A primal-dual interior-point algorithm for nonsymmetric exponential-cone optimization

Joachim Dahl Erling D. Andersen

May 27, 2019

Abstract

A new primal-dual interior-point algorithm applicable to nonsymmetric conic optimization is proposed. It is a generalization of the famous algorithm suggested by Nesterov and Todd for the symmetric conic case, and uses primal-dual scalings for nonsymmetric cones proposed by Tunçel. We specialize Tunçel's primal-dual scalings for the important case of 3 dimensional exponential-cones, resulting in a practical algorithm with good numerical performance, on level with standard symmetric cone (*e.g.*, quadratic cone) algorithms. A significant contribution of the paper is a novel higher-order search direction, similar in spirit to a Mehrotra corrector for symmetric cone algorithms. To a large extent, the efficiency of our proposed algorithm can be attributed to this new corrector.

1 Introduction

In 1984 Karmarkar [11] presented an interior-point algorithm for linear optimization with polynomial complexity. This triggered the interior-point revolution which gave rise to a vast amount research on interior-point methods. A particularly important result was the analysis of so-called *self-concordant* barrier functions, which led to polynomial-time algorithms for linear optimization over a convex domain with self-concordant barrier, provided that the barrier function can be evaluated in polynomial time. This was proved by Nesterov and Nemirovski [18], and as a consequence convex optimization problems with such barriers can be solved efficently by interior-point methods, at least in theory.

However, numerical studies for linear optimization quickly demonstrated that primal-dual interior-point methods were superior in practice, which led researchers to generalize the primal-dual algorithm to general smooth convex problems. A major breakthrough in that direction is the seminal work by Nesterov and Todd (NT) [22, 21] who generalized the primal-dual algorithm for linear optimization to *self-scaled* cones, with the same complexity bound as in the linear case. Güler [8] later showed that the self-scaled cones corresponds to class of symmetric cones, which has since become the most commonly used term. The good theoretical performance of primal-dual methods for symmetric cones has since been confirmed computionally, *e.g.*, by [27, 2].

The class of symmetric cones has been completely characterized and includes 5 different cones, where the three most interesting ones are the linear cone, the quadratic cone, and the cone of symmetric positive semidefinite matrices, as well as products of those three cones. Although working exclusively with the symmetric cones is a limitation, they cover a great number of important applications, see, *e.g.*, Nemirovski [16] for an excellent survey of conic optimization over symmetric cones.

Some convex sets with a symmetric cone representation are more naturally characterized using non-symmetric cones, *e.g.*, semidefinite matrices with chordal sparsity patterns, see [31] for an extensive survey. Thus algorithmic advancements for handling nonsymmetric cones directly could hopefully lead to both simpler modeling and reductions in computational complexity. Many other important convex sets cannot readily be modeled using symmetric cones, *e.g.*, convex sets involving exponentials or logarithms for which no representation using symmetric cones is known. In terms of practical importance, the three dimensional power- and exponential-cones are perhaps the most important nonsymmetric cones; Lubin et.al. [12] showed how all instances in a large benchmark library can modeled using the three symmetric cones as well the three dimensional power- and exponential cone.

Generalizing the NT algorithm to nonsymmetric cones is not straightforward, however. An important feature that holds symmetric cones is the existence of a scaling-point w relating primal and dual variables

x and s as $s = F''(w)x$ where F'' denotes the Hessian of the self-scaled barrier function. Such a scaling-point is not exists in general for a non-symmetric cone, which creates a nontrivial challenge in generalizing the theory and algorithms from [22, 21] to non-symmetric cones. In [23] Nesterov suggest an long-step algorithm using both the primal and dual barriers effectively doubling the size of the linear system solved at each iteration. For small-dimensional cones (such as the exponential cone) this overhead might be acceptable. More recently Nesterov [20] proved the existence of a NT-like primal-dual scaling in the vicinity of the central path, leading to an algorithm that uses only a single barrier, but is restricted to following the central path closely. At each iteration the algorithm has centering phase, which brings the current iterate close to the central path. This is followed by one affine step which brings the iterate close to optimum. From a practical perspective this is a significant drawback of the method since the centering phase is computational costly. Hence, the centering and affine steps should be combined as in the symmetric cone case. Also both algorithms [23, 20] are feasible methods. Hence, they require either a strictly feasible known starting point or some sort of phase-I method to get a feasible starting point. Skajaa and Ye [26] extended these methods with a homogeneous model, which also simplifies infeasibility detection. In practice, however, the method of Skaajaa and Ye is not competitive with other methods [3] due to the centering steps. In more recent work Serrano and Ye [25] improves the algorithm so explicit centering steps are not needed, but instead restricting iterates to a vicinity of the central path. This method has been implemented as part of the ECOS solver [6] and will be used for comparison in §9.

A different approach to non-symmetric conic optimization was proposed by Tunçel [29], who introduced primal-dual scaling (not parametrized by w) defined as low-rank updates to either the Hessian of the primal or dual barrier function, with similar properties as Nesterov-Todd scalings. These ideas were further explored by Myklebust and Tunçel [15] and also in [14] and form the basis of our implementation. A main difficulty with this method is to compute well-bounded scaling matrices for general non-symmetric cones. The complexity result developed in [15, 14] hinges on the existence of bounded scaling matrices, which have not been established for nonsymmetric cones. For three-dimensional cones such scaling matrices are particularly simple and characterized by a single scalar, as shown in §5. Our numerical results indicate that an implementation using the scalings [29] results in an algorithm with a practical iteration complexity similar to the ECOS implementation [6].

It is also possible to develop algorithms for convex optimization specified on functional form. This has been done by several authors, for example by [4, 7, 3, 9] who all solve the KKT optimality conditions of the problem in functional form. The algorithms all require a sort of merit or penalty function, which often require problem specific parameter tuning to work well in practice. Another strain of research is the work Nemirovski and Tunçel [17] and very recently Karimi and Tunçel [10] who advocate a non-linear convex formulation instead of a conic formulations, explicitly using self-concordant barriers for the convex domains. They present theoretical results with the same best-known complexity bounds as the NT algorithm for the symmetric cone case. Whether these methods will be competitive with conic formulations in practice is still an unanswered question, though.

The remaining paper is structured as follows. We define basic properties for the exponential cone in §2, and we discuss the homogeneouous model, the central path and related metrics in §3. In §4 we discuss search-directions assuming a primal-dual scaling is known and some basic convergence properties. Subsequently in §5 we discuss new characterizations of the primal-dual scalings from [29, 15], which reduce to univariate characterizations for three-dimensional cones. It is well known that the Mehrotra predictor-corrector idea [13] leads to vastly improved computational performance in the symmetric cone case. One of our contributions presented in §6 is new corrector applicable to nonsymmetric cones. The corrector shares some similarities with the Mehrotra corrector, and offers a substantial reduction in the number of iterations required to solve the problems in all our numerical studies. In §7 we give a collected overview of the suggested path-following algorithm. This is followed by a discussion of some implementational details. Next in §9 we present numerical results on a moderately large collection of exponential-cone problems. We conclude in §10, and in the appendix we give details of the first-, second- and third-order derivatives of the barrier for the exponential cone.

2 Preliminaries

In this section we list well-known properties of self-concordant and self-scaled barriers, which are used in the remainder of the paper. The proofs can be found in references such as [22, 21]. We consider a pair

of primal and dual linear conic problems

$$\begin{array}{ll} \text{minimize} & c^T x \\ \text{subject to} & Ax = b \\ & x \in K, \end{array} \tag{P}$$

and

$$\begin{array}{ll} \text{maximize} & b^T y \\ \text{subject to} & c - A^T y = s \\ & s \in K^*, \end{array} \tag{D}$$

where $y \in \mathbb{R}^m$, $s \in \mathbb{R}^n$ and $K \subset \mathbb{R}^n$ is a proper cone, *i.e.*, a pointed, closed, convex cone with non-empty interior. The dual cone K^* is

$$K^* = \{z \in \mathbb{R}^n \mid \langle x, z\rangle \geq 0, \forall x \in K\}.$$

If K is proper then K^* is also proper. A cone K is called self-dual if there is positive definite map between K and K^*, *i.e.*, if $TK = K^*$, $T \succ 0$. A function $F : \mathbf{int}(K) \mapsto \mathbb{R}$, $F \in C^3$ is a ν-logarithmically homogeneouos self-concordant barrier (ν-LHSCB) for $\mathbf{int}(K)$ if

$$|F'''(x)[u,u,u]| \leq 2(F''(x)[u,u])^{3/2}$$

and

$$F(\tau x) = F(x) - \nu \log \tau$$

holds for all $x \in \mathbf{int}(K)$ and for all $u \in \mathbb{R}^n$. For a pointed cone $\nu \geq 1$ holds. We will refer to a LHSCB simply as a self-concordant barrier. If F_1 and F_2 are ν_1 and ν_2-self-concordant barriers for K_1 and K_2, respectively, then $F_1(x_1)+F_2(x_2)$ is a $(\nu_1+\nu_2)$-self-concordant barrier for $K_1 \times K_2$. Some straightforward consequences of the homogeneouos property include

$$F'(\tau x) = \frac{1}{\tau} F'(x), \quad F''(\tau x) = \frac{1}{\tau^2} F''(x), \tag{1}$$

$$F''(x)x = -F'(x), \quad F'''(x)x = -2F''(x), \tag{2}$$

$$\langle F'(x), x\rangle = -\nu. \tag{3}$$

If F is a ν-self-concordant barrier for K, then the Fenchel conjugate

$$F_*(s) = \sup_{x \in \mathbf{int}(K)} \{-\langle s, x\rangle - F(x)\} \tag{4}$$

is a ν-self-concordant barrier for K^*. Let

$$\tilde{x} := -F_*'(s), \quad \tilde{s} := -F'(x), \quad \tilde{\mu} := \frac{\langle \tilde{x}, \tilde{s}\rangle}{\nu}.$$

Then $\tilde{x} \in \mathbf{int}(K)$, $\tilde{s} \in \mathbf{int}(K^*)$ and

$$\mu\tilde{\mu} \geq 1 \tag{5}$$

with equality iff $x = -\mu\tilde{x}$ (and $s = \mu\tilde{s}$). For a ν-self-concordant barrier, the Dikin ellipsoid for $r < 1$ is included in the cone, *i.e.*,

$$E(x; r) = \{x \in \mathbb{R}^n \mid \langle F''(x)(y-x), y-x\rangle < r\} \subset \mathbf{int}(K)$$

and F is almost quadratic inside this ellipsoid,

$$(1-r)^2 F''(x) \preceq F''(z) \preceq \frac{1}{(1-r)^2} F''(x) \tag{6}$$

for all $z \in E(x, r)$, $r < 1$. A cone is called self-scaled if it has a ν-self-concordant barrier F such that for all $w, x \in \mathbf{int}(K)$,

$$F''(w)x \in \mathbf{int}(K^*)$$

and

$$F_*(F''(w)) = F(x) - 2F(w) - \nu.$$

Self-scaled cones are equivalent to symmetric cones and they satisfy the stronger long-step Hessian estimation property

$$\frac{1}{(1+\alpha\sigma_x(-p))^2}F''(x) \preceq F''(x-\alpha p) \preceq \frac{1}{(1-\alpha\sigma_x(p))^2}F''(x)$$

for any $\alpha \in [0; \sigma_x(p)^{-1})$ where

$$\sigma_x(p) := \frac{1}{\sup\{\alpha \ : \ x - \alpha p \in K\}}$$

denotes the distance to the boundary. Many properties of symmetric cones follow from the fact that the barriers have negative curvature $F'''(x)[u] \preceq 0$ for all $x \in \mathbf{int}(K)$ and all $u \in K$. An interesting property proven in [19] is that if both the primal and dual barrier has negative curvature then the cone is symmetric.

In addition to the three symmetric cones (*i.e.*, the nonnegative orthant, the quadratic cone and the cone of symmetric positive semidefinite matrices) we mainly consider the nonsymmetric exponential cone studied by Charez [5] in the present work.

$$K_{\exp} = \mathrm{cl}\{x \in \mathbb{R}^3 \mid x_1 \geq x_2 \exp(x_3/x_2),\ x_2 > 0\}, \tag{7}$$

with a 3-self-concordant barrier,

$$F(x) = -\log(x_2 \log(x_1/x_2) - x_3) - \log x_1 - \log x_2. \tag{8}$$

The dual exponential cone is

$$K^*_{\exp} = \mathrm{cl}\{z \in \mathbb{R}^3 \mid e \cdot z_1 \geq -z_3 \exp(z_2/z_3),\ z_1 > 0, z_3 < 0\}. \tag{9}$$

The exponential cone is not self-dual, but $TK_{\exp} = K^*_{\exp}$ for

$$T = \begin{bmatrix} e & 0 & 0 \\ 0 & 0 & -1 \\ 0 & -1 & 0 \end{bmatrix} \not\succ 0. \tag{10}$$

For a ν-self-scaled cone

$$F_*(s) = F(x) - \nu$$

holds. However, for the exponential cone the conjugate barrier $F_*(x)$ or its derivatives cannot be evaluated on closed-form, but it can be evaluated numerically to high accuracy (*e.g.*, with a damped Newton's method) using the definition (4), *i.e.*, if

$$x_s = \arg\min\{-\langle s, x\rangle - F(x) \ : \ x \in \mathbf{int}(K)\} \tag{11}$$

then

$$F'_*(s) = -x_s, \quad F''_*(s) = [F''(x_s)]^{-1}. \tag{12}$$

We conclude this survey of introductory material by listing some of the many convex sets that can be represented using the exponential cone, or a combination of exponential cones and symmetric cones. The epigraph $t \geq e^x$ can be modelled as $(t, 1, x) \in K_{\exp}$ and similarly for the hypograph of the logarithm $t \leq \log x \Leftrightarrow (x, 1, t) \in K_{\exp}$. The hypograph of the entropy function, $t \leq -x \log x$ is equivalent to $(1, x, t) \in K_{\exp}$, and similarly for relative entropy $t \geq x \log(y/x) \Leftrightarrow (y, x, -t) \in K_{\exp}$. The *soft-plus function* $\log(1 + e^x)$ can be thought of as a smooth approximation of $\max\{0, x\}$. Its epigraph can be modelled as $t \geq \log(1 + e^x) \Leftrightarrow u + v = 1,\ (u, 1, x - t), (v, 1, -t) \in K_{\exp}$. The epigraph of the logarithm of a sum exponentials can modelled as

$$t \geq \log(e^{x_1} + \cdots + e^{x_1}) \quad \Longleftrightarrow \quad \sum_{i=1}^{n} u_i = 1,\ (u_i, 1, x_i - t) \in K_{\exp}, i = 1, \ldots, n,$$

These examples all have auxiliary variables and constraints in their conic representations, which might suggest that an algorithm working directly with a barrier of the convex domain (*e.g.*, [10]) is more efficient. However, a conic formulation has the advantage of the nice the conic duality. In particular the all constraints are linear except for the conic constraints. Furthermore, it is easy to exploit the special (sparse) structure created by the additional constraints and variables in the linear algebra implementation, thereby eliminating the potential overhead in a conic formulation.

3 The homogeneouous model and central path

In the simplified homogeneouos model we embed the KKT conditions for (P) and (D) into the homogeneous self-dual model

$$\begin{bmatrix} 0 & A & -b \\ -A^T & 0 & c \\ b^T & -c^T & 0 \end{bmatrix} \begin{bmatrix} y \\ x \\ \tau \end{bmatrix} - \begin{bmatrix} 0 \\ s \\ \kappa \end{bmatrix} = 0 \tag{13}$$
$$x \in K,\ s \in K^*,\ y \in \mathbb{R}^m,\ \tau, \kappa \geq 0.$$

We consider a central path as the solution to

$$\begin{bmatrix} 0 & A & -b \\ -A^T & 0 & c \\ b^T & -c^T & 0 \end{bmatrix} \begin{bmatrix} y_\mu \\ x_\mu \\ \tau_\mu \end{bmatrix} - \begin{bmatrix} 0 \\ s_\mu \\ \kappa_\mu \end{bmatrix} = \mu \begin{bmatrix} r_p \\ r_d \\ r_g \end{bmatrix} \tag{14}$$

$$s_\mu = -\mu F'(x_\mu), \quad x_\mu = -\mu F_*'(s_\mu), \quad \kappa_\mu \tau_\mu = \mu, \tag{15}$$

parametrized by μ where

$$r_p := Ax - b\tau, \qquad r_d := c\tau - A^T y - s, \qquad r_g := \kappa - c^T x + b^T y. \tag{16}$$

If we use only a primal barrier in (15) we can characterize the central path as the solution to

$$\begin{pmatrix} s + \mu F'(x) \\ \kappa - \mu \tau^{-1} \end{pmatrix} = 0.$$

This leads to the neighborhood definition used by Skajaa and Ye [26]

$$\left\| \begin{pmatrix} s + \mu F'(x) \\ \kappa - \mu \tau^{-1} \end{pmatrix} \right\|_{(x;\tau)}^* = \left(\langle s + \mu F'(x), F''(x)^{-1}(s + \mu F'(x)) \rangle + (\kappa\tau - \mu)^2 \right)^{1/2} \leq \beta\mu$$

which we can think of as a generalization of the standard two-norm neighborhood

$$\|XSe - \mu e\|_2 \leq \beta\mu$$

from linear optimization [32]. A similar definition using the conjugate barrier is also possible. An alternative characterization of the central-path due to Nesterov and Todd [21] is $\mu\tilde{\mu} \geq 1$ with equality only on the central path. This leads to a different neighborhood $\mu\tilde{\mu} \leq \rho$ for $\rho \geq 1$, or equivalently

$$\frac{\sum_{i=1}^k \tilde{x}_i^T \tilde{s}_i + (\tau\kappa)^{-1}}{\nu} \leq \rho \frac{\nu}{x^T s + \tau\kappa}$$

where $\nu = \sum_{i=1}^k \nu_i + 1$ is the accumulated barrier-parameter. This is satisfied if

$$(\tau\kappa)^{-1} \leq \rho \frac{\nu}{x^T s + \tau\kappa}, \qquad \tilde{x}_i^T \tilde{s}_i \leq \rho \frac{\nu\nu_i}{x^T s + \tau\kappa},\ i = 1, \ldots, k,$$

leading to another neighborhood definition

$$\mathcal{N}(\beta) = \left\{ (x, s, \tau, \kappa) \in K \times K^* \times \mathbb{R}_+ \times \mathbb{R}_+ \mid \tau\kappa \geq \beta\mu,\ \nu_i \langle F'(x_i), F_*'(s_i) \rangle^{-1} \geq \beta\mu,\ i = 1, \ldots, k \right\}, \tag{17}$$

for $\beta \in (0; 1]$. We use the neighborhood $\mathcal{N}(\beta)$, which can be seen as a generalization of the one-sided ∞-norm neighborhood.

A starting point on the central path can be obtained by solving

$$x^0 = s^0 = -F'(x^0),$$

which are optimality conditions for minimizing

$$f(z) := (1/2) z^T z + F(z).$$

For the exponential cone this can be solved off-line using a backtracking Newton's method to get

$$x^0 = s^0 \approx (1.290928, 0.805102, -0.827838).$$

For the symmetric cones and the three-dimensional power-cone such a central starting point can be found analytically. Hence, it is easy to choose an initial point such that

$$(x^0, s^0, \tau^0, \kappa^0) \in \mathcal{N}(1)$$

For simplicity we let $y^0 = 0$.

4 Search-directions using a primal-dual scaling

In this section we define a search direction, assuming a primal-dual scaling is known; how to compute such scalings is discussed in §5. We assume that x and s are partitioned into k cones,

$$x = (x_1, \ldots, x_k), \qquad s = (s_1, \ldots, s_k),$$

and we consider nonsingular primal-dual scalings W_i satisfying

$$v_i = W_i x_i = W_i^{-T} s_i, \qquad \tilde{v}_i = W_i \tilde{x}_i = W_i^{-T} \tilde{s}_i. \tag{18}$$

We can then express the centrality conditions

$$x_i = \mu \tilde{x}_i, \quad s_i = \mu \tilde{s}_i$$

symmetrically as

$$v_i = \mu \tilde{v}_i,$$

where $\mu := \frac{x^T s + \tau\kappa}{\nu+1}$ and $\nu = \sum_{i=1}^k \nu_i$ is the accumulated barrier parameter. This results in a linearized centrality condition $v = \mu\tilde{v}$ as

$$W_i \Delta x_i + W_i^{-T} \Delta s_i = \mu \tilde{v}_i - v_i, \quad i = 1, \ldots k$$

which is a combination of an affine and centering direction known from the NT algorithm for the symmetric cone case.

If we denote $z := (x, s, y, \tau, \kappa)$ and

$$G(z) := \begin{bmatrix} 0 & A & -b \\ -A^T & 0 & c \\ b^T & -c^T & 0 \end{bmatrix} \begin{bmatrix} y \\ x \\ \tau \end{bmatrix} - \begin{bmatrix} 0 \\ s \\ \kappa \end{bmatrix},$$

we can state the following lemmas.

Lemma 1. *If the affine search-direction Δz^a is the solution to*

$$\begin{gathered} G(\Delta z^a) = -(r_p; r_d; r_g) \\ \tau \Delta\kappa^a + \kappa \Delta\tau^a = -\kappa\tau, \qquad W_i \Delta x_i^a + W_i^{-T} \Delta s_i^a = -v_i,\ i = 1, \ldots, k \end{gathered} \tag{19}$$

then

$$(\Delta x^a)^T \Delta s^a + \Delta\tau^a \Delta\kappa^a = 0. \tag{20}$$

holds.

Proof. We rewrite (19) as $G(z + \Delta z^{\mathrm{a}}) = 0$ and skew-symmetry implies that

$$-\begin{bmatrix} y + \Delta y^{\mathrm{a}} \\ x + \Delta x^{\mathrm{a}} \\ \tau + \Delta\tau^{\mathrm{a}} \end{bmatrix}^T G(z + \Delta z^{\mathrm{a}}) = (x + \Delta x^{\mathrm{a}})^T (s + \Delta s^{\mathrm{a}}) + (\tau + \Delta\tau^{\mathrm{a}})(\kappa + \Delta\kappa^{\mathrm{a}}) = 0. \tag{21}$$

Using (18) and (19) we have that

$$s^T \Delta x^{\mathrm{a}} + x^T \Delta s^{\mathrm{a}} + \tau \Delta\kappa^{\mathrm{a}} + \kappa \Delta\tau^{\mathrm{a}} = -(x^T s + \tau\kappa).$$

This combined with (21) shows (20). □

We also see that if $z + \Delta z^{\mathrm{a}}$ is feasible, then it is also optimal, *i.e.*, a full affine step is optimal.

Lemma 2. *If the centering search-direction Δz^c is the solution to*

$$\begin{gathered} G(\Delta z^c) = (r_p; r_d; r_g) \\ \tau \Delta\kappa^c + \kappa \Delta\tau^c = \mu, \qquad W_i \Delta x_i^c + W_i^{-T} \Delta s_i^c = \mu \tilde{v}_i,\ i = 1, \ldots, k \end{gathered} \tag{22}$$

then

$$(\Delta x^c)^T \Delta s^c + \Delta\tau^c \Delta\kappa^c = 0 \tag{23}$$

holds.

Proof. We rewrite (22) as $G(-z+\Delta z^{\mathrm{c}})=0$ and skew-symmetry implies that

$$-\begin{bmatrix} -y+\Delta y^{\mathrm{c}} \\ -x+\Delta x^{\mathrm{c}} \\ -\tau+\Delta\tau^{\mathrm{c}} \end{bmatrix}^T G(-z+\Delta z^{\mathrm{c}}) = (-x+\Delta x^{\mathrm{c}})^T(-s+\Delta s^{\mathrm{c}})+(-\tau+\Delta\tau^{\mathrm{c}})(-\kappa+\Delta\kappa^{\mathrm{c}}) = 0. \quad (24)$$

Using (18) and (22) we have that

$$s^T\Delta x^{\mathrm{c}} + x^T\Delta s^{\mathrm{c}} + \tau\Delta\kappa^{\mathrm{c}} + \kappa\Delta\tau^{\mathrm{c}} = x^Ts+\tau\kappa$$

together with (24) implies (23). □

As a consequence we have following, which can be proved similarly.

Lemma 3. *Let*

$$\Delta z = \Delta z^a + \gamma\Delta z^c$$

for $\gamma\in[0,1]$. *Then*

$$G(z+\alpha\Delta z) = (1-\alpha(1-\gamma))(r_p;r_d;r_g) \quad (25)$$

$$(x+\alpha\Delta x)^T(s+\alpha\Delta s)+(\tau+\alpha\Delta\tau)(\kappa+\alpha\Delta\kappa) = (1-\alpha(1-\gamma))(x^Ts+\tau\kappa). \quad (26)$$

An immediate consequence is that for a given step-size $\alpha\in(0,1]$ the residuals and the complementarity gap $x^Ts+\tau\kappa$ decrease at the same rate. Hence, convergence follows by proving γ and α can be chosen suffciently positive in every iteration. This is in contrast with [20, 26, 25], where the complementarity gap is not reduced at the same rate. Also, no explicit merit function as in [9] is required.

5 Primal-dual scalings

We next discuss how to compute primal-dual scaling proposed by Tunçel. Symmetric cones have the important notion of a scaling point [21]. For any $x\in\mathbf{int}(K)$ and $s\in\mathbf{int}(K^*)$ there exists a unique $w\in\mathbf{int}(K)$ such that

$$s = F''(w)x.$$

The scaling $F''(w)$ further satisfies

$$F'(x) = F''(w)F'_*(s), \quad F''(x) = F''(w)F''_*(s)F''(w).$$

and is bounded by

$$\frac{1}{\sigma_x(-F'_*(s))}F''(x) \preceq F''(w) \preceq \sigma_x(-F'_*(s))\left[F''_*(s)\right]^{-1}.$$

In a similar vain [29, 15] consider a general scaling $T^2\succ 0$ for nonsymmetric cones satisfying

$$T^2s = x, \qquad T^2(-F'(x)) = -F'_*(s). \quad (27)$$

If the scaling T^2 is also bounded as

$$\frac{\mu}{\xi^\star[\nu(\mu\tilde{\mu}-1)+1]}F''_*(s) \preceq T^2 \preceq \frac{\xi^\star[\nu(\mu\tilde{\mu}-1)+1]}{\mu}\left[F''(x)\right]^{-1} \quad (28)$$

with $\xi^\star=\mathcal{O}(1)$ we have an immediate iteration complexity bound $\mathcal{O}(\sqrt{\nu}\log(1/\epsilon))$. As for self-scaled cones, this is satisfied if $F'''(x)[u]\preceq 0$ for all $x\in\mathbf{int}(K)$ and for $u\in K$, in which case $F'''(x)$ is said to have *negative curvature*. The barrier for the exponential cone does not have negative curvature, however, as can be seen be considering $\hat{x}:=(1,e^{-2},0)\in K_{\exp}$ and $\hat{u}:=(1,0,0)\in K_{\exp}\setminus\mathbf{int}(K_{\exp})$. Then it can be verified using the expressions in the appendix that

$$F'''(\hat{x})[\hat{u}] = \begin{bmatrix} -4 & \frac{e^2}{2} & \frac{e^2}{2} \\ \frac{e^2}{2} & 0 & 0 \\ \frac{e^2}{2} & 0 & -\frac{e^4}{4} \end{bmatrix},$$

which is indefinite, for example

$$F'''(\hat{x})[\hat{h},\hat{v},\hat{v}] = 4$$

for $\hat{v} := (1, 8e^{-2}, 4e^{-2})$.

Scalings such as (27) have been studied in the context of quasi-Newton methods with multiple secant equations. The following theorem by Schnabel [24] is used repeatedly in the following discussion.

Theorem 1. *Let $S, Y \in \mathbb{R}^{n\times p}$ have full rank p. Then there exists $H \succ 0$ such that $HS = Y$ if and only if $Y^T S \succ 0$.*

As a consequence we can write any such $H \succ 0$ as

$$H = Y(Y^T S)^{-1}Y^T + ZZ^T, \quad S^T Z = 0, \quad \mathbf{rank}(Z) = n - p \tag{29}$$

or in factored form $H = W^T W$ with

$$W = \begin{bmatrix} Y(Y^T S)^{-1/2} & Z \end{bmatrix}^T. \tag{30}$$

Given any $\Omega \succ 0$ satisfying $\Omega S = Y$ we have

$$W^{-1} = \begin{bmatrix} S(Y^T S)^{-1/2} & \Omega^{-1}Z(Z^T\Omega^{-1}Z)^{-1} \end{bmatrix}, \tag{31}$$

and therefore

$$H^{-1} = S(Y^T S)^{-1}S^T + VV^T, \quad Y^T V = 0, \quad \mathbf{rank}(V) = n - p, \tag{32}$$

where $VV^T = \Omega^{-1}Z(Z^T\Omega^{-1}Z)^{-2}Z^T\Omega^{-1}$. The two most popular quasi-Newton update rules are the David-Fletcher-Powell (DFP) and Broyden-Fletcher-Goldfarb-Shanno (BFGS) steps summarized below, where $H \succ 0$ denotes an approximation of the Hessian.

$$H_{\mathrm{DFP}} := Y(Y^T S)^{-1}Y^T + (I - Y(Y^T S)^{-1}S^T)H(I - S(Y^T S)^{-1}Y^T) \tag{33}$$

$$H_{\mathrm{BFGS}} := Y(Y^T S)^{-1}Y^T + H - HS(S^T HS)^{-1}S^T H \tag{34}$$

$$H^{-1}_{\mathrm{DFP}} := S(Y^T S)^{-1}S^T + H^{-1} - H^{-1}Y(Y^T H^{-1}Y)^{-1}Y^T H^{-1} \tag{35}$$

$$H^{-1}_{\mathrm{BFGS}} := S(Y^T S)^{-1}S^T + (I - S(Y^T S)^{-1}Y^T)H^{-1}(I - Y(Y^T S)^{-1}S^T) \tag{36}$$

There is an evident symmetry between (33) and (36), if we invert H and reverse Y and S; the DFP update (33) is the solution to

$$\begin{array}{ll} \text{minimize} & \|\Omega^{-1/2}(H_+ - H)\Omega^{-1/2}\|_F \\ \text{subject to} & H_+ S = Y, \\ & H_+ \succ 0 \end{array} \tag{37}$$

and the BFGS update (36) is the solution to

$$\begin{array}{ll} \text{minimize} & \|\Omega^{1/2}(H_+^{-1} - H^{-1})\Omega^{1/2}\|_F \\ \text{subject to} & H_+^{-1} Y = S, \\ & H_+^{-1} \succ 0 \end{array} \tag{38}$$

for any $\Omega \succ 0$ satisfying $\Omega S = Y$. In the following theorem we show how these formulas can be computed using a sequence of updates, similar to quasi-Newton updates with a single secant equation.

Theorem 2. *Given $Y_0, S_0 \in \mathbb{R}^{n\times p}$ with $Y_0^T S_0 \succ 0$. Then*

$$Y_0(Y_0^T S_0)^{-1}Y_0^T = VV^T \tag{39}$$

$$S_0(Y_0^T S_0)^{-1}S_0^T = UU^T, \tag{40}$$

where $V := \begin{pmatrix} v_1 & \cdots & v_p \end{pmatrix}$, $U := \begin{pmatrix} u_1 & \cdots & u_p \end{pmatrix}$ and

$$v_k := \frac{1}{(e_k^T Y_{k-1}^T S_{k-1} e_k)^{1/2}} Y_{k-1} e_k, \qquad Y_k := Y_{k-1} - v_k v_k^T S_{k-1} \tag{41}$$

$$u_k := \frac{1}{(e_k^T Y_{k-1}^T S_{k-1} e_k)^{1/2}} S_{k-1} e_k, \qquad S_k := S_{k-1} - u_k u_k^T Y_{k-1} \tag{42}$$

for $k = 1, \ldots, p$.

Proof. We first note that $Y_k e_j = S_k e_j = 0$ for $k = 1, \ldots, p$, $j \leq k$ which simplifies notation and the proof. Let $\Psi_k := (Y_k^T S_k)_{k+1:p,k+1:p}$, where $\Psi_0 = Y_0^T S_0$ is positive definite by assumption. Expanding $Y_k^T S_k$ we have

$$Y_k^T S_k = Y_{k-1}^T S_{k-1} - \frac{1}{e_k^T Y_{k-1}^T S_{k-1} e_k} (Y_{k-1}^T S_{k-1} e_k)(Y_{k-1}^T S_{k-1} e_k)^T, \tag{43}$$

i.e., Ψ_k is the Schur-complement of $(\Psi_{k-1})_{1,1}$ and therefore positive definite, so the recursion is well-defined. From (43) we have a Cholesky factorization $Y_0^T S_0 = LL^T$ with

$$L = \begin{pmatrix} (e_1^T Y_0^T S_0 e_1)^{-1/2} Y_0^T S_0 e_1 & \cdots & (e_p^T Y_{p-1}^T S_{p-1} e_p)^{-1/2} Y_{p-1}^T S_{p-1} e_p \end{pmatrix}.$$

Then

$$\begin{aligned} LV^T &= Y_0^T - Y_0^T + \frac{S_0^T Y_0 e_1 (Y_0 e_1)^T}{e_1^T Y_0^T S_0 e_1} + \frac{S_1^T Y_1 e_2 (Y_1 e_2)^T}{e_2^T Y_1^T S_1 e_2} + \cdots + \frac{S_{p-1}^T Y_{p-1} e_p (Y_{p-1} e_p)^T}{e_p^T Y_{p-1}^T S_{p-1} e_p} \\ &= Y_0^T - Y_1^T + \frac{S_1^T Y_1 e_2 (Y_1 e_2)^T}{e_2^T Y_1^T S_1 e_2} + \cdots + \frac{S_{p-1}^T Y_{p-1} e_p (Y_{p-1} e_p)^T}{e_p^T Y_{p-1}^T S_{p-1} e_p} \\ &= Y_0^T - Y_p^T = Y_0^T, \end{aligned}$$

i.e., $L^{-1} Y^T = V^T$, which proves (39). Equation (40) follows similarly. □

Myklebust and Tunçel [15] consider such quasi-Newton updates to $\mu F_*''((s + \mu\tilde{s})/2)$ satisfying (27) and give bounds on the difference between T^2 and $\mu F_*''(s)$ near the central path. We are going to derive a factorization of the inverse scaling $W^T W = T^{-2}$ obtained from low-rank updates to $\mu F''(x)$ instead, *i.e.*, we have

$$W^{-T} s = Wx, \qquad -W^{-T} F'(x) = -W F_*'(s), \tag{44}$$

and although there is a symmetry between the pairs (33) and (36) (and similarly between (34) and (35)), that symmetry is lost when we compute low-rank updates to $\mu F''(x)$ rather than $(\mu F_*''(s))^{-1}$ or $(\mu F_*''((s + \mu\tilde{s})/2))^{-1}$. In the context of the scaling (44) we have

$$S := \begin{bmatrix} x & \tilde{x} \end{bmatrix}, \quad Y := \begin{bmatrix} s & \tilde{s} \end{bmatrix},$$

where we assume that $Y^T S \succ 0$. Let $\delta_x := x - \mu\tilde{x}$, $\delta_s := s - \mu\tilde{s}$. Using theorem 2 with $S_0 := S$, $Y_0 := S$ we have

$$U := \begin{bmatrix} (x^T s)^{-1/2} x & (\delta_x^T \delta_s)^{-1/2} \delta_x \end{bmatrix}, \qquad V := \begin{bmatrix} (x^T s)^{-1/2} s & (\delta_x^T \delta_s)^{-1/2} \delta_s \end{bmatrix}$$

and thus

$$\begin{aligned} Y(S^T Y)^{-1} Y^T = VV^T &= \frac{1}{x^T s} ss^T + \frac{1}{\delta_x^T \delta_s} \delta_s \delta_s^T, \\ S(S^T Y)^{-1} S^T = UU^T &= \frac{1}{x^T s} xx^T + \frac{1}{\delta_x^T \delta_s} \delta_x \delta_x^T. \end{aligned}$$

We consider only the BFGS update in the following. The update (36) can be computed as

$$H_{\text{BFGS}}^{-1} = UU^T + (I - UV^T) H^{-1} (I - UV^T)^T \tag{45}$$

$$= \frac{1}{x^T s} xx^T + \frac{1}{\delta_x^T \delta_s} \delta_x \delta_x^T + (I - \frac{1}{x^T s} xs^T - \frac{1}{\delta_x^T \delta_s} \delta_x \delta_s^T) H^{-1} (I - \frac{1}{x^T s} xs^T - \frac{1}{\delta_x^T \delta_s} \delta_x \delta_s^T)^T. \tag{46}$$

To compute (34) we use theorem 2 with $S_0 := S$ and $Y_0 := HS$ to get

$$U := \begin{bmatrix} (x^T H x)^{-1/2} x & (\rho_x^T H \rho_x)^{-1/2} \rho_x \end{bmatrix}, \qquad V := \begin{bmatrix} (x^T H x)^{-1/2} Hx & (\rho_x^T H \rho_x)^{-1/2} H\rho_x \end{bmatrix}$$

where $\rho_x := \tilde{x} - \frac{x^T H \tilde{x}}{x^T H x} x$, and thus

$$HS(S^T HS)^{-1} S^T H = VV^T = \frac{1}{x^T H x} Hx(Hx)^T + \frac{1}{\rho_x^T H \rho_x} H\rho_x (H\rho_x)^T,$$

i.e.,

$$H_{\text{BFGS}} = H + \frac{1}{x^T s} ss^T + \frac{1}{\delta_x^T \delta_s} \delta_s \delta_s^T - \frac{1}{x^T Hx} Hx(Hx)^T - \frac{1}{\rho_x^T H \rho_x} H\rho_x (H\rho_x)^T \tag{47}$$

It is shown in [15] how (47) can be computed alternatively as

$$\begin{aligned} H_1 &:= H + \frac{1}{x^T s} ss^T - \frac{1}{x^T Hx} Hx(Hx)^T \\ H_2 &:= H_1 + \frac{1}{\delta_x^T \delta_s} \delta_s \delta_s^T - \frac{1}{\delta_x^T H_1 \delta_x} H_1 \delta_x (H_1 \delta_x)^T. \end{aligned}$$

To see the equivalence $H_2 = H_{\text{BFGS}}$ we note that $s^T \delta_x = 0$ so

$$H_1 \delta_x = \left(H - \frac{1}{x^T Hx} Hx(Hx)^T \right) \delta_x = -\mu \left(H - \frac{1}{x^T Hx} Hx(Hx)^T \right) \tilde{x} = -\mu H \rho_x.$$

Similarly,

$$\delta_x^T H_1 \delta_x = \mu^2 \tilde{x}^T \left(H - \frac{1}{x^T Hx} Hx(Hx)^T \right) \tilde{x} = \mu^2 \tilde{x}^T H \rho_x = \mu^2 \rho_x^T H \rho_x$$

where we used that $x^T H \rho_x = 0$ in the last equality, which shows the equivalence.

To derive a factorization of the scaling, let $LL^T = H$. Then

$$\begin{aligned} H_{\text{BFGS}} &= \frac{1}{x^T s} ss^T + \frac{1}{\delta_x^T \delta_s} \delta_s \delta_s^T + L \left(I - \frac{L^T x (L^T x)^T}{\|L^T x\|^2} - \frac{L^T \rho_x (L^T \rho_x)^T}{\|L^T \rho_x\|^2} \right) L^T \\ &= \frac{1}{x^T s} ss^T + \frac{1}{\delta_x^T \delta_s} \delta_s \delta_s^T + ZZ^T, \end{aligned}$$

i.e., we have a spectral factorization

$$(L^{-1}Z)(L^{-1}Z)^T = I - \frac{L^T x (L^T x)^T}{\|L^T x\|^2} - \frac{L^T \rho_x (L^T \rho_x)^T}{\|L^T \rho_x\|^2}$$

where $(L^T x) \perp (L^T \rho_x)$. For the special case $n = 3$ we have

$$L^{-1} z (L^{-1} z)^T v = v - \frac{L^T x (L^T x)^T}{\|L^T x\|^2} v - \frac{L^T \rho_x (L^T \rho_x)^T}{\|L^T \rho_x\|^2} v$$

for any $v \in \mathbb{R}^3$. We then obtain the following factor of the scaling

$$W = \begin{bmatrix} (x^T s)^{-1/2} s & (\delta_x^T \delta_s)^{-1/2} \delta_s & z \end{bmatrix}^T, \tag{48}$$

satisfying $W^T W S = Y$. For the inverse factor we consider $v = \Omega^{-1} z / (z^T \Omega^{-1} z)$ for $\Omega \succ 0$ satisfying $\Omega S = Y$. In particular, for $\Omega := H_{\text{BFGS}}$ we have from (46) that

$$v := \frac{\Omega^{-1} z}{z^T \Omega^{-1} z} = \left(I - \frac{xs^T}{x^T s} - \frac{\delta_x \delta_s^T}{\delta_x^T \delta_s} \right) H^{-1} z \tag{49}$$

since $x^T z = \delta_x^T z = 0$ and $\|L^{-1} z\| = 1$, resulting in an expression for the inverse factor

$$W^{-1} = \begin{bmatrix} (x^T s)^{-1/2} x & (\delta_x^T \delta_s)^{-1/2} \delta_x & v \end{bmatrix}. \tag{50}$$

Considering (38) we see that the BFGS update to $H := \mu F''(x)$ has the desirable property of minimizing the right-hand bound in (28)

$$\|W^{-1} W^{-T} - (\mu F''(x))^{-1}\|_\Omega,$$

measured in a weighted norm; for simplicity we can assume that $\Omega = W^T W$.

6 A higher-order corrector

The performance of practical interior-point implementations is often improved by a higher-order corrector, *e.g.*, a Mehrotra corrector [13]. Such a corrector is not easy to generalize from the symmetric cone case to the nonsymmetric case, and currently only limited progress has been made in that regard. In the following we propose a new corrector direction that inherits many of the desirable properties of a Mehrotra corrector for a symmetric path-following scheme. The new corrector is shown numerically to result in a consistent and significant reduction in the iteration count, and we believe that this corrector will be of interest for other conic and non-conic algorithms as well.

To derive a higher-order corrrector we consider the first- and second-order derivatives of $s_\mu = -\mu F'(x_\mu)$ with respect to μ,

$$\dot{s}_\mu + \mu F''(x_\mu)\dot{x}_\mu = -F'(x_\mu), \tag{51}$$
$$\ddot{s}_\mu + \mu F''(x_\mu)\ddot{x}_\mu = -2F''(x_\mu)\dot{x}_\mu - \mu F'''(x_\mu)[\dot{x}_\mu, \dot{x}_\mu]. \tag{52}$$

From (51) we have

$$\mu\dot{x}_\mu = -[F''(x_\mu)]^{-1}(F'(x_\mu) + \dot{s}_\mu) = x_\mu - [F''(x_\mu)]^{-1}\dot{s}_\mu,$$

resulting in an expression for the third-order directional derivative,

$$\mu F'''(x_\mu)[\dot{x}_\mu, \dot{x}_\mu] = F'''(x_\mu)[\dot{x}_\mu, x_\mu] - F'''(x_\mu)[\dot{x}_\mu, (F''(x_\mu))^{-1}\dot{s}_\mu] \tag{53}$$
$$= -2F''(x_\mu)\dot{x}_\mu - F'''(x_\mu)[\dot{x}_\mu, (F''(x_\mu))^{-1}\dot{s}_\mu] \tag{54}$$

where (54) follows from the homogeneity property $F'''(x)[x] = -2F''(x)$. This results in an alternative expression for the second-order derivative of the centrality condition, *i.e.*,

$$\ddot{s}_\mu + \mu F''(x_\mu)\ddot{x}_\mu = F'''(x_\mu)[\dot{x}_\mu, (F''(x_\mu))^{-1}\dot{s}_\mu]. \tag{55}$$

Comparing (19) and (51) we interpret $\Delta s^{\mathrm{a}} = -\dot{s}_\mu$ and $\Delta x^{\mathrm{a}} = -\dot{x}_\mu$ since $W^TW \approx \mu F''(x)$, and we define a corrector search-direction from

$$\begin{aligned} G(\Delta z^{\mathrm{cor}}) &= 0, \\ W_i\Delta x_i^{\mathrm{cor}} + W_i^{-T}\Delta s_i^{\mathrm{cor}} &= -\eta_i, \quad i = 1\ldots, k, \\ \tau\Delta\kappa^{\mathrm{cor}} + \kappa\Delta\tau^{\mathrm{cor}} &= -\Delta\tau^{\mathrm{a}}\Delta\kappa^{\mathrm{a}}, \end{aligned} \tag{56}$$

where $\eta_i := -\frac{1}{2}W_i^{-T}F_i'''(x_i)[\Delta x_i^{\mathrm{a}}, (F_i''(x_i))^{-1}\Delta s_i^{\mathrm{a}}]$. Using the homogeneity properties (2) we have

$$s^T\Delta x^{\mathrm{cor}} + x^T\Delta s^{\mathrm{cor}} = \frac{1}{2}\sum_i x_i^T F_i'''(x)[\Delta x_i^{\mathrm{a}}, (F_i''(x_i))^{-1}\Delta s_i^{\mathrm{a}}] = -(\Delta x^{\mathrm{a}})^T\Delta s^{\mathrm{a}}$$

so

$$s^T\Delta x^{\mathrm{cor}} + x^T\Delta s^{\mathrm{cor}} + \kappa\Delta\tau^{\mathrm{cor}} + \tau\Delta\kappa^{\mathrm{cor}} = -(\Delta x^{\mathrm{a}})^T\Delta s^{\mathrm{a}} - \Delta\tau^{\mathrm{a}}\Delta\kappa^{\mathrm{a}} = 0,$$

which combined with

$$-\begin{bmatrix} \Delta y^{\mathrm{cor}} \\ \Delta x^{\mathrm{cor}} \\ \Delta\tau^{\mathrm{cor}} \end{bmatrix}^T G(\Delta z^{\mathrm{cor}}) = (\Delta x^{\mathrm{cor}})^T\Delta s^{\mathrm{cor}} + \Delta\tau^{\mathrm{cor}}\Delta\kappa^{\mathrm{cor}} = 0$$

shows that

$$(x + \alpha\Delta x^{\mathrm{cor}})^T(s + \alpha\Delta s^{\mathrm{cor}}) + (\tau + \alpha\Delta\tau^{\mathrm{cor}})(\kappa + \alpha\Delta\kappa^{\mathrm{cor}}) = x^Ts + \tau\kappa.$$

We then define a combined centering-corrector direction $\Delta z := \Delta z^{\mathrm{c}} + \Delta z^{\mathrm{cor}}$, *i.e.*,

$$\begin{aligned} G(\Delta z) &= (\gamma - 1)(r_p; r_d; r_g) \\ W_i\Delta x_i + W_i^{-T}\Delta s_i &= \gamma\mu\tilde{v}_i - v_i - \eta_i, \quad i = 1, \ldots, k \\ \tau\Delta\kappa + \kappa\Delta\tau &= \gamma\mu - \kappa\tau - \Delta\tau^{\mathrm{a}}\Delta\kappa^{\mathrm{a}}. \end{aligned} \tag{57}$$

The search-direction (57) is the one used in our algorithm, and preserves the property that all residuals also decrease at the same rate,

$$G(z + \alpha\Delta z) = (1 - \alpha(1-\gamma))(r_p; r_d; r_g)$$
$$(x + \alpha\Delta x)^T(s + \alpha\Delta s) + (\tau + \alpha\Delta\tau)(\kappa + \alpha\Delta\kappa) = (1 - \alpha(1-\gamma))(x^T s + \tau\kappa).$$

To gain some insight into the corrector η we note that in the case of the nonnegative orthant we have the familiar expression

$$\frac{1}{2}F'''(x)[\Delta x^{\mathrm{a}}, (F''(x))^{-1}\Delta s^{\mathrm{a}}] = -\mathbf{diag}(x)^{-1}\mathbf{diag}(\Delta x^{\mathrm{a}})\Delta s^{\mathrm{a}},$$

and similarly for the semidefinite cone we have

$$\frac{1}{2}F'''(x)[\Delta x^{\mathrm{a}}, (F''(x))^{-1}\Delta s^{\mathrm{a}}] = -\frac{1}{2}x^{-1}\Delta x^{\mathrm{a}}\Delta s^{\mathrm{a}} - \frac{1}{2}\Delta s^{\mathrm{a}}\Delta x^{\mathrm{a}}x^{-1} = -(x^{-1}) \circ (\Delta x^{\mathrm{a}}\Delta s^{\mathrm{a}}),$$

using the generalized product associated with the Euclidean Jordan algebra, see, *e.g.*, [30] for a discussion of Jordan algebra in a similar context as ours. For the Lorentz cone we have

$$F'''(x)[(F''(x))^{-1}u] = -\frac{2}{x^T Q x}(ux^T Q + Qxu^T - (x^T u)Q)$$

with $Q = \mathbf{diag}(1, -1, \ldots, -1)$, so that

$$F'''(x)[(F''(x))^{-1}u]e = -2(x^{-1}) \circ u,$$

again using the notation of the generalized product [30]. We defer the derivation and implementation specific details for the exponential cone to the appendix.

7 A primal-dual algorithm for nonsymmetric conic optimization

In this section we give a collected overview of the suggested path-following primal-dual algorithm. The different essential parts of the method are *i*) finding a starting point, *ii*) computing a search-direction and step-size, and *iii*) checking the stopping criteria for termination.

- *Starting point.* Find a starting point on the central path

  $$x_i^0 = s_i^0 = -F_i'(x_i), \quad i = 1, \ldots, k$$

  and $y^0 = 0$, $\tau^0 = \kappa^0 = 1$. Then $z^0 := (x^0, s^0, y^0, \tau^0, \kappa^0) \in \mathcal{N}(1)$.

- *Search-direction and step-size.* At each iteration we compute an affine direction Δz^{a} as the solution to (19). From Δz^{a} we compute correctors η_i, and similar to Mehrotra [13] we define a centering parameter γ as

  $$\gamma := (1 - \alpha_{\mathrm{a}}) \min\{(1 - \alpha_{\mathrm{a}})^2, 1/4\}$$

  where α_{a} is the stepsize to the boundary, *i.e.*,

  $$\alpha_{\mathrm{a}} = \sup\{\alpha \mid (x + \alpha\Delta x^{\mathrm{a}}) \in K,\ (s + \alpha\Delta s^{\mathrm{a}}) \in K^*,\ (\tau + \alpha\Delta\tau^{\mathrm{a}}), (\kappa + \alpha\Delta\kappa^{\mathrm{a}}) \geq 0\}$$

  which we approximate using a bisection procedure. We then compute a combined centering-corrector search direction Δz as the solution to (57), and we update $z := z + \alpha\Delta z$ with the largest step inside a neighborhood $\mathcal{N}(\beta)$ of the central path.

- *Checking termination.* Terminate if the updated iterate satisfies the termination criteria or else take a new step.

8 Implementation

MOSEK is software package for solving large scale linear and conic optimization problem. It can solve problems with a mixture of linear, quadratic and semidefinite cones, and the implementation is based on the homogeneous model, the NT search direction and a Mehrotra like predictor-corrector algorithm [2].

Our implementation has been extended to handle the three dimensional exponential cone using the algorithm above. We use the usual NT scaling for the symmetric cones and the Tunçel scalings for the nonsymmetric cones. Except for small differences in the linearization of the complementarity conditions, the symmetric and nonsymmmetric cones are handled completely analogously. Our extension for nonsymmetric cones also includes the three dimensional power cone, but this is not discussed further here.

8.1 Dualization, presolve and scaling

Occasionally it is worthwhile to dualize the problem before solving it, since it will make the linear algebra more efficient. Whether the primal or dual formulation is more efficient is not easily determined in advance. MOSEK makes a heuristic choice between the two forms, and the dualization is transparent to the user.

Furthermore, a presolve step is applied to the problem, which often leads to a significant reduction in computational complexity [1]. The presolve step removes obviously redundant constraints, tries to remove linear dependencies, *etc.* Finally, many optimization problems are badly scaled, so MOSEK rescales the problem before solving it. The rescaling is very simple, essentially normalizing the rows and columns of the A.

8.2 Computing the search direction

Usually the most expensive operation in each iteration of the primal-dual algorithm is to compute the search direction, *i.e.*, solving the linear system

$$\begin{bmatrix} 0 & A & -b \\ -A^T & 0 & c \\ b^T & -c^T & 0 \end{bmatrix} \begin{bmatrix} \Delta y \\ \Delta x \\ \Delta\tau \end{bmatrix} - \begin{bmatrix} 0 \\ \Delta s \\ \Delta\kappa \end{bmatrix} = \begin{bmatrix} r_p \\ r_d \\ r_g \end{bmatrix}$$
$$W\Delta x + W^{-T}\Delta s = r_{xs}, \qquad \tau\Delta\kappa + \kappa\Delta\tau = r_{\tau\kappa},$$

where W is block-diagonal scaling matrix for a product of cones. Eliminating Δs and $\Delta\kappa$ from the linearized centrality conditions results in the reduced bordered system

$$\begin{bmatrix} W^TW & -A^T & c \\ A & 0 & -b \\ -c^T & b^T & \tau^{-1}\kappa \end{bmatrix} \begin{bmatrix} \Delta x \\ \Delta y \\ \Delta\tau \end{bmatrix} = \begin{bmatrix} r_d + W^T r_{xs} \\ r_p \\ r_g + \tau^{-1} r_{\tau\kappa} \end{bmatrix}, \tag{58}$$

which can be solved in different ways. Given a (sparse) LDL^T factorization of the symmetric matrix

$$\begin{bmatrix} -W^TW & A^T \\ A & 0 \end{bmatrix}$$

it is computationally cheap to compute the search direction. In the case the factorization breaks down due to numerical issues we add regularization to the system, *i.e.*, we modify the diagonal, which is common in interior-point methods. If the resulting search direction is inaccurate (*i.e.*, the residuals are not decreased sufficiently) we use iterative refinement, which in most cases improves the accuracy of the search direction. We omit details of computing the LDT^T factorization, since it is fairly conventional and close to the approach discussed in [2].

Finally, MOSEK and hence the algorithm is implemented in the C programming language. Moreover, the Intel MKL BLAS library is employed for small dense matrix operations; the remaining portions of the code are developed internally, and the most computationally expensive parts have been parallelized.

8.3 The termination criteria

We next discuss the termination criteria employed in MOSEK. Let $(\varepsilon_p, \varepsilon_d, \varepsilon_g, \varepsilon_i) > 0$ be given tolerance levels for the algorithm, and denote by $(x^k, y^k, s^k, \tau^k, \kappa^k)$ the kth interior-point iterate which satisfies

$$(x^k, s^k) \in K \times K^*, \quad (\tau^k, \kappa^k) > 0.$$

Consider the metrics

$$\rho_p^k := \min\left\{\rho \mid \left\|A\frac{x^k}{\tau^k} - b\right\|_\infty \leq \rho\varepsilon_p(1 + \|b\|_\infty)\right\},$$

$$\rho_d^k := \min\left\{\rho \mid \left\|A^T\frac{y^k}{\tau^k} + \frac{s^k}{\tau^k} - c\right\|_\infty \leq \rho\varepsilon_d(1 + \|c\|_\infty)\right\},$$

and

$$\rho_g^k := \min\left\{\rho \mid \min\left(\frac{(x^k)^T s^k}{(\tau^k)^2}, |\frac{c^T x^k}{\tau^k} - \frac{b^T y^k}{\tau^k}|\right) \leq \rho\varepsilon_g \max\left(1, \frac{\min\left(|c^T x^k|, |b^T y^k|\right)}{\tau^k}\right)\right\}.$$

If

$$\max(\rho_p^k, \rho_d^k, \rho_g^k) \leq 1 \tag{59}$$

then

$$\begin{aligned}
\left\|A\frac{x^k}{\tau^k} - b\right\|_\infty &\leq \varepsilon_p(1 + \|b\|_\infty),\\
\left\|A^T\frac{y^k}{\tau^k} + \frac{s^k}{\tau^k} - c\right\|_\infty &\leq \varepsilon_d(1 + \|c\|_\infty),\\
\min\left(\frac{(x^k)^T s^k}{(\tau^k)^2}, |\frac{c^T x^k}{\tau^k} - \frac{b^T y^k}{\tau^k}|\right) &\leq \varepsilon_g \max\left(1, \frac{\min\left(|c^T x^k|, |b^T y^k|\right)}{\tau^k}\right),
\end{aligned}$$

and hence $(x^k, y^k, s^k)/\tau^k$ is an almost primal and dual feasible solution with small duality gap. Clearly, how good the approximation is dependent on the problem and the prespecified tolerances $(\varepsilon_p, \varepsilon_d, \varepsilon_g, \varepsilon_i)$. Therefore, ρ_p^k and ρ_d^k measure how far the kth iterate is from being approximately primal and dual feasible, respectively. Furthermore, ρ_g^k measure how far the kth iterate is from having a zero duality gap.

Similarly, define infeasibility metrics

$$\rho_{pi}^k := \min\left\{\rho \mid \left\|A^T y^k + s^k\right\|_\infty \leq \rho\varepsilon_i b^T y^k,\ b^T y^k > 0\right\},$$

$$\rho_{di}^k := \min\left\{\rho \mid \left\|Ax^k\right\|_\infty \leq -\rho\varepsilon_i c^T x^k,\ c^T x^k < 0\right\},$$

and

$$\rho_{ip}^k := \min\left\{\rho \mid \left\| \begin{array}{c} A^T y^k + s^k \\ Ax^k \end{array} \right\|_\infty \leq \rho\varepsilon_i \left\| \begin{array}{c} y^k \\ s^k \\ x^k \end{array} \right\|_\infty, \left\| \begin{array}{c} y^k \\ s^k \\ x^k \end{array} \right\|_\infty > 0\right\}.$$

If $\rho_{pi} \leq 1$ then

$$\left\|A^T y^k + s^k\right\|_\infty \leq \varepsilon_i b^T y^k, \quad b^T y^k > 0.$$

Thus, for

$$\bar{y} := \frac{y^k}{b^T y^k}, \quad \bar{s} := \frac{s^k}{b^T y^k}$$

we have

$$b^T \bar{y} \geq 1, \quad \left\|A^T \bar{y} + \bar{s}\right\| \leq \varepsilon_i, \quad \bar{s} \in K^*$$

i.e., $(\bar{y}, \bar{s})$ is an approximate certificate of primal infeasibility. Similarly, if $\rho_{di} \leq 1$ we have an approximate certificate of dual infeasibility. Finally, assume that $\rho_{ip} \leq 1$. Then

$$\left\| \begin{array}{c} A^T y^k + s^k \\ Ax^k \end{array} \right\|_\infty \leq \varepsilon_i \left\| \begin{array}{c} y^k \\ s^k \\ x^k \end{array} \right\|_\infty, \left\| \begin{array}{c} y^k \\ s^k \\ x^k \end{array} \right\|_\infty > 0$$

is an approximate certificate of ill-posedness. For example, if $\left\|y^k\right\|_\infty \gg 0$ then a tiny perturbation in b will make the problem infeasible. Hence, the problem is by definition unstable.

9 Numerical results

We investigate the numerical performance of our implementation on a selection of exponential-cone problems from the Conic Benchmark Library (CBLIB) [28]. Some of those problems have integer variables, in which case we solve their continuous relaxations, *i.e.*, we ignore the integrality constraints. In the study we compare the performance of the MOSEK solver both with and without the proposed corrector. In the case without a corrector we also disable the standard Mehrotra corrector effecting linear and quadratic cones; otherwise the residuals will not decrease at the same rate. We also compare our implementation with the open-source solver ECOS [6], which implements the algorithm by Serrano [25]. Since ECOS only supports linear, quadratic and exponential cones we limit the test-set to examples with combinations of those cones; there are also instances in CBLIB with both exponential and semidefinite cones.

Figure 1 shows a summary of the number of iterations needed for solving the collection of test examples, with 158 examples in total. We sort the examples according the number of iterations required for the version of MOSEK using a corrector. We note that MOSEK solves all instances, and that the number of iterations for that version is well-bounded. For two of the instances MOSEK without a corrector could not solve the instance within a limit of 400 iterations, and in general the iteration count for MOSEK without a corrector is significantly higher. The ECOS solver failed with numerical issues for 84 of the problems. For the problems ECOS solved succesfully it generally required more iterations than MOSEK with the proposed corrector, but fewer than MOSEK without the corrector. Failed instances for ECOS and MOSEK without the corrector are not included in figure.

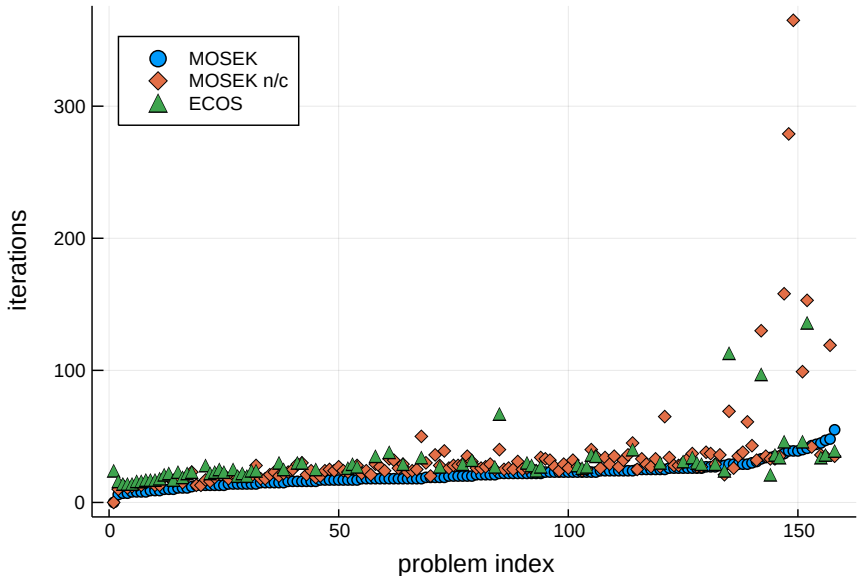

Figure 1: Number of iterations required to solve a selection of exponential-cone problems from CBLIB. "MOSEK" corresponds to the MOSEK solver using a corrector and solved all 158 instances. "MOSEK n/c" corresponds to MOSEK with no corrector, which solved 154 of 158 instances. ECOS solved 74 of 158 instances. Unsolved instances for "MOSEK n/c" and "ECOS" are not shown in the scatter plot.

For the same instances, we show largest violation of primal and dual residuals in Figure 2. In the scatter plot we include all instances regardless of solution status. From the figure we see that there is little difference in solution quality between the two versions of MOSEK. For the ECOS solver, the accuracy is slightly better than MOSEK for some instances, but the residuals are very high for the instances where it failed.

The time required for solving the instances is shown in Figure 3, again discarding instances where the solvers failed. The solvers were run on a standard Linux 64bit computer using a single thread. Most instances are small and solve in less than a second, and with a few exceptions all instances are solved in less than minute. We also note that MOSEK has a larger overhead for very small problems.

In Table 1-Table 3 we give more detailed solution information for a diverse subset of the test problems. For all solvers we report the number of iterations, primal and dual feasiblity measures, as well as the

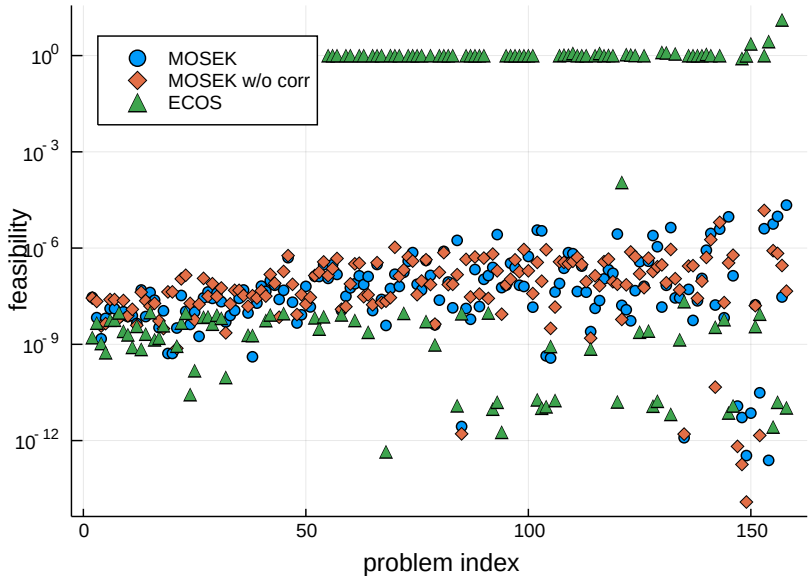

Figure 2: Primal-dual feasibility measures for a selection of exponential-cone problems from CBLIB.

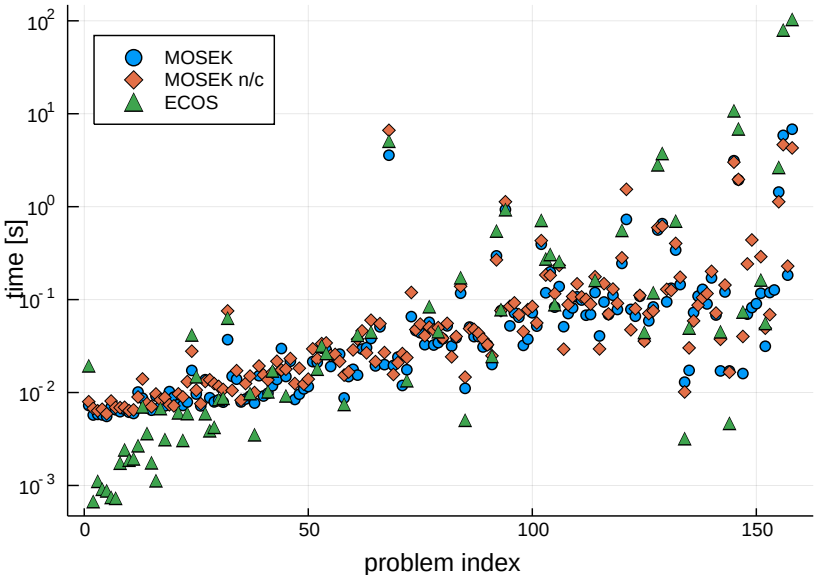

Figure 3: Computation time in seconds for a selection of exponential-cone problems from CBLIB.

relative gap defined by

$$\text{relgap} := \frac{|\text{pobj} - \text{dobj}|}{|\text{pobj}|}$$

for instances with a primal-dual optimal solution. For the MOSEK solver a status key of 1 means that an optimal solution was found, 5 signifies primal infeasibility and 0 means that the solver terminated with an unknown status. For ECOS the exit-codes are 0 for primal-dual optimality, -2 for an unknown status, 1 for primal infeasibility, 10 for primal-dual optimality but inaccuracy detected, and 11 signifies primal infeasibility with inaccuracy.

One open question is if the proposed algorithm has polynomial complexity. For the algorithm using a corrector and the BFGS based primal-dual scaling (48) we evaluated the largest bound (28) using bisection on ξ for all instances in the test set. The largest observed value of ξ was 1.72, and we observed a clear clustering around 1.25. A similar bound of 4/3 is proved in [21] for symmetric cones using a

Problem	status	iter	pfeas	dfeas	relgap	time
cbl_batch	1	29	1.0e-13	1.2e-12	4.0e-10	0.0
cbl_batchs201210m	1	44	3.5e-15	2.4e-13	8.3e-11	0.1
cbl_enpro48	1	33	1.6e-11	1.7e-08	7.2e-11	0.0
cbl_enpro56	1	37	1.2e-13	1.2e-11	3.6e-09	0.0
cbl_ex1223	1	13	5.2e-10	6.9e-11	2.2e-11	0.0
cbl_gams01	1	48	6.5e-11	3.0e-08	3.5e-09	0.2
cbl_ravem	1	41	1.1e-14	3.1e-11	3.2e-10	0.0
cbl_rsyn0805h	1	19	2.2e-08	5.5e-08	1.2e-11	0.0
cbl_syn30h	1	18	2.5e-09	1.1e-08	3.6e-10	0.0
cbl_syn40h	1	18	5.5e-09	2.5e-08	1.1e-09	0.0
cbl_synthes3	1	34	6.8e-10	6.7e-09	1.3e-08	0.0
cbl_rijc787	1	13	3.2e-10	1.0e-08	1.8e-09	0.0
cbl_isil01	5	0	0.0e+00	0.0e+00	-	0.0
cbl_fiac81a	1	10	4.9e-08	2.3e-09	9.6e-09	0.0
cbl_gp_dave_3	1	24	2.5e-09	1.5e-10	7.9e-09	0.1
cbl_cx02-200	1	18	3.9e-09	1.9e-13	9.3e-10	3.6
cbl_beck753	1	11	3.1e-09	3.3e-09	3.7e-09	0.0
cbl_jha88	1	14	4.4e-10	5.0e-09	4.2e-09	0.0
cbl_fiac81b	1	15	4.1e-10	3.2e-11	2.7e-11	0.0
cbl_varun	1	40	1.7e-08	1.1e-10	4.8e-09	0.1
cbl_fang88	1	14	2.1e-08	2.1e-09	5.7e-09	0.0
cbl_demb761	1	14	4.3e-08	4.3e-09	6.3e-09	0.0
cbl_LogExpCR-n500-m1600	1	47	9.7e-06	1.8e-10	2.1e-06	5.9
cbl_mra02	1	25	1.6e-08	3.2e-11	1.8e-08	0.7

Table 1: Summary for MOSEK using corrector.

Problem	status	iter	pfeas	dfeas	relgap	time
cbl_batch	1	69	9.1e-13	1.6e-12	3.1e-08	0.0
cbl_batchs201210m	0	400	1.5e-09	3.2e-08	-	0.5
cbl_enpro48	1	130	8.4e-14	4.6e-11	7.1e-12	0.0
cbl_enpro56	1	158	1.4e-14	6.6e-13	1.0e-08	0.0
cbl_ex1223	1	13	4.3e-08	5.9e-10	2.0e-10	0.0
cbl_gams01	1	119	1.0e-09	2.9e-07	6.6e-08	0.2
cbl_ravem	1	153	3.4e-14	1.4e-12	1.7e-08	0.0
cbl_rsyn0805h	1	30	1.3e-08	2.9e-08	2.4e-11	0.0
cbl_syn30h	1	23	3.9e-09	1.7e-08	6.1e-10	0.0
cbl_syn40h	1	25	4.4e-09	2.0e-08	8.8e-10	0.0
cbl_synthes3	1	33	4.1e-10	2.0e-08	3.9e-08	0.0
cbl_rijc787	1	21	4.6e-10	5.8e-09	2.7e-09	0.0
cbl_isil01	5	0	0.0e+00	0.0e+00	-	0.0
cbl_fiac81a	1	19	4.2e-08	2.0e-09	8.3e-09	0.0
cbl_gp_dave_3	1	45	1.6e-09	3.2e-10	3.9e-09	0.2
cbl_cx02-200	1	50	2.2e-08	1.1e-12	5.1e-09	6.6
cbl_beck753	1	20	4.9e-09	5.4e-09	5.8e-09	0.0
cbl_jha88	1	28	3.5e-10	2.3e-09	3.3e-09	0.1
cbl_fiac81b	1	22	3.3e-08	3.7e-10	3.5e-10	0.0
cbl_varun	1	99	1.6e-08	1.2e-10	5.3e-09	0.3
cbl_fang88	1	20	5.7e-08	5.7e-09	1.5e-08	0.0
cbl_demb761	1	19	3.2e-08	3.2e-09	4.7e-09	0.0
cbl_LogExpCR-n500-m1600	1	35	6.8e-07	5.9e-11	8.3e-07	4.6
cbl_mra02	1	65	6.0e-09	1.3e-11	7.3e-09	1.5

Table 2: Summary for MOSEK without corrector.

Nesterov-Todd scaling. We are optimistic that future work will show that ξ in (28) is indeed bounded for the exponential cone, giving the same worst-case complexity bound as for the symmetric cones.

Problem	exitcode	iter	pfeas	dfeas	relgap	time
cbl_batch	10	113	1.2e-08	2.2e-08	3.9e-11	0.0
cbl_batchs201210m	-2	19	1.0e+00	2.7e+00	-	0.5
cbl_enpro48	0	97	1.7e-09	3.4e-09	9.8e-12	0.0
cbl_enpro56	11	46	8.7e-05	-	-	0.1
cbl_ex1223	-2	10	3.2e-01	4.0e-01	-	0.0
cbl_gams01	-2	10	1.0e+00	1.3e+01	-	0.2
cbl_ravem	0	136	4.8e-09	8.8e-09	1.9e-12	0.1
cbl_rsyn0805h	-2	10	1.0e+00	1.0e+00	-	0.1
cbl_syn30h	-2	14	1.0e+00	5.4e-01	-	0.1
cbl_syn40h	-2	20	1.0e+00	5.4e-01	-	0.1
cbl_synthes3	0	21	5.9e-09	2.8e-09	4.0e-08	0.0
cbl_rijc787	0	23	1.2e-10	1.5e-10	4.0e-10	0.0
cbl_isil01	1	24	1.2e-09	-	-	0.0
cbl_fiac81a	0	22	6.1e-10	7.0e-10	2.4e-09	0.0
cbl_gp_dave_3	0	40	1.4e-10	7.2e-10	5.2e-07	0.2
cbl_cx02-200	0	34	6.5e-14	4.5e-13	3.7e-12	5.1
cbl_beck753	0	22	1.6e-09	1.0e-09	1.6e-09	0.0
cbl_jha88	0	24	8.2e-11	9.3e-11	7.1e-10	0.1
cbl_fiac81b	0	25	1.9e-09	1.7e-09	1.7e-09	0.0
cbl_varun	0	46	4.1e-10	3.6e-09	9.1e-10	0.2
cbl_fang88	0	24	6.5e-09	4.6e-09	1.4e-10	0.0
cbl_demb761	0	20	7.2e-09	5.4e-09	5.1e-12	0.0
cbl_LogExpCR-n500-m1600	0	36	1.2e-13	1.6e-11	2.9e-08	79.7
cbl_mra02	-2	21	1.4e-05	1.1e-04	-	1.2

Table 3: Summary for ECOS.

10 Conclusions

Based on previous work by Tunçel we have presented a generalization of the Nesterov-Todd algorithm for symmetric conic optimization to handle the nonsymmetric exponential cone. Our main contribution is a new Mehrotra like corrector search direction for the to nonsymmetric case, which improves practical performance significantly. Moreover, we presented a practical implementation with extensive computational results documenting the efficiency of proposed algorithm. Indeed the suggested algorithm is significantly more robust and faster than the current state of the art software for nonsymmetric conic optimization ECOS.

Possible future work includes establishing the complexity of the algorithm and applying it to other nonsymmetric cone types, possibly of larger dimensions. One such is example is the nonsymmetric cone of semidefinite matrices with sparse chordal structure [31], which could extend primal-dual solvers like MOSEK with the ability to solve large sparse semidefinite programs.

A Barrier function and derivatives

We consider derivatives up to third order of the exponential-cone barrier (8).

A.1 First-order derivatives

Let $\psi(x) = x_2 \log(x_1/x_2) - x_3$, $g(x) = -\log(\psi(x))$ and $h(x) = -\log x_1 - \log x_2$, *i.e.*, $F(x) = g(x) + h(x)$. Then $F'(x) = g'(x) + h'(x)$ with

$$g'(x) = -\frac{\psi'(x)}{\psi(x)}$$

and $\psi'(x) = (x_2/x_1, \log(x_1/x_2) - 1, -1)$, $h'(x) = -(1/x_1, 1/x_2, 0)$. Sometimes we will omit the arguments for these functions and their derivatives when it is implicitly given.

A.2 Second-order derivatives

$$F''(x) = -\frac{1}{\psi(x)}\left(\psi''(x) - \frac{\psi'(x)\psi'(x)^T}{\psi(x)}\right) + h''(x)$$

with $h''(x) = \mathbf{diag}(1/x_1^2, 1/x_2^2, 0)$ and

$$\psi''(x) = \begin{bmatrix} -\frac{x_2}{x_1^2} & \frac{1}{x_1} & 0 \\ \frac{1}{x_1} & -\frac{1}{x_2} & 0 \\ 0 & 0 & 0 \end{bmatrix}.$$

Let $\hat{\psi}'(x)$ and $\hat{\psi}''(x)$ denote the leading parts of $\psi'(x)$ and $\psi''(x)$, respectively, *i.e.*,

$$\hat{\psi}'(x) := \begin{bmatrix} x_2/x_1 \\ \log(x_1/x_2) - 1 \end{bmatrix}, \quad \hat{\psi}''(x) := \begin{bmatrix} -\frac{x_2}{x_1^2} & \frac{1}{x_1} \\ \frac{1}{x_1} & -\frac{1}{x_2} \end{bmatrix} = -\begin{bmatrix} \frac{\sqrt{x_2}}{x_1} \\ -\frac{1}{\sqrt{x_2}} \end{bmatrix}\begin{bmatrix} \frac{\sqrt{x_2}}{x_1} \\ -\frac{1}{\sqrt{x_2}} \end{bmatrix}^T = -v(x)v(x)^T$$

and similarly for $\hat{h}'$ and $\hat{h}''$. We can then write $F''(x)$ as

$$F''(x) = \frac{1}{\psi(x)^2}\begin{bmatrix} A(x) + \hat{\psi}'(x)\hat{\psi}'(x)^T & -\hat{\psi}'(x) \\ -\hat{\psi}'(x)^T & 1 \end{bmatrix} \tag{60}$$

where

$$A(x) := \psi(x)^2\hat{h}''(x) + \psi(x)v(x)v(x)^T.$$

We can factor $A(x) = V(x)V(x)^T$ with

$$V(x) = \psi(x)\begin{bmatrix} \frac{1-\sqrt{1+2x_2/\psi}}{2x_1} & \frac{1+\sqrt{1+2x_2/\psi}}{2x_1} \\ \frac{1+\sqrt{1+2x_2/\psi}}{2x_2} & \frac{1-\sqrt{1+2x_2/\psi}}{2x_2} \end{bmatrix},$$

which gives a factored expression of $F''(x) = R(x)R(x)^T$ where

$$R(x) = \frac{1}{\psi(x)}\begin{bmatrix} V(x) & \hat{\psi}'(x) \\ 0 & -1 \end{bmatrix} = \begin{bmatrix} \frac{1-\sqrt{1+2x_2/\psi}}{2x_1} & \frac{1+\sqrt{1+2x_2/\psi}}{2x_1} & \frac{1}{\psi}\frac{x_2}{x_1} \\ \frac{1+\sqrt{1+2x_2/\psi}}{2x_2} & \frac{1-\sqrt{1+2x_2/\psi}}{2x_2} & \frac{\log(x_1/x_2)-1}{\psi} \\ 0 & 0 & -\frac{1}{\psi} \end{bmatrix}.$$

A.3 Third-order directional derivatives

We have $F'''(x)[u] = \frac{d}{dt}F''(x+tu)\Big|_{t=0}$ with $F''(x) = g''(x) + h''(x)$. Then

$$F'''(x)[u] = -2\frac{\langle\psi'(x),u\rangle}{\psi(x)}g''(x) - \frac{\langle\psi'(x),u\rangle}{\psi(x)^2}\psi''(x) - \frac{1}{\psi(x)}\psi'''(x)[u] + \frac{1}{\psi(x)^2}\left(\psi'(x)u^T\psi''(x) + \psi''(x)u\psi'(x)^T\right) + h'''(x)[u] \tag{61}$$

with

$$\hat{h}'''(x)[u] = \begin{bmatrix} \frac{d\hat{h}''(x)}{dx_1}\hat{u} & \frac{d\hat{h}''(x)}{dx_2}\hat{u} \end{bmatrix} = -2\begin{bmatrix} \frac{u_1}{x_1^3} & 0 \\ 0 & \frac{u_2}{x_2^3} \end{bmatrix}$$

$$\hat{\psi}'''(x)[u] = \begin{bmatrix} \frac{d\hat{\psi}''(x)}{dx_1}\hat{u} & \frac{d\hat{\psi}''(x)}{dx_2}\hat{u} \end{bmatrix} = \begin{bmatrix} \frac{2x_2u_1}{x_1^3} - \frac{u_2}{x_1^2} & -\frac{u_1}{x_1^2} \\ -\frac{u_1}{x_1^2} & \frac{u_2}{x_2^2} \end{bmatrix}.$$

Considering (60) we can express the inverse Hessian of F as

$$F''(x)^{-1} = \psi(x)^2\begin{bmatrix} A(x)^{-1} & A(x)^{-1}\hat{\psi}'(x) \\ \hat{\psi}'(x)^TA(x)^{-1} & 1 + \hat{\psi}'(x)^TA(x)^{-1}\hat{\psi}'(x) \end{bmatrix}, \tag{62}$$

with

$$A(x)^{-1} = \frac{1}{\psi(x)^2}\left(\hat{h}''(x)^{-1} + \frac{\hat{h}''(x)^{-1}\hat{\psi}''(x)\hat{h}''(x)^{-1}}{\psi(x)+2x_2}\right). \tag{63}$$

From (62) we have that

$$F''(x)^{-1}\psi'(x) = (0,0,-\psi(x)^2), \tag{64}$$

and thus

$$\langle \psi'(x), F''(x)^{-1}u\rangle = -\psi(x)^2 u_3. \tag{65}$$

By expressing

$$\psi''(x) = \psi(x)(h''(x) - F''(x)) + \frac{\psi'(x)\psi'(x)^T}{\psi(x)}$$

we then have from (63) and (64) that

$$\psi''(x)F''(x)^{-1}u = \psi(x)\begin{pmatrix} \hat{u}+\hat{\psi}'(x)u_3 \\ 0 \end{pmatrix} + \frac{\psi(x)\psi''(x)}{\psi(x)+2x_2}\begin{pmatrix} \hat{h}''(x)^{-1}(\hat{u}+\hat{\psi}'(x)u_3) \\ 0 \end{pmatrix} - \psi(x)u - \psi(x)\psi'(x)u_3$$

which simplifies to

$$\psi''(x)F''(x)^{-1}u = \frac{\psi(x)}{\psi(x)+2x_2}\psi''(x)\begin{pmatrix} \hat{h}''(x)^{-1} & \\ & 0 \end{pmatrix}(u+\psi'(x)u_3). \tag{66}$$

Using (65) and (66) in (61) significantly improves the numerically accuracy of the corrector using standard double precision floating-point arithmetic.

References

[1] E. D. Andersen and K. D. Andersen. "Presolving in linear programming". In: *Math. Programming* 71.2 (1995), pp. 221–245.

[2] E. D. Andersen, C. Roos, and T. Terlaky. "On implementing a primal-dual interior-point method for conic quadratic optimization". In: *Math. Programming* 95.2 (Feb. 2003).

[3] Erling D. Andersen and Yinyu Ye. "On a homogeneous algorithm for the monotone complementarity problem". In: *Mathematical Programming* 84.2 (1999), pp. 375–399.

[4] K. Anstreicher and J. -P. Vial. "On the convergence of an infeasible primal-dual interior-point method for convex programming". In: *Optimization Methods and Software* 3 (1994), pp. 273–283.

[5] Peter R. Chares. "Cones and interior-point algorithms for structured convex optimization involving powers and exponentials". PhD thesis. Université Catholique de Louvain, Louvain-la-Neuve, 2009.

[6] A. Domahidi, E. Chu, and S. Boyd. "ECOS: An SOCP solver for embedded systems". In: *European Control Conference (ECC)*. 2013, pp. 3071–3076.

[7] A. S. El-Bakry et al. "On the formulation and theory of the primal-dual Newton interior-point method for nonlinear programming". In: *J. Optim. Theory Appl.* 89.3 (June 1996), pp. 507–541.

[8] Osman Güler. "Barrier functions in interior point methods". In: *Mathematics of Operations Research* 21.4 (1996), pp. 860–885.

[9] Kuo-Ling Huang and Sanjay Mehrotra. *A Modified Potential Reduction Algorithm for Monotone Complementarity and Convex Programming Problems and Its Performance*. Tech. rep. Northwestern University, 2012.

[10] Mehdi Karimi and Levent Tunçel. "Primal-Dual Interior-Point Methods for Domain-Driven Formulations: Algorithms". In: *arXiv preprint arXiv:1804.06925* (2018).

[11] N. Karmarkar. "A polynomial-time algorithm for linear programming". In: *Combinatorica* 4 (1984), pp. 373–395.

[12] M. Lubin and E. Yamangil and R. Bent and J. P. Vielma. "Extended Formulations in Mixed-integer Convex Programming". In: *Integer Programming and Combinatorial Optimization. IPCO 2016. Lecture Notes in Computer Science, Volume 9682*. Ed. by Q. Louveaux and M. Skutella. Springer, Cham, 2016, pp. 102–113.

[13] S. Mehrotra. "On the implementation of a primal-dual interior point method". In: *SIAM J. on Optim.* 2.4 (1992), pp. 575–601.

[14] Tor G. J. Myklebust. "On primal-dual interior-point algorithms for convex optimisation". PhD thesis. University of Waterloo, 2015.

[15] Tor G. J. Myklebust and Levent Tunçel. "Interior-point algorithms for convex optimization based on primal-dual metrics". In: *arXiv preprint arXiv:1411.2129* (2014).

[16] Arkadi Nemirovski. "Advances in convex optimization: conic programming". In: *International Congress of Mathematicians*. Vol. 1. 2007, pp. 413–444.

[17] Arkadi Nemirovski and Levent Tunçel. "Cone-free primal-dual path-following and potential-reduction polynomial time interior-point methods". In: *Mathematical Programming* 102.2 (2005), pp. 261–294.

[18] Y. Nesterov and A. Nemirovskii. *Interior-Point Polynomial Algorithms in Convex Programming*. 1st ed. Philadelphia, PA: SIAM, 1994.

[19] Yu. Nesterov and L. Tuncel. *Local superlnear convergence of polynomial-time interior-point methods for hyperbolicity cone optimization problems*. Tech. rep. Revised August 2015. CORE, Lovain-la-Neuve, 2009.

[20] Yurii Nesterov. "Towards non-symmetric conic optimization". In: *Optimization methods and software* 27.4-5 (2012), pp. 893–917.

[21] Yurii Nesterov and Michael J. Todd. "Primal-dual interior-point methods for self-scaled cones". In: *SIAM Journal on Optimization* 8.2 (1998), pp. 324–364.

[22] Yurii Nesterov and Michael J. Todd. "Self-scaled barriers and interior-point methods for convex programming". In: *Mathematics of Operations research* 22.1 (1997), pp. 1–42.

[23] Yurii Nesterov, Michael J. Todd, and Yinyu Ye. "Infeasible-start primal-dual methods and infeasibility detectors for nonlinear programming problems". In: *Mathematical Programming* 84.2 (1999), pp. 227–267.

[24] Robert B. Schnabel. *Quasi-Newton Methods Using Multiple Secant Equations.* Tech. rep. Colorado University at Boulder, 1983.

[25] Santiago A. Serrano. "Algorithms for unsymmetric cone optimization and an implementation for problems with the exponential cone". PhD thesis. Stanford University, 2015.

[26] Anders Skajaa and Yinyu Ye. "A homogeneous interior-point algorithm for nonsymmetric convex conic optimization". In: *Mathematical Programming* 150.2 (2015), pp. 391–422.

[27] J. F. Sturm. "SeDuMi 1.02, a MATLAB toolbox for optimizing over symmetric cones". In: *Optimization Methods and Software* 11–12 (1999), pp. 625–653.

[28] *The Conic Benchmark Library.* `http://cblib.zib.de/`. [Online; accessed 01-December-2018]. 2018.

[29] Levent Tunçel. "Generalization of Primal-Dual Interior-Point Methods to Convex Optimization Problems in Conic Form". In: *Foundations of computational mathematics* 1.3 (2001), pp. 229–254.

[30] Lieven Vandenberghe. "The CVXOPT linear and quadratic cone program solvers". In: *Online: http://cvxopt. org/documentation/coneprog.pdf* (2010).

[31] Lieven Vandenberghe and Martin S. Andersen. "Chordal graphs and semidefinite optimization". In: *Foundations and Trends® in Optimization* 1.4 (2015), pp. 241–433.

[32] Stephen J. Wright. *Primal-dual interior-point methods.* Vol. 54. Siam, 1997.