) + \operatorname{Arw}\big(g_i(x) \circ \lambda_i(x)\big)Jg_i(x) \end{aligned}$$

for all i. Moreover, since

$$\begin{aligned} Jg(x)\nabla_x L(x, \lambda(x), \mu(x)) &= \big[Jg_i(x)\nabla_x L(x, \lambda(x), \mu(x))\big]_{i=1}^r \\ &= \left[\sum_{j=1}^{m_i} e_j^{m_i} \nabla g_{i,j}(x)^\top \nabla_x L(x, \lambda(x), \mu(x))\right]_{i=1}^r, \end{aligned}$$

differentiating the whole expression (3.13) yields

$$-R_1(x) + \Big[Jg(x)Jg(x)^\top + \zeta_2^2\alpha(x)I_m + \zeta_1^2 \operatorname{Arw}(g(x))^2\Big] J\lambda(x) - Jg(x)Jh(x)^\top J\mu(x) = 0.$$

Analogously, from (3.14), we obtain

$$R_2(x) - Jh(x)Jg(x)^\top J\lambda(x) + \Big[Jh(x)Jh(x)^\top + \zeta_2^2\alpha(x)I_p\Big] J\mu(x) = 0.$$

These two equations give the result. $\square$

Now let us show precisely the idea for building a continuously differentiable exact penalty function for SOCP. The main idea was given by Di Pillo and Grippo [11, 12] for nonlinear programming and it consists in incorporating a multipliers estimate in an augmented Lagrangian function. In the case of the nonlinear SOCP, the *augmented Lagrangian* [26, 37] is given by

$$L_c(x,\lambda,\mu) = f(x) + \langle h(x),\mu\rangle + \frac{c}{2}\|h(x)\|^2 + \frac{1}{2c}\sum_{i=1}^{r}\Big[\|P_{\mathcal{K}_i}(\lambda_i - cg_i(x))\|^2 - \|\lambda_i\|^2\Big],$$

where $c > 0$ is a penalty parameter. It is not difficult to see that this function is actually an extension of the classical augmented Lagrangian for nonlinear programming [22, 33, 36]. We define the incorporation of the estimates $\lambda(x), \mu(x)$, given in (3.8), in the augmented Lagrangian as

$$\begin{aligned} w_c(x) \doteq L_c(x,\lambda(x),\mu(x)) &= f(x) + \langle h(x),\mu(x)\rangle + \frac{c}{2}\|h(x)\|^2 \\ &\quad + \frac{1}{2c}\sum_{i=1}^{r}\Big[\|P_{\mathcal{K}_i}(\lambda_i(x) - cg_i(x))\|^2 - \|\lambda_i(x)\|^2\Big]. \end{aligned} \tag{3.15}$$

Note that the function $w_c\colon \mathbb{R}^n \to \mathbb{R}$ is continuously differentiable because all functions involved in its formula are continuously differentiable, in particular $\lambda(\cdot)$ and $\mu(\cdot)$ from Proposition 3.3(d). Besides, since $P_{\mathcal{K}_i}(\tilde{c}y) = \tilde{c}P_{\mathcal{K}_i}(y)$ for all $\tilde{c} > 0$ and $y \in \mathbb{R}^{m_i}$, we can write its gradient as follows:

$$\begin{aligned} \nabla w_c(x) &= \nabla f(x) + Jh(x)^\top\mu(x) + J\mu(x)^\top h(x) + cJh(x)^\top h(x) \\ &\quad + \sum_{i=1}^{r}\left(\frac{1}{c}J\lambda_i(x) - Jg_i(x)\right)^\top P_{\mathcal{K}_i}\big(\lambda_i(x) - cg_i(x)\big) - \frac{1}{c}J\lambda(x)^\top\lambda(x) \\ &= \nabla_x L(x,\lambda(x),\mu(x)) + \big(J\mu(x) + cJh(x)\big)^\top h(x) + \big(J\lambda(x) - cJg(x)\big)^\top y_c(x), \end{aligned} \tag{3.16}$$

where

$$y_c(x) \doteq \left[P_{\mathcal{K}_i}\left(\frac{\lambda_i(x)}{c} - g_i(x)\right)\right]_{i=1}^{r} - \frac{\lambda(x)}{c}. \tag{3.17}$$

Recall that $g_i\colon \mathbb{R}^n \to \mathbb{R}^{m_i}$, $i = 1,\ldots,r$, and so $Jg(x)^\top = \big[Jg_1(x)^\top,\ldots,Jg_r(x)^\top\big]$. The function $y_c\colon \mathbb{R}^n \to \mathbb{R}^m$ has, in turn, the following property.

PROPOSITION 3.4. *For each $x \in \mathbb{R}^n$, $y_c(x) = 0$ if and only if $\lambda_i(x), g_i(x) \in \mathcal{K}_i$, and $\lambda_i(x) \circ g_i(x) = 0$ for $i = 1,\ldots,r$.*

Proof. It follows directly from Lemma 2.2 and the fact that $c > 0$. ☐

In section 4, we will actually prove that, under some reasonable assumptions, w_c is an exact penalty function for (SOCP). Moreover, since $P_{\mathcal{K}_i}$ is (strongly) semismooth (see [10, Proposition 7] or [21, Proposition 4.5]), ∇w_c is also semismooth. Thus, w_c is an SC^1 function. This fact allows the use of a generalized Newton method [16, 15, 31].

4. Exactness results. In this section, we prove that the function w_c defined in (3.15) is an *exact penalty* function for (SOCP), which means that a solution of the unconstrained problem

$$\min_{x\in\mathbb{R}^n} w_c(x) \tag{4.1}$$

recovers a solution of (SOCP) when c is greater than a threshold value. Following the structure presented in [3, 2, 12], we show first that KKT conditions of (SOCP) are equivalent, under some reasonable conditions, to the system of equations $\nabla w_c(x) = 0$.

In the remainder of the paper, we suppose that Assumption 3.1 holds. The first half of the equivalence is given below.

PROPOSITION 4.1. *If* (x^*, λ^*, μ^*) *is a KKT triple for* (SOCP), *then* $\nabla w_c(x^*) = 0$ *for all* $c > 0$.

Proof. From Proposition 3.3(c), we have $\lambda^* = \lambda(x^*)$. Also, from Proposition 3.4, we obtain $y_c(x^*) = 0$. This result, along with the KKT conditions (3.1) and the formula of ∇w_c in (3.16), gives

$$\nabla_x L(x^*, \lambda^*, \mu^*) + \big(J\mu(x^*) + cJh(x^*)\big)^\top h(x^*) + \big(J\lambda(x^*) - cJg(x^*)\big)^\top y_c(x^*) = 0$$

for all $c > 0$, and the result follows. ∎

As to the converse, we will see that it may hold when the penalty parameter c is large enough. If it does not hold, then, instead of a KKT point, we find an infeasible point that is stationary for the function α, defined in (3.3). We recall that α is actually a feasibility measure for (SOCP).

PROPOSITION 4.2. *Let* $\{x^k\} \subset \mathbb{R}^n$, *and let* $\{c_k\} \subset \mathbb{R}_{++}$ *be sequences such that* $x^k \to \hat{x}$, $c_k \to \infty$, *and* $\nabla w_{c_k}(x^k) = 0$ *for all* k. *Then,* $\hat{x}$ *is a stationary point of the feasibility measure* α, *defined in* (3.3). *In other words,*

$$\nabla \alpha(\hat{x}) = Jh(\hat{x})^\top h(\hat{x}) - Jg(\hat{x})^\top \big[P_{\mathcal{K}_i}(-g_i(\hat{x}))\big]_{i=1}^r = 0.$$

Proof. Using (3.16) and dividing the equation $\nabla w_{c_k}(x^k) = 0$ by c_k, we have

$$\begin{aligned} &\frac{\nabla_x L(x^k, \lambda(x^k), \mu(x^k))}{c_k} + \left(\frac{J\mu(x^k)}{c_k} + Jh(x^k)\right)^\top h(x^k) \\ &+ \left(\frac{J\lambda(x^k)}{c_k} - Jg(x^k)\right)^\top y_{c_k}(x^k) \;=\; 0. \end{aligned}$$

Since all the functions involved in the above equality are continuous, we can take the limit $k \to \infty$, and, taking into account the definition (3.17) of $y_c(x)$, we obtain $Jh(\hat{x})^\top h(\hat{x}) - Jg(\hat{x})^\top \big[P_{\mathcal{K}_i}(-g_i(\hat{x}))\big]_{i=1}^r = 0$. Thus, the proof is complete. ∎

Before presenting the next result, we observe that the following inequality holds:

$$\|u - v\|^2 \geq \frac{\|u\|^2}{2} - \|v\|^2 \quad \text{for all } u, v. \tag{4.2}$$

In fact, for all u, v,

$$\|u - v\|^2 - \frac{\|u\|^2}{2} + \|v\|^2 = \frac{\|u\|^2}{2} + 2\|v\|^2 - 2\langle u, v\rangle = \frac{\|u - 2v\|^2}{2} \geq 0.$$

PROPOSITION 4.3. *Let* $\hat{x} \in \mathbb{R}^n$ *be a feasible point for* (SOCP). *Then, there exist* $\hat{c}, \hat{\delta} > 0$ *such that if* $\|x - \hat{x}\| \leq \hat{\delta}$, $c \geq \hat{c}$, *and* $\nabla w_c(x) = 0$, *then* $(x, \lambda(x), \mu(x))$ *is a KKT triple associated to* (SOCP).

Proof. For any $c > 0$, let us define

$$\tilde{y}_c(x) \doteq \left[P_{\mathcal{K}_i}\left(-\frac{\lambda_i(x)}{c} + g_i(x)\right)\right]_{i=1}^r.$$

Recalling the definition (3.17) of $y_c(x)$ and Lemma 2.3(a), we have

$$\tilde{y}_c(x) = y_c(x) + g(x). \tag{4.3}$$

For all $i = 1, \ldots, r$, from Lemma 2.3(b), we have $\langle P_{\mathcal{K}_i}(\lambda_i(x)/c - g_i(x)), [\tilde{y}_c(x)]_i \rangle = 0$. Since both $P_{\mathcal{K}_i}(\lambda_i(x)/c - g_i(x))$ and $[\tilde{y}_c(x)]_i$ belong to $\mathcal{K}_i$, it follows from Lemma 2.2 that $[\tilde{y}_c(x)]_i \circ P_{\mathcal{K}_i}(\lambda_i(x)/c - g_i(x)) = 0$ for all $i = 1, \ldots, r$. That is,

$$\operatorname{Arw}(\tilde{y}_c(x)) \left[P_{\mathcal{K}_i} \left(\frac{\lambda_i(x)}{c} - g_i(x) \right) \right]_{i=1}^{r} = 0,$$

where $\operatorname{Arw}(\tilde{y}_c(x)) \doteq \operatorname{diag}(\operatorname{Arw}([\tilde{y}_c(x)]_i))$ is a block diagonal matrix with $\operatorname{Arw}([\tilde{y}_c(x)]_i)$ as its entries. This, along with the definition of $y_c(x)$ in (3.17), implies that

$$\operatorname{Arw}(\tilde{y}_c(x)) y_c(x) = -\operatorname{Arw}(\tilde{y}_c(x)) \frac{\lambda(x)}{c}. \tag{4.4}$$

Moreover, observe that

$$\begin{aligned} -\frac{1}{c}\operatorname{Arw}(g(x))^2 \lambda(x) &= \frac{1}{c}\operatorname{Arw}(g(x)) \Big(\operatorname{Arw}(\tilde{y}_c(x)) - \operatorname{Arw}(g(x)) \Big) \lambda(x) \\ &\quad - \frac{1}{c}\operatorname{Arw}(g(x)) \operatorname{Arw}(\tilde{y}_c(x)) \lambda(x). \end{aligned}$$

From (4.3) and (4.4), we obtain

$$\begin{aligned} -\frac{1}{c}\operatorname{Arw}(g(x))^2 \lambda(x) &= \frac{1}{c}\operatorname{Arw}(g(x)) \operatorname{Arw}(y_c(x)) \lambda(x) + \operatorname{Arw}(g(x)) \operatorname{Arw}(\tilde{y}_c(x)) y_c(x) \\ &= \left(\frac{1}{c}\operatorname{Arw}(g(x)) \operatorname{Arw}(\lambda(x)) + \operatorname{Arw}(g(x)) \operatorname{Arw}(\tilde{y}_c(x)) \right) y_c(x), \end{aligned} \tag{4.5}$$

where the last equality holds from the commutativity of the Jordan product.

Now, from the formula (3.16) of $\nabla w_c(x)$ and the equality (3.13), we have

$$\begin{aligned} -\frac{1}{c} Jg(x) \nabla w_c(x) &= -\frac{1}{c} Jg(x) \nabla_x L(x, \lambda(x), \mu(x)) - Jg(x) \left(\frac{J\mu(x)}{c} + Jh(x) \right)^{\top} h(x) \\ &\quad + Jg(x) \left(Jg(x) - \frac{J\lambda(x)}{c} \right)^{\top} y_c(x) \\ &= -\frac{1}{c} \zeta_1^2 \operatorname{Arw}(g(x))^2 \lambda(x) - Jg(x) \left(\frac{J\mu(x)}{c} + Jh(x) \right)^{\top} h(x) \\ &\quad + Jg(x) \left(Jg(x) - \frac{J\lambda(x)}{c} \right)^{\top} y_c(x) - \frac{1}{c} \zeta_2^2 \alpha(x) \lambda(x). \end{aligned}$$

Using equality (4.5), we have

$$-\frac{1}{c} Jg(x) \nabla w_c(x) = \tilde{N}_c(x) y_c(x) - Jg(x) \left(\frac{J\mu(x)}{c} + Jh(x) \right)^{\top} h(x) - \frac{1}{c} \zeta_2^2 \alpha(x) \lambda(x), \tag{4.6}$$

where

$$\begin{aligned} \tilde{N}_c(x) \doteq Jg(x) \left(Jg(x) - \frac{J\lambda(x)}{c} \right)^{\top} + \frac{\zeta_1^2}{c} \operatorname{Arw}(g(x)) \operatorname{Arw}(\lambda(x)) \\ + \zeta_1^2 \operatorname{Arw}(g(x)) \operatorname{Arw}(\tilde{y}_c(x)). \end{aligned}$$

In a similar way, from equality (3.14), we obtain

$$\begin{aligned}\frac{1}{c}Jh(x)\nabla w_c(x) &= Jh(x)\left(\frac{J\lambda(x)}{c} - Jg(x)\right)^\top y_c(x)\\ &\quad + Jh(x)\left(\frac{J\mu(x)}{c} + Jh(x)\right)^\top h(x) - \frac{1}{c}\zeta_2^2\alpha(x)\mu(x).\end{aligned} \tag{4.7}$$

The above equalities (4.6) and (4.7) can be rewritten as

$$\frac{1}{c}\begin{bmatrix} -Jg(x) \\ Jh(x)\end{bmatrix}\nabla w_c(x) = N_c(x)\begin{bmatrix} y_c(x) \\ h(x)\end{bmatrix} - \frac{1}{c}\zeta_2^2\alpha(x)\begin{bmatrix}\lambda(x)\\ \mu(x)\end{bmatrix}, \tag{4.8}$$

where

$$N_c(x) \doteq \begin{bmatrix} \tilde{N}_c(x) & -Jg(x)\left(J\mu(x)/c + Jh(x)\right)^\top \\ Jh(x)\left(J\lambda(x)/c - Jg(x)\right)^\top & Jh(x)\left(J\mu(x)/c + Jh(x)\right)^\top\end{bmatrix}.$$

Now, denoting $\sigma_{m+p}(N_c(x))$ as the smallest singular value of $N_c(x)$, we have

$$\begin{aligned}\left\|N_c(x)\begin{bmatrix} y_c(x) \\ h(x)\end{bmatrix}\right\|^2 &\geq \sigma_{m+p}(N_c(x))^2\left\|\begin{bmatrix} y_c(x) \\ h(x)\end{bmatrix}\right\|^2\\ &= \sigma_{m+p}(N_c(x))^2\big(\|y_c(x)\|^2 + \|h(x)\|^2\big).\end{aligned} \tag{4.9}$$

Furthermore, from the definition of $y_c(x)$ in (3.17) and Lemma 2.3(c), we obtain

$$\alpha(x) = \frac{1}{2}\left(\|h(x)\|^2 + \sum_{i=1}^{r}\|P_{\mathcal{K}_i}(-g_i(x))\|^2\right) \leq \frac{1}{2}\left(\|y_c(x)\|^2 + \|h(x)\|^2\right). \tag{4.10}$$

Thus, taking the square of the norm in (4.8) and using the inequality (4.2), we have

$$\begin{aligned}&\frac{1}{c^2}\left\|\begin{bmatrix} -Jg(x) \\ Jh(x)\end{bmatrix}\nabla w_c(x)\right\|^2\\ &\quad\geq \frac{1}{2}\left\|N_c(x)\begin{bmatrix} y_c(x) \\ h(x)\end{bmatrix}\right\|^2 - \frac{\zeta_2^4}{c^2}\alpha(x)^2\big(\|\lambda(x)\|^2 + \|\mu(x)\|^2\big)\\ &\quad\geq \left[\frac{1}{2}\sigma_{m+p}(N_c(x))^2 - \frac{\zeta_2^4}{2c^2}\alpha(x)\big(\|\lambda(x)\|^2 + \|\mu(x)\|^2\big)\right]\big(\|y_c(x)\|^2 + \|h(x)\|^2\big),\end{aligned} \tag{4.11}$$

where the second inequality comes from (4.9) and (4.10).

Observe that if $c \to \infty$, then $\tilde{y}_c(\hat{x}) \to [P_{\mathcal{K}_i}(g_i(\hat{x}))]_{i=1}^r = g(\hat{x})$ because $\hat{x}$ is feasible. Then, taking $c \to \infty$ and recalling that $\alpha(\hat{x}) = 0$, we have $N_c(\hat{x}) \to N(\hat{x})$, with $N(\hat{x})$ defined as in (3.7). Since $N(\hat{x})$ is nonsingular from Proposition 3.3(a), there exist $\hat{\delta}, \hat{c}, \hat{\rho} > 0$ such that, for any $x \in \mathbb{R}^n$ with $\|x - \hat{x}\| \leq \hat{\delta}$ and $c \geq \hat{c}$,

$$\frac{1}{2}\sigma_{m+p}(N_c(x))^2 - \frac{\zeta_2^4}{2c^2}\alpha(x)\big(\|\lambda(x)\|^2 + \|\mu(x)\|^2\big) \geq \hat{\rho} > 0. \tag{4.12}$$

Now, consider any $x \in \mathbb{R}^n$ and $c \in \mathbb{R}_{++}$ such that $\|x - \hat{x}\| \leq \hat{\delta}$, $c \geq \hat{c}$, and $\nabla w_c(x) = 0$. We conclude from (4.11) and (4.12) that $y_c(x) = 0$ and $h(x) = 0$. So, by Proposition 3.4, x is feasible and the complementarity condition holds. Plugging these equalities into $\nabla w_c(x)$ gives $\nabla_x L(x, \lambda(x), \mu(x)) = 0$, and the proof is complete. ∎

Now we are ready to state the main result concerning the system of equations $\nabla w_c(x) = 0$ and KKT points of (SOCP).

THEOREM 4.4. *Let $\{x^k\} \subset \mathbb{R}^n$ and $\{c_k\} \subset \mathbb{R}_{++}$ be sequences with $c_k \to \infty$ and $\nabla w_{c_k}(x^k) = 0$ for all k. Suppose that there is a subsequence $\{x^{k_j}\}$ of $\{x^k\}$ such that $x^{k_j} \to \hat{x}$ for some $\hat{x} \in \mathbb{R}^n$. Then, either there exists $\hat{k}$ such that $(x^{k_j}, \lambda(x^{k_j}), \mu(x^{k_j}))$ is a KKT triple associated to* (SOCP) *for all $k_j > \hat{k}$, or $\hat{x}$ is a stationary point of α that is infeasible for* (SOCP).

Proof. Observe that Proposition 4.2 guarantees that $\hat{x}$ is a stationary point of α. Considering the case that $\hat{x}$ is feasible, we can use Proposition 4.3 to conclude that there exists $\hat{k}$ such that $(x^{k_j}, \lambda(x^{k_j}), \mu(x^{k_j}))$ is a KKT triple for all $k_j > \hat{k}$. ◻

Let us consider now two additional results concerning the values of the objective function f and the function w_c at feasible points and KKT points.

LEMMA 4.5. *If (x, λ, μ) is a KKT triple for* (SOCP)*, then $w_c(x) = f(x)$ for all $c > 0$.*

Proof. From Proposition 3.3(c), we have $\lambda_i(x) = \lambda_i$ for all i. Then, the result follows because $h(x) = 0$ and, from Lemma 2.2, $\lambda_i = P_{\mathcal{K}_i}(\lambda_i - cg_i(x))$ for all i. ◻

LEMMA 4.6. *If $x \in \mathbb{R}^n$ is feasible for* (SOCP)*, then $w_c(x) \le f(x)$ for all $c > 0$.*

Proof. Since $h(x) = 0$, we have only to prove that $\|P_{\mathcal{K}_i}(\lambda_i(x) - cg_i(x))\| \le \|\lambda_i(x)\|$ for $i = 1, \ldots, r$. Recalling that $P_{\mathcal{K}_i}$ is nonexpansive, we obtain

$$\|P_{\mathcal{K}_i}(\lambda_i(x) - cg_i(x)) - P_{\mathcal{K}_i}(-cg_i(x))\| \le \|(\lambda_i(x) - cg_i(x)) + cg_i(x)\|.$$

Since $g_i(x) \in \mathcal{K}_i$ for all i and $c > 0$, we have $P_{\mathcal{K}_i}(-cg_i(x)) = 0$, and the proof is complete. ◻

Let us prove now that w_c is an exact penalty function for (SOCP). The definition considered here is the same as the one given in [2] for nonlinear programming, which in turn is the same as the one studied in [12] without its extra compact set.

DEFINITION 4.7. *Let $\mathcal{G}_f$ ($\mathcal{L}_f$) and $\mathcal{G}_w(c)$ ($\mathcal{L}_w(c)$) be the sets of global (local) minimizers of* (SOCP) *and* (4.1)*, respectively. The function $w_c \colon \mathbb{R}^n \to \mathbb{R}$ is an* exact penalty *function for* (SOCP) *if there exists $\hat{c} > 0$ such that $\mathcal{G}_f = \mathcal{G}_w(c)$ and $\mathcal{L}_w(c) \subseteq \mathcal{L}_f$ for all $c > \hat{c}$.*

First, we proceed with the equivalence of the sets of global minimizers.

PROPOSITION 4.8. *Assume that $\mathcal{G}_f \neq \emptyset$ and let $\{x^k\} \subset \mathbb{R}^n$, and let $\{c_k\} \subset \mathbb{R}_{++}$ be sequences such that $x^k \to \hat{x}$ for some $\hat{x} \in \mathbb{R}^n$, $c_k \to \infty$, and $x^k \in \mathcal{G}_w(c_k)$ for all k. Then, there exists $\hat{k}$ such that $x^k \in \mathcal{G}_f$ for all $k > \hat{k}$.*

Proof. Let $\tilde{x} \in \mathcal{G}_f$. Since $\tilde{x}$ is a KKT point, from Lemma 4.5, $w_c(\tilde{x}) = f(\tilde{x})$ for all $c > 0$. Thus, since $x^k \in \mathcal{G}_w(c_k)$, we obtain $w_{c_k}(x^k) \le w_{c_k}(\tilde{x}) = f(\tilde{x})$ for all k. Taking the supremum limit in this inequality yields

$$\limsup_{k \to \infty} w_{c_k}(x^k) \le f(\tilde{x}). \tag{4.13}$$

Recall that, by (3.15), we have

$$\begin{aligned} w_{c_k}(x^k) = {} & f(x^k) + \langle h(x^k), \mu(x^k) \rangle + \frac{c_k}{2}\|h(x^k)\|^2 \\ & + \frac{c_k}{2} \sum_{i=1}^{r} \left[\left\| P_{\mathcal{K}_i}\left(\frac{\lambda_i(x^k)}{c_k} - g_i(x^k) \right) \right\|^2 - \left\| \frac{\lambda_i(x^k)}{c_k} \right\|^2 \right]. \end{aligned}$$

Then, it follows from (4.13) and the continuity of the involved functions that $h(\hat{x}) = 0$ and $P_{\mathcal{K}_i}(-g_i(\hat{x})) = 0$ for all i, which in turn implies that $g_i(\hat{x}) \in \mathcal{K}_i$ for all i. In other words, $\hat{x}$ is feasible for (SOCP). Moreover, since $c_k > 0$ and the norm is nonnegative, we have $w_{c_k}(x^k) \geq f(x^k) + \langle h(x^k), \mu(x^k) \rangle - \|\lambda(x^k)\|^2/(2c_k)$ for all k. Once again, taking the supremum limit, we obtain $f(\hat{x}) \leq \limsup w_{c_k}(x^k)$. Thus, recalling (4.13), we have $f(\hat{x}) \leq f(\tilde{x})$, and hence we conclude that $\hat{x} \in \mathcal{G}_f$.

Now, take $\hat{c}, \hat{\delta} > 0$ as in Proposition 4.3, which exist because of the feasibility of $\hat{x}$. Let $\hat{k}$ be large enough so that $\|x^k - \hat{x}\| \leq \hat{\delta}$, $c_k \geq \hat{c}$, and $x^k \in \mathcal{G}_w(c_k)$ for all $k > \hat{k}$. Since $\nabla w_{c_k}(x^k) = 0$ and using Proposition 4.3, we have that x^k is a KKT point for all $k > \hat{k}$. Thus, once again from Lemma 4.5, we obtain $f(x^k) = w_{c_k}(x^k) \leq f(\tilde{x})$ for all $k > \hat{k}$, which means that $x^k \in \mathcal{G}_f$ for all $k > \hat{k}$, as desired. ◻

PROPOSITION 4.9. *Assume that $\mathcal{G}_f \neq \emptyset$ and that there exists $\tilde{c} > 0$ such that $\tilde{S} \doteq \bigcup_{c \geq \tilde{c}} \mathcal{G}_w(c)$ is bounded. Then, there exists some $\bar{c} > 0$ such that $\mathcal{G}_w(c) = \mathcal{G}_f$ for all $c \geq \bar{c}$.*

Proof. Let $\{x^k\} \subset \tilde{S}$ and $\{c_k\} \subset \mathbb{R}_{++}$ be sequences such that $c_k \geq \tilde{c}$, $c_k \to \infty$, and $x^k \in \mathcal{G}_w(c_k)$ for all k. Since $\tilde{S}$ is bounded, we can also assume that $\{x^k\}$ converges to some accumulation point. Then, Proposition 4.8 shows that there exists $\hat{k}$ such that $x^k \in \mathcal{G}_f$ for all $k > \hat{k}$, and so $\mathcal{G}_w(c) \subset \mathcal{G}_f$ for all $c \geq c_{\hat{k}}$. Now, let $c \geq c_{\hat{k}}$ and take $\tilde{x} \in \mathcal{G}_w(c) \subset \mathcal{G}_f$. From Lemma 4.5, we have $w_c(\tilde{x}) = f(\tilde{x})$. Choose $\hat{x} \in \mathcal{G}_f$ arbitrarily. Then, once again from Lemma 4.5, we have $w_c(\hat{x}) = f(\hat{x}) = f(\tilde{x}) = w_c(\tilde{x})$. Therefore, $\hat{x} \in \mathcal{G}_w(c)$ for all $c \geq c_{\hat{k}}$. Since $\hat{x} \in \mathcal{G}_f$ is arbitrary, we have $\mathcal{G}_f \subset \mathcal{G}_w(c)$ for all $c \geq c_{\hat{k}}$. Hence, we conclude that w_c is a weakly exact penalty function for (SOCP). ◻

For local minimizers, the inclusion $\mathcal{L}_w(c) \subseteq \mathcal{L}_f$ is not generally guaranteed, because a local minimum of (4.1) is not necessarily feasible for (SOCP). Here, we establish the results as in [2], which admit that we may end up with a stationary point of the feasibility measure α that is infeasible for (SOCP).

THEOREM 4.10. *Let $\{x^k\} \subset \mathbb{R}^n$ and $\{c_k\} \subset \mathbb{R}_{++}$ be sequences such that $c_k \to \infty$ and $x^k \in \mathcal{L}_w(c_k)$ for all k. Suppose that there is a subsequence $\{x^{k_j}\}$ of $\{x^k\}$ such that $x^{k_j} \to \hat{x}$ for some $\hat{x} \in \mathbb{R}^n$. Then, either there exists $\hat{k}$ such that $x^k \in \mathcal{L}_f$ for all $k > \hat{k}$, or $\hat{x}$ is a stationary point of α that is infeasible.*

Proof. In view of Theorem 4.4, there exists $\hat{k}$ such that x^{k_j} is a KKT point for all $k_j > \hat{k}$ or $\hat{x}$ is a stationary point of α that is infeasible. Consider the first assertion and fix $k_j > \hat{k}$. From Lemma 4.5 and the fact that $x^{k_j} \in \mathcal{L}_w(c_{k_j})$, there exists a neighborhood $\mathcal{N}(x^{k_j})$ of x^{k_j} such that

$$f(x^{k_j}) = w_{c_{k_j}}(x^{k_j}) \leq w_{c_{k_j}}(x) \quad \text{for all } x \in \mathcal{N}(x^{k_j}). \tag{4.14}$$

Now take an arbitrary $x \in \mathcal{N}(x^{k_j})$ that is feasible for (SOCP). Then, Lemma 4.6 guarantees that $w_{c_{k_j}}(x) \leq f(x)$, which, together with (4.14), shows that $f(x^{k_j}) \leq f(x)$. Since $x \in \mathcal{N}(x^{k_j})$ is arbitrary, we have $x^{k_j} \in \mathcal{L}_f$. Thus, the proof is complete. ◻

5. Generalized newton method. As we mentioned in the last paragraph of section 3, w_c is an SC^1 function. So, a generalized Newton method with line search strategy [16, 31, 34] can be used to solve the unconstrained problem (4.1). Let $c > 0$ be fixed and sufficiently large. We recall that the method is iterative and generates a sequence $\{x^k\}$ by

$$\begin{aligned} x^{k+1} &\doteq x^k + t_k d^k, \\ d^k &\doteq -V_k^{-1} \nabla w_c(x^k), \quad V_k \in \partial_B \nabla w_c(x^k), \end{aligned} \tag{5.1}$$

where d^k is the search direction and t_k is the step-size determined, for example, by an Armijo-type rule. We note that a convergence theorem of the generalized Newton method [35] shows that if all the elements of the B-subdifferential $\partial_B \nabla w_c(x^*)$ of ∇w_c at a KKT point x^* are positive definite (with large enough c), then the method, with unit step-size $t_k \doteq 1$ converges locally with a superlinear rate. If, in addition, $\nabla^2 f$, $\nabla^2 g_{i,j}$, and $\nabla^2 h_j$ are locally Lipschitz continuous, then ∇w_c is strongly semismooth and the convergence rate is quadratic.

Before establishing the conditions that guarantee the positive definiteness of the elements of $\partial_B \nabla w_c(x^*)$, let us first recall a couple of results and definitions. Recalling the notation $I_I(x)$, $I_0(x)$, and $I_B(x)$ given in (3.6), for a KKT triple (x^*, λ^*, μ^*), we define the following sets of indices:

$$\begin{array}{ll} \left.\begin{array}{lll} I^*_{I0} & \doteq & \{i \in I_I(x^*) : \lambda^*_i = 0\} \\ I^*_{0I} & \doteq & \{i \in I_0(x^*) : \lambda^*_i \in \mathrm{int}(\mathcal{K}_i)\} \\ I^*_{BB} & \doteq & \{i \in I_B(x^*) \colon \lambda^*_i \in \mathrm{bd}^+(\mathcal{K}_i)\} \end{array}\right\} & \begin{array}{l}\text{satisfies} \\ \text{strict complementarity,}\end{array} \\ \left.\begin{array}{lll} I^*_{B0} & \doteq & \{i \in I_B(x^*) \colon \lambda^*_i = 0\} \\ I^*_{0B} & \doteq & \{i \in I_0(x^*) : \lambda^*_i \in \mathrm{bd}^+(\mathcal{K}_i)\} \\ I^*_{00} & \doteq & \{i \in I_0(x^*) : \lambda^*_i = 0\} \end{array}\right\} & \begin{array}{l}\text{does not satisfy} \\ \text{strict complementarity.}\end{array} \end{array} \tag{5.2}$$

Note that the above sets of indices constitute a partition for $\{1, \ldots, r\}$. Also, if the *strict complementarity* condition holds, that is, if $g_i(x^*) + \lambda^*_i \in \mathrm{int}(\mathcal{K}_i)$, $i = 1, \ldots, r$, then $I^*_{B0} = I^*_{0B} = I^*_{00} = \emptyset$. The following result gives us a characterization of the KKT points of (SOCP).

LEMMA 5.1. *Let (x^*, λ^*, μ^*) be a KKT triple for* (SOCP). *Then, for each $i = 1, \ldots, r$, one of the following conditions holds:* (a) $g_i(x^*) = 0$, (b) $\lambda^*_i = 0$, *or* (c) $\lambda^*_i = \kappa_i\big(g_{i0}(x^*), -\overline{g_i(x^*)}\big)$, *where* $\kappa_i \doteq \lambda^*_{i0}/g_{i0}(x^*)$.

Proof. See [1, Lemma 15]. ∎

Now, we consider the formula of the B-subdifferential of the projection mapping onto second-order cones. It will be used further to characterize the elements of the B-subdifferential of ∇w_c.

LEMMA 5.2. *For each $i = 1, \ldots, r$, let $M_i \colon \mathbb{R} \times \mathbb{R}^{m_i - 1} \to \mathbb{R}^{m_i \times m_i}$ be defined as*

$$M_i(\xi, u) \doteq \frac{1}{2} \begin{bmatrix} 1 & u^\top \\ u & (1+\xi) I_{m_i - 1} - \xi u u^\top \end{bmatrix}.$$

Then, for each $i = 1, \ldots, r$, the B-subdifferential $\partial_B P_{\mathcal{K}_i}(z)$ is given as follows:

(a) *If $z_0 < -\|\bar{z}\|$, then $\partial_B P_{\mathcal{K}_i}(z) = \big\{0\big\}$.*

(b) *If $z_0 > \|\bar{z}\|$, then $\partial_B P_{\mathcal{K}_i}(z) = \big\{I_{m_i}\big\}$.*

(c) *If $-\|\bar{z}\| < z_0 < \|\bar{z}\|$, then $\partial_B P_{\mathcal{K}_i}(z) = \left\{M_i\left(\frac{z_0}{\|\bar{z}\|}, \frac{\bar{z}}{\|\bar{z}\|}\right)\right\}$.*

(d) *If $z_0 = \|\bar{z}\| \neq 0$, then $\partial_B P_{\mathcal{K}_i}(z) = \left\{I_{m_i}, M_i\left(1, \frac{\bar{z}}{\|\bar{z}\|}\right)\right\}$.*

(e) *If $z_0 = -\|\bar{z}\| \neq 0$, then $\partial_B P_{\mathcal{K}_i}(z) = \left\{0, M_i\left(-1, \frac{\bar{z}}{\|\bar{z}\|}\right)\right\}$.*

(f) *If $z_0 = 0$ and $\bar{z} = 0$, then $\partial_B P_{\mathcal{K}_i}(z) = \{0, I_{m_i}\} \cup \{M_i(\xi, u) \colon |\xi| \leq 1, \|u\| = 1\}$.*

Proof. See [32, Lemma 14] or [21, Proposition 4.8]. ∎

Notice that $M_i(\xi, u)$, defined in the above lemma, is symmetric and positive semidefinite whenever $|\xi| \leq 1$ and $\|u\| = 1$ [24, Lemma 2.8].

LEMMA 5.3. *Let (x^*, λ^*, μ^*) be a KKT triple for* (SOCP) *and $c > 0$. For each $i = 1, \ldots, r$, the B-subdifferential $\partial_B P_{\mathcal{K}_i}\big(\lambda^*_i - c g_i(x^*)\big)$ is given as follows:*

(a) *If $i \in I^*_{I0}$, then $\partial_B P_{\mathcal{K}_i}\big(\lambda^*_i - c g_i(x^*)\big) = \big\{0\big\}$.*

(b) *If $i \in I^*_{0I}$, then $\partial_B P_{\mathcal{K}_i}\big(\lambda^*_i - c g_i(x^*)\big) = \big\{I_{m_i}\big\}$.*

(c) *If* $i \in I^*_{BB}$, *then* $\partial_B P_{\mathcal{K}_i}\big(\lambda^*_i - cg_i(x^*)\big) = \left\{ M_i\left(\frac{\kappa_i - c}{\kappa_i + c}, \frac{\bar{\lambda}^*_i}{\|\bar{\lambda}^*_i\|}\right)\right\}$, *where* κ_i *is defined as in Lemma* 5.1.

(d) *If* $i \in I^*_{0B}$, *then* $\partial_B P_{\mathcal{K}_i}\big(\lambda^*_i - cg_i(x^*)\big) = \left\{ I_{m_i}, M_i\left(1, \frac{\bar{\lambda}^*_i}{\|\bar{\lambda}^*_i\|}\right)\right\}$.

(e) *If* $i \in I^*_{B0}$, *then* $\partial_B P_{\mathcal{K}_i}\big(\lambda^*_i - cg_i(x^*)\big) = \left\{ 0, M_i\left(-1, -\frac{\overline{g_i(x^*)}}{\|\overline{g_i(x^*)}\|}\right)\right\}$.

(f) *If* $i \in I^*_{00}$, *then* $\partial_B P_{\mathcal{K}_i}\big(\lambda^*_i - cg_i(x^*)\big) = \{0, I_{m_i}\} \cup \{M_i(\xi, u) \colon |\xi| \le 1, \|u\| = 1\}$.

Proof. The results follow from Lemma 5.2 and Lemma 5.1 (for item (c)). ∎

The next proposition and corollary give a characterization of the B-subdifferential of ∇w_c at an arbitrary point and a KKT point, respectively.

PROPOSITION 5.4. *Let* $x \in \mathbb{R}^n$ *and* $c > 0$. *Then, any* $V \in \partial_B \nabla w_c(x)$ *is expressed as* $V \doteq \tilde{V} + \phi_c(x)$, *where* $\tilde{V}$ *is given by*

$$\begin{aligned} \tilde{V} \doteq\ & \nabla^2_{xx} L(x, \lambda(x), \mu(x)) + Jh(x)^\top\big(J\mu(x) + cJh(x)\big) + J\mu(x)^\top Jh(x) \\ & - \frac{1}{c} J\lambda(x)^\top J\lambda(x) + \frac{1}{c}\sum_{i \in \mathcal{I}_I} \big(J\lambda_i(x) - cJg_i(x)\big)^\top \big(J\lambda_i(x) - cJg_i(x)\big) \\ & + \frac{1}{c}\sum_{i \in \mathcal{I}_M} \big(J\lambda_i(x) - cJg_i(x)\big)^\top M_i(\xi_i, u_i)\big(J\lambda_i(x) - cJg_i(x)\big) \end{aligned} \tag{5.3}$$

for some sets of indices $\mathcal{I}_I, \mathcal{I}_M \subseteq \{1, \dots, r\}$, *some* $\xi_i \in \mathbb{R}$ *and* $u_i \in \mathbb{R}^{m_i - 1}$ *satisfying* $|\xi_i| \le 1$ *and* $\|u_i\| = 1$ *for each* $i \in \mathcal{I}_M$, *and*

$$\phi_c(x) \doteq \sum_{i=1}^r \sum_{j=1}^{m_i} [y_c(x)]_{i,j}\big(\nabla^2 \lambda_{i,j}(x) - c\nabla^2 g_{i,j}(x)\big) + \sum_{j=1}^p h_j(x)\big(\nabla^2 \mu_j(x) + c\nabla^2 h_j(x)\big), \tag{5.4}$$

with $\nabla^2 g_{i,j}(x)$ *and* $\nabla^2 \lambda_{i,j}(x)$ *denoting the Hessians of the* j*th component of* g_i *and* λ_i *at* x, *respectively, and* $[y_c(x)]_{i,j}$ *denoting the* j*th entry of* $[y_c(x)]_i$.

Proof. Let $V \in \partial_B \nabla w_c(x)$ be arbitrarily given. Then, from the first equality in (3.16), there exist $R_i \in \partial_B P_{\mathcal{K}_i}(\lambda_i(x) - cg_i(x))$, $i = 1, \dots, r$, such that

$$\begin{aligned} V = \ & \nabla^2 f(x) + \sum_{j=1}^p \mu_j(x)\nabla^2 h_j(x) + Jh(x)^\top J\mu(x) + \sum_{j=1}^p h_j(x)\nabla^2\mu_j(x) + J\mu(x)^\top Jh(x) \\ & + c\sum_{j=1}^p h_j(x)\nabla^2 h_j(x) + cJh(x)^\top Jh(x) - \frac{1}{c}\sum_{i=1}^r\sum_{j=1}^{m_i} \lambda_{i,j}(x)\nabla^2\lambda_{i,j}(x) \\ & - \frac{1}{c}J\lambda(x)^\top J\lambda(x) + \sum_{i=1}^r\sum_{j=1}^{m_i}\big[P_{\mathcal{K}_i}\big(\lambda_i(x) - cg_i(x)\big)\big]_j \left(\frac{1}{c}\nabla^2\lambda_{i,j}(x) - \nabla^2 g_{i,j}(x)\right) \\ & + \sum_{i=1}^r \left(\frac{1}{c}J\lambda_i(x) - Jg_i(x)\right)^\top R_i\big(J\lambda_i(x) - cJg_i(x)\big), \end{aligned}$$

where $\big[P_{\mathcal{K}_i}(\lambda_i(x) - cg_i(x))\big]_j$ and $\lambda_{i,j}(x)$ are the jth entries of $P_{\mathcal{K}_i}(\lambda_i(x) - cg_i(x))$ and $\lambda_i(x)$, respectively. From definition (3.17) of $y_c(x)$, we obtain

$$\begin{aligned} V = \ & \nabla^2_{xx} L(x, \lambda(x), \mu(x)) + Jh(x)^\top\big(J\mu(x) + cJh(x)\big) + J\mu(x)^\top Jh(x) \\ & - \frac{1}{c}J\lambda(x)^\top J\lambda(x) + \frac{1}{c}\sum_{i=1}^r \big(J\lambda_i(x) - cJg_i(x)\big)^\top R_i\big(J\lambda_i(x) - cJg_i(x)\big) + \phi_c(x), \end{aligned}$$

where $\phi_c(x)$ is defined as in (5.4). Now, define $\mathcal{I}_I, \mathcal{I}_M, \mathcal{I}_0$ as the sets of indices

$$
\begin{aligned}
\mathcal{I}_I &\doteq \big\{ i \in \{1,\dots,r\} \colon R_i = I_{m_i} \big\}, \\
\mathcal{I}_0 &\doteq \big\{ i \in \{1,\dots,r\} \colon R_i = 0 \big\}, \\
\mathcal{I}_M &\doteq \big\{ i \in \{1,\dots,r\} \colon R_i = M_i(\xi,u) \text{ for some } \xi \in \mathbb{R} \text{ and } u \in \mathbb{R}^{m_i-1} \\
&\qquad\qquad\qquad\qquad\qquad\qquad\qquad \text{with } |\xi| \le 1 \text{ and } \|u\| = 1 \big\}.
\end{aligned} \tag{5.5}
$$

From Lemma 5.2, $\mathcal{I}_I \cup \mathcal{I}_0 \cup \mathcal{I}_M = \{1,\dots,r\}$, and the result follows. ◻

From the above result, we observe that $V \in \partial_B \nabla w_c(x)$ contains $\nabla^2 \lambda_{i,j}$ and $\nabla^2 \mu_{i,j}$ in its formula, which in turn contain third-order derivatives of f, $g_{i,j}$, and h_j (see Proposition 3.3(d)). When these functions are only twice continuously differentiable, a way to avoid such computation is required. This will be discussed in section 6.

COROLLARY 5.5. *Let (x^*, λ^*, μ^*) be a KKT triple for* (SOCP) *and let $c > 0$. Then, any let $V \in \partial_B \nabla w_c(x^*)$ is expressed as*

$$
\begin{aligned}
V = {}& \nabla^2_{xx} L(x^*, \lambda^*, \mu^*) + Jh(x^*)^\top \big(J\mu(x^*) + cJh(x^*) \big) + J\mu(x^*)^\top Jh(x^*) \\
&- \frac{1}{c} J\lambda(x^*)^\top J\lambda(x^*) + \frac{1}{c} \sum_{i \in \mathcal{I}_I} \big(J\lambda_i(x^*) - cJg_i(x^*) \big)^\top \big(J\lambda_i(x^*) - cJg_i(x^*) \big) \\
&+ \frac{1}{c} \sum_{i \in \mathcal{I}_M} \big(J\lambda_i(x^*) - cJg_i(x^*) \big)^\top M_i(\xi_i, u_i) \big(J\lambda_i(x^*) - cJg_i(x^*) \big)
\end{aligned}
$$

for some sets of indices $\mathcal{I}_I, \mathcal{I}_M \subseteq \{1,\dots,r\}$, and some $\xi_i \in \mathbb{R}$ and $u_i \in \mathbb{R}^{m_i-1}$ satisfying $|\xi_i| \le 1$ and $\|u_i\| = 1$ for each $i \in \mathcal{I}_M$.

Proof. From Proposition 3.3(c), $\lambda(x^*) = \lambda^*$ and $\mu(x^*) = \mu^*$. Moreover, the KKT conditions give $h(x^*) = 0$ and $y_c(x^*) = 0$ (see Proposition 3.4). Thus, by (5.4), $\phi_c(x^*) = 0$. Hence, the formula of $V \in \partial_B \nabla w_c(x^*)$ follows from Proposition 5.4. ◻

We are now ready to show that the generalized Newton method (5.1), with unit step-size $t_k \doteq 1$, results in superlinear convergence, under the following assumption [7].

Assumption 5.6. Let (x^*, λ^*, μ^*) be a KKT triple for (SOCP). The *strong second-order sufficient condition* holds, that is,

$$
\left\langle \left(\nabla^2_{xx} L(x^*, \lambda^*, \mu^*) + \sum_{i=1}^{r} H_i(x^*, \lambda^*) \right) d, d \right\rangle \ge \eta_0 \|d\|^2
$$

for all $d \in \operatorname{aff}(C(x^*))$, where $\eta_0 > 0$,

$$
\operatorname{aff}(C(x^*)) \doteq \left\{ d \in \mathbb{R}^n \colon \begin{array}{ll} Jh(x^*)d = 0, & \\ Jg_i(x^*)d = 0, & i \in I^*_{0I}, \\ \langle Jg_i(x^*)d, \lambda^*_i \rangle = 0, & i \in I^*_{BB}, \\ Jg_i(x^*)d \in \{ \nu(\lambda^*_{i0}, -\bar{\lambda}^*_i) \colon \nu \in \mathbb{R} \}, & i \in I^*_{0B} \end{array} \right\}
$$

is the affine hull of the critical cone at x^*, and

$$
H_i(x^*, \lambda^*) \doteq \begin{cases} -\kappa_i Jg_i(x^*)^\top \begin{bmatrix} 1 & 0^\top \\ 0 & -I_{m_i-1} \end{bmatrix} Jg_i(x^*) & \text{if } i \in I^*_{BB}, \\ 0 & \text{otherwise}, \end{cases} \tag{5.6}
$$

with κ_i defined as in Lemma 5.1.

THEOREM 5.7. *Let $x^* \in \mathbb{R}^n$ be a KKT point for* (SOCP) *and suppose that Assumption* 5.6 *holds. Then, all matrices in the B-subdifferential $\partial_B \nabla w_c(x^*)$ are positive definite for any $c > 0$ sufficiently large.*

Proof. Let us assume for the purpose of contradiction that there are sequences $\{c_k\} \subset \mathbb{R}_{++}$ and $\{V_k\} \subset \mathbb{R}^{n\times n}$ such that $c_k \to +\infty$ and $V_k \in \partial_B \nabla w_{c_k}(x^*)$ is not positive definite for all k. Then, there exists $\{d^k\} \subset \mathbb{R}^n$ such that $\|d^k\| = 1$ and $\langle V_k d^k, d^k\rangle \le 0$ for all k. Without loss of generality, suppose that $\{d^k\}$ converges to some vector $d^* \in \mathbb{R}^n$ such that $\|d^*\| = 1$. Then, from Corollary 5.5, for each k, there exist sets of indices $\mathcal{I}_I^k$, $\mathcal{I}_M^k$ such that

$$
\begin{aligned}
\big\langle V_k d^k, d^k\big\rangle = {} & \left\langle \left(\nabla^2_{xx} L(x^*,\lambda^*,\mu^*) + \sum_{i=1}^r H_i(x^*,\lambda^*)\right) d^k, d^k \right\rangle \\
& - \sum_{i\in I^*_{BB}} \big\langle H_i(x^*,\lambda^*)d^k, d^k\big\rangle + 2\big\langle Jh(x^*)d^k, J\mu(x^*)d^k\big\rangle + c_k\big\|Jh(x^*)d^k\big\|^2 \\
& + c_k \sum_{i\in\mathcal{I}_I^k} \left\| \left(\frac{J\lambda_i(x^*)}{c_k} - Jg_i(x^*)\right) d^k \right\|^2 - \frac{1}{c_k}\big\|J\lambda(x^*)d^k\big\|^2 \\
& + c_k \sum_{i\in\mathcal{I}_M^k\setminus I^*_{BB}} \left\| M_i(\xi_i,u_i)^{1/2}\left(\frac{J\lambda_i(x^*)}{c_k} - Jg_i(x^*)\right) d^k \right\|^2 \\
& + \frac{1}{c_k}\sum_{i\in I^*_{BB}} \Bigg\langle \big(J\lambda_i(x^*) - c_k Jg_i(x^*)\big)d^k, \\
& \qquad M_i\left(\frac{\kappa_i - c_k}{\kappa_i + c_k}, \frac{\bar\lambda_i^*}{\|\bar\lambda_i^*\|}\right)\big(J\lambda_i(x^*) - c_k Jg_i(x^*)\big)d^k \Bigg\rangle,
\end{aligned} \tag{5.7}
$$

where $\xi_i \in \mathbb{R}$ and $u_i \in \mathbb{R}^{m_i-1}$ satisfy $|\xi_i| \le 1$ and $\|u_i\| = 1$ for each $i \in \mathcal{I}_M^k \setminus I^*_{BB}$, $H_i(x^*,\lambda^*)$ is defined by (5.6), and the fact that $H_i(x^*,\lambda^*) = 0$ for all $i \notin I^*_{BB}$ is used. Also, recalling (5.5) and Lemma 5.3, we notice that for $i \in \mathcal{I}_M^k \setminus I^*_{BB}$, $M_i(\xi_i,u_i)$ is symmetric positive semidefinite and $M_i(\xi_i,u_i) = M_i(\xi_i,u_i)^{1/2}M_i(\xi_i,u_i)^{1/2}$.

For simplicity, for each $i \in I^*_{BB}$, define $\beta_i \colon \mathbb{R}_{++}\times\mathbb{R}^n \to \mathbb{R}$ with

$$
\beta_i(c,d) \doteq \beta_i^{(1)}(d) + \beta_i^{(2)}(c,d) + \beta_i^{(3)}(c,d) + \beta_i^{(4)}(c,d),
$$

where $\beta_i^{(1)} \colon \mathbb{R}^n \to \mathbb{R}$ and $\beta_i^{(2)}, \beta_i^{(3)}, \beta_i^{(4)} \colon \mathbb{R}_{++}\times\mathbb{R}^n \to \mathbb{R}$ are given by

$$
\begin{aligned}
\beta_i^{(1)}(d) &\doteq - \big\langle H_i(x^*,\lambda^*)d, d\big\rangle, \\
\beta_i^{(2)}(c,d) &\doteq c\left\langle Jg_i(x^*)d, M_i\left(\frac{\kappa_i - c}{\kappa_i + c}, \frac{\bar\lambda_i^*}{\|\bar\lambda_i^*\|}\right) Jg_i(x^*)d\right\rangle, \\
\beta_i^{(3)}(c,d) &\doteq -2\left\langle Jg_i(x^*)d, M_i\left(\frac{\kappa_i - c}{\kappa_i + c}, \frac{\bar\lambda_i^*}{\|\bar\lambda_i^*\|}\right) J\lambda_i(x^*)d\right\rangle, \\
\beta_i^{(4)}(c,d) &\doteq \frac{1}{c}\left\langle J\lambda_i(x^*)d, M_i\left(\frac{\kappa_i - c}{\kappa_i + c}, \frac{\bar\lambda_i^*}{\|\bar\lambda_i^*\|}\right) J\lambda_i(x^*)d\right\rangle.
\end{aligned}
$$

Since there are only finitely many subsets $\mathcal{I}_I^k$ and $\mathcal{I}_M^k$, we may assume that $\mathcal{I}_I^k = \mathcal{I}_I$ and $\mathcal{I}_M^k = \mathcal{I}_M$ for all k. Then, observing that the matrix $M_i\big((\kappa_i - c_k)/(\kappa_i + c_k), \bar\lambda_i^*/\|\bar\lambda_i^*\|\big)$ is symmetric, we can rewrite (5.7) as

$$
\begin{aligned}
\langle V_k d^k, d^k \rangle = {} & \left\langle \left(\nabla^2_{xx} L(x^*, \lambda^*, \mu^*) + \sum_{i=1}^{r} H_i(x^*, \lambda^*) \right) d^k, d^k \right\rangle \\
& + 2 \langle Jh(x^*) d^k, J\mu(x^*) d^k \rangle + c_k \| Jh(x^*) d^k \|^2 - \frac{1}{c_k} \| J\lambda(x^*) d^k \|^2 \\
& + c_k \sum_{i \in \mathcal{I}_I} \left\| \left(\frac{J\lambda_i(x^*)}{c_k} - Jg_i(x^*) \right) d^k \right\|^2 \\
& + c_k \sum_{i \in \mathcal{I}_M \setminus I^*_{BB}} \left\| M_i(\xi_i, u_i)^{1/2} \left(\frac{J\lambda_i(x^*)}{c_k} - Jg_i(x^*) \right) d^k \right\|^2 + \sum_{i \in I^*_{BB}} \beta_i(c_k, d^k).
\end{aligned} \tag{5.8}
$$

Let us fix $i \in I^*_{BB}$ and analyze the expression of $\beta_i(c, d)$ for any $c \in \mathbb{R}_{++}$ and $d \in \mathbb{R}^n$. For simplicity, we also define $s_i \doteq Jg_i(x^*)d$ and $u_i \doteq J\lambda_i(x^*)d$. Observe that, by (5.6),

$$
\beta_i^{(1)}(d) = \kappa_i \left\langle s_i, \begin{bmatrix} 1 & 0^\top \\ 0 & -I_{m_i - 1} \end{bmatrix} s_i \right\rangle = \kappa_i \left[(s_{i0})^2 - \|\bar{s}_i\|^2 \right].
$$

For the term $\beta_i^{(2)}(c, d)$, recall that $\lambda^*_i \in \mathrm{bd}^+(\mathcal{K}_i)$ (i.e., $\lambda^*_{i0} = \|\bar{\lambda}^*_i\| > 0$) and note that

$$
\langle s_i, \lambda^*_i \rangle^2 = \langle \bar{\lambda}^*_i, \bar{s}_i \rangle^2 + 2 \|\bar{\lambda}^*_i\| s_{i0} \langle \bar{\lambda}^*_i, \bar{s}_i \rangle + \|\bar{\lambda}^*_i\|^2 (s_{i0})^2. \tag{5.9}
$$

Thus, we have

$$
\begin{aligned}
\beta_i^{(2)}(c, d) &= \frac{c}{2} \left[s_{i0}^2 + \frac{2 s_{i0}}{\|\bar{\lambda}^*_i\|} \langle \bar{\lambda}^*_i, \bar{s}_i \rangle + \left(\frac{2\kappa_i}{c + \kappa_i} \right) \|\bar{s}_i\|^2 + \left(\frac{c - \kappa_i}{c + \kappa_i} \right) \frac{\langle \bar{\lambda}^*_i, \bar{s}_i \rangle^2}{\|\bar{\lambda}^*_i\|^2} \right] \\
&= \frac{c\kappa_i}{c + \kappa_i} \left[\|\bar{s}_i\|^2 - \frac{\langle \bar{\lambda}^*_i, \bar{s}_i \rangle^2}{\|\bar{\lambda}^*_i\|^2} \right] + \frac{c}{2} \left[s_{i0}^2 + \frac{2 s_{i0}}{\|\bar{\lambda}^*_i\|} \langle \bar{\lambda}^*_i, \bar{s}_i \rangle + \frac{\langle \bar{\lambda}^*_i, \bar{s}_i \rangle^2}{\|\bar{\lambda}^*_i\|^2} \right] \\
&= \frac{c\kappa_i}{c + \kappa_i} \left[\|\bar{s}_i\|^2 - \frac{\langle \bar{\lambda}^*_i, \bar{s}_i \rangle^2}{\|\bar{\lambda}^*_i\|^2} \right] + \frac{c}{2\|\bar{\lambda}^*_i\|^2} \langle s_i, \lambda^*_i \rangle^2,
\end{aligned} \tag{5.10}
$$

where the last equality follows from (5.9). Finally, we have

$$
\begin{aligned}
\beta_i^{(3)}(c, d) = -2 \Bigg[& u_{i0} s_{i0} + \frac{u_{i0}}{\|\bar{\lambda}^*_i\|} \langle \bar{\lambda}^*_i, \bar{s}_i \rangle + \frac{s_{i0}}{\|\bar{\lambda}^*_i\|} \langle \bar{\lambda}^*_i, \bar{u}_i \rangle \\
& + \left(\frac{2\kappa_i}{c + \kappa_i} \right) \langle \bar{s}_i, \bar{u}_i \rangle + \left(\frac{c - \kappa_i}{c + \kappa_i} \right) \frac{\langle \bar{\lambda}^*_i, \bar{s}_i \rangle \langle \bar{\lambda}^*_i, \bar{u}_i \rangle}{\|\bar{\lambda}^*_i\|^2} \Bigg],
\end{aligned}
$$

and, analogously to $\beta_i^{(2)}(c, d)$ (see line (5.10)), we obtain

$$
\beta_i^{(4)}(c, d) = \frac{\kappa_i}{c(c + \kappa_i)} \left[\|\bar{u}_i\|^2 - \frac{\langle \bar{\lambda}^*_i, \bar{u}_i \rangle^2}{\|\bar{\lambda}^*_i\|^2} \right] + \frac{1}{2c} \left[u_{i0}^2 + \frac{2 u_{i0}}{\|\bar{\lambda}^*_i\|} \langle \bar{\lambda}^*_i, \bar{u}_i \rangle + \frac{\langle \bar{\lambda}^*_i, \bar{u}_i \rangle^2}{\|\bar{\lambda}^*_i\|^2} \right]. \tag{5.11}
$$

Let us go back to (5.8). Note that $\langle V_k d^k, d^k \rangle \leq 0$ for all k. Thus, we divide (5.8) by c_k and take the limit $k \to \infty$ to conclude that

$$
\begin{array}{lll}
\text{(a) } Jh(x^*) d^* = 0, & \text{(b) } M_i(\xi_i, u_i)^{1/2} Jg_i(x^*) d^* = 0, & i \in \mathcal{I}_M \setminus I^*_{BB}, \\
\text{(c) } Jg_i(x^*) d^* = 0, i \in \mathcal{I}_I, & \text{(d) } \langle Jg_i(x^*) d^*, \lambda^*_i \rangle = 0, & i \in I^*_{BB}.
\end{array} \tag{5.12}
$$

We notice that (5.12)(d) comes from the last term of $\beta_i^{(2)}(c_k, d^k)$. Now, let us show that $d^* \in \operatorname{aff}(C(x^*))$. Observe that we have only to prove that for each $i \in I_{0B}^*$, $Jg_i(x^*)d^* = \nu(\lambda_{i0}^*, -\bar{\lambda}_i^*)$ for some $\nu \in \mathbb{R}$. First, note that $I_{0B}^* \subset \mathcal{I}_I \cup \mathcal{I}_M$ from Lemma 5.3. If $i \in I_{0B}^* \cap \mathcal{I}_I$, then (5.12)(c) shows that $Jg_i(x^*)d^* = 0 = \nu(\lambda_{i0}^*, -\bar{\lambda}_i^*)$ with $\nu = 0$. Consider now the case $i \in I_{0B}^* \cap \mathcal{I}_M$. It is easy to show that $M_i(1, \bar{\lambda}_i^*/\|\bar{\lambda}_i^*\|)^{1/2} = M_i(1, \bar{\lambda}_i^*/\|\bar{\lambda}_i^*\|)$. Therefore, defining $s_i^* \doteq Jg_i(x^*)d^*$, and from (5.12)(b), we have

$$M_i(1, \bar{\lambda}_i^*/\|\bar{\lambda}_i^*\|)^{1/2} s_i^* = \frac{1}{2} \begin{bmatrix} s_{i0}^* + \dfrac{\langle \bar{\lambda}_i^*, \bar{s}_i^* \rangle}{\|\bar{\lambda}_i^*\|} \\ \left(\dfrac{s_{i0}^*}{\|\bar{\lambda}_i^*\|} - \dfrac{\langle \bar{\lambda}_i^*, \bar{s}_i^* \rangle}{\|\bar{\lambda}_i^*\|^2} \right) \bar{\lambda}_i^* + 2\bar{s}_i^* \end{bmatrix} = 0.$$

This implies that $s_{i0}^* = -\langle \bar{\lambda}_i^*, \bar{s}_i^* \rangle / \|\bar{\lambda}_i^*\|$ and

$$\left(\frac{s_{i0}^*}{\|\bar{\lambda}_i^*\|} - \frac{\langle \bar{\lambda}_i^*, \bar{s}_i^* \rangle}{\|\bar{\lambda}_i^*\|^2} \right) \bar{\lambda}_i^* + 2\bar{s}_i^* = \frac{2 s_{i0}^*}{\|\bar{\lambda}_i^*\|} \bar{\lambda}_i^* + 2\bar{s}_i^* = 0.$$

Since $i \in I_{0B}^*$ ($\lambda_{i0}^* = \|\bar{\lambda}_i^*\| > 0$), the above equality shows that $\bar{s}_i^* = -\big(s_{i0}^*/\lambda_{i0}^*\big)\bar{\lambda}_i^*$. Thus, $s_i^* = Jg_i(x^*)d^* = \nu(\lambda_{i0}^*, -\bar{\lambda}_i^*)$, with $\nu = s_{i0}^*/\lambda_{i0}^*$. Consequently, we conclude that $d^* \in \operatorname{aff}(C(x^*))$.

We now claim that for every d sufficiently close to d^* and for c_k large enough, we have $\langle V_k d, d \rangle > 0$, which will be a contradiction to the hypothesis $\langle V_k d^k, d^k \rangle \leq 0$ with $d^k \to d^*$. Let $i \in I_{BB}^*$ and define $s_i^* \doteq Jg_i(x^*)d^*$ and $u_i^* \doteq J\lambda_i(x^*)d^*$. Observe that $\lambda_{i0}^* = \|\bar{\lambda}_i^*\| > 0$ and that $\langle s_i^*, \lambda_i^* \rangle = 0$ implies $\langle \bar{s}_i^*, \bar{\lambda}_i^* \rangle = -s_{i0}^* \lambda_{i0}^*$. Also, note that

$$\begin{aligned} &\beta_i^{(1)}(d^*) + \beta_i^{(2)}(c_k, d^*) \\ &\quad = \kappa_i \Big[\big(s_{i0}^*\big)^2 - \big\|\bar{s}_i^*\big\|^2 \Big] + \frac{c_k \kappa_i}{c_k + \kappa_i} \left[\big\|\bar{s}_i^*\big\|^2 - \frac{\big(s_{i0}^*\big)^2 \big(\lambda_{i0}^*\big)^2}{\|\bar{\lambda}_i^*\|^2} \right] \\ &\quad = \frac{\kappa_i^2}{c_k + \kappa_i} \Big[\big(s_{i0}^*\big)^2 - \big\|\bar{s}_i^*\big\|^2 \Big] \end{aligned} \tag{5.13}$$

and

$$\begin{aligned} &\beta_i^{(3)}(c_k, d^*) \\ &= -2 \Bigg[u_{i0}^* s_{i0}^* + \frac{u_{i0}^*}{\|\bar{\lambda}_i^*\|} \big\langle \bar{\lambda}_i^*, \bar{s}_i^* \big\rangle + \frac{s_{i0}^*}{\|\bar{\lambda}_i^*\|} \big\langle \bar{\lambda}_i^*, \bar{u}_i^* \big\rangle \\ &\qquad + \left(\frac{2\kappa_i}{c_k + \kappa_i} \right) \big\langle \bar{s}_i^*, \bar{u}_i^* \big\rangle + \left(\frac{c_k - \kappa_i}{c_k + \kappa_i} \right) \frac{\big\langle \bar{\lambda}_i^*, \bar{s}_i^* \big\rangle \big\langle \bar{\lambda}_i^*, \bar{u}_i^* \big\rangle}{\|\bar{\lambda}_i^*\|^2} \Bigg] \\ &= -2 \left[\frac{s_{i0}^*}{\|\bar{\lambda}_i^*\|} \big\langle \bar{\lambda}_i^*, \bar{u}_i^* \big\rangle + \left(\frac{2\kappa_i}{c_k + \kappa_i} \right) \big\langle \bar{s}_i^*, \bar{u}_i^* \big\rangle - \left(\frac{c_k - \kappa_i}{c_k + \kappa_i} \right) \frac{s_{i0}^* \lambda_{i0}^* \big\langle \bar{\lambda}_i^*, \bar{u}_i^* \big\rangle}{\|\bar{\lambda}_i^*\|^2} \right] \\ &= -2 \left[\left(\frac{2\kappa_i}{c_k + \kappa_i} \right) \frac{s_{i0}^*}{\|\bar{\lambda}_i^*\|} \big\langle \bar{\lambda}_i^*, \bar{u}_i^* \big\rangle + \left(\frac{2\kappa_i}{c_k + \kappa_i} \right) \big\langle \bar{s}_i^*, \bar{u}_i^* \big\rangle \right]. \end{aligned} \tag{5.14}$$

So, since $d^* \in \operatorname{aff}(C(x^*))$ (and satisfies (5.12)), in a similar way to (5.8), we can write

$$\begin{aligned}\langle V_k d^*, d^*\rangle &= \left\langle \left(\nabla^2_{xx} L(x^*, \lambda^*, \mu^*) + \sum_{i=1}^{r} H_i(x^*, \lambda^*)\right) d^*, d^* \right\rangle \\ &\quad - \frac{1}{c_k}\left\|J\lambda(x^*)d^*\right\|^2 + \frac{1}{c_k}\sum_{i\in\mathcal{I}_I}\left\|J\lambda_i(x^*)d^*\right\|^2 + \sum_{i\in I^*_{BB}} \beta_i(c_k, d^*) \\ &= \left\langle \left(\nabla^2_{xx} L(x^*, \lambda^*, \mu^*) + \sum_{i=1}^{r} H_i(x^*, \lambda^*)\right) d^*, d^* \right\rangle - \frac{1}{c_k}\sum_{i\notin\mathcal{I}_I}\left\|J\lambda_i(x^*)d^*\right\|^2 \\ &\quad - \sum_{i\in I^*_{BB}}\left[-\left(\beta_i^{(1)}(d^*) + \beta_i^{(2)}(c_k, d^*)\right) - \beta_i^{(3)}(c_k, d^*) - \beta_i^{(4)}(c_k, d^*)\right].\end{aligned}$$

Let $\eta_0 > 0$ be a constant as specified in Assumption 5.6. Recalling (5.11) (with $c = c_k$ and $d = d^*$), (5.13), and (5.14), when c_k is sufficiently large, we have

$$-\left(\beta_i^{(1)}(d^*) + \beta_i^{(2)}(c_k, d^*)\right) - \beta_i^{(3)}(c_k, d^*) - \beta_i^{(4)}(c_k, d^*) < \frac{\eta_0}{3\,|I^*_{BB}|}$$

for any $i \in I^*_{BB}$, and

$$\frac{\left\|J\lambda_i(x^*)d^*\right\|^2}{c_k} < \frac{\eta_0}{3\,|\{1,\ldots,r\}\setminus\mathcal{I}_I|}$$

for any $i \notin \mathcal{I}_I$. Recall that $\|d^*\| = 1$. These inequalities, together with the strong second-order sufficient condition, give

$$\langle V_k d^*, d^*\rangle > \eta_0 - \frac{\eta_0}{3} - \frac{\eta_0}{3} = \frac{\eta_0}{3} > 0$$

for all c_k large enough. It then follows from continuity that $\langle V_k d, d\rangle > 0$ for all d sufficiently close to d^*. This gives the desired contradiction. ∎

We point out that the above result can be established if we replace the strong second-order sufficient condition by the strict complementarity, together with the second-order sufficient condition [7]. In such a case, from (5.2), (5.5), and Corollary 5.5, $\partial_B \nabla w_c(x^*)$ is a singleton, and the proof is analogous to the one above.

6. The algorithm. Let us now present a way to choose the penalty parameter c. The idea was given in [19] and also used in [3, 2]. Observe that $\nabla w_c(x) = 0$ is actually a reformulation of the KKT system (3.1) from Proposition 4.1 and Theorem 4.4. Then, we introduce a function, called a *test function*, that measures the risk of computing a zero of ∇w_c that is not a KKT point. We define the function $\mathcal{T}_c\colon \mathbb{R}^n \to \mathbb{R}$ by

$$\mathcal{T}_c(x) \doteq -\|\nabla w_c(x)\|^2 + \frac{1}{c^\gamma}\left(\|y_c(x)\|^2 + \|h(x)\|^2\right), \tag{6.1}$$

where γ is an arbitrary positive number (e.g., $\gamma = 2$). The following result shows that $\mathcal{T}_c$ is in fact a test function.

PROPOSITION 6.1. *The following statements are equivalent:* (a) *x is a KKT point of* (SOCP)*;* (b) *$\nabla w_c(x) = 0$, $y_c(x) = 0$, and $h(x) = 0$;* (c) *$\nabla w_c(x) = 0$ and $\mathcal{T}_c(x) \le 0$.*

Proof. It follows directly from expression (3.16) of $\nabla w_c(x)$ and the definitions (3.17) and (6.1) of $y_c(x)$ and $\mathcal{T}_c(x)$, respectively, along with Proposition 3.4. ∎

Since the results associated to $\mathcal{T}_c$ and their proofs are similar to the ones presented in [3, section 3] and [2, section 4], we just state here the main result.

PROPOSITION 6.2. *For any $\hat{x} \in \mathbb{R}^n$, either $\hat{x}$ is a stationary point of the feasibility measure α that is infeasible for* (SOCP)*, or there exist $\hat{c}, \hat{\delta} > 0$ such that if $c \geq \hat{c}$ and $\|x - \hat{x}\| \leq \delta$, then $\mathcal{T}_c(x) \leq 0$.*

This result shows a way to update the parameter c. More precisely, from Proposition 6.1, while we approximately compute a zero of ∇w_c, we increase the value of c if the test function $\mathcal{T}_c$ at a point is greater than zero.

ALGORITHM 6.3 (dynamical update of the penalty parameter).

1. *Let $\mathcal{A}_c(x)$ be the iteration function of an algorithm that computes a zero of $\nabla w_c(x)$. Initialize $x^0 \in \mathbb{R}^n$, $c_0 > 0$, $\sigma > 1$, and $\gamma > 0$. Set $k \doteq 0$.*
2. *If x^k is an approximate KKT point of the problem, stop.*
3. *If $\mathcal{T}_{c_k}(x^k) \leq 0$, then go to step* 5.
4. *Set $c_k \doteq \tau c_k$ and go to step* 3.
5. *Compute $x^{k+1} \doteq \mathcal{A}_{c_k}(x^k)$, set $k \doteq k+1$, and go to step* 2.

THEOREM 6.4. *Let $\{x^k\} \subset \mathbb{R}^n$ be a sequence computed by Algorithm* 6.3*. If $\{x^k\}$ is bounded and infinite, then for each one of its accumulation points, either it is a KKT point, or it is a stationary point of the measure α that is infeasible for* (SOCP)*.*

The above theorem, along with Propositions 6.1 and 6.2, shows that when the sequence computed by Algorithm 6.3 converges to a KKT point, then c_k stays constant for large enough k. Now, consider the case when $\{x^k\}$ converges to a KKT point, and let $c_k = c$ for large enough k. Observe that at each iteration of the generalized Newton method (5.1), an element V_k of the B-subdifferential $\partial_B \nabla w_c(x^k)$ is needed. But Propositions 5.4 and 3.3(d) show that V_k contains third-order derivatives of the problem functions f, $g_{i,j}$, and h_j. From the numerical point of view, it is desirable to replace V_k by another matrix that does not contain third-order derivatives. Here, following the idea given in [14], we choose a matrix $\tilde{V}_k$ with the following property: If $\{x^k\}$ converges to a KKT point, then

$$\lim_{k\to\infty} \left\{ \min \left\|\tilde{V}_k - V_k\right\| : V_k \in \partial_B \nabla w_c(x^k) \right\} = 0. \tag{6.2}$$

We suggest using matrix $\tilde{V}_k$ given in (5.3) with $x = x^k$. Note that $\tilde{V}_k$ satisfies

$$\tilde{V}_k = V_k - \phi_c(x^k) \quad \text{for some } V_k \in \partial_B \nabla w_c(x^k),$$

where $\phi_c(x)$ is defined as in (5.4). From continuity, it follows that $\lim_{k\to\infty} \phi_c(x^k) = 0$ when $\{x^k\}$ converges to a KKT point. Then, condition (6.2) holds.

The algorithm is stated as follows.

ALGORITHM 6.5 (a generalized Newton method for nonlinear SOCP based on an exact penalty function).

1. *Choose $x^0 \in \mathbb{R}^n$, $c_0 > 0$, $\tau > 1$, $\varepsilon_1 \geq 0$, $\varepsilon_2, \varepsilon_3 \in (0,1)$, $\sigma \in (0, 1/2)$. Set $k \doteq 0$.*
2. *If $\|\nabla w_{c_k}(x^k)\| \leq \varepsilon_1$, stop.*
3. *If $\mathcal{T}_{c_k}(x^k) \leq 0$, then go to step* 5.
4. *Set $c_k \doteq \tau c_k$ and go to step* 3.
5. *Choose $\tilde{V}_k$ satisfying* (6.2) *and compute d^k such that $\tilde{V}_k d^k = -\nabla w_{c_k}(x^k)$.*
6. *If $w_{c_k}(x^k + d^k) < w_{c_k}(x^k)$, then set $t_k \doteq 1$ and go to step* 9.
7. *If $\langle \nabla w_{c_k}(x^k), d^k \rangle > -\varepsilon_2 \|d^k\| \|\nabla w_{c_k}(x^k)\|$ or $\|d^k\| < \varepsilon_3 \|\nabla w_{c_k}(x^k)\|$, then set $d^k \doteq -\nabla w_{c_k}(x^k)$.*

8. *Find* $t_k \in (0,1]$ *such that* $w_{c_k}(x^k + t_k d^k) \le w_{c_k}(x^k) + \sigma t_k \langle \nabla w_{c_k}(x^k), d^k \rangle$ *with a backtracking strategy.*
9. *Set* $x^{k+1} \doteq x^k + t_k d^k$, $k \doteq k+1$, *and go to step* 2.

Some comments about this algorithm are in order. In step 7, the steepest descent direction is taken instead of the Newton direction if the latter is not a sufficient descent direction or if the norm condition is not satisfied. In step 8, an Armijo-type line search is used in order to find a step-size t_k. At every backtracking step of the line search, the evaluation of $w_{c_k}(x^k + t_k d^k)$ requires the computation of $\lambda(x^k + t_k d^k)$ and $\mu(x^k + t_k d^k)$, which means that a linear least squares problem has to be solved. Since it is computationally expensive, another strategy that can obviate frequent evaluations of w_{c_k} is desired. We leave this question as a future topic of research.

From Theorem 6.4, any accumulation point of a sequence produced by Algorithm 6.5 is a KKT point of (SOCP), or it is a stationary point of α that is infeasible. The next theorem states that the proposed method achieves superlinear convergence.

THEOREM 6.6. *Let* $\{x^k\} \subset \mathbb{R}^n$ *be a sequence produced by Algorithm* 6.5. *Assume that* $\{x^k\}$ *converges to a KKT point of* (SOCP) *that satisfies the strong second-order sufficient condition. Then, eventually, only the Newton direction is taken as the search direction, and* $t_k = 1$ *satisfies the condition in step* 8, *implying that the convergence rate is superlinear.*

Proof. The result follows from Theorem 5.7, the property (6.2), and [14, Proposition 8.1]. See also the results given in [16]. □

7. Numerical experiments. In this section, we present some simple numerical experiments to validate the results described above. Our main objective is to verify that the superlinear convergence rate is actually attained numerically. We have implemented Algorithm 6.5 in Python using the scientific library Scipy. Whenever the Newton direction is rejected in step 7, the Cauchy direction with the spectral step-size described in [6] is used. Moreover, the code employs a nonmonotone Armijo line search, more specifically the variation due to Grippo, Lapariello, and Lucidi [20], which is essential to ensure the effectiveness of the spectral step-size and is known to improve the behavior of a Newton-type method.

To obtain a multipliers estimate in (3.2), we used $\zeta_1 = 2$ and kept $\zeta_2 = 10^{-4}$ small as suggested in [2]. To define the test function in (6.1), we set $\gamma = 2$. In Algorithm 6.5, we set $c_0 = 100$, $\tau = 10$, $\varepsilon_1 = \varepsilon_2 = \varepsilon_3 = 10^{-8}$, and $\sigma = 10^{-4}$. The Armijo search decreases the step-size by a factor of 0.5 whenever needed. The memory parameter in the nonmonotone search was 10, and the maximum number of iterations is 500. The experiments were carried out on a Dell E6410 laptop with a core i5 M250 processor running at 2.4GHz and 4GB of RAM. The operating system was Linux (Ubuntu 11.04). We have borrowed our test problems from [24, 25].

Table 7.1 shows the performance of the method when solving Example 4.1 from [24]. This is a simple problem with a nonconvex quadratic objective function and the constraint $x \in \mathcal{K}^3$, where $\mathcal{K}^3$ denotes[2] the second-order cone in $\mathbb{R}^3$. It is easy to observe that superlinear, probably quadratic, convergence was attained. The (supremum) norm of ∇w_{c_k} goes from the order of 10^{-3} to 10^{-12} in the last three iterations. It is also interesting to point out that the method did converge to the global solution of the problem given in [24] and not to an arbitrary KKT point. Observe also that the

[2]Note the difference between $\mathcal{K}^\ell$ and $\mathcal{K}_\ell$. The first denotes the second-order cone in $\mathbb{R}^\ell$, and the second means the ℓth second-order cone in the Cartesian product $\mathcal{K} = \mathcal{K}_1 \times \cdots \times \mathcal{K}_r$.

TABLE 7.1
Numerical results for Example 4.1 *from* [24].

k	$\|\nabla w_{c_k}\|_\infty$	w_{c_k}
0	3.914233e+01	1.845668e+01
1	6.027272e–01	1.179886e+00
2	6.434275e+00	1.442102e+00
3	9.356229e–02	1.010681e+00
4	1.230339e+00	1.015349e+00
5	6.824498e–03	1.000047e+00
6	4.673036e–03	1.000000e+00
7	2.065192e–07	1.000000e+00
8	4.220489e–12	1.000000e+00

TABLE 7.2
Numerical results for Examples 4.2 *(left) and* 4.3 *(right) from* [24].

k	$\|\nabla w_{c_k}\|_\infty$	w_{c_k}
0	6.642553e+03	9.538828e+03
1	3.411402e+01	3.756326e+00
2	5.233446e+01	2.976846e+01
3	6.830832e+00	3.373946e+00
4	9.349042e–01	3.331225e+00
5	6.022119e–01	3.325817e+00
6	5.004713e–01	3.317168e+00
7	1.373289e–01	3.305990e+00
8	1.554650e–02	3.304765e+00
9	1.374576e–04	3.304749e+00
10	4.485237e–09	3.304749e+00

k	$\|\nabla w_{c_k}\|_\infty$	w_{c_k}
0	2.217257e+02	8.048864e+02
1	2.898601e+01	1.271508e+01
2	2.097331e+01	7.505792e+00
3	6.844309e–01	2.836546e+00
4	2.825392e–03	2.828442e+00
5	7.474542e–04	2.828427e+00
6	4.770868e–09	2.828427e+00

value of w_c increases in iterations 2 and 4. This is allowed due to the nonmonotone nature of the line search employed in the algorithm.

Next, we turn our attention to Examples 4.2 and 4.3 of [24]; both problems are convex. Example 4.2 has a highly nonlinear objective function containing exponentials and polynomials of order up to 4. It is a problem in $\mathbb{R}^5$, with constraints $x \in \mathcal{K}^3 \times \mathcal{K}^2$. Example 4.3 is a linear SOCP problem of finding the point that minimizes the maximal Euclidean distance to three points fixed in the plane. This last example is particularly interesting because strict complementarity does not hold [24]. The results are presented in Table 7.2. Both problems have only linear constraints. Once again the superlinear convergence takes place in the final iterations.

In all these first tests, the globalization strategy did not have an important role in convergence. The pure Newton method, i.e., the method without the globalization strategy, was also able to find the same solutions in almost the same number of steps. We then turn to Experiment 2 in [25]. This problem has highly nonlinear constraints that can easily lead a pure Newton method to diverge. This problem allows for variable dimensions and a number of second-order cone constraints. As in [25], we solved problems with $\mathcal{K} = \mathcal{K}^5 \times \mathcal{K}^5$, $\mathcal{K} = \mathcal{K}^5 \times \mathcal{K}^5 \times \mathcal{K}^{20}$, and $\mathcal{K} = \mathcal{K}^5 \times \mathcal{K}^5 \times \mathcal{K}^{20} \times \mathcal{K}^{20}$. For each $\mathcal{K}$, 10 random problems with random starting points were generated and solved. The results are shown in Table 7.3.

Observe that the pure Newton method failed in almost all instances, solving only two problems of the smallest dimension. The globalized Newton method (Algorithm 6.5) was able to solve 27 problems out of 30, failing only in three of the largest ones. The main difficulty for the pure Newton method is that it diverges to points with very large objective values. This is a direct result of the lack of globalization.

TABLE 7.3

Numerical results for Experiment 2 from [25]. *It shows the median, maximum, and minimum number of iterations needed to solve the problems when trials were successful, and the number of failures.*

$\mathcal{K}$	Globalized Newton				Pure Newton			
	Median	Max	Min	Fail	Median	Max	Min	Fail
$\mathcal{K}^5 \times \mathcal{K}^5$	29	110	15	0	67	82	52	8
$\mathcal{K}^5 \times \mathcal{K}^5 \times \mathcal{K}^{20}$	105	357	35	0	–	–	–	10
$\mathcal{K}^5 \times \mathcal{K}^5 \times \mathcal{K}^{20} \times \mathcal{K}^{20}$	141	495	70	3	–	–	–	10

Anyhow, the globalization strategy still failed, although not often, suggesting that such a naive globalization scheme may not be robust enough to deal with difficult, highly nonlinear problems. Moreover, we point out that convergence to infeasible points, but stationary to the feasibility measure α, did not occur in all these tests.

Finally we would like to stress that, whenever convergence occurred, the Newton direction with unit step-size was accepted in the last few iterations, resulting in superlinear convergence. Actually, in many instances, the typical behavior was slow progress with frequent rejections of the Newton direction until the Newton convergence basin was achieved. In a few cases, this was possible only after updating the penalty parameter from 100 to 1000, which was the largest c value used (step 4 of Algorithm 6.5). Such slow progress in the early stage of iterations also suggests that better globalization strategies, together with improved criteria to update the penalty parameter sooner, must be investigated.

8. Conclusions.

We have proposed a method for solving nonlinear SOCPs that uses a continuously differentiable exact penalty function as a base. Under the nondegeneracy assumption and the strong second-order sufficient condition, we have proved that the method has global and superlinear convergence. Preliminary numerical experiments have been carried out to confirm the theoretical properties. Some investigations should be done in future research, including comparison with other methods.

Acknowledgments. We would like to thank Roberto Andreani and the referees for their helpful comments and suggestions.

REFERENCES

[1] F. ALIZADEH AND D. GOLDFARB, *Second-order cone programming*, Math. Program., 95 (2003), pp. 3–51.

[2] R. ANDREANI, E. H. FUKUDA, AND P. J. S. SILVA, *A Gauss-Newton approach for solving constrained optimization problems using differentiable exact penalties*, J. Optim. Theory Appl., published online June 29, 2012; doi:10.1007/s10957-012-0114-6.

[3] T. A. ANDRÉ AND P. J. S. SILVA, *Exact penalties for variational inequalities with applications to nonlinear complementarity problems*, Comput. Optim. Appl., 47 (2010), pp. 401–429.

[4] A. AUSLENDER AND M. TEBOULLE, *Lagrangian duality and related multiplier methods for variational inequality problems*, SIAM J. Optim., 10 (2000), pp. 1097–1115.

[5] D. P. BERTSEKAS, *Constrained Optimization and Lagrange Multipliers Methods*, Academic Press, New York, 1982.

[6] E. G. BIRGIN, J. M. MARTÍNEZ, AND M. RAYDAN, *Nonmonotone spectral projected gradient method on convex sets*, SIAM J. Optim., 10 (2000), pp. 1196–1211.

[7] J. F. BONNANS AND H. RAMÍREZ C., *Perturbation analysis of second-order cone programming problems*, Math. Program., 104 (2005), pp. 205–227.

[8] J. F. BONNANS AND A. SHAPIRO, *Perturbation Analysis of Optimization Problems*, Springer-Verlag, New York, 2000.

[9] S. BOYD AND L. VANDENBERGHE, *Convex Optimization*, Cambridge University Press, Cambridge, uk, 2004.

[10] J.-S. Chen, X. Chen, and P. Tseng, *Analysis of nonsmooth vector-valued functions associated with second-order cones*, Math. Program., 101 (2004), pp. 95–117.
[11] G. Di Pillo and L. Grippo, *A continuously differentiable exact penalty function for nonlinear programming problems with inequality constraints*, SIAM J. Control Optim., 23 (1985), pp. 72–84.
[12] G. Di Pillo and L. Grippo, *Exact penalty functions in constrained optimization*, SIAM J. Control Optim., 27 (1989), pp. 1333–1360.
[13] G. Di Pillo, G. Liuzzi, S. Lucidi, and L. Palagi, *A truncated Newton method in an Augmented Lagrangian framework for nonlinear programming*, Comput. Optim. Appl., 45 (2010), pp. 311–352.
[14] G. Di Pillo and S. Lucidi, *An augmented Lagrangian function with improved exactness properties*, SIAM J. Optim., 12 (2001), pp. 376–406.
[15] F. Facchinei and J. S. Pang, *Finite-Dimensional Variational Inequalities and Complementarity Problems*, Springer Ser. Oper. Res., Springer-Verlag, New York, 2003.
[16] F. Facchinei, *Minimization of SC^1 functions and the Maratos effect*, Oper. Res. Lett., 17 (1995), pp. 131–137.
[17] R. Fletcher, *A class of methods for nonlinear programming with termination and convergence properties*, in Integer and Nonlinear Programming, J. Abadie, ed., North-Holland, Amsterdam, Amsterdam, 1970, pp. 157–173.
[18] M. Fukushima, Z.-Q. Luo, and P. Tseng, *Smoothing functions for second-order-cone complementarity problems*, SIAM J. Optim., 12 (2001), pp. 436–460.
[19] T. Glad and E. Polak, *A multiplier method with automatic limitation of penalty growth*, Math. Program., 17 (1979), pp. 140–155.
[20] L. Grippo, F. Lampariello, and S. Lucidi, *A nonmonotone line search technique for Newton's method*, SIAM J. Numer. Anal., 23 (1986), pp. 707–716.
[21] 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), pp. 593–615.
[22] M. R. Hestenes, *Multiplier and gradient methods*, J. Optim. Theory Appl., 4 (1969), pp. 303–320.
[23] J.-P. Hiriart-Urruty and C. Lemaréchal, *Fundamentals of Convex Analysis*, Springer, Berlin, 2004.
[24] 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*, SIAM J. Optim., 20 (2009), pp. 297–320.
[25] H. Kato and M. Fukushima, *An SQP-type algorithm for nonlinear second-order cone programs*, Optim. Lett., 1 (2007), pp. 129–144.
[26] Y. Z. Liu and L. W. Zhang, *Convergence analysis of the augmented Lagrangian method for nonlinear second-order cone optimization problems*, Nonlinear Anal., 67 (2007), pp. 1359–1373.
[27] Y. Z. Liu and L. W. Zhang, *Convergence of the augmented Lagrangian method for nonlinear optimization problems over second-order cones*, J. Optim. Theory Appl., 139 (2008), pp. 557–575.
[28] M. S. Lobo, L. Vandenberghe, S. Boyd, and H. Lebret, *Applications of second-order cone programming*, Linear Algebra Appl., 284 (1998), pp. 193–228.
[29] S. Lucidi, *New results on a continuously differentiable exact penalty function*, SIAM J. Optim., 2 (1992), pp. 558–574.
[30] H. Mukai and E. Polak, *A quadratically convergent primal-dual algorithm with global convergence properties for solving optimization problems with equality constraints*, Math. Program., 9 (1975), pp. 336–349.
[31] J. S. Pang and L. Qi, *Nonsmooth equations: Motivation and algorithms*, SIAM J. Optim., 3 (1993), pp. 443–465.
[32] J. S. Pang, D. Sun, and J. Sun, *Semismooth homeomorphisms and strong stability of semidefinite and Lorentz complementarity problems*, Math. Oper. Res., 28 (2003), pp. 39–63.
[33] M. J. D. Powell, *A method for nonlinear constraints in minimization problems*, in Optimization, R. Fletcher, ed., Academic Press, New York, 1969, pp. 283–298.
[34] L. Qi and J. Sun, *A nonsmooth version of Newton's method*, Math. Program., 58 (1993), pp. 353–367.
[35] L. Qi, *Convergence analysis of some algorithms for solving nonsmooth equations*, Math. Oper. Res., 18 (1993), pp. 227–244.
[36] R. T. Rockafellar, *Augmented Lagrange multiplier functions and duality in nonconvex programming*, SIAM J. Control Optim., 12 (1974), pp. 268–285.

[37] D. SUN, J. SUN, AND L. ZHANG, *The rate of convergence of the augmented Lagrangian method for nonlinear semidefinite programming*, Math. Program., 114 (2008), pp. 349–391.

[38] Y. WANG AND L. ZHANG, *Properties of equation reformulation of the Karush-Kuhn-Tucker condition for nonlinear second order cone optimization problems*, Math. Methods Oper. Res., 70 (2009), pp. 195–218.

[39] H. YAMASHITA AND H. YABE, *A primal-dual interior point method for nonlinear optimization over second-order cones*, Optim. Methods Softw., 24 (2009), pp. 407–426.

[40] W. I. ZANGWILL, *Nonlinear programming via penalty functions*, Manage. Sci., 13 (1967), pp. 344–358.