# An Efficient IPM Implementation for A Class of Nonsymmetric Cones

Yuwen Chen and Paul Goulart

May 23, 2023

#### **Abstract**

We present an efficient implementation of interior point methods for a family of non-symmetric cones, including generalized power cones, power mean cones and relative entropy cones, by exploiting underlying low-rank and sparse properties of Hessians of homogeneous self-concordant barrier functions. We prove that the augmented linear system in our interior point method is sparse and quasi-definite, enabling the use of sparse LDL factorization with a dual scaling strategy for nonsymmetric cones. Numerical results show that our proposed implementation for nonsymmetric cones performs much faster than the state-of-art solvers for spare problems and scales well for large problems.

# 1 Literature Review

Convex optimization has been widely used in fields such as control engineering [BEGFB94], signal processing [LY06] and finance [CPT18], and includes common problem classes such as linear programming (LP), quadratic programming (QP), second-order cone programming (QP) and semidefinite programming (SDP) problems, as well as other problems classes defined over more exotic cone constraints. Common modern approaches for general convex optimization include first-order operator-splitting methods [OCPB16, GCG21], and second-order interior-point methods (IPMs) [NT98], where the latter is preferred when high accuracy solutions are required. Interior point methods commonly employ logarithmically-homogeneous self-concordant barrier (LHSCB) functions, which apply penalties on convex inequality constraints defined over cones. The self-concordant parameter  $\nu$  is related to the convergence speed of an IPM, where the total iteration number is known to be  $\mathcal{O}(\sqrt{\nu} \ln(1/\epsilon))$  in theory.

State-of-the-art conic interior-point solvers commonly employ an infeasible starting-point method where the initial iterate is chosen on the interior of any conic inequality constraints, with linear equalities not necessarily satisfied until the interior-point algorithm converges to optimality. Since general purpose optimization methods must allow for the possibility of infeasible problems being posed, IPMs require a mechanism that will work either by providing a solution (in the feasible case) or a certificate of infeasibility (in the infeasible

case). This requirement motivated the development of the homogeneous self-dual embedding (HSDE) [YTM94], which augments the original problem to a new one that is always feasible. The new augmented problem is always solvable, and from its solution one can recover either a solution or certificate of infeasibility for the original problem. The HSDE method is very popular and is widely used in many solvers, e.g. SCS [OCPB16], CVXOPT [Van10], ECOS [DCB13] and Mosek [ApS21].

The great majority of research on efficient IPMs has focussed on problems with conic constraints that are self-dual (also called self-scaled), e.g. linear programming (LP), quadratic programming (QP), second order cone programming (SOCP) and semidefinite programming (SDP). Cones that do not possess the self-duality property are called nonsymmetric and are a topic of more recent interest interior point methods [Nes12]. The extension of the homogeneous self-dual model for the nonsymmetric case was proposed in [SY15] with a variant implemented in ECOS [Ser15], and then a complementary proof for  $\mathcal{O}(\sqrt{\nu} \ln(1/\epsilon))$  convergence provided in [PY17]. So far, the exponential cone and the power cone are the two most studied and are available in the commercial solver Mosek [ApS21] and some open-source solvers like ECOS [DCB13] and Alfonso [PY22]. It has also been shown that some existing conic optimization problems can be solved more efficiently by exploiting their special structure through the lens of nonsymmetric conic optimization, like sparse SDPs [ADV10] and sum of squares (SOS) programs [PY19]. A Matlab-based solver DDS [KT19] and a Julia-based solver Hypatia introduce some nonsymmetric cones that solve many optimization problems more efficiently than their extended formulations based on three-dimensional exponential cones or power cones.

However, nonsymmetric cones are not self-scaled, so one can't compute Nesterov-Todd (NT) scaling points as is possible in the symmetric case. Instead, the dual scaling strategy is commonly used for nonsymmetric cones in solvers like ECOS and Hypatia. Recently, a primal-dual scaling algorithm motivated by [Tun01] was proposed in [DA21] and implemented in Mosek, which claims to be the fastest solver for exponential and power cones at present. Both the dual scaling and the primal-dual scaling, require conjugate gradients of conic barrier functions which usually don't have closed forms for nonsymmetric cones. The numerical implementation for computing conjugate gradients of some nonsymmetric cones is detailed in [KAV22].

The power cone is a powerful tool to model various cones such as the p-norm cone, the power mean cone and the generalized power cone [Cha09]. The PhD thesis of Chares [Cha09] describes a 3-self-concordant barrier function and conjectured a (n+1)-self-concordant barrier for (n+1)-dimensional power cone  $\mathcal{K}_{\alpha}^{(n)} := \{(x,z) \in \mathbb{R}_{+}^{n} \times \mathbb{R} : \prod_{i=1}^{n} x_{i}^{\alpha_{i}} \geq |z| \}$  of  $\alpha \in \mathbb{R}_{\geq}^{n}$  and  $\sum_{i=1}^{n} \alpha_{i} = 1$ , which was finally validated in the context of a more general (n,m)-generalized power cone  $\mathcal{K}_{\alpha}^{(n,m)} := \{(x,z) \in \mathbb{R}_{+}^{n} \times \mathbb{R}^{m} : \prod_{i=1}^{n} x_{i}^{\alpha_{i}} \geq ||z||_{2} \}$  [RX22]. The relative entropy cone can be regarded as the extension of three-dimensional exponential cone or second-order cone and is suitable to model many problems in circuit design, statistical learning and power control in communication systems, etc. [CS17].

In this paper, our contribution are: (1) to propose a sparse decomposition of generalized power cones, power mean cones, relative entropy cones that can be used in direct sparse LDL factorization; (2) to show how to compute the conjugate barrier functions for these three nonsymmetric cones to measure the proximity to the central path.

The sketch of this paper is given as follows: We introduce basic definitions of homogeneous self-dual embedding and interior point methods in Section 2. Then, we propose a general structure that is suitable for sparse LDL factorization in Section 3. The proposed augmented sparse decomposition of Hessians for nonsymmetric cones are shown in Section 4 and the centrality check for IPMs with nonsymmetric cones is detailed in Section 5. In Section 7, numerical results show that our proposed sparse implementation for nonsymmetric cones performs better than the state-of-art solvers. Finally, the conclusion is summarized in Section 8.

**Notation**  $\mathbb{R}^d$  is the d-dimensional space of reals and  $x \in \mathbb{R}^d$  ( $\mathbb{R}^d_+$ ) denotes  $x \in \mathbb{R}^d$  is nonnegative (positive) elementwise.  $X \in \mathbb{S}^n$  denotes X is a symmetric matrix in  $\mathbb{R}^{n \times n}$ .  $X \in \mathbb{S}^n_+$  ( $\mathbb{S}^n_{++}$ ) denotes X is positive semidefinite (positive definite). Functions  $f(\cdot)$  and  $f^*(\cdot)$  consist of a convex conjugate pair. The Hessian of f at x is denoted by f''(x) and the 3rd-order directional derivative along direction h by  $f'''(x)[h] \in \mathbb{R}^{n \times n}$ . The set of interior points of C is denoted by  $\inf(C)$ . The inner product of vectors x, y in the Euclidean space  $\mathbb{R}^n$  is denoted by  $\inf(C)$  and the Euclidean norm of x by  $\inf(x)$ . The product set of  $\inf(C_1, C_2)$  is denoted  $\inf(C_1, C_2)$  is denoted  $\inf(C_1, C_2)$  is abbreviated as  $\inf(C_1, C_2)$  and  $\inf(C_1, C_2)$  is abbreviated as  $\inf(C_1, C_2)$  is denoted by  $\inf(C_1, C_2)$  is denoted by  $\inf(C_1, C_2)$  is denoted by  $\inf(C_1, C_2)$  is abbreviated as  $\inf(C_1, C_2)$  is denoted by  $\inf(C_1, C_2)$  is

$$\delta_{i,j} = \begin{cases} 1, & i = j, \\ 0, & \text{otherwise.} \end{cases}$$

# 2 Problem Description

Throughout, we will consider the following primal-dual pair of conic optimization problems:

$$\begin{aligned} & \underset{x,s}{\min} & c^{\top}x & & \underset{y,z}{\max} & -h^{\top}y - b^{\top}z \\ & \text{s.t.} & Gx = h, & (\mathcal{P}) & & \text{s.t.} & A^{\top}z + G^{\top}y + c = 0, & (\mathcal{D}) \\ & & Ax + s = b, \ s \in \mathcal{K}, & & z \in \mathcal{K}^*, \end{aligned}$$

where  $G \in \mathbb{R}^{p \times n}$ ,  $A \in \mathbb{R}^{m \times n}$ ,  $c \in \mathbb{R}^n$ ,  $h \in \mathbb{R}^p$ ,  $b \in \mathbb{R}^m$ ,  $\mathcal{K}$  is a nonempty, closed convex cone and  $\mathcal{K}^*$  is the dual cone of  $\mathcal{K}$ . We assume that  $\mathcal{K}$  does not contain any straight lines, i.e. it is *pointed*. We call problem  $(\mathcal{P})$  the *primal* problem and  $(\mathcal{D})$  the *dual* problem, and will assume throughout that strong duality holds.

The problem  $(\mathcal{P})$  and its dual  $(\mathcal{D})$  are quite general, and cover a wide range of standard problem types in numerical optimization. If  $\mathcal{K}$  is the nonnegative orthant, i.e.  $\mathcal{K} = \mathbb{R}_+$ ,

then  $(\mathcal{P})$  represents a linear program. Likewise, if  $\mathcal{K}$  is the second-order cone  $\mathcal{K}_{soc}$  or the positive semidefinite cone  $\mathcal{S}^n_+$ , then  $(\mathcal{P})$  represents a second-order cone program (SOCP) or semidefinite program (SDP), respectively. In each of these cases, or where  $\mathcal{K}$  is a composition of cones of these types, the cone  $\mathcal{K}$  is symmetric and satisfies the self-dual property  $\mathcal{K} = \mathcal{K}^*$ .

In this paper we focus on problems in which the cone  $\mathcal{K}$  is non-symmetric so that the self-dual property for  $\mathcal{K}$  does not hold, and will consider in detail a family of non-symmetric cones for which we can exploit special structure within an interior point method. In the remainder of this section, we sketch the basic structure of an interior point method for such cones, and subsequently consider in detail the issue of sparsity exploitation for our cones of interest.

#### Logarithmically homogeneous self-concordant barrier (LHSCB) functions

A function  $f: \mathbb{R}^n \to \mathbb{R}$  is called  $\nu$ -LHSCB for a convex cone  $\mathcal{K} \subset \mathbb{R}^n$ , if it satisfies

$$|\nabla^3 f(x)[r,r,r]| \le 2 \left(\nabla^2 f(x)[r,r]\right)^{3/2} \quad \forall x \in \text{int} \mathcal{K}, r \in \mathbb{R}^n,$$
  
$$f(\lambda x) = f(x) - \nu \ln(\lambda) \qquad \forall x \in \text{int} \mathcal{K}, \lambda > 0,$$
 (2)

for some scalar  $\nu \geq 1$ . We first recall several standard properties of such a barrier function; the reader is referred to [NT97] for details.

The gradient q of f satisfies

$$\langle g(x), x \rangle = -\nu, \quad \forall x \in \text{int} \mathcal{K}.$$
 (3)

The convex conjugate  $f^*$ , defined as

$$f^*(y) := \max_{x \in \text{int} \mathcal{K}} \{ -\langle y, x \rangle - f(x) \}, \tag{4}$$

is also  $\nu$ -LHSCB in the interior of  $\mathcal{K}^*$ , and we call it the *conjugate barrier*. The gradient  $g^*$  of  $f^*$  is the solution of

$$g^*(y) := -\arg\max_{x \in \text{int}\mathcal{K}} \{-\langle y, x \rangle - f(x)\}. \tag{5}$$

Likewise, we also call f is the conjugate and g is the conjugate gradient of  $f^*$ . From (3)–(5) it can be verified that

$$f^*(y) = -\langle y, -g^*(y) \rangle - f(-g^*(y)) = -\nu - f(-g^*(y)), \ \forall y \in \text{int} \mathcal{K}^*.$$
 (6)

Finally, we have

$$-g^*(-g(x)) = x, \ \forall x \in \text{int}\mathcal{K}, \quad -g(-g^*(y)) = y, \ \forall y \in \text{int}\mathcal{K}^*.$$
 (7)

# 2.1 Homogeneous self-dual IPM for nonsymmetric cones

We next introduce several basic building blocks for an IPM based on the homogeneous self-dual embedding (HSDE) for nonsymmetric cones. In the simplified HSDE model the KKT condition for  $(\mathcal{P})$  and  $(\mathcal{D})$  are modified to

$$\begin{bmatrix} 0 \\ 0 \\ s \\ \kappa \end{bmatrix} = \begin{bmatrix} 0 & G^{\top} & A^{\top} & c \\ -G & 0 & 0 & h \\ -A & 0 & 0 & b \\ -c^{\top} & -h^{\top} & -b^{\top} & 0 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ \tau \end{bmatrix}$$
(8)

$$(s, z, \kappa, \tau) \in \mathcal{F}, x \in \mathbb{R}^n, y \in \mathbb{R}^p,$$

where  $\mathcal{F} := \mathcal{K} \times \mathcal{K}^* \times \mathbb{R}_+ \times \mathbb{R}_+$  and the cone  $\mathcal{K}$  may represent the composition of multiple standard cone types. We refer to (8) as an HSDE model for  $(\mathcal{P})$ – $(\mathcal{D})$  since the problem of finding a feasible point for (8) is dual to itself<sup>1</sup>, its feasible points form a cone, and any feasible point  $(x^*, y^*, z^*, s^*, \tau^*, \kappa^*)$  of (8) can be used to extract either a solution to  $(\mathcal{P})$ – $(\mathcal{D})$  or a certificate of infeasibility. In particular, if  $\tau^* > 0$  then  $(\mathcal{P})$ – $(\mathcal{D})$  is feasible with solution  $(x^*/\tau^*, y^*/\tau^*, z^*/\tau^*, s^*/\tau^*)$ . Our IPM approach amounts to a Newton-like method for finding a feasible point for (8).

The optimality condition in a convex problem include a complementary slackness condition, which is replaced by a *central path* condition in an IPM whose details depend on the scaling strategy employed. For symmetric cones, e.g. nonnegative, second-order and positive-semidefinite cones, we can exploit the self-scaled property [NT97] that ensures a unique scaling point called the Nesterov-Todd (NT) scaling [NT97] and is obtained from the combination of primal and dual variables. In the case of nonsymmetric cones, the dual scaling is a popular choice [Nes12, SY15, ADV10] and is implemented in ECOS [Ser15]. A primal-dual scaling with a 3rd-order correction is also effective and is implemented in Mosek [DA21]. In this paper, we will employ the dual scaling strategy so that we can exploit the low-rank information or sparsity of Hessians of dual barrier functions directly.

For the dual scaling strategy, given some initial point  $(x^0, y^0, s^0, z^0, \tau^0, \kappa^0)$  with  $(s^0, z^0) \in \mathcal{K} \times \mathcal{K}^*$ , the central path is parametrized by  $\mu \in (0, 1]$  and the gradient  $g^*(z)$  of the dual barrier function  $f^*(z)$ , satisfying

$$\begin{bmatrix} 0 \\ 0 \\ s \\ \kappa \end{bmatrix} = \begin{bmatrix} 0 & G^{\top} & A^{\top} & c \\ -G & 0 & 0 & h \\ -A & 0 & 0 & b \\ -c^{\top} & -h^{\top} & -b^{\top} & 0 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ \tau \end{bmatrix} + \mu \begin{bmatrix} q_x \\ q_y \\ q_z \\ q_\tau \end{bmatrix}$$

$$(s, z, \kappa, \tau) \in \mathcal{F},$$

$$s = -\mu g^*(z), \quad \tau \kappa = \mu,$$

$$(9)$$

<sup>&</sup>lt;sup>1</sup>NB: this does *not* require that the cone  $\mathcal{K}$  should be self-dual.

where

$$\begin{split} q_x &= -G^\top y^0 - A^\top z^0 - c\tau^0, & q_y &= Gx^0 - h\tau^0, \\ q_\tau &= \kappa + c^\top x^0 + h^\top y^0 + b^\top z^0, & q_z &= s^0 + Ax^0 - b\tau^0 \end{split}$$

For an IPM with nonsymmetric cones, we require that every iterate should remain close to the central path. Standard metrics for this distance include the proximity measure in [Nes12, SY15] and one defined in terms of shadow iterates [Tun01, DA21].

#### 2.1.1 Computation of the KKT system

In order to solve (8) we adopt the widely-used affine-centering (predictor-corrector) framework due to its excellent practical performance. We provide implementation details for this method in Section 6. Note that the step equations for both the predictor and the corrector steps can be written in the unified general form

$$\begin{bmatrix} 0 & G^{\top} & A^{\top} & c \\ -G & 0 & 0 & h \\ -A & 0 & 0 & b \\ -c^{\top} & -h^{\top} & -b^{\top} & 0 \end{bmatrix} \begin{bmatrix} \Delta x \\ \Delta y \\ \Delta z \\ \Delta \tau \end{bmatrix} - \begin{bmatrix} 0 \\ 0 \\ \Delta s \\ \Delta \kappa \end{bmatrix} = \begin{bmatrix} d_x \\ d_y \\ d_z \\ d_{\tau} \end{bmatrix},$$

$$\mu H^*(z) \Delta z + \Delta s = -d_s, \ \tau \Delta \kappa + \kappa \Delta \tau = -d_{\kappa},$$

$$(10)$$

where only the choice of right-hand side term  $d_x, d_y, d_z, d_s, d_\tau, d_\kappa$  differs between the affine and the centering step directions and a given iterate.  $H^*(z)$  is the Hessian of the dual barrier  $f^*(z), \forall z \in \text{int} \mathcal{K}$ .

We typically expect the data matrices G and A to be sparse, particularly for large problems, so we adopt the same approach as [Van10, DCB13] to preserve this sparsity while reducing the system to one in for which we can perform a symmetric linear solve. First, we substitute the last two equations into the first one to obtain

$$\begin{bmatrix} 0 & G^{\top} & A^{\top} & c \\ G & 0 & 0 & -h \\ A & 0 & -\mu H^*(z) & -b \\ -c^{\top} & -h^{\top} & -b^{\top} & \frac{\kappa}{\tau} \end{bmatrix} \begin{bmatrix} \Delta x \\ \Delta y \\ \Delta z \\ \Delta \tau \end{bmatrix} = \begin{bmatrix} d_x \\ -d_y \\ d_s - d_z \\ d_{\tau} - d_{\kappa}/\tau \end{bmatrix}.$$

This can then be solved by solving a symmetric linear system twice with different right-hand sides, i.e.

$$K\begin{bmatrix} x_1 \\ y_1 \\ z_1 \end{bmatrix} = \begin{bmatrix} d_x \\ -d_y \\ d_s - d_z \end{bmatrix}, \qquad K\begin{bmatrix} x_2 \\ y_2 \\ z_2 \end{bmatrix} = \begin{bmatrix} -c \\ h \\ b \end{bmatrix}, \tag{11}$$

where K is defined as

$$K := \begin{bmatrix} 0 & G^{\top} & A^{\top} \\ G & 0 & 0 \\ A & 0 & -H_s \end{bmatrix}$$
 (12)

with  $H_s = \mu H^*(z)$ . Only the lower right-hand block  $H_s$ , which may be dense in general, would be different if we had instead employed a primal-dual scaling strategy [DA21]. Finally, we obtain the solution of (10) by first solving for  $\Delta \tau$  as

$$\Delta \tau = \frac{d_{\tau} - d_{\kappa} / \tau + c^{\top} x_1 + h^{\top} y_1 + b^{\top} z_1}{\kappa / \tau - c^{\top} x_2 - h^{\top} y_2 - b^{\top} z_2},$$

followed by

$$\Delta x = x_1 + \Delta \tau x_2, \qquad \Delta s = -d_s - H_s \Delta z,$$
  

$$\Delta y = y_1 + \Delta \tau y_2, \qquad \Delta \kappa = -\left(d_\kappa + \kappa \Delta \tau\right) / \tau.$$
  

$$\Delta z = z_1 + \Delta \tau z_2,$$

Observe that we only need to factor the matrix K once at each iteration before performing the two linear solves in (11). We utilize sparse LDL factorization in [Dav05] with static ordering to factorize the matrix K. We add a small regularization term to the diagonal of K so that it becomes symmetric quasidefinite and the LDL factorization always exists with diagonal D even after any symmetric permutation P [Van95], i.e.  $K = PLDL^{\top}P^{\top}$ .

# 3 Augmented Sparsity

We will consider a class of nonsymmetric cones for which the term  $H_s$  that appears as the lower right-hand block in (12) is the sum of a sparse matrix plus a few low-rank dense terms, and satisfies a certain quasi-definiteness condition.

**Definition 1** Suppose the matrix H is of the form

$$H := D + \sum_{i=1}^{n_1} u_i u_i^{\top} - \sum_{j=1}^{n_2} v_i v_i^{\top}, \tag{13}$$

where  $D \in \mathbb{S}^{n \times n}$  is sparse and  $n_1, n_2 \ll n$ . We will call H augmented-sparse if  $D - \sum_{i=1}^{n_2} v_i v_i^{\top} \succ 0$ . The corresponding augmented sparse matrix is

$$H_{ ext{aug}} := \left[ egin{array}{ccc} D & V & U \ V^ op & I_{n_2} \ U^ op & -I_{n_1} \end{array} 
ight]$$

with  $V = [v_1, \dots, v_{n_2}], U = [u_1, \dots, u_{n_1}].$ 

**Lemma 1** If H is augmented-sparse then  $H_{\text{aug}}$  is quasidefinite.

*Proof.* A sufficient condition is for the upper left  $2\times 2$  block of H to be positive definite [Van95]. This follows immediately since the Schur complement  $D-VV^T$  is positive definite by definition.

If  $H_s$  in (12) is augmented-sparse, we can solve (11) via an equivalent linear system. For example, the last row of the left equation in (11), i.e.  $Ax - H_s z_1 = d_s - d_z$  becomes

$$\begin{bmatrix} A \\ 0 \\ 0 \end{bmatrix} \Delta x - \underbrace{\begin{bmatrix} D & V & U \\ V^{\top} & I_{n_2} \\ U^{\top} & -I_{n_1} \end{bmatrix}}_{H_{\text{S,alig}}} \begin{bmatrix} \Delta z_1 \\ t_v \\ t_u \end{bmatrix} = \begin{bmatrix} d_s - d_z \\ 0 \\ 0 \end{bmatrix}, \tag{14}$$

which is both larger and sparser than the equivalent term in (11). The same applies to the last row of the right equation in (11). For (14), the number of nonzero entries in  $H_{s,\text{aug}}$  is  $[\text{nnz}(D) + n(n_1 + n_2 + 2)]$ , which is much smaller than the number of entries in a dense H if  $n_1, n_2 \ll n$ . We therefore expect that an LDL factorization of  $H_{\text{aug}}$  will have significantly sparser factors than would be obtained by direct factorization of H. Such a property has been exploited for SOCPs in ECOS under the NT (primal-dual) scaling strategy [DCB13]. We will show how this approached can also be utilized in the dual scaling strategy for nonsymmetric cones.

# 4 Sparse Decomposition of $H_s$ in Nonsymmetric Cones

In this section, we describe several non-symmetric cones that can be shown to have *augmented-sparse* structure in the sense of Definition 1. We introduce fundamental definitions and barrier functions for these nonsymmetric cones in Section 4.1, and then detail the underlying augmented-sparse structures in Section 4.2.1-4.2.3, respectively.

## 4.1 Three nonsymmetric cones

We consider three nonsymmetric cones that can exploit sparse structure: generalized power cones, power mean cones and relative entropy cones. They are defined as follows:

**Definition 2 (Generalized Power Cone)** The generalized power cone is parametrized by  $\alpha \in \mathbb{R}^{d_1}_{++}$  such that  $\sum_{i \in [d_1]} \alpha_i = 1$  and is defined as

$$\mathcal{K}_{\text{gpow}(\alpha, d_1, d_2)} = \left\{ (u, w) \in \mathbb{R}_+^{d_1} \times \mathbb{R}^{d_2} : \prod_{i \in [d_1]} u_i^{\alpha_i} \ge ||w|| \right\}.$$
 (15a)

Its dual cone is

$$\mathcal{K}_{\mathrm{gpow}(\alpha,d_1,d_2)}^* = \left\{ (u,w) \in \mathbb{R}_+^{d_1} \times \mathbb{R}^{d_2} : \prod_{i \in [d_1]} \left( \frac{u_i}{\alpha_i} \right)^{\alpha_i} \ge ||w|| \right\}. \tag{15b}$$

**Definition 3 (Power Mean Cone)** The power mean cone is parametrized by  $\alpha \in \mathbb{R}^{d_1}_{++}$  such that  $\sum_{i \in [d]} \alpha_i = 1$  and is defined as

$$\mathcal{K}_{\text{powm}(\alpha,d)} = \left\{ (u, w) \in \mathbb{R}^d_+ \times \mathbb{R} : \prod_{i \in [d]} u_i^{\alpha_i} \ge w \right\}.$$
 (16a)

Its dual cone is

$$\mathcal{K}_{\text{powm}(\alpha,d)}^* = \left\{ (u, w) \in \mathbb{R}_+^d \times \mathbb{R}_- : \prod_{i \in [d]} \left( \frac{u_i}{\alpha_i} \right)^{\alpha_i} \ge -w \right\}.$$
 (16b)

Note that the geometric mean cone [CKV22] is a special case of the power mean cone where  $\alpha_1 = \cdots = \alpha_d = \frac{1}{d}$ .

**Definition 4 (Relative Entropy Cone)** The relative entropy cone is defined as

$$\mathcal{K}_{\text{rel}} = cl \left\{ (u, v, w) \in \mathbb{R} \times \mathbb{R}^d_{++} \times \mathbb{R}^d_{++} : u \ge \sum_{i \in [d]} w_i \ln \left( \frac{w_i}{v_i} \right) \right\}.$$
 (17a)

Its dual cone is

$$\mathcal{K}_{\text{rel}}^* = cl\left\{(u, v, w) \in \mathbb{R}_{++} \times \mathbb{R}_{++}^d \times \mathbb{R}^d : w_i \ge u\left(\ln\left(\frac{u}{v_i}\right) - 1\right), \forall i \in [d]\right\}. \tag{17b}$$

Note that a generalized power cone reduce to a second-order cone when  $d_1 = 1$  and it becomes self-dual, i.e.  $\mathcal{K}_{\text{gpow }(1,1,d_2)} = \mathcal{K}_{\text{gpow }(1,1,d_2)}^*$ , while it looks similar to a power mean cone when  $d_2 = 1$ . This motivates us to consider whether we can exploit sparsity of these nonsymmetric cones using an approach similar to the one that was described for second-order cones in [DCB13]. The answer is *yes* if we choose the dual scaling strategy for these cones in a IPM. Meanwhile, the dual of the relative entropy cone has a sparse Hessian that we can exploit and which also fits our Definition 1.

We therefore choose the dual scaling strategy in our IPM throughout the paper. We consequently must define  $\nu$ -LHSCB conjugate barrier functions for each of the cones  $\mathcal{K}^*_{\text{gpow}}(\alpha,d_1,d_2)$ ,  $\mathcal{K}^*_{\text{powm}}(\alpha,d)$  and  $\mathcal{K}^*_{\text{rel}}$ . The results are summarized in the next theorem.

**Theorem 1** For the dual cones of generalized power cones, power mean cones and relative entropy cones:

i) The function

$$f_{gpow}^*(u, w) = -\ln\left(\prod_{i \in [d_1]} \left(\frac{u_i}{\alpha_i}\right)^{2\alpha_i} - \|w\|^2\right) - \sum_{i \in [d_1]} (1 - \alpha_i) \ln\left(\frac{u_i}{\alpha_i}\right)$$
(18)

is a  $(d_1 + 1)$ -LHSCB function of  $\mathcal{K}^*_{gpow(\alpha,d_1,d_2)}$  where  $\alpha, u \in \mathbb{R}^{d_1}_{++}, w \in \mathbb{R}^{d_2}$ .

ii) The function

$$f_{\text{powm}}^*(u, w) = -\ln\left(\prod_{i \in [d]} \left(\frac{u_i}{\alpha_i}\right)^{\alpha_i} + w\right) - \sum_{i \in [d]} (1 - \alpha_i) \ln\left(\frac{u_i}{\alpha_i}\right) - \ln(-w)$$
 (19)

is a (d+1)-LHSCB function of  $\mathcal{K}^*_{powm(\alpha,d)}$  where  $\alpha, u \in \mathbb{R}^d_{++}, w < 0$ .

iii) The function

$$f_{\text{rel}}^*(u, v, w) = -\sum_{i \in [d]} \left( w_i - u \ln \left( \frac{u}{v_i} \right) + u \right) - d \ln(u) - \sum_{i \in [d]} \ln(v_i)$$
 (20)

is a 3d-LHSCB function of  $\mathcal{K}_{rel}^*$  where u > 0 and  $v \in \mathbb{R}_{++}^d$ ,  $w \in \mathbb{R}^d$ .

*Proof.* See Appendix A.

# 4.2 Sparsity exploitation

For the dual-scaling strategy at iteration k, the scaling matrix  $H_s$  is set to  $H_s^k = \mu^k H^*(z^k)$ , where  $\mu^k > 0$  is the centering parameter and  $H^*(z^k)$  is the Hessian of the dual barrier functions at  $z^k \in \mathcal{K}^*$ . Hence, we can exploit the augmented-sparse structure of the scaling matrix  $H_s$  as long as  $H^*(z^k)$  is augmented-sparse. For the remainder of this Section, we will show that the Hessians  $H^*(z)$  of each of the dual barrier functions defined in Theorem 1 are all augmented-sparse, and thus so are the related scaling matrices  $H_s$ .

### 4.2.1 Generalized power cone

We start by establishing the augmented-sparse property for generalized power cones:

**Theorem 2** The Hessian of the barrier function (18) for the generalized power cone satisfies Definition 1 with  $n_1 = 1, n_2 = 2$ , i.e.

$$H^*(z) = D + pp^{\top} - qq^{\top} - rr^{\top},$$

where z =: (u, w) and  $D - qq^{\top} - rr^{\top} \succ 0$ . The parameters D, p, q, r are given by

$$D = \begin{bmatrix} \ddots & & & \\ & \frac{\tau_{i}\varphi}{\zeta u_{i}} + \frac{1-\alpha_{i}}{u_{i}^{2}} & & \\ & & \ddots & \\ & & & \frac{2}{\zeta} \cdot I_{d_{2}} \end{bmatrix}, \ p = \begin{bmatrix} p_{0} \cdot \frac{\tau}{\zeta} \\ p_{1} \cdot \frac{w}{\zeta} \end{bmatrix}, \ q = \begin{bmatrix} q_{0} \cdot \frac{\tau}{\zeta} \\ 0 \end{bmatrix}, \ r = \begin{bmatrix} 0 \\ r_{1} \cdot \frac{w}{\zeta} \end{bmatrix},$$

$$(21a)$$

with

$$p_0 = \sqrt{\frac{\varphi(\varphi + ||w||^2)}{2}}, \qquad p_1 = -2\sqrt{\frac{2\varphi}{\varphi + ||w||^2}},$$

$$q_0 = \sqrt{\frac{\zeta\varphi}{2}}, \qquad \qquad r_1 = 2\sqrt{\frac{\zeta}{\varphi + ||w||^2}},$$
(21b)

where 
$$\varphi = \prod_{i \in [d_1]} \left(\frac{u_i}{\alpha_i}\right)^{2\alpha_i}, \tau_i = \frac{2\alpha_i}{u_i}, \forall i \in [d_1], \text{ and } \zeta = \prod_{i \in [d_1]} \left(\frac{u_i}{\alpha_i}\right)^{2\alpha_i} - \|w\|^2.$$

Proof of Theorem 2. The gradient and Hessians for  $f_{\text{gpow}}^*(u, w)$  (18) are given by

$$\nabla_{u_i} f^* = -\frac{\tau_i \varphi}{\zeta} - \frac{1 - \alpha_i}{u_i}, \quad \forall i \in [d_1],$$

$$\nabla_{w_i} f^* = \frac{2w_i}{\zeta}, \quad \forall i \in [d_2],$$

$$\nabla^2_{u_i, u_j} f^* = \frac{\tau_i \tau_j \varphi}{\zeta} \left(\frac{\varphi}{\zeta} - 1\right) + \delta_{i,j} \left(\frac{\tau_i \varphi}{\zeta u_i} + \frac{1 - \alpha_i}{u_i^2}\right), \quad \forall i, j \in [d_1],$$

$$\nabla^2_{u_i, w_j} f^* = -\frac{2\tau_i \varphi w_j}{\zeta^2}, \quad \forall i \in [d_1], \forall j \in [d_2],$$

$$\nabla^2_{w_i, w_j} f^* = \frac{4w_i w_j}{\zeta^2} + \delta_{i,j} \frac{2}{\zeta}, \quad \forall i, j \in [d_2].$$

In order to satisfy Definition 1,  $p_0, p_1, q_0, r_1$  need to satisfy the conditions

$$p_0^2 - q_0^2 = \varphi(\varphi - \zeta) = \varphi ||w||^2, \quad p_0 p_1 = -2\varphi, \quad p_1^2 - r_1^2 = 4,$$

to match the coefficients in the Hessian. Given the parameter choices in (21), we find  $D - qq^{\top} - rr^{\top}$  is  $2 \times 2$  block diagonal and the condition  $D - qq^{\top} - rr^{\top} \succ 0$  becomes

$$D_1 - q_0^2 \left(\frac{\tau}{\zeta}\right) \cdot \left(\frac{\tau}{\zeta}\right)^{\top} \succ 0 \text{ and } D_2 - r_1^2 \left(\frac{w}{\zeta}\right) \cdot \left(\frac{w}{\zeta}\right)^{\top} \succ 0,$$

which is equivalent to

$$1 - \left(\frac{\tau}{\zeta}\right)^{\top} D_1^{-1} \left(\frac{\tau}{\zeta}\right) = 1 - \sum_{i \in [d_1]} \frac{q_0^2 \cdot \frac{\tau_i^2}{\zeta^2}}{\frac{\tau_i \varphi}{\zeta u_i} + \frac{1 - \alpha_i}{u_i^2}} = 1 - \sum_{i \in [d_1]} \frac{4\alpha_i^2 q_0^2}{\zeta (2\alpha_i \varphi + (1 - \alpha_i)\zeta)} > 0, \quad (22a)$$

and

$$1 - \left(\frac{w}{\zeta}\right)^{\top} D_2^{-1} \left(\frac{w}{\zeta}\right) = 1 - r_1^2 \frac{\|w\|^2}{\zeta^2} \cdot \frac{\zeta}{2} > 0, \tag{22b}$$

by Schur complement. If we set  $q_0 = \sqrt{\frac{\zeta \varphi}{2}}$ ,  $r_1 = 2\sqrt{\frac{\zeta}{\varphi + ||w||^2}}$ , (22a) is satisfied by

$$1 - \sum_{i \in [d_1]} \frac{4\alpha_i^2 q_0^2}{\zeta(2\alpha_i \varphi + (1 - \alpha_i)\zeta)} > 1 - \sum_{i \in [d_1]} \frac{4\alpha_i^2 q_0^2}{2\zeta\alpha_i \varphi} = 1 - \frac{2q_0^2}{\zeta\varphi} = 0,$$

and (22b) is satisfied by

$$1 - r_1^2 \frac{\|w\|^2}{\zeta^2} \cdot \frac{\zeta}{2} = 1 - \frac{2\|w\|^2}{\varphi + \|w\|^2} = \frac{2\zeta}{\varphi + \|w\|^2} > 0.$$

Hence, 
$$D - qq^{\top} - rr^{\top} \succ 0$$
.

Since the scaling matrix  $H_s = \mu H^*(z)$  is augmented-sparse, we can employ an expanded sparse linear system

$$\begin{bmatrix} A \\ 0 \\ 0 \\ 0 \end{bmatrix} \Delta x - \mu \begin{bmatrix} D & q & r & p \\ q^{\top} & 1 & 0 & 0 \\ r^{\top} & 0 & 1 & 0 \\ p^{\top} & 0 & 0 & -1 \end{bmatrix} \begin{bmatrix} \Delta z_1 \\ t_q \\ t_r \\ t_p \end{bmatrix} = \begin{bmatrix} d_s - d_z \\ 0 \\ 0 \\ 0 \end{bmatrix}, \tag{23}$$

in our interior point step equation (11), where  $K_{\text{aug}}$  is quasi-definite and strongly factorizable. Instead of (23), we solve the linear system

$$\begin{bmatrix} A \\ 0 \\ 0 \\ 0 \end{bmatrix} \Delta x - \begin{bmatrix} \mu D & \sqrt{\mu}q & \sqrt{\mu}r & \sqrt{\mu}p \\ \sqrt{\mu}q^{\top} & 1 & 0 & 0 \\ \sqrt{\mu}r^{\top} & 0 & 1 & 0 \\ \sqrt{\mu}p^{\top} & 0 & 0 & -1 \end{bmatrix} \begin{bmatrix} \Delta z_1 \\ t'_q \\ t'_r \\ t'_p \end{bmatrix} = \begin{bmatrix} d_s - d_z \\ 0 \\ 0 \\ 0 \end{bmatrix}, \tag{24}$$

which yields the same  $\Delta x$ ,  $\Delta z_1$  as (23). During direct factorization we find that the pivoting for (24) is more numerically stable than for (23) since the magnitude of diagonal terms of  $K_{\text{aug}}$  increases from  $\mu$  to 1 and the ratio of off-diagonal to diagonal terms also decreases, e.g. q to  $\sqrt{\mu}q$ . In addition, since half of entries in q, r are zeros, the memory required for q, r together is the same as p in the sparse data form.

#### 4.2.2 Power mean cone

We next establish the augmented-sparse property for the power mean cone:

**Theorem 3** The Hessian of the barrier function (19) for the power mean cone satisfies Definition 1 with  $n_1 = 1, n_2 = 2$ , i.e.

$$H^*(z) = D + pp^{\top} - qq^{\top} - rr^{\top},$$

where z =: (u, w) and  $D - qq^{\top} - rr^{\top} \succ 0$ . The parameters D, p, q, r are given by

$$D = \begin{bmatrix} \ddots & & & \\ & \frac{\tau_i \varphi}{u_i} + \frac{1 - \alpha_i}{u_i^2} & & \\ & & \ddots & \\ & & & \theta \end{bmatrix}, \ p = \begin{bmatrix} p_0 \cdot \tau \\ p_1 \cdot \frac{1}{\zeta} \end{bmatrix}, \ q = \begin{bmatrix} q_0 \cdot \tau \\ 0 \end{bmatrix}, \ r = \begin{bmatrix} 0 \\ r_1 \cdot \frac{1}{\zeta} \end{bmatrix}, \quad (25a)$$

with

$$\theta = 1 + \frac{1}{w^2}, \quad p_0 = \varphi, \qquad p_1 = 1,$$

$$q_0 = \sqrt{\zeta \varphi}, \quad r_1 = \zeta,$$
(25b)

where we define  $\varphi = \prod_{i \in [d]} \left(\frac{u_i}{\alpha_i}\right)^{\alpha_i}$ ,  $\zeta = \varphi + w$  and  $\tau \in \mathbb{R}^d$  with  $\tau_i = \frac{\alpha_i}{u_i \zeta}, \forall i \in [d]$ .

Proof of Theorem 3. The gradient of (19) is

$$\nabla_{u_i} f^* = -\tau_i \varphi - \frac{1 - \alpha_i}{u_i}, \ \forall i \in [d], \quad \nabla_w f^* = -\frac{1}{\zeta} - \frac{1}{w},$$

and the corresponding Hessian is

$$\nabla_{u_{i},u_{j}}^{2} f^{*} = -\varphi w \tau_{i} \tau_{j} + \delta_{i,j} \left( \frac{\tau_{i} \varphi}{u_{i}} + \frac{1 - \alpha_{i}}{u_{i}^{2}} \right), \quad \forall i, j \in [d],$$

$$\nabla_{u_{i},w}^{2} f^{*} = \frac{\tau_{i} \varphi}{\zeta}, \quad \forall i \in [d],$$

$$\nabla_{w,w}^{2} f^{*} = \frac{1}{\zeta^{2}} + \frac{1}{w^{2}}.$$
(26)

Our choices for  $\theta$ ,  $p_0$ ,  $p_1$ ,  $q_0$ ,  $r_1$  satisfy

$$p_0^2 - q_0^2 = \varphi(\varphi - \zeta) = -\varphi w, \quad p_0 p_1 = \varphi, \quad \theta + \frac{p_1^2}{\zeta^2} - \frac{r_1^2}{\zeta^2} = \frac{1}{\zeta^2} + \frac{1}{w^2},$$

which match coefficients in the dual Hessian (26). Since  $D - qq^{\top} - rr^{\top}$  is  $2 \times 2$  block diagonal, the condition  $D - qq^{\top} - rr^{\top} \succ 0$  is equivalent to  $\theta - \frac{r_1^2}{\zeta^2} = \frac{1}{w^2} > 0$  and

$$D_1 - q_0^2 \cdot \tau \tau^{\top} \succ 0, \iff 1 - \tau^{\top} D_1^{-1} \tau > 0,$$

which can be shown by writing

$$1 - \tau^{\top} D_1^{-1} \tau = 1 - \sum_{i \in [d]} \frac{q_0^2 \tau_i^2}{\frac{\tau_i \varphi}{u_i} + \frac{1 - \alpha_i}{u_i^2}} = 1 - \sum_{i \in [d]} \frac{\alpha_i^2 q_0^2}{\zeta(\alpha_i \varphi + (1 - \alpha_i)\zeta)}$$
$$> 1 - \sum_{i \in [d]} \frac{\alpha_i^2 q_0^2}{\zeta \alpha_i \varphi} = 1 - \frac{q_0^2}{\zeta \varphi} = 0.$$

Hence, 
$$D - qq^{\top} - rr^{\top} \succ 0$$
.

Note that the augmented-sparse structure of power mean cones is similar to the case of generalized power cones in Theorem 3, where both are composed of a diagonal matrix plus a rank-3 update and the memory requirement for the rank-3 update is equivalent to a rank-2 update due to the presence of zeros in q, r. Therefore, we can use an expanded sparse linear system as (23) with a numerically stable variant similar to (24) for interior point step equations for the power mean cone.

#### 4.2.3 Relative entropy cone

The relative entropy cone has a slightly different structure than generalized power cones and power mean cones, but it still satisfies the augmented-sparse property as in Definition 1.

**Theorem 4** The Hessian of the barrier function (20) for the relative entropy cone satisfies Definition 1 with  $n_1 = n_2 = 0$ , i.e.

$$H^*(z) = \left[ \begin{array}{ccc} \nabla^2_{u,u} f^* & \nabla^2_{u,v} f^* & \nabla^2_{u,w} f^* \\ \nabla^2_{v,u} f^* & D_{v,v} & D_{v,w} \\ \nabla^2_{w,u} f^* & D_{w,v} & D_{w,w} \end{array} \right],$$

where z =: (u, v, w) and  $H^*(z) \succ 0$  sparse for  $d \gg 1$ . The lower right  $2 \times 2$  blocks of  $H^*(z)$  are given by the diagonal matrices

$$D_{v,v} = \operatorname{Diag}\left(\left[\frac{u(\gamma_1 + u)}{\gamma_1^2 v_1^2} + \frac{1}{v_1^2}, \dots, \frac{u(\gamma_d + u)}{\gamma_d^2 v_d^2} + \frac{1}{v_d^2}\right]\right),$$

$$D_{w,w} = \operatorname{Diag}\left(\left[\frac{1}{\gamma_1^2}, \dots, \frac{1}{\gamma_d^2}\right]\right),$$

$$D_{v,w} = D_{w,v} = \operatorname{Diag}\left(\left[\frac{u}{\gamma_1^2 v_1}, \dots, \frac{u}{\gamma_d^2 v_d}\right]\right),$$

$$v_{v,w} = \left[\frac{u}{v_1^2}, \dots, \frac{u}{v_d^2}\right], \quad (27)$$

with  $\gamma_i = w_i - u \ln\left(\frac{u}{v_i}\right) + u > 0, \forall i \in [d].$ 

*Proof of Theorem 4.* The proof is straightforward. The gradient of (20) is

$$\nabla_{u}f^{*} = \sum_{i \in [d]} \frac{\tau_{i}}{\gamma_{i}} - \frac{d}{u},$$

$$\nabla_{v_{i}}f^{*} = -\frac{u}{\gamma_{i}v_{i}} - \frac{1}{v_{i}}, \quad \forall i \in [d],$$

$$\nabla_{w_{i}}f^{*} = -\frac{1}{\gamma_{i}}, \quad \forall i \in [d],$$

where Let  $\tau_i = \ln\left(\frac{u}{v_i}\right), \forall i \in [d]$ . The Hessian of (20) is

$$\begin{split} \nabla^2_{u,u} f^* &= \frac{d}{u^2} + \sum_{i \in [d]} \frac{1}{u \gamma_i} + \sum_{i \in [d]} \left(\frac{\tau_i}{\gamma_i}\right)^2, \\ \nabla^2_{v_i,u} f^* &= \nabla^2_{u,v_i} f^* = -\frac{1}{\gamma_i v_i} - \frac{u \tau_i}{\gamma_i^2 v_i}, \quad \forall i \in [d], \\ \nabla^2_{w_i,u} f^* &= \nabla^2_{u,w_i} f^* = -\frac{\tau_i}{\gamma_i^2}, \quad \forall i \in [d], \\ \nabla^2_{v_i,v_j} f^* &= \delta_{i,j} \left(\frac{u (\gamma_i + u)}{\gamma_i^2 v_i^2} + \frac{1}{v_i^2}\right), \quad \forall i, j \in [d], \\ \nabla^2_{w_i,w_j} f^* &= \delta_{i,j} \frac{1}{\gamma_i^2}, \quad \forall i, j \in [d], \\ \nabla^2_{v_i,w_j} f^* &= \delta_{i,j} \frac{u}{\gamma_i^2 v_i}, \quad \forall i, j \in [d]. \end{split}$$

# 5 Functional Proximity Measures

After obtaining a search direction  $(\Delta x^k, \Delta s^k, \Delta z^k)$ , we need to compute the step size  $\alpha$  such that the new iterate  $(x^{k+1}, s^{k+1}, z^{k+1}) = (x^k, s^k, z^k) + \alpha(\Delta x^k, \Delta s^k, \Delta z^k)$  satisfies conic feasibility and stays close to the central path. The latter condition is crucial for the convergence of an IPM. There are several ways of defining the proximity to the central path depending on the scaling strategy used in an IPM [NT98, DA21, CKV22].

In this paper we use the functional proximity measure [NT98, Ser15],

$$\Phi(s,z) := \nu \ln(s^{\top}z) + f(s) + f^{*}(z) - \nu \ln(\nu) + \nu, \tag{28}$$

for the proximity measurement, which requires evaluation of *both* the barrier f and its conjugate  $f^*$ . The sequence  $(s^k, z^k)$  generated by an IPM converges to a global optimum if  $\Phi(s^k, z^k) \leq 1, \forall k$ . The property of  $\Phi(s, z)$  is mentioned in Lemma 5.2.2 and the connection between them is discussed in Theorem 7.0.6 of [Ser15].

For symmetric cones, the conjugate barrier function is the same as the original one, i.e.  $f = f^*$ . Hence, we have analytical solutions for both  $f(s^k)$  and  $f^*(z^k)$  at each interior point iteration. For nonsymmetric cones, we generally will not have analytical expressions for both f(s) and  $f^*(z)$ , but will rather have one with an analytical solution and the other computed iteratively, which amounts to computation of the conjugate gradient.

# 5.1 Computation of conjugate gradient

For an interior point algorithm in which the dual barrier value  $f^*(z)$  is available from Theorem 1,the corresponding primal barrier value f(s) can be computed via the dual barrier function via (6), i.e. using

$$f(s) = -\nu - f^*(-g(s)).$$

Hence, the main effort is to compute the primal conjugate gradient via (5), i.e. using

$$g(s) = \arg \sup_{w \in \mathcal{K}^*} \{ -\langle s, w \rangle - f^*(w) \}.$$
 (29)

We next detail how to compute the conjugate gradient of dual barrier function efficiently for generalized power cones, power mean cones and relative entropy cones.

#### 5.1.1 Generalized power cone

The barrier function of  $\mathcal{K}^*_{\text{gpow }(\alpha,d_1,d_2)}$  is defined in (18) with  $\nu = d_1 + 1$ . We denote  $\varphi(u) = \prod_{i \in [d_1]} (u_i/\alpha_i)^{2\alpha_i}$ , and  $\zeta(u,w) = \varphi(u) - ||w||^2$ . Our goal is to find a conjugate gradient  $g_{(p,r)}$  such that  $-g^*(-g_{(p,r)}) = (p,r) \in \mathcal{K}_{\text{gpow}}(\alpha,d_1,d_2)$ , which is summarized as follows:

**Theorem 5** Suppose  $(p,r) \in \mathcal{K}_{\text{gpow}(\alpha,d_1,d_2)}$ . The corresponding primal conjugate gradient  $g_{(p,r)} = (g_p, g_r)$  can be obtained via

$$g_{p_i} = -\frac{(1+\alpha_i) + \alpha_i g_{\|r\|} \|r\|}{p_i}, \ \forall i \in [d_1], \qquad g_r = \frac{g_{\|r\|} \cdot r}{\|r\|},$$

where  $g_{||r||}$  is the root of

$$h(x) = \sum_{i \in [d_1]} 2\alpha_i \ln\left(\|r\| \cdot x + \frac{1 + \alpha_i}{\alpha_i}\right) - \ln\left(\frac{2x}{\|r\|} + x^2\right) - \sum_{i \in [d_1]} 2\alpha_i \ln(p_i).$$
 (30)

The root of h(x) is computable via the 1-dimensional Newton-Raphson method, which converges quadratically if we choose the initial point  $x_0$  as

$$x_0 = -\frac{1}{\|r\|} + \frac{\psi \|r\| + \sqrt{\chi \left(\frac{\chi}{\|r\|^2} + \psi^2 - 1\right)}}{\chi - \|r\|^2},$$

where  $\chi = \prod_{i \in [d_1]} p_i^{2\alpha_i}, \psi = 1/(\sum_{i \in [d_1]} \alpha_i^2)$ .

*Proof of Theorem 5.* Following the preceding discussion, the dual gradient is

$$g_{u_i}^* = \frac{-2\alpha_i \varphi(u)}{u_i \zeta(u, w)} - \frac{1 - \alpha_i}{u_i},\tag{31a}$$

$$g_{w_i}^* = \frac{2w_i}{\zeta(u, w)}. (31b)$$

If r = 0, we have  $-\frac{2(-g_{r_i})}{\zeta(p,r)} = r_i = 0, \forall i \in [d_2]$  and then

$$g_{p_i} = -\frac{1 + \alpha_i}{p_i}, \quad \forall i \in [d_1],$$
  
$$g_{r_i} = 0, \quad \forall i \in [d_2],$$

from  $\varphi(-g_p) = \zeta(-g_p, 0)$ .

Suppose instead  $r \neq 0$ . We will denote  $\zeta = \zeta(-g_p, -g_r)$  for simplicity of notation and will assume  $r \geq 0$  without loss of generality. Suppose Q is an orthonormal transformation that maps  $r \in \mathbb{R}^{d_2}$  to a vector of zeros except one at the first entry that is equal to ||r||, i.e.  $[||r||, 0, \ldots]^{\top}$ . Then the conjugate gradient under the transformation Q satisfies  $g_r = Q^{\top} g_{Qr}$  due to (29), where the first entry of  $g_{Qr}$  is  $g_{||r||}$  and all other entries are zeros. From (31b) and (7), have

$$g_r = \frac{\zeta r}{2}.\tag{32}$$

Due to the invariance of  $\zeta$  under orthonormal transformations, we then obtain

$$\zeta := \zeta(-g_p, -g_r) = \zeta(-g_p, -Q^{\top}g_{Qr}) = \zeta(-g_p, [-g_{\parallel r \parallel}; 0; \dots]) \stackrel{\text{(32)}}{=} \frac{2g_{\parallel r \parallel}}{\parallel r \parallel},$$

which can be substituted back into (32) to produce

$$g_r = \frac{g_{\|r\|} \cdot r}{\|r\|}.$$

From (31a) and (7) we can write the elements of the subvector p as

$$p_i = -\left(\frac{-2\alpha_i \varphi(-g_p)}{-g_{p_i}\zeta} - \frac{1-\alpha_i}{-g_{p_i}}\right), \ \forall i \in [d_1].$$
(33)

which can be rearranged to

$$p_i(-g_{p_i})\zeta = 2\alpha_i \varphi(-g_p) + (1 - \alpha_i)\zeta. \tag{34}$$

Applying  $\varphi(\cdot)$  over both sides of (34) for all i, we obtain

$$\prod_{i \in [d_1]} \alpha_i^{2\alpha_i} \cdot \varphi(p)\varphi(-g_p)\zeta^2 = \varphi(2\alpha_i\varphi(-g_p) + (1-\alpha_i)\zeta)$$

Substituting

$$\varphi(-g_p) = \zeta + \|g_r\|^2 = \zeta + \|Q^{\top}g_{Qr}\|^2 = \zeta + g_{\|r\|}^2 = \frac{2g_{\|r\|}}{\|r\|} + g_{\|r\|}^2$$
(35)

yields

$$\Longrightarrow \prod_{i \in [d_{1}]} \alpha_{i}^{2\alpha_{i}} \cdot \varphi(p) \left( \frac{2g_{\|r\|}}{\|r\|} + g_{\|r\|}^{2} \right) \left( \frac{2g_{\|r\|}}{\|r\|} \right)^{2} = \varphi \left( 2\alpha_{i} \left( \frac{2g_{\|r\|}}{\|r\|} + g_{\|r\|}^{2} \right) + (1 - \alpha_{i}) \frac{2g_{\|r\|}}{\|r\|} \right)$$

$$\Longrightarrow \prod_{i \in [d_{1}]} p_{i}^{2\alpha_{i}} \left( \frac{2g_{\|r\|}}{\|r\|} + g_{\|r\|}^{2} \right) \cdot \frac{4g_{\|r\|}^{2}}{\|r\|^{2}} = \prod_{i \in [d_{1}]} \left( 2g_{\|r\|}^{2} + \frac{(1 + \alpha_{i})}{\alpha_{i}} \frac{2g_{\|r\|}}{\|r\|} \right)^{2\alpha_{i}}$$

$$\Longrightarrow \prod_{i \in [d_{1}]} p_{i}^{2\alpha_{i}} \left( \frac{2g_{\|r\|}}{\|r\|} + g_{\|r\|}^{2} \right) = \prod_{i \in [d_{1}]} \left( g_{\|r\|} \cdot \|r\| + \frac{1 + \alpha_{i}}{\alpha_{i}} \right)^{2\alpha_{i}},$$

where computing  $g_{\|r\|}$  reduces to finding the root of the function h(x) defined in (30).

We next show that h(x) is convex and monotonically decreasing. The gradient and Hessian of h are

$$h'(x) = \sum_{i \in [d_1]} \frac{2\alpha_i \|r\|}{\|r\| \cdot x + \frac{1+\alpha_i}{\alpha_i}} - \frac{2(\|r\| + \frac{1}{x})}{\|r\| \cdot x + 2}$$

$$< \sum_{i \in [d_1]} \frac{2\alpha_i \|r\|}{\|r\| \cdot x + 2} - \frac{2(\|r\| + \frac{1}{x})}{\|r\| \cdot x + 2} = -\frac{\frac{2}{x}}{\|r\| \cdot x + 2} < 0$$

$$h''(x) = -\sum_{i \in [d_1]} \frac{2\alpha_i \|r\|^2}{(\|r\| \cdot x + \frac{1+\alpha_i}{\alpha_i})^2} - \frac{-\frac{2}{x^2}(\|r\| \cdot x + 2) - 2(\|r\| + \frac{1}{x})\|r\|}{(\|r\| \cdot x + 2)^2}$$

$$= -\sum_{i \in [d_1]} \frac{2\alpha_i^3}{(\alpha_i x + \frac{1+\alpha_i}{\|r\|})^2} + \frac{\frac{4\|r\|}{x} + \frac{4}{x^2} + 2\|r\|^2}{(\|r\| \cdot x + 2)^2}.$$

Since h'(x) < 0 holds for x > 0 and  $h(\hat{x}) > 0$ , the root of h is unique over x > 0.

Consider next the function  $g(\beta) = -\frac{\beta^3}{\left(\beta x + \frac{1+\beta}{\|r\|}\right)^2}$ , which is concave over the interval [0, 1] when x > 0. We also have g(0) = 0, which implies

$$g(t\beta) \ge tg(\beta) + (1-t)g(0) = tg(\beta), \ \forall t, \beta \in [0,1].$$

Therefore,

$$h''(x) = 2\sum_{i \in [d_1]} g(\alpha_i) + \frac{\frac{4||r||}{x} + \frac{4}{x^2} + 2||r||^2}{(||r|| \cdot x + 2)^2} \ge 2\sum_{i \in [d_1]} \alpha_i g(1) + \frac{\frac{4||r||}{x} + \frac{4}{x^2} + 2||r||^2}{(||r|| \cdot x + 2)^2}$$
$$= -\frac{2}{(x + \frac{2}{||r||})^2} + \frac{\frac{4||r||}{x} + \frac{4}{x^2} + 2||r||^2}{(||r|| \cdot x + 2)^2} = \frac{\frac{4||r||}{x} + \frac{4}{x^2}}{(||r|| \cdot x + 2)^2} > 0$$

shows h is convex over x > 0. Since h is convex and decreasing, the root of h(x) is unique and a root finding Newton-Raphson method converges quadratically if it starts with  $0 < x_0 \le \hat{x}$  where  $h(\hat{x}) > 0$  [SM03, Thm. 1.9]. Since  $q(t) = \ln(\|r\| \cdot x + \frac{1}{t} + 1)$  is convex when  $t \in [0, 1], x \ge 0$ , we can apply the Jensen's inequality and obtain

$$h(x) \ge 2\ln(\|r\| \cdot x + \frac{1}{\sum_{i \in [d_1]} \alpha_i^2} + 1) - \ln\left(\frac{2x}{\|r\|} + x^2\right) - \sum_{i \in [d_1]} 2\alpha_i \ln(p_i),$$

where a positive root of the right part above is

$$\hat{x} = -\frac{1}{\|r\|} + \frac{\psi \|r\| + \sqrt{\chi \left(\frac{\chi}{\|r\|^2} + \psi^2 - 1\right)}}{\chi - \|r\|^2}$$

with  $\chi = \prod_{i \in [d_1]} p_i^{2\alpha_i}$ ,  $\psi = 1/(\sum_{i \in [d_1]} \alpha_i^2)$ , which implies  $h(\hat{x}) \geq 0$  and we can set it as the starting point of the Newton-Raphson method. After obtaining  $g_{\|r\|}$  from (30), we get  $\zeta = \frac{2g_{\|r\|}}{\|r\|}$  and the primal conjugate gradient

$$g_r = \frac{g_{\|r\|} \cdot r}{\|r\|}, \quad g_{p_i} = -\frac{1}{p_i \zeta} \left( (1 - \alpha_i) \zeta + 2\alpha_i (\zeta + g_{\|r\|}^2) \right) = -\frac{1}{p_i} (1 + \alpha_i + \alpha_i g_{\|r\|} \|r\|),$$

via (33) and (35).

#### 5.1.2 Power mean cone

For the dual barrier function  $f_{\text{powm}}^*(u, w)$  defined in (19), we denote  $\varphi(u) = \prod_{i \in [d]} \left(\frac{u_i}{\alpha_i}\right)^{\alpha_i}$ ,  $\zeta(u, w) = \varphi(u) + w$  and  $\tau_i = \frac{\alpha_i}{u_i \zeta}$ ,  $\forall i \in [d]$ . The computation of the primal conjugate gradient  $g_{(p,r)}$  is summarized in the theorem below.

**Theorem 6** Suppose  $(p,r) \in \mathcal{K}_{powm(\alpha,d)}$ . The corresponding primal conjugate gradient  $g_{(p,r)} = (g_p, g_r)$  can be obtained via

$$g_{p_i} = -\frac{1}{p_i} - \frac{\alpha_i}{p_i} \left( 1 + rg_r \right),$$

where  $g_r$  can be computed numerically depending on the sign of r:

1.  $\mathbf{r} > \mathbf{0}$ : We can obtain  $g_r = 1/\hat{x}$ , where  $\hat{x}$  is the root of

$$h(x) = \sum_{i \in [d]} \alpha_i \ln \left( \frac{1 + \alpha_i}{\alpha_i p_i} \cdot x + \frac{r}{p_i} \right) - \ln \left( 1 + \frac{1}{r + x} \right).$$

The root of h(x) can be computed via the 1-dimensional Newton-Raphson method, which converges quadratically if we start from  $x_0 = 0$ .

2.  $\mathbf{r} < \mathbf{0}$ : We can obtain  $g_r = \left(\varphi - \sqrt{\varphi^2 + 4/r^2}\right)/2 - 1/r$ , where  $\varphi = 1/\hat{x}$  and  $\hat{x}$  is the root of

$$h(x) = \sum_{i \in [d]} \alpha_i \ln \left( \frac{x}{\alpha_i p_i} + \frac{r + \sqrt{r^2 + 4x^2}}{2p_i} \right).$$

We can compute the root of h(x) via the 1-dimensional Newton-Raphson method, which converges quadratically if we start from  $x_0 = \left(\frac{\alpha_i p_i}{1+\alpha_i}\right)^{\alpha_i}$ .

3.  $\mathbf{r} = \mathbf{0}$ : We have

$$g_{p_i} = -\frac{1}{p_i} - \frac{\alpha_i}{p_i}, \qquad g_r = \varphi(-g_p)/2.$$

*Proof of Theorem 6.* The dual gradient of (19) is

$$g_u^* = -\tau_i \varphi - \frac{1 - \alpha_i}{u_i}$$
$$g_w^* = -\frac{1}{\zeta} - \frac{1}{w}.$$

Our goal is to find  $g_{(p,r)}$  such that  $-g^*\left(-g_{(p,r)}\right) = \begin{pmatrix} p \\ r \end{pmatrix} \in \mathcal{K}_{powm}$ , which is

$$p = -\frac{\alpha_i \varphi(-g_{(p,r)})}{g_{p_i} \zeta(-g_{(p,r)})} - \frac{1 - \alpha_i}{g_{p_i}} = -\frac{1}{g_{p_i}} - \frac{\alpha_i g_r}{g_{p_i} \zeta(-g_{(p,r)})},$$

$$r = \frac{1}{\zeta(-g_{(p,r)})} - \frac{1}{g_r}$$

with  $g_r > 0$ . We then obtain the primal gradient  $g_{(p,r)}$  as

$$g_{p_i} = -\frac{1}{p_i} - \frac{\alpha_i}{p_i} (1 + rg_r),$$

$$\zeta(-g_{(p,r)}) = \frac{1}{r + \frac{1}{q_r}} = \frac{g_r}{1 + rg_r} > 0.$$
(36)

Due to the definition of  $\zeta(\cdot)$  and the two equalities above, we obtain

$$\frac{g_r}{1 + rg_r} = \prod_{i \in [d]} \left( \frac{1}{\alpha_i p_i} + \frac{1}{p_i} (1 + rg_r) \right)^{\alpha_i} - g_r.$$
 (37)

We compute  $g_p, g_r$  in three different ways based on the value of r:

1. When r > 0, (37) reduces to finding the root  $\hat{x}$  of h(x) if we regard  $1/g_r$  as x,

$$h(x) := \sum_{i \in [d]} \alpha_i \ln \left( \frac{1 + \alpha_i}{\alpha_i p_i} \cdot x + \frac{r}{p_i} \right) - \ln \left( 1 + \frac{1}{r + x} \right).$$

The gradient and the hessian of h(x) are

$$h'(x) = \sum_{i \in [d]} \frac{\alpha_i}{x + \frac{\alpha_i r}{1 + \alpha_i}} + \frac{1}{(r+x)(r+x+1)} > 0,$$
  
$$h''(x) = -\sum_{i \in [d]} \frac{\alpha_i}{(x + \frac{\alpha_i r}{1 + \alpha_i})^2} - \frac{1}{(r+x)^2} + \frac{1}{(r+x+1)^2} < 0,$$

when  $x \geq 0$ , which implies h(x) is concave and increasing. Moreover, we have

$$h(0) = \sum_{i \in [d]} \alpha_i \ln\left(\frac{r}{p_i}\right) - \ln\left(1 + \frac{1}{r}\right) = \ln\left(\frac{r^2}{(1+r)\prod_{i=1}^d p_i^{\alpha_i}}\right) \le \left(\frac{r^2}{(1+r)r}\right) < 0.$$

Hence, the root of h(x) is unique and a root finding Newton-Raphson method converges quadratically when it starts with  $x_0 = 0$  (Theorem 1.9 in [SM03]). Finally, we obtain the primal gradient  $g_{(p,r)}$  via (36), given  $g_r = 1/\hat{x}$ .

2. When r < 0, we design a different strategy to compute the conjugate gradient. Due to the definition of  $\varphi(\cdot)$  and (36), we have

$$\varphi(-g_p) = \frac{g_r}{1 + rq_r} + g_r,$$

which (abbreviating  $\varphi(-g_p)$  as  $\varphi$ ) can be written as

$$g_r = \frac{\varphi - \sqrt{\varphi^2 + \frac{4}{r^2}}}{2} - \frac{1}{r}$$
 (38)

given  $g_r \ge -1/r$ . Substituting (38) into (37) yields

$$\varphi = \prod_{i \in [d]} \left( \frac{1}{\alpha_i p_i} + \frac{r\varphi + \sqrt{r^2 \varphi^2 + 4}}{2p_i} \right)^{\alpha_i},$$

which is equivalent to finding the root of h(x) if we regard  $1/\varphi$  as x, where

$$h(x) = \sum_{i \in [d]} \alpha_i \ln \left( \frac{x}{\alpha_i p_i} + \frac{r + \sqrt{r^2 + 4x^2}}{2p_i} \right). \tag{39}$$

We can obtain

$$h'(x) = \sum_{i \in [d]} \alpha_i \cdot \frac{1 + \frac{2\alpha_i x}{\sqrt{r^2 + 4x^2}}}{x + \frac{\alpha_i}{2} (r + \sqrt{r^2 + 4x^2})} > 0$$

and h''(x) < 0 (See Appendix B) when x > 0. Moreover,

$$h(x_0) = \sum_{i \in [d]} \alpha_i \ln \left( \frac{x_0}{\alpha_i p_i} + \frac{2x_0^2}{p_i (\sqrt{r^2 + 4x_0^2 - r})} \right) < \sum_{i \in [d]} \alpha_i \ln \left( \frac{x_0}{\alpha_i p_i} + \frac{x_0}{p_i} \right)$$

$$\leq \ln(x_0) + \sum_{i \in [d]} \alpha_i \ln \left( \frac{1 + \alpha_i}{\alpha_i p_i} \right) = 0.$$

Hence, the root  $\hat{x}$  of h(x) is unique and a root finding Newton-Raphson method converges quadratically when it starts with  $x_0 = \left(\frac{\alpha_i p_i}{1+\alpha_i}\right)^{\alpha_i}$  with  $h(x_0) < 0$ . Given  $\varphi = 1/\hat{x}$ , we obtain the primal gradient  $g_r$  via (38) and  $g_p$  (36).

3. When r = 0, (36) becomes

$$g_{p_i} = -\frac{1}{p_i} - \frac{\alpha_i}{p_i}, \qquad \zeta(-g_{(p,r)}) = g_r.$$

Then,  $\varphi(-g_p) - g_r = \zeta(-g_{(p,r)}) = g_r$  due to the definition of  $\varphi(\cdot)$ , which yields  $g_r = \varphi(-g_p)/2$ .

## 5.1.3 Relative entropy cone

For the barrier function  $f_{\text{rel}}^*(u, v, w)$  defined in (20), the primal conjugate gradient can be obtained from the following theorem.

**Theorem 7** Suppose we have a point  $(p,q,r) \in \mathcal{K}_{rel}$  where  $p \in \mathbb{R}, q, r \in \mathbb{R}^d$ . The corresponding primal conjugate gradient  $g_{(p,q,r)} = (g_p, g_q, g_r)$  can be obtained via

$$g_{q_i} = \frac{g_p r_i - 1}{q_i}, \qquad g_{r_i} = g_p \ln \left(\frac{g_p}{g_{q_i}}\right) - g_p - \frac{1}{r_i}, \ \forall i \in [d],$$

where  $g_p = -1/\hat{x}$ .  $\hat{x}$  is the root of

$$h(x) := d \cdot x + \sum_{i \in [d]} r_i \ln \left( \frac{r_i}{q_i} + \frac{x}{q_i} \right) - p,$$

which can be solved via the 1-dimensional Newton-Raphson method, which converges quadratically if we choose the initial point  $x_0 = 0$ .

Proof of Theorem 7. We aim to find the conjugate gradient  $g_{(p,q,r)}$  such that  $-g^*\left(-g_{(p,q,r)}\right) = (p,q,r) \in \mathcal{K}_{rel}$ , which becomes

$$p = -\sum_{i \in [d]} \frac{\ln\left(\frac{g_p}{g_{q_i}}\right)}{\gamma_i} - \frac{d}{g_p},$$

$$q_i = \frac{g_p}{\gamma_i g_{q_i}} - \frac{1}{g_{q_i}},$$

$$r_i = \frac{1}{\gamma_i},$$

where  $\gamma_i$  is the abbreviation of  $\gamma_i(-g_{(p,q,r)})$  defined in Section 4.2.3. We can then obtain

$$-g_{q_i} = \frac{-g_p r_i + 1}{q_i}, \qquad \gamma_i = \frac{1}{r_i}.$$
 (40)

with

$$p = -\frac{d}{g_p} - \sum_{i \in [d]} r_i \ln \left( \frac{g_p q_i}{g_p r_i - 1} \right)$$
$$= -\frac{d}{g_p} + \sum_{i \in [d]} r_i \ln \left( \frac{r_i}{q_i} - \frac{1}{g_p q_i} \right),$$

which is a nonlinear equation w.r.t.  $g_p$ . Suppose we define h(x) as

$$h(x) := d \cdot x + \sum_{i \in [d]} r_i \ln \left( \frac{r_i}{q_i} + \frac{x}{q_i} \right) - p.$$

We can compute the root  $\hat{x}$  of h(x) and  $g_p$  is indeed  $-\hat{x}^{-1}$ . Since h(x) is concave and increasing, the root of g(x) is unique and a root finding Newton-Raphson method converges quadratically when it starts with  $0 \le x_0 \le \hat{x}$  where  $g(x_0) < 0$  (Theorem 1.9 in [SM03]). A feasible choice is  $x_0 = 0$  as  $h(0) = \sum_{i \in [d]} r_i \ln(r_i/q_i) < p$  due to  $(p, q, r) \in \mathcal{K}_{rel}$ . Finally, we can obtain  $g_{q_i}, g_{r_i}, \forall i \in [d]$  via (40) and

$$g_{r_i} = g_p \ln \left(\frac{g_p}{g_{q_i}}\right) - g_p - \frac{1}{r_i}$$

due to the definition of  $\gamma_i$ .

# 6 A dual-scaling IPM for nonsymmetric conic optimization

We summarize the implementation of our dual-scaling IPM algorithm in this section.

# 6.1 Initialization of $s^0, z^0$

The initialization of symmetric cones and three-dimensional nonsymmetric cones, i.e. power and exponential cones, follows the Mosek setting as discussed in [DA21], where  $x^0 = 0$ ,  $\kappa^0 = \tau^0 = 1$  and  $s^0$ ,  $z^0$  are set to

$$s^0 = z^0 = -g^*(z^0), (41)$$

which are on the central path with  $\mu^0 = 1$ . This strategy holds for  $\mathcal{K}_{\text{gpow}(\alpha,d_1,d_2)}$ , and we initialize  $s^0, z^0$  by

$$s^{0} = z^{0} = \left( \left( \sqrt{1 + \alpha_{i}} \right)_{i \in [d_{1}]}, 0_{d_{2}} \right).$$

For  $\mathcal{K}_{\text{powm}(\alpha,d)}$ ,  $\mathcal{K}_{\text{rel}}$ , we can not find  $s^0, z^0$  satisfying (41), so we initialize  $s^0, z^0$  as in Hypatia [CKV22], where  $s^0, z^0$  satisfy  $s^0 = -g^*(z^0)$ . Note that  $\mu^0 = 1$  still holds in this setting.

# 6.2 Algorithmic sketch

Our dual-scaling IPM proceeds as follows at each iteration k:

1. Update residuals, gap and check termination or infeasibility: We compute residuals by

$$\begin{split} r_x &= -G^\top y^k - A^\top z^k - c\tau^k, & r_y &= Gx^k - h\tau^k, \\ r_\tau &= \kappa^k + c^\top x^k + h^\top y^k + b^\top z^k, & r_z &= s^k + Ax^k - b\tau^k, \end{split}$$

and  $\mu^k = \frac{s^{k^\top}z^k + \kappa^k\tau^k}{\nu+1}$ . The computation of duality gap, the termination check and the infeasibility detection follows ECOS [DCB13].

2. Affine step: The affine direction (predictor) is the solution of

$$\begin{bmatrix} 0 & G^{\top} & A^{\top} & c \\ -G & 0 & 0 & h \\ -A & 0 & 0 & b \\ -c^{\top} & -h^{\top} & -b^{\top} & 0 \end{bmatrix} \begin{bmatrix} \Delta x_a \\ \Delta y_a \\ \Delta z_a \\ \Delta \tau_a \end{bmatrix} - \begin{bmatrix} 0 \\ 0 \\ \Delta s_a \\ \Delta \kappa_a \end{bmatrix} = \begin{bmatrix} r_x \\ r_y \\ r_z \\ r_{\tau} \end{bmatrix}$$
$$\mu^k H^*(z^k) \Delta z_a + \Delta s_a = -s^k, \quad \tau^k \Delta \kappa_a + \kappa^k \Delta \tau_a = -\tau^k \kappa^k$$

which tries to remove residuals of the linearized model at iteration k. We can compute the maximal step size  $\alpha_a$  such that  $(s^k + \alpha_a \Delta s_a, z^k + \alpha_a \Delta z_a, \tau^k + \alpha_a \Delta \tau_a, \kappa^k + \alpha_a \Delta \kappa_a)$  resides in  $\mathcal{F}$ .

3. Update the sparse decomposition of  $H^*(z)$ : We update the augmented sparse decomposition of the dual Hessian  $H^*(z)$  for nonsymmetric cones as discussed in Section 3 and Section 4.

4. Combined step: The weight of centrality  $\sigma$  is set to  $(1 - \alpha_a)^3$  empirically and then used for the computation of centering direction (corrector) via

$$\begin{bmatrix} 0 & G^{\top} & A^{\top} & c \\ -G & 0 & 0 & h \\ -A & 0 & 0 & b \\ -c^{\top} & -h^{\top} & -b^{\top} & 0 \end{bmatrix} \begin{bmatrix} \Delta x_c \\ \Delta y_c \\ \Delta z_c \\ \Delta \tau_c \end{bmatrix} - \begin{bmatrix} 0 \\ 0 \\ \Delta s_c \\ \Delta \kappa_c \end{bmatrix} = (1 - \sigma) \begin{bmatrix} r_x \\ r_y \\ r_z \\ r_{\tau} \end{bmatrix}$$
$$\mu^k H^*(z^k) \Delta z_c + \Delta s_c = -s^k - \sigma \mu^k g^*(z^k) - \eta(\Delta s_a, \Delta z_a),$$
$$\tau^k \Delta \kappa_c + \kappa^k \Delta \tau_c = -\tau^k \kappa^k + \sigma \mu^k - \Delta \tau_a \Delta \kappa_a,$$

where  $\Delta \tau_a \Delta \kappa_a$  and the function  $\eta(\Delta s_a, \Delta z_a)$  are higher-order corrections given information from the affine directions. We only use Mehrotra's correction [Meh92] for symmetric cones. Likewise, we compute the largest step size  $\alpha_c$  ensuring  $(s^k + \alpha_c \Delta s_c, z^k + \alpha_c \Delta z_c, \tau^k + \alpha_c \Delta \tau_c, \kappa^k + \alpha_c \Delta \kappa_c)$  stays inside  $\mathcal{F}$  and the proximity measure satisfies  $\Phi(s^k + \alpha_c \Delta s_c, z^k + \alpha_c \Delta z_c) \leq 1$  as discussed in Section 5.

5. Update iterates: At the end of each iteration k, we obtain the new iterate  $(x^{k+1}, y^{k+1}, s^{k+1}, z^{k+1}, x^{k+1}, \tau^{k+1})$  by

$$(x^{k+1}, y^{k+1}, s^{k+1}, z^{k+1}, \kappa^{k+1}, \tau^{k+1}) := (x^k, y^k, s^k, z^k, \kappa^k, \tau^k) + \alpha_c(\Delta x_c, \Delta y_c, \Delta s_c, \Delta z_c, \Delta \kappa_c, \Delta \tau_c).$$

# 7 Experiments

In our experiments, we choose two examples from Hypatia [CKV23] to show the effectiveness of our sparse implementation for generalized power cones, the discrete maximum likelihood and the maximum volume hypercube problems. We test these examples with 4 different solver configurations:

- 1. Clarabel-GenPow is the proposed sparse implementation of generalized power cones in Clarabel;
- 2. Clarabel-Pow uses Clarabel but transforms the generalized power cone to a product of three-dimensional power cones as formulated in [Cha09, ApS21];
- 3. Hypatia uses the Hypatia solver, which supports the generalized power cone constraints directly;
- 4. Mosek denotes the use of Mosek with transforming the generalized power cone to a product of three-dimensional power cones.

The convergence tolerance is set to  $\epsilon = 10^{-8}$  for all solvers in all tests. We utilize the dual scaling strategy (same as ECOS) for Clarabel in our test for fair comparison between the direct support of generalized power cones (configuration (1)) and the transformation to a product of three-dimensional power cones (configuration (2)).

We also test the sparse implementation of the power mean cone, called Clarabel-PowM, on these same two examples, since it produces the same results as the generalized power cone for both cases.

In addition, we choose the entropy maximization problem from Mosek's Cookbook [ApS21] for testing relative entropy cones. We test it with 4 different solver settings:

- 1. Clarabel-Entropy is the proposed sparse implementation of relative entropy cones via Clarabel;
- 2. Clarabel-Exp uses Clarabel but transforms the relative entropy cone to a product of three-dimensional exponential cones;
- 3. ECOS uses the ECOS solver, which supports three-dimensional exponential cones;
- 4. Mosek denotes the use of Mosek solver with three-dimensional exponential cones.

The termination accuracy is set to be  $\epsilon = 10^{-8}$  and both setting (1) and (2) use the dual scaling strategy for this test.

#### 7.1 Discrete maximum likelihood

### 7.1.1 Test of generalized power cones

The discrete maximum likelihood problem has the form

s.t. 
$$\sum_{i \in [n]} x_i = 1,$$

$$(x, t) \in \mathcal{K}_{\text{gpow }(\alpha, n, 1)},$$

$$(42)$$

and we test Clarabel-GenPow and Hypatia on it directly. Following [ApS21], we can rewrite (42) by replacing the generalized power cone constraint with the products of three-dimensional power cones where  $p_i = \alpha_i / (\sum_{1 \leq j \leq i} \alpha_j)$ . This is the form used by Clarabel-Pow and Mosek,

$$\max t$$
s.t. 
$$\sum_{i \in [n]} x_i = 1, x \ge 0,$$

$$x_1^{1-p_2} x_2^{p_2} \ge |z_3|,$$

$$z_3^{1-p_3} x_3^{p_3} \ge |z_4|,$$

$$\dots$$

$$z_{n-1}^{1-p_{n-1}} x_{n-1}^{p_{n-1}} \ge |z_n|,$$

$$z_n^{1-p_n} x_n^{p_n} \ge |t|.$$
(43)

Computational results are given in Table 1 and Table 2 for results of dimension n from 100 to 2500, which record the total computational time and the number of iterations for different dimensions n of the generalized power cone respectively.

n	Clarabel-GenPow	Clarabel-Pow	Mosek	Hypatia
100	3.96	7.67	16	33
500	6.42	23.9	31	334
2500	23.7	142	125	22190

**Table 1:** Time (ms) w.r.t. dimensions (generalized power cones)

n	Clarabel-GenPow	Clarabel-Pow	Mosek	Hypatia
100	14	19	11	22
500	13	20	14	37
2500	16	24	16	67

**Table 2:** Iteration number w.r.t. dimensions (generalized power cones)

Table 1 shows that our sparse LDL implementation for generalized power cones is very efficient relative to the other methods. It can solve the problem within a few tens of milliseconds even for dimensions up to n=2500. In comparison, Clarabel-Pow and Mosek take more than 100 ms to solve it, which is about 4-5 times slower than the proposed sparse implementation in Section 4. This can be explained by the transformation of a generalized power cone to a product of three-dimensional cones. Firstly, the transformation increases the total number of variables in (43), which results in more non-zero entries in LDL factorization at each iteration. Secondly, it takes less time to compute the conjugate gradient for a generalized power cone than the computation for an equivalent collection of three-dimensional cones, and hence less time for centrality check of a generalized power cone in (42) than that in (43). Thirdly, Table 2 says Clarabel-GenPow takes fewer iteration numbers than Clarabel-Pow regardless of the choice of dimension n, which is consistent with the link between the self-concordant parameter  $\nu$  and the complexity of the total iteration number in theory, i.e.  $\mathcal{O}(\sqrt{\nu} \ln(1/\epsilon))$ .

#### 7.1.2 Test of power mean cones

For the same problem as in the previous section we can replace the generalized power cone in (42) with a power mean cone,

s.t. 
$$\sum_{i \in [n]} x_i = 1,$$
$$(x, t) \in \mathcal{K}_{powm(\alpha, n)},$$

which should yield the same solution as (42) since the optimum of (42) must be  $t \geq 0$ . Since Mosek doesn't support power mean cones directly we compare our spare implementation only with Hypatia. We vary the dimension n as for the test of generalized power cones. Table 3 shows that our sparse implementation for power mean cones is significantly faster than Hypatia which is based on the Cholesky decomposition on the reduced normal system,

both in total time and the averaged time per iteration. Although the results are slower than our sparse implementation of generalized power cones, it still performs better than other solvers on the discrete maximum likelihood problem. This demonstrates the efficacy of our sparse augmentation approach. Compared to  ${\tt Hypatia}$ ,  ${\tt Clarabel-PowM}$  also scales well when the dimension n becomes large.

n	Clarabel-PowM	Hypatia
100	4.4	23
500	10.8	320
2500	49.1	20926

n	Clarabel-PowM	Hypatia
100	13	17
500	14	33
2500	16	66

- (a) Time (ms) w.r.t. dimensions
- **(b)** Iteration number w.r.t. dimensions

**Table 3:** The maximum likelihood problem via power mean cones

## 7.2 Maximum volume hypercube

## 7.2.1 Test of generalized power cones

The problem of finding the maximum volume is given as follows,

max 
$$t$$
  
s.t.  $(x,t) \in \mathcal{K}_{\text{gpow }(\alpha,n,1)}, \alpha = \left[\frac{1}{n}, \dots, \frac{1}{n}\right],$  (44)  
 $Ax \in \mathcal{B}_1(\gamma), Ax \in \mathcal{B}_{\infty}(\gamma),$ 

where  $\mathcal{B}_1(\gamma) := \{x \mid ||x||_1 \leq \gamma\}$  is a 1-norm ball,  $\mathcal{B}_{\infty}(\gamma) := \{x \mid ||x||_{\infty} \leq \gamma\}$  is an  $\infty$ -norm ball and  $\gamma$  is a given constant. Both norm constraints can be transformed to equivalent linear constraints and we use the native transformation from MathOptInterface.jl [LDDGL21] in our implementation.

We set A to the identity matrix in order to reduce the effect of constraints other than the generalized power cone. The solver settings are kept the same as in the previous example, along with the same transformation of the generalized power cone to a product of three-dimensional power cones as in (43). We again vary the dimension n from 100 to 2500 and present the computational results in Table 4 and Table 5.

n	Clarabel-GenPow	Clarabel-Pow	Mosek	Hypatia
100	6.41	9.79	<10	26
500	16.8	39.4	32	436
2500	90.6	256	141	23413

**Table 4:** Time (ms) w.r.t. dimensions (generalized power cones)

n	Clarabel-GenPow	Clarabel-Pow	Mosek	Hypatia
100	18	22	9	11
500	18	27	11	16
2500	18	29	13	20

**Table 5:** Iteration number w.r.t. dimensions (generalized power cones)

Once again, Table 4 shows the superior performance of Clarabel-GenPow over Clarabel-Pow, Mosek and Hypatia. Our sparse decomposition can solve (44) within 100 ms even up to dimension n=2500. Though the speedup seems to be less for the maximum volume hypercube problem than the discrete maximum likelihood example, it mainly comes from increased computational time due to the introduction of norm constraints rather than the efficiency of our sparse decomposition for generalized power cones. Meanwhile, we also list numerical results for both (42) and (44) in Hypatia, but it seems to take much more time than other three solvers under any setting. This is due to the fact that Hypatia uses the Cholesky factorization of the reduced system that factorizes a  $(n+1) \times (n+1)$  dense positive semidefinite matrix for each iteration and doesn't utilize the sparsity of problem (42) and (44). It was verified by our additional experiments that we tried to increase the density of A in (44) given a fixed dimension n and found increased computational time for all settings except Hypatia.

#### 7.2.2 Test of power mean cones

We can again replace the generalized power cone in (44) with a power mean cone,

max 
$$t$$
  
s.t.  $(x,t) \in \mathcal{K}_{powm(\alpha,n)}, \alpha = \left[\frac{1}{n}, \dots, \frac{1}{n}\right],$   
 $Ax \in \mathcal{B}_1(\gamma), Ax \in \mathcal{B}_{\infty}(\gamma),$ 

which should yields the same solution as (44) since the optimum of (44) must be  $t \geq 0$ . We test it with n = 100, 500, 2500 and Table 6 shows that the proposed implementation of power mean cones also performs much better than Hypatia. Clarabel-PowM also performs slightly slower than our sparse implementation of generalized power cones but better than Mosek in solving the maximum volume hypercube problem.

n	Clarabel-PowM	Hypatia
100	6.1	27.0
500	20.6	370
2500	120.4	23272

n	Clarabel-PowM	Hypatia
100	15	15
500	18	17
2500	20	21

(a) Time (ms) w.r.t. dimensions

**(b)** Iteration number w.r.t. dimensions

**Table 6:** The maximum volume problem via power mean cones

## 7.3 Entropy maximization

The entropy maximization problem can be reformulated to the problem as follows:

min 
$$t$$
  
s.t. 
$$\sum_{i \in [d]} p_i = 1, \ p_i \ge 0,$$

$$(r_i, q_i, p_i) \in \mathcal{K}_{exp}, \forall i \in [d]$$

$$-\sum_{i \in [d]} r_i \le t,$$

$$p \in \mathcal{P}, q \in \mathcal{Q},$$

$$(45)$$

where  $\mathcal{P}$  defines additional constraints over the distribution p. Suppose q is a prior distribution, (45) becomes minimizing Kullback-Leiber divergence. We can reformulate (45) as

min 
$$t$$
  
s.t. 
$$\sum_{i \in [d]} p_i = 1,$$

$$(t, q, p) \in \mathcal{K}_{rel},$$

$$p \in \mathcal{P}, q \in \mathcal{Q},$$

$$(46)$$

where  $\mathcal{K}_{rel}$  is the relative entropy cone defined in (17a).

d	Clarabel-Entropy	Clarabel-Exp	Mosek	ECOS
1000	21.0	38.9	15	34.0
2000	51.1	77.6	31	66.2
5000	139	220	79	189.1

**Table 7:** Time (ms) w.r.t. dimensions (relative entropy cones)

d	Clarabel-Entropy	Clarabel-Exp	Mosek	ECOS
1000	16	17	10	18
2000	16	16	10	18
5000	16	17	10	20

Table 8: Iteration number w.r.t. dimensions (relative entropy cones)

If we set  $\mathcal{Q}$  to a randomly generated distribution and  $\mathcal{P} = \emptyset$ , the results are shown in Table 7 and Table 8. We found the sparse exploitation can save computational time but not as much as the implementation for generalized power cones. Clarabel-Entropy is about 30% faster than Clarabel-Exp or ECOS but is slower than Mosek which uses the primal-dual scaling strategy with higher-order correction. This could be potentially explained by: (1)

Reformulating a relative entropy cone as a product of three-dimensional exponential cones doesn't substantially increase the number of nonzero entries in matrix factorization; (2) the self-concordant parameter decreases from 3d + 1 to 3d, which doesn't reduce as much as the generalized power cone; (3) The initial point for relative entropy cones is approximated and needs to be improved.

# 8 Conclusion

In this paper, we have presented an efficient sparse decomposition of generalized power cones, power mean cones and relative entropy cones for the dual scaling interior point method. We have proposed logarithmically-homogeneous self-concordant barrier functions for the dual cones of three nonsymmetric cones respectively. We have detailed the computation of corresponding gradients and Hessians and then augmented the linear system to a sparse form by exploiting the low-rank property inside Hessian matrices of these barrier functions, giving a set of decomposition that ensures the quasidefiniteness of the augmented linear system and makes it factorizable under the LDL factorization with static pivoting. The experimental results demonstrated numerical stability of our sparse decomposition for these cones and the efficiency of our method over the state-of-art solvers when an optimization problem is sparse.

## References

- [ADV10] Martin S. Andersen, Joachim Dahl, and Lieven Vandenberghe. Implementation of nonsymmetric interior-point methods for linear optimization over sparse matrix cones. *Mathematical Programming Computation*, 2(3):167–201, 2010.
- [ApS21] MOSEK ApS. MOSEK Modeling Cookbook, 2021.
- [BEGFB94] Stephen Boyd, Laurent El Ghaoui, Eric Feron, and Venkataramanan Balakrishnan. *Linear Matrix Inequalities in System and Control Theory*. Society for Industrial and Applied Mathematics, 1994.
- [Cha09] Peter Robert Chares. Cones and interior-point Algorithms for Structured Convex Optimization involving Powers and Exponentials. PhD thesis, Universit'e catholique de Louvain, 2009.
- [CKV22] Chris Coey, Lea Kapelevich, and Juan Pablo Vielma. Solving natural conic formulations with Hypatia.jl. *INFORMS Journal on Computing*, 34(5):2686–2699, 2022.
- [CKV23] Chris Coey, Lea Kapelevich, and Juan Pablo Vielma. Performance enhancements for a generic conic interior point algorithm. *Mathematical Programming Computation*, 15(1):53–101, 2023.
- [CPT18] Gérard Cornuéjols, Javier Peña, and Reha Tütüncü. Optimization Methods in Finance. Cambridge University Press, 2 edition, 2018.
- [CS17] Venkat Chandrasekaran and Parikshit Shah. Relative entropy optimization and its applications. *Mathematical Programming*, 161(1):1–32, 2017.
- [DA21] Joachim Dahl and Erling D. Andersen. A primal-dual interior-point algorithm for nonsymmetric exponential-cone optimization. *Mathematical Programming*, 2021.
- [Dav05] Timothy A. Davis. Algorithm 849: A concise sparse cholesky factorization package. *ACM Trans. Math. Softw.*, 31(4):587–591, 2005.
- [DCB13] Alexander Domahidi, Eric Chu, and Stephen Boyd. Ecos: An socp solver for embedded systems. In 2013 European Control Conference (ECC), pages 3071–3076, 2013.
- [GCG21] Michael Garstka, Mark Cannon, and Paul Goulart. COSMO: A conic operator splitting method for convex conic problems. *Journal of Optimization Theory and Applications*, 190(3):779–810, 2021.
- [KAV22] Lea Kapelevich, Erling D. Andersen, and Juan Pablo Vielma. Computing conjugate barrier information for nonsymmetric cones. *Journal of Optimization Theory and Applications*, 2022.

- [KT19] Mehdi Karimi and Levent Tunçel. Domain-driven solver (dds) version 2.0: a matlab-based software package for convex optimization problems in domain-driven form, 2019.
- [LDDGL21] Benoît Legat, Oscar Dowson, Joaquim Dias Garcia, and Miles Lubin. MathOpt-Interface: a data structure for mathematical optimization problems. *INFORMS Journal on Computing*, 34(2):672–689, 2021.
- [LY06] Zhi-Quan Luo and Wei Yu. An introduction to convex optimization for communications and signal processing. *IEEE Journal on Selected Areas in Communications*, 24(8):1426–1438, 2006.
- [Meh92] Sanjay Mehrotra. On the implementation of a primal-dual interior point method. SIAM Journal on Optimization, 2(4):575–601, 1992.
- [Nes12] Yurii Nesterov. Towards non-symmetric conic optimization. Optimization Methods and Software, 27(4-5):893–917, 2012.
- [NN94] Yurii Nesterov and Arkadii Nemirovskii. Interior-Point Polynomial Algorithms in Convex Programming. Society for Industrial and Applied Mathematics, 1994.
- [NT97] Y. E. Nesterov and M. J. Todd. Self-scaled barriers and interior-point methods for convex programming. *Mathematics of Operations Research*, 22(1):1–42, 1997.
- [NT98] Y. E. Nesterov and M. J. Todd. Primal-dual interior-point methods for self-scaled cones. SIAM Journal on Optimization, 8(2):324–364, 1998.
- [OCPB16] Brendan O'Donoghue, Eric Chu, Neal Parikh, and Stephen Boyd. Conic optimization via operator splitting and homogeneous self-dual embedding. *Journal of Optimization Theory and Applications*, 169(3):1042–1068, 2016.
- [PY17] Dávid Papp and Sercan Yıldız. On "a homogeneous interior-point algorithm for non-symmetric convex conic optimization", 2017.
- [PY19] Dávid Papp and Sercan Yildiz. Sum-of-squares optimization without semidefinite programming. SIAM Journal on Optimization, 29(1):822–851, 2019.
- [PY22] Dávid Papp and Sercan Yıldız. Alfonso: Matlab package for nonsymmetric conic optimization. *INFORMS Journal on Computing*, 34(1):11–19, 2022.
- [RX22] Scott Roy and Lin Xiao. On self-concordant barriers for generalized power cones. *Optimization Letters*, 16(2):681–694, 2022.
- [Ser15] Santiago Akle Serrano. Algorithms for unsymmetric cone optimization and an implementation for problems with the exponential cone. PhD thesis, 2015.
- [SM03] Endre Süli and David F. Mayers. An Introduction to Numerical Analysis. Cambridge University Press, 2003.

- [SY15] Anders Skajaa and Yinyu Ye. A homogeneous interior-point algorithm for non-symmetric convex conic optimization. *Mathematical Programming*, 150(2):391–422, 2015.
- [Tun01] Levent Tunçel. Generalization of primal-dual interior-point methods to convex optimization problems in conic form. Foundations of Computational Mathematics, 1(3):229–254, 2001.
- [Van95] Robert J. Vanderbei. Symmetric quasidefinite matrices. SIAM Journal on Optimization, 5(1):100–113, 1995.
- [Van10] Lieven Vandenberghe. The CVXOPT linear and quadratic cone program solvers, 2010.
- [YTM94] Yinyu Ye, Michael J. Todd, and Shinji Mizuno. An  $O(\sqrt{nL})$ -iteration homogeneous and self-dual linear programming algorithm. *Mathematics of Operations Research*, 19(1):53–67, 1994.

# A Proof of Theorem 1

*Proof.* We split the proof into three parts corresponding to each cone mentioned in Theorem 1.

i)  $\mathcal{K}_{\text{gpow }(\alpha,d_1,d_2)}$  has a valid  $(d_1+1)$ -LHSCB function (Theorem 1 in [RX22]),

$$f_{\text{gpow}}(u, w) = -\ln \left( \prod_{i \in [d_1]} (u_i)^{2\alpha_i} - ||w||^2 \right) - \sum_{i \in [d_1]} (1 - \alpha_i) \ln (u_i),$$

which implies

$$|\langle f'''_{\text{gpow}}(u,w)[h]h,h\rangle| \le 2\langle f''_{\text{gpow}}(u,w)h,h\rangle^{3/2}, \ \forall h \in \mathbb{R}^{d_1+d_2}.$$

Note that  $f_{\text{gpow}}^*(u, w) = f_{\text{gpow}}(\alpha^{-1}u, w)$  where  $\alpha^{-1}u := (\cdots, u_i/\alpha_i, \cdots)$  denotes element-wise division. We then have

$$|\langle f_{\text{gpow}}^{*}'''(u,w)[h]h,h\rangle| = |\langle f_{\text{gpow}}'''(\alpha^{-1}u,w)[D_{1}h]D_{1}h,D_{1}h\rangle|$$

$$\leq 2\langle f_{\text{gpow}}''(\alpha^{-1}u,w)D_{1}h,D_{1}h\rangle^{3/2}$$

$$= 2\langle f_{\text{gpow}}^{*}''(\alpha^{-1}u,w)h,h\rangle^{3/2}$$

for any  $h \in \mathbb{R}^{d_1+d_2}$  with a linear transformation

$$D_1 := \operatorname{diag}(1/\alpha_1, \cdots, 1/\alpha_i, \cdots, 1/\alpha_{d_1}, \mathbf{1}^{d_2}).$$

The conjugate barrier  $f_{\text{gpow}}^*(u, w)$  satisfies

$$f_{\text{gpow}}^*(tu, tw) = f_{\text{gpow}}^*(u, w) - (d_1 + 1)\ln(t), \ \forall (u, w) \in \mathcal{K}_{\text{gpow}}^*(\alpha, d_1, d_2), t > 0,$$

hence  $f_{\text{gpow}}^*(u, w)$  is a  $(d_1 + 1)$ -LHSCB function of  $\mathcal{K}_{\text{gpow}}^*(\alpha, d_1, d_2)$  due to (2).

ii) The same reasoning applies to  $f_{\text{powm}}^*$  since we already know that

$$f(u, w) = -\ln \left( \prod_{i \in [d]} u_i^{\alpha_i} - w \right) - \sum_{i \in [d]} (1 - \alpha_i) \ln (u_i) - \ln(w)$$

is a (d+1)-LHSCB function of the nonnegative power cone

$$\mathcal{K}_{\alpha}^{(n,+)} := \{(u,w) \in \mathbb{R}_+^d \times \mathbb{R}_+ : \prod_{i \in [d]} u_i^{\alpha_i} \ge w\}$$

from Theorem 2 in [RX22], which differs from  $f_{\text{powm}}^*(u, w)$  by the transformation  $D_2 := \text{diag}(1/\alpha_1, \dots, 1/\alpha_i, \dots, 1/\alpha_d, -1)$ , like its counterparts for the generalized power cones.

iii) For the proof w.r.t.  $f_{\mathrm{rel}}^*(u,v,w)$ , we define the map  $A:\mathbb{R}^2\to\mathbb{R}$ 

$$A(x,y) := -x \ln\left(\frac{x}{y}\right) + x.$$

Following the same argument for relative entropy cones in [KT19, Appendix E], we have A(x,y) is  $(\mathbb{R}_+,1)$ -compatible with the barrier  $-\ln(x)-\ln(y)$  and then t+A(x,y) is  $(\mathbb{R}_+,1)$ -compatible with the barrier  $-\ln(x)-\ln(y)$ . Hence  $-\ln(t+A(x,y))-\ln(x)-\ln(y)$  is a 3-self-concordant barrier due to Proposition 5.1.7 in [NN94]. Moreover, it is easy to verify  $-\ln(t+A(x,y))-\ln(x)-\ln(y)$  is 3-logarithmically homogeneous and is therefore a 3-LHSCB function.

therefore a 3-LHSCB function. This implies that  $-\left(w_i - u \ln\left(\frac{u}{v_i}\right) + u\right) - \ln(u) - \ln(v_i)$  is a 3-self-concordant barrier by replacing t, x, y with  $w_i, u, v_i, \forall i \in [d]$ . According to the summation rule (Proposition 5.1.3) in [NN94], we can prove  $f_{\rm rel}^*(u, v, w)$  is 3d-LHSCB.

Note that the LHSCB parameter for a relative entropy cone is  $\nu = (2d+1)$  in [KT19, CKV22], which is smaller than our choice  $\nu = 3d$  for the dual of relative entropy cone. A LHSCB function with a smaller  $\nu$  is possible and can reduce the total number of iterations of an IPM in theory, but it is out of the scope of this paper.

# B Root finding for power mean cones

Given r < 0, x > 0 and  $\alpha_i \in (0, 1), \forall i \in [d]$ , the Hessian of (39) is

$$h''(x) = \sum_{i \in [d]} \alpha_i \cdot \frac{\frac{2r^2}{(r^2 + 4x^2)^{3/2}} \left( x/\alpha_i + \left( r + \sqrt{r^2 + 4x^2} \right)/2 \right) - \left( 1/\alpha_i + 2x/\sqrt{r^2 + 4x^2} \right)^2}{\left( x/\alpha_i + \left( r + \sqrt{r^2 + 4x^2} \right)/2 \right)^2}$$

$$= \sum_{i \in [d]} \alpha_i \cdot \frac{2r^2 x/\alpha_i + r^3 + r^2 \sqrt{r^2 + 4x^2} - (r^2 + 4x^2)^{3/2}/\alpha_i^2 - 4x(r^2 + 4x^2)/\alpha_i - 4x^2 \sqrt{r^2 + 4x^2}}{(r^2 + 4x^2)^{3/2} \left( x/\alpha_i + \left( r + \sqrt{r^2 + 4x^2} \right)/2 \right)^2}$$

$$= \sum_{i \in [d]} \alpha_i \cdot \frac{r^3 + \sqrt{r^2 + 4x^2} \left( r^2 - (r^2 + 4x^2)/\alpha_i^2 \right) - 2xr^2/\alpha_i - 16x^3/\alpha_i - 4x^2 \sqrt{r^2 + 4x^2}}{(r^2 + 4x^2)^{3/2} \left( x/\alpha_i + \left( r + \sqrt{r^2 + 4x^2} \right)/2 \right)^2}$$

$$< \sum_{i \in [d]} \alpha_i \cdot \frac{r^3 - 2xr^2/\alpha_i - 16x^3/\alpha_i - 4x^2\sqrt{r^2 + 4x^2}}{(r^2 + 4x^2)^{3/2} \left( x/\alpha_i + \left( r + \sqrt{r^2 + 4x^2} \right)/2 \right)^2} < 0.$$