

## LDL<sup>T</sup> Direction in interior point method for semidefinite programming

Raghunathan, A.U.; Biegler, L.T.

TR2018-037 March 2018

### Abstract

We present an interior point method for semidefinite programming where the semidefinite constraints on a matrix  $X$  are formulated as nonnegative constraints on  $d[1](X), \dots, d[n](X)$  obtained from the LDLT factorization  $X = L\text{Diag}(d[1](X), \dots, d[n](X))L^T$ . The approach was first proposed by Fletcher who also provided analytic expressions for the derivatives of the factors in terms of  $X$  and the approach was subsequently utilized in an interior point algorithm by Benson and Vanderbei. However, the evaluation of first and second derivatives of  $d[i](X)$  has been a bottleneck in such an algorithm. In this paper, we: (i) derive formulae for the first and second derivatives of  $d[i](X)$  that are efficient and numerically stable to compute, (ii) show that the LDLT based search direction can be viewed in the standard framework of interior point methods for semidefinite programs with comparable computational cost per iteration, (iii) characterize the central path, and (iv) analyze the numerical conditioning of the linear system arising in the algorithm. We provide detailed numerical results on 79 SDP instances from the SDPLIB test set.

*SIAM Journal on Optimization*

This work may not be copied or reproduced in whole or in part for any commercial purpose. Permission to copy in whole or in part without payment of fee is granted for nonprofit educational and research purposes provided that all such whole or partial copies include the following: a notice that such copying is by permission of Mitsubishi Electric Research Laboratories, Inc.; an acknowledgment of the authors and individual contributions to the work; and all applicable portions of the copyright notice. Copying, reproduction, or republishing for any other purpose shall require a license with payment of fee to Mitsubishi Electric Research Laboratories, Inc. All rights reserved.



# $LDL^T$ DIRECTION INTERIOR POINT METHOD FOR SEMIDEFINITE PROGRAMMING\*

ARVIND U. RAGHUNATHAN<sup>†</sup> AND LORENZ. T. BIEGLER<sup>‡</sup>

**Abstract.** We present an interior point method for semidefinite programming where the semidefinite constraints on a matrix  $X$  are formulated as nonnegative constraints on  $d_{[1]}(X), \dots, d_{[n]}(X)$  obtained from the  $LDL^T$  factorization  $X = L\text{Diag}(d_{[1]}(X), \dots, d_{[n]}(X))L^T$ . The approach was first proposed by Fletcher [15] who also provided analytic expressions for the derivatives of the factors in terms of  $X$  and the approach was subsequently utilized in an interior point algorithm by Benson and Vanderbei [6]. However, the evaluation of first and second derivatives of  $d_{[i]}(X)$  has been a bottleneck in such an algorithm. In this paper, we: (i) derive formulae for the first and second derivatives of  $d_{[i]}(X)$  that are efficient and numerically stable to compute, (ii) show that the  $LDL^T$  based search direction can be viewed in the standard framework of interior point methods for semidefinite programs with comparable computational cost per iteration, (iii) characterize the central path, and (iv) analyze the numerical conditioning of the linear system arising in the algorithm. We provide detailed numerical results on 79 SDP instances from the SDPLIB test set.

**Key words.** Semidefinite Programming,  $LDL^T$  Factorization, Interior Point Method, Central path, Conditioning of Schur complement.

**AMS subject classifications.** 65K05, 90C22, 90C51

**1. Introduction.** In this paper we are interested in the solution of semidefinite programs (SDPs) of the form

$$(1) \quad \begin{aligned} \min_{X \in \mathbb{S}^n} \quad & C \bullet X \\ \text{s.t.} \quad & \mathcal{A}(X) = b \\ & X \succeq 0 \end{aligned}$$

where  $\mathbb{S}^n$  denotes the set of  $n \times n$  symmetric matrices,  $C \in \mathbb{S}^n$ ,  $\mathcal{A} : \mathbb{S}^n \rightarrow \mathbb{R}^m$ ,  $b \in \mathbb{R}^m$  are input data,  $A \bullet B = \sum_{i=1}^n \sum_{j=1}^n A_{[ij]}B_{[ij]}$  is the trace inner product for symmetric matrices with  $A_{[ij]}$  denoting the  $(i, j)$ th entry of matrix  $A$  and  $\succeq$  denotes the positive semidefinite constraint on  $X$ . The linear map  $\mathcal{A}$  performs the following linear transformation,  $X \mapsto (A_1 \bullet X, \dots, A_m \bullet X)$  where  $A_k \in \mathbb{S}^n$  is the  $k$ th constraint matrix.

The seminal work of Alizadeh [1] and Nesterov and Nemirovskii [35] laid the foundation for the development of theory and algorithms for SDPs. Interior point algorithms for SDPs generate search directions at each iteration by solving the linearization of the following system [48]:

$$(2) \quad \begin{aligned} C + \mathcal{A}^*(\lambda) - S &= 0 \\ \mathcal{A}(X) &= b \\ H_P(XS) &= \mu I_n \end{aligned}$$

where,  $\mathcal{A}^* : \mathbb{R}^m \rightarrow \mathbb{S}^n$  is the adjoint of the linear map  $\mathcal{A}$  defined as  $\mathcal{A}^*(\lambda) = \sum_{k=1}^m \lambda_{[k]}A_k$ ,  $\lambda_{[k]}$  denotes the  $k$ th element of vector  $\lambda$ ,  $S \succeq 0 \in \mathbb{S}^n$  is the multiplier matrix for the positive semidefinite constraint,  $I_n \in \mathbb{R}^{n \times n}$  the identity matrix,

---

\*Submitted to the editors DATE.

<sup>†</sup>Mitsubishi Electric Research Laboratories, 201 Broadway, Cambridge, MA 02139. (raghunanathan@merl.com)

<sup>‡</sup>Carnegie Mellon University, 5000 Forbes Avenue, Pittsburgh, PA 15213. (lb01@andrew.cmu.edu)

$H_P$  is the symmetrization operation defined as

$$(3) \quad H_P(Y) := \frac{1}{2} (PYP^{-1} + (PYP^{-1})^T)$$

for  $P \in \mathbb{R}^{n \times n}$  an invertible matrix and  $\mu > 0$  the barrier parameter. The interior point algorithms eventually drive  $\mu$  to 0, thus recovering an optimal solution to (1). The interior point algorithms for SDPs give rise to different search directions based on the symmetrization operator (choice of  $P$ ) employed for the complementarity constraints. The three important directions and their respective choices for  $P$  are: the AHO direction [2] with  $P = I_n$ , the HKM direction [23, 25, 30] with  $P = X^{-\frac{1}{2}}$  (or  $S^{\frac{1}{2}}$ ) and NT direction [36] with  $P = \left( X^{\frac{1}{2}} (X^{\frac{1}{2}} S X^{\frac{1}{2}})^{-\frac{1}{2}} X^{\frac{1}{2}} \right)^{-\frac{1}{2}}$ . We refer the interested reader to the survey article on search directions in SDP by Todd [42]. A comprehensive collection of theory, algorithms and applications can also be found in Monteiro [31], Wolkowicz et al. [46] and in Anjos and Lasserre [4].

An alternate formulation to (1), first proposed by Fletcher [15], is

$$(4) \quad \begin{aligned} \min_{X \in \mathbb{S}^n} \quad & C \bullet X \\ \text{s.t.} \quad & \mathcal{A}(X) = b \\ & d_{[i]}(X) \geq 0 \quad \forall i = 1, \dots, n \end{aligned}$$

where  $d_{[i]}(X)$  is the  $i$ th diagonal entry of the diagonal matrix  $D$  in the  $LDL^T$  factorization [19] of  $X$  and  $L \in \mathbb{R}^{n \times n}$  is unit lower triangular. The  $LDL^T$  factorization is uniquely defined only for  $X \succ 0 \in \mathbb{S}^n$ , i.e.  $X$  is positive definite. Fletcher [15] assumed that the rank of the solution was known and the data matrices had been permuted so that the solution matrix can be parameterized as

$$X = \begin{bmatrix} L_{11}(X) & \mathbf{0}_{p \times (n-p)} \\ L_{21}(X) & I_{n-p} \end{bmatrix} \begin{bmatrix} D_1(X) & \mathbf{0}_{p \times (n-p)} \\ \mathbf{0}_{(n-p) \times p} & D_2(X) \end{bmatrix} \begin{bmatrix} L_{11}(X)^T & L_{21}(X)^T \\ \mathbf{0}_{(n-p) \times p} & I_{n-p} \end{bmatrix}$$

where  $p$  is the rank of  $X$  at the solution,  $L_{11}(X)$  is unit lower triangular and  $D_1(X)$  is diagonal with positive entries. Substituting this parameterization in (4) and replacing the inequalities with  $D_2(X) = 0$  yields the formulation in [15]. Analytical expressions were derived for the first and second derivatives of  $D_2(X)$ . Fletcher [15] proposed an active-set based sequential quadratic programming (SQP) algorithm with exact derivatives to solve (4). Globalization of the algorithm was achieved using the exact  $\ell_1$  penalty function and limited computational results were reported. Benson and Vanderbei [6] considered the solution of (4) using an interior point algorithm for nonlinear programs. The authors used the expressions derived in [15] to provide exact first and second derivative information to the interior point algorithm. Limited numerical results were provided on small instances. The cost of evaluating the first and second derivatives of  $d_{[i]}(X)$  as provided in [15] are prohibitive and this also affected the solution times of the search direction computation. However, no analysis of the approach or its relation to standard interior point methods for SDPs was presented. A related approach proposed by Burer, Monteiro and Zhang [10] replaced the matrix  $X$  by nonnegativity constraints on the diagonal of the lower triangular matrix  $\hat{L}$  in the Cholesky factorization  $X = \hat{L}\hat{L}^T$  [19]. They also derived gradient formulas for the objective of the resulting nonlinear program. Encouraging numerical results were presented for a first-order interior point method using this transformation in [9]. Burer [8], Srijungtongsiri and Vavasis [40], and Dahl et al. [12] presented efficient

procedures for computing the first and second derivatives of the barrier function based on the  $LDL^T$  factorization and the sparsity in the problem data. However, these approaches do not employ the formulation in (4).

**1.1. Focus of the paper and our contributions.** In this paper we study the solution of SDP (1) through the  $LDL^T$  formulation in (4). As mentioned earlier, Fletcher [15] derived derivative expressions for  $d_{[i]}(X)$  but these require the inverse of principal submatrices of  $X$  and several matrix-vector products. This can be computationally expensive and result in round-off error. To address this, we provide an elegant derivation of the first and second derivatives of  $d_{[i]}(X)$ , the  $i$ th element of  $d(X)$ , that is efficient and numerically stable to evaluate in §3. This derivation allows us to view the first-order stationary conditions of  $LDL^T$  formulation as a new search direction for SDP. Using the first derivative expressions, we show that the interior point method for  $LDL^T$  formulation (4) satisfies the following equations

$$\begin{aligned}
 (5) \quad C + \mathcal{A}^*(\lambda) - L(X)^{-T} \text{Diag}(z)L(X)^{-1} &= 0 \\
 \mathcal{A}(X) &= b \\
 d_{[i]}(X)z_{[i]} &= \mu \quad \forall i = 1, \dots, n
 \end{aligned}$$

where  $z \geq 0 \in \mathbb{R}^n$  are the multipliers,  $z_{[i]}$  denotes the  $i$ th element of vector  $z$ , and  $L(X)$  is the unit lower triangular matrix in the  $LDL^T$  factorization of  $X$ . Using the derived expressions for the second derivative of  $d_{[i]}(X)$ , we show in §4 that the Newton step for (5) can be computed in a manner identical to standard SDP interior point algorithms. In particular, we show that the computational cost per iteration is the same as the NT direction [36]. We also show that the Newton step for (5) is well defined for all points in the interior of the feasible region, i.e.  $d_{[i]}(X), z_{[i]} > 0$ . Further, even though (5) involves the inverse of  $L(X)$  we never require this as part of the step computation.

From (2) and (5), we have that the multiplier matrix  $S$  can be identified with  $L(X)^{-T} \text{Diag}(z)L(X)^{-1}$ . In fact this representation of  $S$  can be viewed as an  $UDU^T$  factorization of  $S$  where  $U(S) = L(X)^{-T}$  is unit upper triangular. With this choice it is also readily verified that if  $D(X)\text{Diag}(z) = \alpha I_n$  for  $\alpha \in \mathbb{R}$  then matrices  $X$  and  $S$  commute, i.e.  $XS = SX$ . It is well known that matrices  $X, S \in \mathbb{S}_+^n$  commute if and only if  $X$  and  $S$  have the same set of eigenvectors. The identification of multiplier matrix as  $S = L(X)^{-T} \text{Diag}(z)L(X)^{-1}$  restates that commutative property in terms of the  $LDL^T, UDU^T$  factorizations of  $X$  and  $S$ , respectively. We prove this in §2. We provide precise correspondence between the solutions of standard formulation and  $LDL^T$  formulation in §5. We establish a homeomorphism between the central paths of the  $LDL^T$  direction and the central path defined for  $XS = \mu I_n$  in §5.

We provide a characterization of the ill-conditioning of the Schur complement matrix in the  $LDL^T$  direction. In particular, we show that the linear equations defining the Newton step for the  $LDL^T$  formulation are equivalent to a *lifted* linear system for all points in the interior of the feasible region. Under *primal* and *dual nondegeneracy* assumptions, we also establish the non-singularity of the *lifted* linear system at the solution to SDP. The nondegeneracy conditions were first stated by Alizadeh, Haeberly and Overton [3] using the spectral decomposition of matrices. We derive the *primal* and *dual nondegeneracy* conditions for SDPs based on the  $LDL^T$  factorization. Under the nondegeneracy assumptions, we show that the conditioning of Schur complement matrix is similar to that for the AHO direction [2].

We present an interior point algorithm using the  $LDL^T$  direction and provide detailed numerical results on the performance of the approach on the SDPLIB [7] test



set. A comparison of the performance of the  $LDL^T$  direction against the standard SDP directions and SDP solvers is also presented.

**1.2. Organization of the paper.** The rest of the paper is organized as follows. We begin by providing a brief description of the  $LDL^T$  factorization in §2. §3 presents computational formulae for the first and second derivatives of  $d_{[i]}(X)$  and establishes that the formulation (4) is a convex program. §4 presents the interior point algorithm corresponding to (4) and the step computation. §5 presents the correspondence between solutions to the  $LDL^T$  formulation and the standard SDP formulation. Uniqueness and existence of the central path are also established. §6 characterizes the conditioning of the Schur complement matrix. Numerical implementation and results are described in §7 followed by conclusions in §8.

**1.3. Notation.** We denote by  $\mathbb{R}, \mathbb{R}_+, \mathbb{R}_{++}$  the set of real, nonnegative real, positive real numbers, respectively, and by  $\mathbb{R}^n$  the set of  $n \times 1$  vectors endowed with standard inner product and Euclidean norm ( $\|\cdot\|$ ). The set of  $n \times n$  symmetric matrices is denoted by  $\mathbb{S}^n$ , and  $(\mathbb{S}_+^n) \mathbb{S}_{++}^n$  denotes the set of symmetric positive (semi-) definite matrices. For a matrix  $A \in \mathbb{S}^n$ ,  $A \succeq 0$  and  $A \succ 0$  respectively denote the positive semi-definiteness and positive definiteness of  $A$ . For a vector  $a$ ,  $a_{[i]}$  denotes the  $i$ th component of  $a$  and for a matrix  $A$ ,  $A_{[ij]}$  denotes the  $(i, j)$ th entry of  $A$ . The  $i$ th row and column of matrix  $A$  are denoted by  $A_{[i\cdot]}$  and  $A_{[\cdot i]}$ , respectively. The space  $\mathbb{S}^n$  is endowed with the trace inner product  $A \bullet B = \sum_{i=1}^n \sum_{j=1}^n A_{[ij]} B_{[ij]}$  for  $A, B \in \mathbb{S}^n$  and  $\|A\|_F = \sqrt{A \bullet A}$  denotes the Frobenius norm. For a matrix  $A \in \mathbb{R}^{n \times n}$ ,  $\|A\|$  denotes the Euclidean norm. For a matrix  $A \in \mathbb{S}_{++}^n$ , the  $LDL^T$  factorization of the matrix will be denoted by  $A = L(A)D(A)L(A)^T$  where  $L(A)$  is unit lower triangular and  $D(A) \succ 0$  is diagonal. We will also refer to  $L(A), D(A)$  as the  $LDL^T$  factors of  $A$ . The argument  $A$  in  $L(A), D(A)$  is suppressed when the dependence is clear from the context. The kronecker product between matrices  $A, B \in \mathbb{S}^n$  is denoted as  $A \otimes B$ . Given  $A \in \mathbb{R}^{n \times n}$ ,  $\text{vec}(A) \in \mathbb{R}^{n^2 \times 1}$  is a vector resulting from column-wise stacking of  $A$  and  $\text{mat}(\cdot)$  denotes the reverse operation that takes a vector of  $\mathbb{R}^{n^2 \times 1}$  to  $\mathbb{R}^{n \times n}$ . Note that  $\text{mat}(\text{vec}(A)) = A$ . For a vector  $a \in \mathbb{R}^n$ ,  $\text{Diag}(a) \in \mathbb{R}^{n \times n}$  is a diagonal matrix with  $(\text{Diag}(a))_{[ii]} = a_{[i]}$ . For a matrix  $A \in \mathbb{R}^{n \times n}$ ,  $\text{diag}(A) \in \mathbb{R}^n$  is a vector with  $(\text{diag}(A))_{[i]} = A_{[ii]}$ . For two vectors  $a, b \in \mathbb{R}^n$ ,  $a \circ b$  denotes the element-wise multiplication and  $a \circ^{-1} b$  with  $b_{[i]} \neq 0$  denotes element-wise division. For two matrices  $A, B \in \mathbb{S}^n$ ,  $A \circ B$  denotes the element-wise multiplication and  $A \circ^{-1} B$  with  $B_{[ij]} \neq 0$  denotes element-wise division. The vector  $e_i \in \mathbb{R}^n$  denotes the  $i$ th unit vector and  $I_n \in \mathbb{R}^{n \times n}$  denotes the identity matrix.

**2.  $LDL^T$  Factorization.** In this section, we review the  $LDL^T$  factorization for positive definite matrices and also show that such a factorization exists even when the matrix is only positive semidefinite. We begin with the following result on existence of the  $LDL^T$  factorization for positive definite matrices from Golub and Van Loan [19].

LEMMA 2.1. *Given  $X \in \mathbb{S}_{++}^n$  the matrix can be factorized uniquely as  $X = LDL^T$  where  $L$  is unit lower triangular and  $D$  is diagonal, positive definite matrix.*

We introduce a partitioning of the matrices  $X$  and the factors  $L, D$ ,

(6)

$$X = \begin{bmatrix} X_{i-1} & x_i & * \\ x_i^T & X_{[ii]} & * \\ * & * & * \end{bmatrix}, L = \begin{bmatrix} L_{i-1} & \mathbf{0}_{i-1 \times 1} & * \\ l_i^T & 1 & * \\ * & * & * \end{bmatrix}, D = \begin{bmatrix} D_{i-1} & \mathbf{0}_{(i-1) \times 1} & * \\ \mathbf{0}_{1 \times (i-1)} & d_{[i]} & * \\ * & * & * \end{bmatrix}$$

where  $X_i, L_i, D_i \in \mathbb{R}^{i \times i}$  are the  $i$ th principal minor of  $X, L, D$ , respectively and  $x_i, l_i \in \mathbb{R}^{i-1}$ . The LDL<sup>T</sup> procedure from [19] is presented in Algorithm 1. Since  $L$  is

---

**Algorithm 1** LDL<sup>T</sup> Factorization

---

**input**  $X \in \mathbb{S}_{++}^n$   
 Set  $L_{[11]} = 1, d_{[1]} = X_{[11]}$   
**for**  $i = 2, \dots, n$  **do**  
     Set  $D_{i-1} = \text{Diag}(d_{[1]}, \dots, d_{[i-1]})$   
     Compute  $l_i = D_{i-1}^{-1} L_{i-1}^{-1} x_i, d_{[i]} = X_{[ii]} - l_i^T D_{i-1} l_i$   
**end for**  
**return** Factors  $L$  - unit lower triangular,  $d$  - diagonal of  $D$

---

unit lower triangular, it follows that  $L_i$  is also unit lower triangular. Hence,

$$(7) \quad X_i = L_i D_i L_i^T.$$

The relation in (7) implies that  $L_i, D_i$  are the LDL<sup>T</sup> factors of  $X_i$ . Using the notation in (6) it can be easily verified that

$$(8) \quad L_i^{-1} = \begin{bmatrix} L_{i-1}^{-1} & \mathbf{0}_{(i-1) \times 1} \\ l_i^T & 1 \end{bmatrix}^{-1} = \begin{bmatrix} L_{i-1}^{-1} & \mathbf{0}_{(i-1) \times 1} \\ -l_i^T L_{i-1}^{-1} & 1 \end{bmatrix} \quad \forall i = 1, \dots, n.$$

Substituting  $i = n$  in (8) yields  $L^{-1}$ . Further, we denote by  $\widehat{X}_i^{-1}, \widehat{D}_i^{-1} \in \mathbb{R}^{n \times n}$  the lifting of the  $i \times i$ -matrices  $X_i^{-1}, D_i^{-1} \in \mathbb{R}^{i \times i}$  to  $n \times n$ -matrices. The lifted matrices  $\widehat{X}_i^{-1}, \widehat{D}_i^{-1}$  for  $i = 1, \dots, n$  are given by

$$(9) \quad \widehat{X}_i^{-1} = \begin{bmatrix} X_i^{-1} & \mathbf{0}_{i \times (n-i)} \\ \mathbf{0}_{(n-i) \times i} & \mathbf{0}_{(n-i) \times (n-i)} \end{bmatrix}, \quad \widehat{D}_i^{-1} = \begin{bmatrix} D_i^{-1} & \mathbf{0}_{i \times (n-i)} \\ \mathbf{0}_{(n-i) \times i} & \mathbf{0}_{(n-i) \times (n-i)} \end{bmatrix}$$

The inverse of  $X_i$  can be expressed using (7) as

$$(10) \quad X_i^{-1} = (L_i D_i L_i^T)^{-1} = L_i^{-T} D_i^{-1} L_i^{-1} \implies \widehat{X}_i^{-1} = L^{-T} \widehat{D}_i^{-1} L^{-1}$$

where the implication follows from (6), (8) and (9).

In the case of positive semidefinite matrices, we can still define a LDL<sup>T</sup> factorization. However, the unit lower triangular matrix is not unique. The lemma below characterizes the existence of such a decomposition. We do not provide a complete algorithm but only existence of such a decomposition. The following is a modification of Theorem 10.9 in Higham [24]. The uniqueness claims in Lemma 2.2(a) are our contribution. We also note that Benson and Vanderbei [6, Theorem 1(b)] only claimed that  $D$  is unique but did not provide a proof.

LEMMA 2.2. Suppose  $X \in \mathbb{S}_+^n$  with  $\text{rank}(X) = p$ .

- (a) There exists a unit lower triangular matrix  $L$  and diagonal matrix with nonnegative entries  $D$  such that  $X = LDL^T$ . Further,  $D$  is unique and  $L_{[\cdot, i]}$  is unique for  $D_{[ii]} > 0$ .
- (b) There exists a permutation matrix  $\Pi$  such that

$$(11) \quad \Pi^T X \Pi = LDL^T$$

with  $L = \begin{bmatrix} L_{11} & \mathbf{0}_{p \times (n-p)} \\ L_{21} & L_{22} \end{bmatrix}, D = \begin{bmatrix} D_1 & \mathbf{0}_{p \times (n-p)} \\ \mathbf{0}_{(n-p) \times p} & \mathbf{0}_{(n-p) \times (n-p)} \end{bmatrix}$



where  $L_{11} \in \mathbb{R}^{p \times p}$ ,  $L_{22} \in \mathbb{R}^{(n-p) \times (n-p)}$  are unit lower triangular,  $L_{21} \in \mathbb{R}^{(n-p) \times p}$  and  $D_1 \succ 0 \in \mathbb{R}^{p \times p}$  is diagonal. Further,  $L_{11}, L_{21}, D_1$  are unique for the specified permutation matrix  $\Pi$ .

*Proof.* Refer to Appendix A.  $\square$

REMARK 2.1. The factorizations in the section have all been stated in terms of unit lower triangular matrices. However, these can be stated equivalently in terms of unit upper triangular matrices. In other words, Lemmas 2.1 and 2.2 can be restated as showing existence of unit upper triangular matrix  $U$  and diagonal matrix  $D$  such that  $X = UDU^T$ . For the case of positive definite matrices, the loop in Algorithm 1 will be executed in reverse starting from element  $X_{[nn]}$ .

We establish certain properties of the matrices  $X, S$  that satisfy  $XS = \mu I_n$ . The lemma below relates the  $LDL^T$  factorization of  $X$  to the  $UDU^T$  factorization of  $S$ .

LEMMA 2.3. Suppose  $\mu > 0$ ,  $X, S \in \mathbb{S}_{++}^n$  and  $X = L(X)D(X)L(X)^T$ . Then  $XS = \mu I_n$  if and only if  $S = U(S)D(S)U(S)^T$  where  $U(S) = L(X)^{-T}$  and  $D(X)D(S) = \mu I_n$ . Further,  $U(S)$  is unique.

*Proof.* Refer to Appendix B.  $\square$

Lemma 2.3 reveals a nice dual relationship in the  $LDL^T$  factorization of  $X$  and the  $UDU^T$  factorization of  $S$  for all  $XS = \mu I_n$  with  $\mu > 0$ . While the unit lower triangular factor is unique for  $\mu > 0$ , it is not necessarily so in the case of  $\mu = 0$ . However, we show next that we can still define a factorization.

LEMMA 2.4. Suppose  $X, S \in \mathbb{S}_+^n$ . Then,  $XS = 0$  if and only if there exist unit lower triangular matrix  $L(X)$  such that  $X = L(X)D(X)L(X)^T$  and  $S = U(S)D(S)U(S)^T$  with  $U(S) = L(X)^{-T}$  and  $D(X)D(S) = 0$ .

*Proof.* Refer to Appendix C.  $\square$

**3. First and Second Derivatives of  $d_{[ij]}(X)$ .** In section 3.1, we present an elegant derivation of the first and second derivatives of  $d_{[ij]}(X)$ . The derivative expressions provided here are equivalent to those in Fletcher [15] and Benson and Vanderbei [6]. We also show that the functions  $d_{[ij]}(X)$  are concave on  $\mathbb{S}_{++}^n$  and hence, prove that the  $LDL^T$  formulation (4) is convex in Theorem 3.3. Benson and Vanderbei [6, Theorem 3] showed that the Hessian of  $d_{[ij]}(X)$  is negative semidefinite for  $X \in \mathbb{S}_{++}^n$ . Finally, we also derive an expression for the action of the Hessian of the Lagrangian on a symmetric matrix. The expressions in [6] required the computation of several inner products involving inverses of principal minors and their summation to compute even an element of the Hessian of the Lagrangian. This is susceptible to numerical errors especially when approaching the solution to the SDP. In this sense, our derivative expressions are computationally stable. In fact, as we show in the next section, inversion of the principal minors is not necessary for computing the Newton step.

**3.1. Expressions for  $\nabla_X d_{[ij]}(X), \nabla_X^2 d_{[ij]}(X)$ .** We start here by recalling the definition of matrix derivatives [37]. Given a function  $g(X) : \mathbb{S}^n \rightarrow \mathbb{R}$ , the first derivative of  $g$  at  $X$  is denoted by the linear map  $\nabla_X g(X) : \mathbb{S}^n \rightarrow \mathbb{S}^n$  satisfying

$$(12a) \quad \lim_{\|\Delta X\| \rightarrow 0} \frac{|g(X + \Delta X) - g(X) - \nabla_X g(X) \bullet \Delta X|}{\|\Delta X\|} = 0.$$



The second derivative of  $g$  at  $X$  is denoted by the linear map  $\nabla_X^2 g(X) : \mathbb{S}^n \rightarrow \mathbb{S}^n$  satisfying

$$(12b) \quad \lim_{\|\Delta X\| \rightarrow 0} \frac{\|\nabla_X g(X + \Delta X) - \nabla_X g(X) - \text{mat}(\nabla_X^2 g(X) \text{vec}(\Delta X))\|}{\|\Delta X\|} = 0.$$

Following the definitions in (12), the first and second derivatives of  $\ln(\det(X))$  w.r.t.  $X$  (see for e.g. [37]) are

$$\nabla_X \ln(\det(X)) = X^{-1}, \quad \nabla_X^2 \ln(\det(X)) = -X^{-1} \otimes X^{-1}.$$

Hence, by the definition of  $X_i$  (6), the derivatives of  $\ln(\det(X_i))$  w.r.t.  $X_i$  are

$$\nabla_{X_i} \ln(\det(X_i)) = X_i^{-1}, \quad \nabla_{X_i}^2 \ln(\det(X_i)) = -X_i^{-1} \otimes X_i^{-1}.$$

The derivatives of  $\ln(\det(X_i))$  w.r.t.  $X$  can be expressed as

$$(13) \quad \nabla_X \ln(\det(X_i)) = \widehat{X_i^{-1}}, \quad \nabla_X^2 \ln(\det(X_i)) = -\widehat{X_i^{-1}} \otimes \widehat{X_i^{-1}}$$

To verify this, note that,

$$\begin{aligned} \nabla_{X_i} \ln(\det(X_i)) \bullet \Delta X_i &= \nabla_X \ln(\det(X_i)) \bullet \Delta X \\ (\nabla_{X_i}^2 \ln(\det(X_i)) \Delta X_i) \bullet \Delta X_i &= (\nabla_X^2 \ln(\det(X_i)) \Delta X) \bullet \Delta X \end{aligned}$$

where  $\Delta X_i$  is the  $i$ th principal minor of  $\Delta X$ .

From the unit lower triangularity of  $L_i$  and diagonality of  $D_i$ , we have that

$$\det(X_i) = \det(L_i D_i L_i^T) = \det(L_i) \det(D_i) \det(L_i^T) = \prod_{j=1}^i d_{[j]}.$$

This provides us with the key observation

$$(14) \quad d_{[1]}(X) = \det(X_1), \quad d_{[i]}(X) = \frac{\det(X_i)}{\det(X_{i-1})} \quad \forall i = 2, \dots, n$$

that simplifies the derivation of the expressions for the gradient and Hessian of  $d_{[i]}(X)$ . We are ready to present the derivation of the gradient and Hessian of  $d_{[i]}(X)$ .

LEMMA 3.1. *Let  $X \in \mathbb{S}_{++}^n$  and  $X = L(X)D(X)L(X)^T$ . Then,*

$$(15) \quad \begin{aligned} \nabla d_{[1]}(X) &= d_{[1]}(X) \widehat{X_1^{-1}} = L^{-T} (e_1 e_1^T) L^{-1} = e_1 e_1^T \\ \nabla d_{[i]}(X) &= d_{[i]}(X) \left( \widehat{X_i^{-1}} - \widehat{X_{i-1}^{-1}} \right) = L^{-T} (e_i e_i^T) L^{-1} \quad \forall i = 2, \dots, n \end{aligned}$$

where  $e_i \in R^n$  is the unit vector with 1 at the  $i$ th component and zero otherwise.

*Proof.* Since  $d_{[1]}(X) = X_{[11]}$  (refer Algorithm 1),

$$\begin{aligned} \nabla_{X_1} d_{[1]}(X) &= 1 = d_{[1]}(X) X_1^{-1} \\ \implies \nabla d_{[1]}(X) &= d_{[1]}(X) \widehat{X_1^{-1}} = L^{-T} e_1 e_1^T L^{-1} = e_1 e_1^T \end{aligned}$$

where the implication follows from (9) and (10). Consider  $i \geq 2$ . Taking logarithms on both sides of (14),  $\ln(d_{[i]}(X)) = \ln(\det(X_i)) - \ln(\det(X_{i-1}))$ , and then differentiating once obtain

$$\begin{aligned} & \nabla \ln d_{[i]}(X) = \nabla \ln(\det(X_i)) - \nabla \ln(\det(X_{i-1})) \\ \implies & \frac{1}{d_{[i]}(X)} \nabla d_{[i]}(X) = \widehat{X_i^{-1}} - \widehat{X_{i-1}^{-1}} \\ \implies & \nabla d_{[i]}(X) = d_{[i]}(X) \left( \widehat{X_i^{-1}} - \widehat{X_{i-1}^{-1}} \right). \end{aligned}$$

where the second equality follows from (13). Substituting (10) into the above yields

$$\nabla d_{[i]}(X) = d_{[i]}(X) L^{-T} \left( \widehat{D_i^{-1}} - \widehat{D_{i-1}^{-1}} \right) L^{-1} = L^{-T} (e_i e_i^T) L^{-1}$$

completing the proof.  $\square$

We now present the derivation of the second derivatives of  $d_{[i]}(X)$ .

LEMMA 3.2. *Let  $X \in \mathbb{S}_{++}^n$  and  $X = L(X)D(X)L(X)^T$ . Then,*

$$(16) \quad \begin{aligned} & \nabla^2 d_{[1]}(X) = 0 \\ & \nabla^2 d_{[i]}(X) = -d_{[i]}(X) \left( \widehat{X_i^{-1}} \otimes \widehat{X_{i-1}^{-1}} \right) - d_{[i]}(X) \left( \widehat{X_{i-1}^{-1}} \otimes \widehat{X_i^{-1}} \right) \\ & \quad + 2d_{[i]}(X) \left( \widehat{X_{i-1}^{-1}} \otimes \widehat{X_{i-1}^{-1}} \right) \quad \forall i = 2, \dots, n. \end{aligned}$$

*Proof.* From Lemma 3.1, we have that  $\nabla d_{[1]}(X)$  is constant and hence,  $\nabla^2 d_{[1]}(X) = 0$ . For the case of  $i \geq 2$ , consider taking two derivatives of (14)

$$\begin{aligned} & \nabla^2 \ln(d_{[i]}(X)) = \nabla^2 \ln(\det(X_i)) - \nabla^2 \ln(\det(X_{i-1})) \\ \implies & \nabla \left( \frac{1}{d_{[i]}(X)} \nabla d_{[i]}(X) \right) = - \left( \widehat{X_i^{-1}} \otimes \widehat{X_{i-1}^{-1}} \right) + \left( \widehat{X_{i-1}^{-1}} \otimes \widehat{X_i^{-1}} \right) \\ \implies & \frac{1}{d_{[i]}(X)} \nabla^2 d_{[i]}(X) - \frac{1}{d_{[i]}(X)^2} (\nabla d_{[i]}(X) \otimes \nabla d_{[i]}(X)) \\ & = - \left( \widehat{X_i^{-1}} \otimes \widehat{X_{i-1}^{-1}} \right) + \left( \widehat{X_{i-1}^{-1}} \otimes \widehat{X_i^{-1}} \right) \\ \implies & \frac{1}{d_{[i]}(X)} \nabla^2 d_{[i]}(X) - \left( \widehat{X_i^{-1}} - \widehat{X_{i-1}^{-1}} \right) \otimes \left( \widehat{X_{i-1}^{-1}} - \widehat{X_i^{-1}} \right) \\ & = - \left( \widehat{X_i^{-1}} \otimes \widehat{X_{i-1}^{-1}} \right) + \left( \widehat{X_{i-1}^{-1}} \otimes \widehat{X_i^{-1}} \right) \end{aligned}$$

where the second equality follows from the substitution of (13). The last equality follows from substitution of expression for the first derivative from (15). Distributing terms in the kronecker product on the left hand side of the fourth equality and rearranging yields the claim in (16).  $\square$

THEOREM 3.3. *The optimization problem in (4) is convex.*

*Proof.* The convexity of linear constraint in (4) is obvious. We focus on the feasible set defined by the inequality constraints,  $\mathcal{S} = \{X \mid d_{[i]}(X) \geq 0\}$  in the rest of the proof. We show that for any  $X, Y \in \text{int}(\mathcal{S}) = \mathbb{S}_{++}^n$  the following inequality holds

$$(17) \quad d_{[i]}(Y) \leq d_{[i]}(X) + \nabla d_{[i]}(X) \bullet (Y - X)$$

which proves concavity of  $d_{[i]}(X)$  for  $X \in \text{int}(\mathcal{S})$ . This proves the convexity of  $\text{int}(\mathcal{S})$ . The convexity of  $\mathcal{S}$  follows since the set  $\mathcal{S}$  is the closure of  $\text{int}(\mathcal{S})$ .

The remainder of the proof shows that (17) holds. Let  $D(X), L(X)$  and  $D(Y), L(Y)$  be the unique LDL<sup>T</sup> factorizations for  $X$  and  $Y$ , respectively, by Lemma 2.1. Substituting this into the the right hand side of (17) obtain,

$$\begin{aligned} d_{[i]}(X) + \nabla d_{[i]}(X) \bullet (Y - X) &= d_{[i]}(X) + (L(X)^{-T} e_i e_i^T L(X)^{-1}) \bullet (Y - X) \\ &= d_{[i]}(X) + (e_i e_i^T) \bullet (L(X)^{-1} Y L(X)^{-T} - D(X)) = (e_i e_i^T) \bullet (L(X)^{-1} Y L(X)^{-T}) \\ &= (e_i e_i^T) \bullet \underbrace{(L(X)^{-1} L(Y))}_{=:L} D(Y) L(Y)^T L(X)^{-T} = \sum_{j=1}^n d_{[j]}(Y) (e_i e_i^T) \bullet (L_{[j]} L_{[j]}^T) \\ &= \sum_{j=1}^n d_{[j]}(Y) L_{[ij]}^2 = \sum_{j=1, j \neq i}^n d_{[j]}(Y) L_{[ij]}^2 + d_{[i]}(Y) \geq d_{[i]}(Y) \end{aligned}$$

where the first equality follows from Lemma 3.1, the second equality from the invariance of trace inner product under cyclic permutations, the third equality from  $e_i^T D(X) e_i = d_{[i]}(X)$ , the fourth equality from the substitution of LDL<sup>T</sup> factorization of  $Y$ , the fifth equality from diagonality of  $D(Y)$ , the sixth equality from the simplification of trace inner product and the final equality follows by noting that  $L = L(X)^{-1} L(Y)$  is also unit lower triangular and  $d_{[j]}(Y) > 0$  for all  $j = 1, \dots, n$ . Thus, (17) holds for  $X, Y \in \text{int}(\mathcal{S})$  and the claim follows.  $\square$

To close this section, we also derive an expression for the action of the Hessian of the Lagrangian on a matrix. This will be used in the derivation of the Newton step for the LDL<sup>T</sup> formulation in §4.1.

LEMMA 3.4. For  $X \in \mathbb{S}_{++}^n$ , let  $X = LDL^T$  denote the LDL<sup>T</sup> factorization of  $X$ . Then, for any  $z \in \mathbb{R}^n$  and any symmetric matrix  $G \in \mathbb{S}^n$

$$(18) \quad \text{mat} \left( - \sum_{i=1}^n z_{[i]} \nabla^2 d_{[i]}(X) \text{vec}(G) \right) = L^{-T} \left( (K - \text{Diag}(z \circ^{-1} d)) \circ (L^{-1} G L^{-T}) \right) L^{-1}$$

where  $K_{[ij]} = \frac{z_{[\max(i,j)]}}{d_{[\min(i,j)]}}$ .

*Proof.* The matrix  $\widehat{X}_i^{-1}$  can be expressed using (15) as

$$\widehat{X}_i^{-1} = \widehat{X}_{i-1}^{-1} + (1/d_{[i]}) L^{-T} e_i e_i^T L^{-1}.$$

The Hessian expression in (16) for  $i \geq 2$  can then be simplified as

$$\nabla^2 d_{[i]}(X) = -L^{-T} e_i e_i^T L^{-1} \otimes \widehat{X}_{i-1}^{-1} - \widehat{X}_{i-1}^{-1} \otimes L^{-T} e_i e_i^T L^{-1}.$$

Substituting this expression for the Hessian of  $d_{[i]}(X)$  into the Hessian of the Lagrangian and using the identity  $\text{mat}((A \otimes B) \text{vec}(G)) = BGA$  for  $A, B \in \mathbb{S}^n$  we have

that

$$\begin{aligned}
& \text{mat} \left( - \sum_{i=1}^n z_{[i]} \nabla^2 d_{[i]}(X) \text{vec}(G) \right) \\
&= \sum_{i=2}^n z_{[i]} \left( L^{-T} e_i e_i^T L^{-1} G \widehat{X}_{i-1}^{-1} + \widehat{X}_{i-1}^{-1} G L^{-T} e_i e_i^T L^{-1} \right) \\
(19) \quad &= \sum_{i=2}^n z_{[i]} L^{-T} \left( e_i e_i^T \underbrace{(L^{-1} G L^{-T})}_{=: \widehat{G}} \widehat{D}_{i-1}^{-1} + \widehat{D}_{i-1}^{-1} (L^{-1} G L^{-T}) e_i e_i^T \right) L^{-1} \\
&= L^{-T} \left( \sum_{i=2}^n z_{[i]} \underbrace{(e_i e_i^T \widetilde{G} \widehat{D}_{i-1}^{-1} + \widehat{D}_{i-1}^{-1} \widetilde{G} e_i e_i^T)}_{=: \widehat{G}_i} \right) L^{-1} \\
&= L^{-T} \left( \sum_{i=2}^n z_{[i]} (\widehat{G}_i + \widehat{G}_i^T) \right) L^{-1}
\end{aligned}$$

where we have used  $\nabla^2 d_{[1]}(X) = 0$  in the first equality. The second equality follows by the substitution of  $\widehat{X}_{i-1}^{-1}$  from (10), and the third equality from collecting terms and rearranging. The matrix  $\widehat{G}_i$  satisfies

$$\begin{aligned}
\widehat{G}_i &= e_i e_i^T \widetilde{G} \widehat{D}_{i-1}^{-1} = e_i e_i^T \widetilde{G} \left( \sum_{l=1}^{i-1} \frac{1}{d_{[l]}} e_l e_l^T \right) = e_i \left( \sum_{l=1}^{i-1} \frac{\widetilde{G}_{[il]}}{d_{[l]}} e_l^T \right) \\
\Rightarrow \widehat{G}_{i[kl]} &= \begin{cases} \frac{\widetilde{G}_{[kl]}}{d_{[l]}} & \text{if } k = i, l \leq i-1 \\ 0 & \text{otherwise.} \end{cases}
\end{aligned}$$

In other words,  $\widehat{G}_i$  is a matrix in which only the first  $i-1$  elements of the  $i$ th row are possibly non-zero. Thus,

$$\begin{aligned}
\sum_{i=2}^n z_{[i]} (\widehat{G}_i + \widehat{G}_i^T) &= \begin{bmatrix} 0 & \frac{z_{[2]}}{d_{[1]}} & \dots & \frac{z_{[n-1]}}{d_{[1]}} & \frac{z_{[n]}}{d_{[1]}} \\ \frac{z_{[2]}}{d_{[1]}} & 0 & \dots & \frac{z_{[n-1]}}{d_{[2]}} & \frac{z_{[n]}}{d_{[2]}} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ \frac{z_{[n-1]}}{d_{[1]}} & \frac{z_{[n-1]}}{d_{[2]}} & \dots & 0 & \frac{z_{[n]}}{d_{[n-1]}} \\ \frac{z_{[n]}}{d_{[1]}} & \frac{z_{[n]}}{d_{[2]}} & \dots & \frac{z_{[n]}}{d_{[n-1]}} & 0 \end{bmatrix} \circ \widetilde{G} \\
&= (K - \text{Diag}(z \circ^{-1} d)) \circ \widetilde{G}.
\end{aligned}$$

Substituting the above in (19) yields the claim.  $\square$

**4. Interior Point Method.** The classical barrier formulation for SDP in (4) is

$$\begin{aligned}
(20) \quad & \min C \bullet X - \mu \sum_{i=1}^n \ln(d_{[i]}(X)) \\
& \text{s.t. } \mathcal{A}(X) = b
\end{aligned}$$



where  $\mu > 0$  is the barrier parameter. Since the determinant of  $X$  is  $\det(X) = \prod_{i=1}^n d_{[i]}(X)$ , the barrier term in (20) is the standard self-concordant barrier function for positive semidefinite matrices [35]. Introducing dual variables  $z_{[i]}$  for  $\mu/d_{[i]}(X)$ , the primal-dual first-order stationary conditions for (20) can be written as

$$(21a) \quad C + \mathcal{A}^*(\lambda) - \sum_{i=1}^n z_{[i]} \nabla d_{[i]}(X) = 0$$

$$(21b) \quad \mathcal{A}(X) = b$$

$$(21c) \quad d_{[i]}(X)z_{[i]} = \mu \quad \forall i = 1, \dots, n.$$

Substituting the expression for  $\nabla d_{[i]}(X)$  in (21) yields precisely the form in (5) as alluded to in §1.

Interior point algorithms for SDPs compute a search direction by solving the linearization of the first order stationary conditions such as in (2). The choice of  $P$  (the third equation in (2)) leads to different directions. Further, the choice also dictates if the resulting direction satisfies properties such as primal-dual symmetry, scale invariance and uniqueness of the search direction; refer to [43] for a discussion.

For the LDL<sup>T</sup> direction, there is no such ambiguity in the choice of the step direction; refer to the complementarity constraints in (21c). Given an iterate  $(X, \lambda, z)$  with  $X \succ 0, z > 0$  (possibly infeasible for (21a),(21b)), the interior point algorithm for the LDL<sup>T</sup> formulation computes the search direction by solving a linearization of the equations in (21). The linear system for computing the step  $(\Delta X, \Delta \lambda, \Delta z)$  is

$$(22a) \quad \begin{array}{rcl} \text{mat} \left( - \sum_{i=1}^n z_{[i]} \nabla^2 d_{[i]}(X) \text{vec}(\Delta X) \right) & + \mathcal{A}^*(\Delta \lambda) & - \sum_{i=1}^n \Delta z_{[i]} \nabla d_{[i]}(X) = R_d \\ & \mathcal{A}(\Delta X) & = r_p \\ z_{[i]} (\nabla d_{[i]}(X) \bullet \Delta X) & & + d_{[i]}(X) \Delta z_{[i]} = r_{c[i]} \\ & & \forall i = 1, \dots, n \end{array}$$

with  $R_d, r_p, r_c$  defined as

$$(22b) \quad \begin{aligned} R_d &= -C - \mathcal{A}^*(\lambda) + \sum_{i=1}^n z_{[i]} \nabla d_{[i]}(X), \quad r_p = b - \mathcal{A}(X), \\ \text{and } r_c &= \mu \mathbf{1}_n - D(X)z \end{aligned}$$

where  $\mathbf{1}_n \in \mathbb{R}^n$  is a vector of all ones. Interior point algorithms employ block Gaussian elimination to: (i) eliminate the dual variables for the semidefinite constraint using the linearized complementarity constraint (the third equation in (22a)), (ii) eliminate the primal variables using the dual stationary conditions (first equation in (22a)), and finally, (iii) obtain a reduced system in the dual variables using the equality constraints in the SDP. We also employ the block Gaussian elimination strategy to form the Schur complement system and also describe an efficient method for computing the Schur complement.

The remainder of the section is organized as follows. §4.1 describes the block Gaussian elimination to compute the step. We also prove that the linear system leads to a unique search direction. A comparison of the LDL<sup>T</sup> formulation with the existing SDP search directions is presented in §4.2.

**4.1. Block Gaussian Elimination of the Linear System.** Suppose that  $d_{[i]}(X), z_{[i]} > 0$  for all  $i$ . As a first step, we eliminate  $\Delta z_{[i]}$  using the last equation

in (22a) as

$$(23) \quad \Delta z_{[i]} = \frac{r_{c[i]}}{d_{[i]}(X)} - \frac{z_{[i]}}{d_{[i]}(X)} (\nabla d_{[i]}(X) \bullet \Delta X).$$

Substituting for  $\Delta z_{[i]}$  from (23) into the first equation in (22a) results in,

$$(24) \quad \begin{aligned} & \text{mat} \left( - \sum_{i=1}^n z_{[i]} \nabla^2 d_{[i]}(X) \text{vec}(\Delta X) \right) \\ & + \mathcal{A}^*(\Delta \lambda) + \sum_{i=1}^n \frac{z_{[i]}}{d_{[i]}(X)} (\nabla d_{[i]}(X) \bullet \Delta X) \nabla d_{[i]}(X) \\ & = R_d + \sum_{i=1}^n \frac{r_{c[i]}}{d_{[i]}(X)} \nabla d_{[i]}(X) = R_d + L^{-T} \text{Diag}(r_c \circ^{-1} d(X)) L^{-1}. \end{aligned}$$

The simplification on the right hand side of (24) is due to (15). The left hand side of (24) can be simplified using (18) as

$$(25) \quad \begin{aligned} & \text{mat} \left( - \sum_{i=1}^n z_{[i]} \nabla^2 d_{[i]}(X) \text{vec}(\Delta X) \right) + \sum_{i=1}^n \frac{z_{[i]}}{d_{[i]}(X)} (\nabla d_{[i]}(X) \bullet \Delta X) \nabla d_{[i]}(X) \\ & = L^{-T} \left( (K - \text{Diag}(z \circ^{-1} d)) \circ (L^{-1} \Delta X L^{-T}) \right) L^{-1} \\ & \quad + \sum_{i=1}^n \frac{z_{[i]}}{d_{[i]}(X)} (e_i^T L^{-1} \Delta X L^{-T} e_i) L^{-T} e_i e_i^T L^{-1} \\ & = L^{-T} \left( K \circ (L^{-1} \Delta X L^{-T}) \right) L^{-1}. \end{aligned}$$

Substituting the simplification above into the left hand side of (24), multiplying on the left and right by  $L^T$  and  $L$  respectively we obtain

$$(26) \quad K \circ \widetilde{\Delta X} + \widetilde{\mathcal{A}}^*(\Delta \lambda) = L^T R_d L + \text{Diag}(r_c \circ^{-1} d)$$

where  $\widetilde{\Delta X}, \widetilde{\mathcal{A}}, \widetilde{\mathcal{A}}^*$  are defined as,

$$(27) \quad \begin{aligned} \widetilde{\Delta X} &= L^{-1} \Delta X L^{-T} \\ \widetilde{A}_j &= L^T A_j L \forall j = 1, \dots, m \\ \widetilde{\mathcal{A}}(\widetilde{\Delta X}) &= \left( \widetilde{A}_1 \bullet \widetilde{\Delta X}, \dots, \widetilde{A}_m \bullet \widetilde{\Delta X} \right) \\ \widetilde{\mathcal{A}}^*(\Delta \lambda) &= \sum_{j=1}^m \Delta \lambda_{[j]} \widetilde{A}_j. \end{aligned}$$

The reduced system obtained by eliminating  $\Delta z$  can be written using (26) and the scaled quantities in (27) as

$$(28) \quad \begin{aligned} K \circ \widetilde{\Delta X} + \widetilde{\mathcal{A}}^*(\Delta \lambda) &= L^T R_d L + \text{Diag}(r_c \circ^{-1} d) \\ \widetilde{\mathcal{A}}(\widetilde{\Delta X}) &= r_p \end{aligned}$$

where we have used  $\widetilde{\mathcal{A}}(\widetilde{\Delta X}) = \mathcal{A}(\Delta X)$ , which holds since  $\widetilde{A}_j \bullet \widetilde{\Delta X} = (L^T A_j L) \bullet (L^{-1} \Delta X L^{-T}) = (A_j L L^{-1}) \bullet (\Delta X L^{-T} L^T) = A_j \bullet \Delta X$ . The linear system in (28) can

be cast in a matrix form as

$$(29) \quad \begin{bmatrix} \text{Diag}(\text{vec}(K)) & \tilde{\mathbf{A}}^T \\ \tilde{\mathbf{A}} & 0 \end{bmatrix} \begin{bmatrix} \tilde{\Delta x} \\ \Delta \lambda \end{bmatrix} = \begin{bmatrix} \tilde{r}_d \\ r_p \end{bmatrix}$$

where  $\tilde{r}_d = \text{vec}(L^T R_d L + \text{Diag}(r_c \circ^{-1} d))$ ,  $\tilde{\mathbf{A}} \in \mathbb{R}^{n^2 \times m}$  with  $\tilde{\mathbf{A}}_{[j,\cdot]} = \text{vec}(\tilde{A}_j)^T$ , and  $\tilde{\Delta x} = \text{vec}(\tilde{\Delta X})$ . For all  $X \succ 0, z > 0$  the matrix  $\text{Diag}(\text{vec}(K))$  is positive definite and invertible. The Schur complement matrix  $M$  is defined as

$$(30) \quad M = \tilde{\mathbf{A}} \text{Diag}(\text{vec}(K))^{-1} \tilde{\mathbf{A}}^T.$$

If  $\tilde{\mathbf{A}}^T$  has full column rank of  $m$ , then the Schur complement matrix  $M$  is nonsingular. The search direction  $(\Delta \lambda, \tilde{\Delta x})$  can be computed as

$$(31) \quad \begin{aligned} \Delta \lambda &= M^{-1} \left( \tilde{\mathbf{A}} \text{Diag}(\text{vec}(K))^{-1} \tilde{r}_d - r_p \right) \\ \tilde{\Delta x} &= \text{Diag}(\text{vec}(K))^{-1} \left( \tilde{r}_d - \tilde{\mathbf{A}}^T \Delta \lambda \right). \end{aligned}$$

The step  $\Delta X$  is obtained as  $L \text{mat}(\tilde{\Delta x}) L^T$  and  $\Delta z$  from substituting in (23). Algorithm 2 summarizes the step computation in the LDL<sup>T</sup> formulation. Observe that in the algorithm only the evaluation of  $R_d$  requires the computation of  $L^{-1}$ . However, the step computation only uses  $L^T R_d L$  which can be obtained without computing  $L^{-1}$ . Thus, the step computation of LDL<sup>T</sup> direction can be performed without inverting the lower triangular factor of  $X$ . The theorem states the main properties of the LDL<sup>T</sup> step computation.

**THEOREM 4.1.** *If the constraint matrices  $\{A_1, \dots, A_m\}$  are linearly independent and  $X \succ 0, z > 0$ , then  $\tilde{\mathbf{A}} \text{Diag}(\text{vec}(K))^{-1} \tilde{\mathbf{A}}^T$  is positive definite and the search direction  $(\Delta X, \Delta \lambda, \Delta z)$  is unique and  $\Delta X$  is symmetric.*

*Proof.* Since  $X \succ 0, z > 0$ ,  $\text{Diag}(\text{vec}(K))$  is positive definite. Further,  $\tilde{r}_d, \tilde{\mathbf{A}}$  are vectorizations of symmetric matrices and the matrix  $K$  is also symmetric. Thus, it is easily verified that  $\text{mat}(\tilde{\Delta x})$  is also symmetric. From the linear independence of  $A_j$ 's and non-singularity of  $L$ , we have that  $\tilde{\mathbf{A}}^T$  has full column rank of  $m$ . Combining this with the positive definiteness of  $\text{Diag}(\text{vec}(K))$ , we have that  $\tilde{\mathbf{A}} \text{Diag}(\text{vec}(K))^{-1} \tilde{\mathbf{A}}^T$  is positive definite. Hence,  $\Delta \lambda$  is unique and so also,  $\tilde{\Delta x}$ . Consequently,  $\Delta X$  is also unique while the symmetry of  $\Delta X$  follows from earlier arguments. The uniqueness of  $\Delta z$  follows from (23), and uniqueness of  $\Delta X$  and  $\Delta \lambda$ .  $\square$

**4.2. Comparison with AHO, HKM, NT Directions.** We derive the computational complexity of the LDL<sup>T</sup> direction based on Toh [44, Appendix A]. Let  $A_k = U_k + U_k^T$  where  $U_k$  is upper triangular. Then,  $\tilde{A}_k = (L^T U_k L) + (L^T U_k L)^T$  can be computed in  $n^3 + n^2$  flops. The product of upper triangular matrices  $L^T U_k$  can be computed in  $n^3/3$  flops while the product of upper and lower triangular matrices  $L^T U_k L$  can be computed in  $2n^3/3$  flops (refer [44, Appendix A]). The addition of  $L^T U_k L$  and  $(L^T U_k L)^T$  requires another  $n^2$  flops. The computation of  $\tilde{B}_l$  requires  $n^2$  flops. The  $(k, l)$ -entry of Schur complement matrix  $M_{[kl]} = \tilde{\mathbf{A}}_{[k,\cdot]} \tilde{\mathbf{B}}_{[l,\cdot]}^T$  can be computed in  $n^2$  flops since it is an inner product of two vectors of length  $n^2$ . All elements of the Schur complement matrix can be computed in  $\frac{1}{2}m(m+1) \cdot n^2$  flops since the matrix  $M$  (30) is symmetric, given  $\tilde{\mathbf{A}}, \tilde{\mathbf{B}}$ . The computation of Schur complement matrix

---

**Algorithm 2** Step computation for the  $LDL^T$  formulation.

---

**input**  $(X, \lambda, z) \in \mathbb{S}_{++}^n \times \mathbb{R}^m \times \mathbb{R}_{++}^n$   
 Compute  $L^T R_d L, r_p, r_c$  using (22b)  
 Compute:  
 •  $\tilde{r}_d = \text{vec}(L^T R_d L + \text{Diag}(r_c \circ^{-1} d))$   
 •  $K$  with  $K_{[kl]} = \frac{z_{[\max(k,l)]}}{d_{[\min(k,l)]}} \forall k, l = 1, \dots, n$   
 •  $\tilde{A}_j = L^T A_j L, \tilde{B}_j = \tilde{A}_j \circ^{-1} K \forall j = 1 \dots, m$   
 Set  $\tilde{\mathbf{A}}^T = [\text{vec}(\tilde{A}_1) \dots \text{vec}(\tilde{A}_m)], \tilde{\mathbf{B}}^T = [\text{vec}(\tilde{B}_1) \dots \text{vec}(\tilde{B}_m)]$   
 Compute:  $M = \tilde{\mathbf{A}} \tilde{\mathbf{B}}^T, \tilde{r} = \tilde{\mathbf{B}}^T \tilde{r}_d$   
 Solve:  
 •  $\Delta \lambda = M^{-1}(\tilde{r} - r_p)$   
 •  $\tilde{\Delta} x = (\tilde{r}_d \circ^{-1} \text{vec}(K) - \tilde{\mathbf{B}}^T \Delta \lambda)$   
 •  $\Delta X = L \text{mat}(\tilde{\Delta} x) L^T$   
 •  $\Delta z = (r_c - z \circ \text{diag}(\text{mat}(\tilde{\Delta} x))) \circ^{-1} d$   
**return** Step  $(\Delta X, \Delta \lambda, \Delta z) \in \mathbb{S}^n \times \mathbb{R}^m \times \mathbb{S}^n$

---

requires  $(0.5m^2n^2 + m(n^3 + n^2))$  flops or  $(0.5m^2n^2 + mn^3)$  flops ignoring the lower order  $mn^2$  term. Thus, the total computation cost is identical to that for the NT direction and lower than the HKM, AHO directions. We summarize the properties of the different directions in Table 1. The computational complexity for AHO, HKM and NT directions are from [44, Table 1] and the rest of the information is from Todd, Toh and Tütüncü [43, Table 1].

Table 1: Summary of SDP search directions.

Directions	Primal-dual symmetry	Scale invariance	Uniquely defined Directions	Computational complexity
$LDL^T$	no	no	yes	$mn^3 + 0.5m^2n^2$
AHO	yes	no	no	$4mn^3 + m^2n^2$
HKM	no	yes	yes	$2mn^3 + m^2n^2$
NT	yes	yes	yes	$mn^3 + 0.5m^2n^2$

The computational complexities reported in Table 1 assume that the data matrices  $A_k$  are dense. In the case of sparse matrices, HKM and NT directions can exploit the sparsity [16] to further reduce the computational cost. Fujisawa, Kojima and Nakata [16] described different approaches for reducing the flops involved in computing the elements of the Schur complement matrix. In particular, the most efficient approach in [16] required that the elements of the Schur complement were of the form  $M_{[kl]} = (EA_k F) \bullet A_l$  with the matrices  $E, F$  being dense. The  $LDL^T$  direction does not satisfy this form (refer (30)). The AHO direction also has a form similar to the  $LDL^T$  direction and is limited in its ability to exploit sparsity. Thus, the  $LDL^T$  direction also suffers from this limitation. However, the matrix completion techniques introduced in Fukuda et al. [17] provide an alternate approach to exploiting sparsity and have been shown to reduce the computational times of SDP algorithms [34, 47]. We believe that the computations in the  $LDL^T$  direction will benefit from exploiting



the matrix completion techniques and will be explored in a separate study.

A direction for SDP is said to be primal-dual symmetric [42] if the same direction is generated when the method is applied to the primal and dual formulations of the SDP (1). The directions in the primal formulation are in the space  $\mathbb{S}^n \times \mathbb{R}^m \times \mathbb{R}^n$ . On the other hand, the directions in the dual formulation will be in the space  $\mathbb{R}^n \times \mathbb{R}^m \times \mathbb{S}^n$ . Due to the incompatibility of the direction dimensions for the corresponding variables, the LDL<sup>T</sup> direction is not primal-dual symmetric (refer Table 1).

A direction for SDP is said to be scale invariant if the scaling of the data matrices  $C, A_k$  by an invertible matrix  $Q$  as  $QCQ^T, QA_kQ^T$  results in the step being transformed from  $(\Delta X, \Delta \lambda, \Delta S)$  to  $(Q^{-T}\Delta XQ^{-1}, \Delta \lambda, Q\Delta SQ^T)$ . The LDL<sup>T</sup> direction does not have this property when  $Q$  is allowed to be a general matrix. If we restrict the matrix  $Q$  to be unit upper triangular, then the scale invariance property does hold for the LDL<sup>T</sup> direction. We refer the interested reader to Appendix D for details on this derivation.

**5. Central Path.** The central path in the LDL<sup>T</sup> formulation is the set

$$(32) \quad \left\{ (X^\mu, \lambda^\mu, z^\mu) \in \mathbb{S}_{++}^n \times \mathbb{R}^m \times \mathbb{R}_{++}^n \mid \begin{array}{l} (X^\mu, \lambda^\mu, z^\mu) \text{ satisfies (5)} \\ \text{for some } \mu > 0 \end{array} \right\}.$$

It is well known that the central path for the standard SDP formulation in (2) is unique. We show the analogous result for the LDL<sup>T</sup> formulation by exhibiting a homeomorphism between the solutions to (2) and (21). In the rest of the section when referring to (2), we assume that  $P = I_n$  in the symmetrization operator  $H_P(\cdot)$ . Then,  $H_P(XS) = \frac{1}{2}(XS + SX)$  and

$$\frac{1}{2}(XS + SX) = \mu I_n \iff XS = \mu I_n$$

since  $X, S$  share the eigenvectors.

**THEOREM 5.1.** *Suppose  $\mu > 0$ ,  $X^\mu \in \mathbb{S}_{++}^n$  and  $X^\mu = L^\mu D^\mu (L^\mu)^T$ . The following are true.*

- (i)  $(X^\mu, \lambda^\mu, S^\mu) \in \mathbb{S}_{++}^n \times \mathbb{R}^m \times \mathbb{S}_{++}^n$  solves (2) if and only if  $(X^\mu, \lambda^\mu, z^\mu) \in \mathbb{S}_{++}^n \times \mathbb{R}^m \times \mathbb{R}_{++}^n$  solves (21) where  $\text{Diag}(z^\mu) = (L^\mu)^T S^\mu L^\mu$ .
- (ii) The central path in (32), if it exists, is unique.

*Proof.* Consider (i). From Lemma 3.1, we have that

$$\sum_{i=1}^n z_{[i]}^\mu \nabla d_{[i]}(X^\mu) = (L^\mu)^{-T} \text{Diag}(z^\mu) (L^\mu)^{-1}.$$

Lemma 2.3 yields that  $X^\mu S^\mu = \mu I_n \iff D^\mu z^\mu = \mu \mathbf{1}_n$ ,  $S^\mu = (L^\mu)^{-T} \text{Diag}(z^\mu) (L^\mu)^{-1}$  and the claim follows.

Consider (ii). The central path for the formulation in (2), with  $P = I_n$ , is the set

$$(33) \quad \left\{ (X^\mu, \lambda^\mu, S^\mu) \mid C + \mathcal{A}^*(\lambda^\mu) - S^\mu = 0, \mathcal{A}(X^\mu) = b \text{ and } X^\mu S^\mu = \mu I_n \right\}.$$

Kojima, Shindoh and Hara [25] have shown the existence and uniqueness of (33), provided there exists a point  $(\hat{X}, \hat{\lambda}, \hat{S}) \in \mathbb{S}_{++}^n \times \mathbb{R}^m \times \mathbb{S}_{++}^n$  that satisfies primal and dual feasibility, i.e.

$$\hat{X}, \hat{S} \succ 0, C + \mathcal{A}^*(\hat{\lambda}) - \hat{S} = 0, \text{ and } \mathcal{A}(\hat{X}) = b.$$

The claim in (i) establishes a homeomorphism between the central path (32) of the  $LDL^T$  formulation and the central path defined by (33). From Theorem 4.1, we have that the Newton step computation in (22a) is unique under linear independence of  $\{A_k\}$  for all  $(X, \lambda, z) \in \mathbb{S}_{++}^n \times \mathbb{R}^m \times \mathbb{R}_{++}^n$ . Hence, the Jacobian of (21) is nonsingular at all points on the central path. The claim on the uniqueness of the central path follows from the application of the implicit function theorem.  $\square$

Given the central path in (32), it is reasonable to ask whether the central path can be extended to  $\mu = 0$  and if so, does the path possess analytic properties. Such questions have been the subject of much investigation for the standard formulation (2) by [11, 13, 18, 21, 22, 26, 27, 28, 33]. Given the homeomorphism between the solutions of (2) and (21), we believe that the results for (2) can be extended to the  $LDL^T$  formulation. One of the main difficulties is that the derivative for  $d_{[i]}(X)$  for  $X \in \mathbb{S}_+^n$  are not defined. We set aside a full analysis for a future study. We close the section with the following remark.

REMARK 5.1. *A key requirement for the limit of  $\lim_{\mu \rightarrow 0}(X^\mu, \lambda^\mu, z^\mu)$  to satisfy (5) with  $\mu = 0$  is that*

$$(34) \quad \lim_{\mu \rightarrow 0} L(X^\mu) = L^\circ, \quad \lim_{\mu \rightarrow 0} U(C + \mathcal{A}^*(\lambda^\mu)) = U^\circ, \quad \text{and} \quad (L^\circ)^T U^\circ = I_n.$$

*If the above is true, then it can be readily seen that*

$$\lim_{\mu \rightarrow 0} (X^\mu, \lambda^\mu, z^\mu) \text{ if it exists is a solution of (5) for } \mu = 0.$$

*However,  $L^\circ$  is not unique since the solution  $X^*$  of SDP (1) is typically only positive semidefinite. The conditions in (34) impose certain limiting behavior on both  $L(X^\mu)$  and  $U(C + \mathcal{A}^*(\lambda^\mu))$ . This is indeed restrictive and is unclear if this can be expected to hold in practice for general SDPs.*

**6. Conditioning of the Schur Complement Matrix.** In this section, we investigate the conditioning of the Schur complement matrix (30) of the  $LDL^T$  direction. We need to define certain regularity properties of the solution to the SDP (1) which will be stated in the context of the  $LDL^T$  factorization. The interested reader is referred to Appendix E and F for the equivalence of these conditions to those originally provided in Alizadeh, Haeberly and Overton [3].

For the purposes of the analysis in this section, we make the following assumptions on the optimal solution  $(X^*, \lambda^*, z^*)$  to the SDP (4) satisfying the conditions in (5).

**Assumptions**

(A1)  $(X^*, z^*)$  satisfy strict complementarity and  $X^*$  has rank  $p$  with  $L^*, D^*$  as given in (11) with

$$\begin{aligned} d_{[1]}^* &\geq \cdots \geq d_{[p]}^* > d_{[p+1]}^* = \cdots = d_{[n]}^* = 0 \\ \text{and } 0 &= z_{[1]}^* = \cdots = z_{[p]}^* < z_{[p+1]}^* \leq \cdots \leq z_{[n]}^*. \end{aligned}$$

(A2)  $X^*$  is primal nondegenerate, i.e.

$$\left\{ \left[ \begin{array}{cc} (\tilde{A}_k^*)_{11} & (\tilde{A}_k^*)_{21}^T \\ (\tilde{A}_k^*)_{21} & \mathbf{0}_{n-p \times n-p} \end{array} \right] \right\}_{k=1}^m \text{ are linearly independent,}$$

where  $\tilde{A}_k^* = (L^*)^T A_k L^*$ ,  $(\cdot)_{11} \in \mathbb{S}^p$ ,  $(\cdot)_{21} \in \mathbb{R}^{n-p \times p}$  refer to the subblocks consistent with the notation in (11).

(A3)  $(\lambda^*, (L^*)^{-T} \text{Diag}(z^*)(L^*)^{-1})$  is *dual nondegenerate*, i.e.

$$\left\{ (\tilde{A}_k^*)_{11} \right\}_{k=1}^m \text{ span } \mathbb{S}^p.$$

Assumptions (A2) and (A3) have been shown to be *generic* in [3, Definition 19]. Informally, a property is said to be *generic* for an SDP if it holds in a measure theoretic-sense for almost all instances of the SDP, where an instance is defined by  $(C, \mathcal{A}, b)$ . Assumptions (A1)-(A3) render the solution unique, which we state next.

**THEOREM 6.1.** *Suppose that (A2)-(A3) hold. Then the optimal solution  $(X^*, \lambda^*, z^*)$  to (4) is unique. In addition, if (A1) holds then there exists a unique  $L^*$  for which (5) holds.*

*Proof.* Uniqueness of multipliers  $(\lambda^*, z^*)$  and  $X^*$  follow from Lemma E.2 and Lemma F.2, respectively. Suppose  $L^*$  is not unique and that there exists  $L^\circ \neq L^*$  such that (5) is satisfied. From (52a) in the proof of Lemma F.2

$$\begin{aligned} L^* \neq L^\circ &\implies L_{22}^* \neq L_{22}^\circ \\ (35a) \quad \implies (L^*)^{-T} &= \begin{bmatrix} (L_{11}^*)^{-T} & -(L_{11}^*)^{-T}(L_{21}^*)^T(L_{22}^*)^{-T} \\ \mathbf{0}_{(n-p) \times p} & (L_{22}^*)^{-T} \end{bmatrix}, \text{ and} \\ (L^\circ)^{-T} &= \begin{bmatrix} (L_{11}^\circ)^{-T} & -(L_{11}^\circ)^{-T}(L_{21}^\circ)^T(L_{22}^\circ)^{-T} \\ \mathbf{0}_{(n-p) \times p} & (L_{22}^\circ)^{-T} \end{bmatrix}. \end{aligned}$$

Uniqueness of  $(\lambda^*, z^*)$  and satisfaction of (5) with  $L^*$  and  $L^\circ$  imply that

$$(35b) \quad (L^*)^{-T} \text{Diag}(z^*)(L^*)^{-1} = (L^\circ)^{-T} \text{Diag}(z^*)(L^\circ)^{-1} (= S^*).$$

The equality in (35b) can be simplified using Assumption (A1) as

$$(35c) \quad \begin{aligned} &(L^*)^{-T} \text{Diag}(0, \dots, 0, z_{[p+1]}^*, \dots, z_{[n]}^*)(L^*)^{-1} \\ &= (L^\circ)^{-T} \text{Diag}(0, \dots, 0, z_{[p+1]}^*, \dots, z_{[n]}^*)(L^\circ)^{-1}. \end{aligned}$$

From (35a) and (35c), we have that the matrix  $S^*$  in (35b) has two  $UDU^T$  factorizations. Further, the upper triangular factors differ in the columns that correspond to nonzero diagonal entries  $(z_{[p+1]}^*, \dots, z_{[n]}^*)$ , contradicting Lemma 2.2. Hence,  $L_{22}^* = L_{22}^\circ$  proving that  $L^*$  is unique.  $\square$

To analyze the conditioning of the Schur complement matrix, we will focus on the scaled transformation of the linear system in (22a). Substituting the expression for the Hessian of Lagrangian (18) into (22a) and using the definition of scaled quantities (27), we obtain

$$(36) \quad \begin{aligned} (K - \text{Diag}(z \circ^{-1} d)) \circ \widetilde{\Delta X} + \widetilde{\mathcal{A}}^*(\Delta \lambda) - \text{Diag}(\Delta z) &= L^T R_d L \\ \widetilde{\mathcal{A}}(\widetilde{\Delta X}) &= r_p \\ z \circ \text{diag}(\widetilde{\Delta X}) + d \circ \Delta z &= r_c \end{aligned}$$

where we have used  $\nabla d_{[i]}(X) \bullet \Delta X = (L^{-T} e_i e_i^T L^{-1}) \bullet \Delta X = e_i^T L^{-1} \Delta X L^{-T} e_i = \widetilde{\Delta X}_{[ii]}$ . Defining the matrices  $D_K$  and  $Z_K$  by

$$(37) \quad D_{K[ij]} = d_{[\min(i,j)]}, Z_{K[ij]} = z_{[\max(i,j)]} \text{ we have that } K = Z_K \circ^{-1} D_K.$$



Using  $D_K$  and  $Z_K$ , the linear system in (36) can be equivalently represented in a lifted space as

$$(38) \quad \begin{aligned} \tilde{\mathcal{A}}^*(\Delta\lambda) & - \Delta Z = L^T R_d L \\ \tilde{\mathcal{A}}(\widetilde{\Delta X}) & = r_p \\ Z_K \circ \widetilde{\Delta X} & + D_K \circ \Delta Z = R_c \end{aligned}$$

where  $R_c = \text{Diag}(r_c)$ . We have introduced  $\Delta Z \in \mathbb{S}^n$  as an unknown in (38) in the place of  $\Delta z$ . We show the equivalence between (36) and (38), and relate  $\Delta Z$  and  $\Delta z$  in the following. Since  $R_c = \text{Diag}(r_c)$ , the third equation in (38) can be equivalently written as

$$(37) \quad \begin{cases} \text{diag}(Z_K) \circ \text{diag}(\widetilde{\Delta X}) + \text{diag}(D_K) \circ \text{diag}(\Delta Z) = r_c \\ Z_{K[ij]} \widetilde{\Delta X}_{[ij]} + D_{K[ij]} \Delta Z_{[ij]} = 0 \quad i \neq j \\ z \circ \text{diag}(\widetilde{\Delta X}) + d \circ \Delta v = r_c \\ \Delta Z = \text{Diag}(\Delta v) - (K - \text{Diag}(z \circ^{-1} d)) \circ \widetilde{\Delta X} \end{cases}$$

where  $\Delta v$  represents the diagonal elements in  $\Delta Z$ . The above equivalence can be used to replace the third equation in (38). Further, the variable  $\Delta Z$  in the first equation of (38) can be eliminated to obtain a linear system in  $(\widetilde{\Delta X}, \Delta\lambda, \Delta v)$  that is identical to (36). Thus, the step computation in (38) is equivalent to the computation in (22a) for all  $X \succ 0, z \succ 0$ . We state this in the following without proof for brevity.

LEMMA 6.2. *Suppose  $X \succ 0, z \succ 0$ . Then  $(\widetilde{\Delta X}, \Delta\lambda, \Delta z)$  solves (36) if and only if  $(\widetilde{\Delta X}, \Delta\lambda, \Delta Z)$  solves (38) with  $\text{diag}(\Delta Z) = \Delta z$ .*

We show in the remainder of the section that system in (38) is nonsingular in the limit. Note that this does not imply that the system in (22a) is nonsingular. However, the proof techniques used in showing non-singularity of (38) mirror those used for the AHO direction in [2]. This serves as a precursor to obtaining results on conditioning of the Schur complement matrix.

We first consider the structure of the matrices  $Z_K, D_K$ ,

$$(39) \quad Z_K = \begin{bmatrix} z_{[1]} & z_{[2]} & \cdots & z_{[n-1]} & z_{[n]} \\ z_{[2]} & z_{[2]} & \cdots & z_{[n-1]} & z_{[n]} \\ \vdots & & \ddots & & \vdots \\ z_{[n-1]} & z_{[n-1]} & \cdots & z_{[n-1]} & z_{[n]} \\ z_{[n]} & z_{[n]} & \cdots & z_{[n]} & z_{[n]} \end{bmatrix}, D_K = \begin{bmatrix} d_{[1]} & d_{[1]} & \cdots & d_{[1]} & d_{[1]} \\ d_{[1]} & d_{[2]} & \cdots & d_{[2]} & d_{[2]} \\ \vdots & & \ddots & & \vdots \\ d_{[1]} & d_{[2]} & \cdots & d_{[n-1]} & d_{[n-1]} \\ d_{[1]} & d_{[2]} & \cdots & d_{[n-1]} & d_{[n]} \end{bmatrix}$$

Under (A1), we have that  $Z_K^*$  and  $D_K^*$  have the following block-structure

$$(40) \quad Z_K^* = \begin{bmatrix} \mathbf{0}_{p \times p} & (Z_{21}^*)^T \\ Z_{21}^* & Z_{22}^* \end{bmatrix}, D_K^* = \begin{bmatrix} D_{11}^* & (D_{21}^*)^T \\ D_{21}^* & \mathbf{0}_{(n-p) \times (n-p)} \end{bmatrix}$$

where  $D_{11}^* \in \mathbb{R}^{p \times p}, Z_{22}^* \in \mathbb{R}^{(n-p) \times (n-p)}, D_{21}^*, Z_{21}^* \in \mathbb{R}^{(n-p) \times p}$  all having element-wise positive entries. The form of the step computation in (38) bears resemblance to that of standard interior point methods. In particular, this is similar to the AHO-direction  $H_P(XS) = \frac{1}{2}(XS + SX)$  presented in Alizadeh, Haeberly and Overton [2]. The precise correspondence between the  $LDL^T$  and AHO directions is provided in



Table 2: Relating the quantities in the LDL<sup>T</sup> and AHO directions.

	LDL <sup>T</sup>	AHO
$\widetilde{\Delta X}$	$(L^*)^{-1} \Delta X (L^*)^{-T}$	$(Q^*)^T \Delta X Q^*$
$\widetilde{A}_k$	$(L^*)^T A_k L^*$	$(Q^*)^T A_k Q^*$
$(Z_K^*)_{[ij]}$	$z_{[\max(i,j)]}^*$	$\frac{1}{2}(\omega_{[i]}^* + \omega_{[j]}^*)$
$(D_K^*)_{ij}$	$d_{[\min(i,j)]}^*$	$\frac{1}{2}(\lambda_{[i]}^* + \lambda_{[j]}^*)$

Table 2. We refer the interested reader to Appendix G for details on the AHO direction. In Table 2,  $Q^*$  is the matrix composed of eigenvectors for  $X^*$ , and  $\lambda_{[i]}^*, \omega_{[i]}^*$  are the eigenvalues for  $X^*$  and the multiplier matrix  $S^* (= (L^*)^{-T} \text{Diag}(z^*) (L^*)^{-1})$ , respectively. Observe that the non-zero block structure of  $Z_K^*, D_K^*$  in (40) for the LDL<sup>T</sup> method is identical to that for the AHO direction. The rest of quantities in (38) are non-singular transformations of the quantities in the step computation of AHO direction. Hence, the arguments in the proof of [2, Theorem 3.1] can be invoked to obtain the main result on the conditioning of the linear system.

**THEOREM 6.3.** *Suppose (A1)-(A3) hold at solution  $(X^*, \lambda^*, z^*)$  to SDP (4). Then the linear system in (38) is well-conditioned at the solution. Let  $M^\mu$  be the Schur complement matrix (30) defined on the central path  $(X^\mu, \lambda^\mu, z^\mu)$ . Then*

$$\lim_{\mu \rightarrow 0} (\mu M^\mu) \text{ exists and has rank } \frac{1}{2}p(p+1).$$

If, in addition,  $m > \frac{1}{2}p(p+1) > 0$ , then

$$\text{cond}(M^\mu) \geq \frac{c}{\mu}$$

for some constant  $c > 0$ , where  $\text{cond}(\cdot)$  denotes the condition number of matrix.

*Proof.* It can be verified that the arguments in proof of [2, Theorem 3.1] carry over to the linear system in (38) to yield that

$$(41) \quad \left. \begin{array}{l} \widetilde{A}^*(\Delta\lambda) - \Delta Z = 0 \\ \widetilde{A}(\widetilde{\Delta X}) = 0 \\ Z_K^* \circ \widetilde{\Delta X} + D_K^* \circ \Delta Z = 0 \end{array} \right\} \implies (\widetilde{\Delta X}, \Delta\lambda, \Delta Z) = 0.$$

This proves the claim on the linear system in (38). Since  $d_{[i]}^\mu z_{[i]}^\mu = \mu$  holds on the central path, then  $K_{[kl]} = z_{[\max(k,l)]}^\mu / d_{[\min(k,l)]}^\mu = \mu / (d_{[k]}^\mu d_{[l]}^\mu)$ . Hence,

$$\mu M^\mu_{[kl]} = \widetilde{A}_k^\mu \bullet \left( (d^\mu (d^\mu)^T) \circ \widetilde{A}_l^\mu \right) = \widetilde{A}_k^\mu \bullet \left( \text{Diag}(d^\mu) \widetilde{A}_k \text{Diag}(d^\mu) \right) = A_k X^\mu \bullet A_l X^\mu$$

which is identical to the AHO direction. The remaining claims on the Schur complement matrix follow from Theorems 4.1 and 4.2 in [2].  $\square$

**7. Implementation & Results.** The standard interior point algorithms for SDPs compute a step by solving a linearization of (2). The nonlinearity in the step computation is restricted to the complementarity conditions. It is typically easy to satisfy the dual stationary (the first equation in (2)) and primal feasibility within a

**Algorithm 3**  $LDL^T$ -based Interior Point Method ( $LDL^T$ )

- 
- 1: Let  $\epsilon \in (0, 1)$  be a desired convergence tolerance.
  - 2: Let  $\mathbf{v}^0 = (X^0, \lambda^0, z^0)$  be an initial iterate with  $X^0 \succ 0, z^0 > 0$ .
  - 3: Choose  $\{\delta, \kappa, \tau\} \subset (0, 1)$ .
  - 4: Set  $l = 0$ .
  - 5: **repeat**
  - 6:   Set  $k = 0$ .
  - 7:   **repeat**
  - 8:     Compute  $\Delta \mathbf{v}^k = (\Delta X^k, \Delta \lambda^k, \Delta z^k)$  as described in Algorithm 2.
  - 9:     Set  $\alpha_p^k := \begin{cases} \frac{\tau}{\sigma_{\max}(-(X^k)^{-1}\Delta X^k)} & \text{if } \sigma_{\max}(-(X^k)^{-1}\Delta X^k) > 0 \\ 1 & \text{otherwise} \end{cases}$ , where  
 $\sigma_{\max}(-(X^k)^{-1}\Delta X^k)$  denotes the largest eigenvalue of the matrix  $-(X^k)^{-1}\Delta X^k$ .
  - 10:     Set  $\alpha_d^k := \begin{cases} \frac{\tau}{\max_i(-\Delta z_{[i]}^k/z_{[i]}^k)} & \text{if } \max_i(-\Delta z_{[i]}^k/z_{[i]}^k) > 0 \\ 1 & \text{otherwise.} \end{cases}$
  - 11:     Set  $\mathbf{v}^{k+1} = (X^k + \alpha_p^k \Delta X^k, \lambda^k + \alpha_d^k \Delta \lambda^k, z^k + \alpha_d^k \Delta z^k)$ .
  - 12:     Set  $k = k + 1$ .
  - 13:   **until**  $\theta(\mathbf{v}^k; \mu^l) \leq \kappa \mu^l$
  - 14:   Set  $\mu^{l+1} = \delta \mu^l$ .
  - 15:   Set  $\mathbf{v}(\mu^l) = \mathbf{v}^k, \mathbf{v}^0 = \mathbf{v}(\mu^l)$ .
  - 16:   Set  $l = l + 1$
  - 17: **until**  $\theta(\mathbf{v}(\mu^{l-1}); 0) \leq \epsilon$
  - 18: **return**  $\mathbf{v}(\mu^{l-1})$
- 

few iterations of the interior point method. The remaining iterations of the algorithm are spent in converging on the complementarity constraints.

The  $LDL^T$  formulation (21) results in nonlinearity in the dual stationarity conditions and the complementarity conditions (the third equation of (21)). As a result, the  $LDL^T$  formulation typically satisfies the dual stationarity conditions only in the limit. In this sense, the  $LDL^T$  formulation has similarity to the classical barrier formulation for SDPs. Unlike the standard formulation, the dual variables are restricted to a space in which the primal variables  $X$  and multiplier matrix  $L(X)^{-T} \text{Diag}(z)L(X)^{-1}$  conform to a specific structure.

We implemented an *infeasible interior point algorithm* employing a *monotone* update strategy for the barrier parameter  $\mu$ . The interior point algorithm is summarized in Algorithm 3. For the sake of brevity, we denote by  $\mathbf{v} := (X, \lambda, z) \in \mathbb{S}_{++}^n \times \mathbb{R}^m \times \mathbb{R}_{++}^n$  and  $\Phi(\mathbf{v}; \mu) = 0$  the first-order stationary conditions in (5). The termination criterion in the algorithm is defined as

$$(42) \quad \theta(\mathbf{v}; \mu) := \max \left( \|\text{vec}(D^{\frac{1}{2}} L^T R_d L D^{\frac{1}{2}})\|_{\infty}, \|r_p\|_{\infty}, \max_{i \in \{1, \dots, n\}} \left( \frac{|d_{[i]} z_{[i]} - \mu|}{1 + d_{[i]} + z_{[i]}} \right) \right)$$

where  $R_d, r_p, r_c$  are as defined in (22b). The particular scaling of  $R_d$  is motivated by our analysis of the central path in the limit; refer Remark 5.1. The scaling of  $L^T R_d L$  by  $D^{\frac{1}{2}}$  removes the need for  $L(X(\mu))^{-T}$  to converge to  $U(C + \mathcal{A}^*(\lambda(\mu)))$  in the limit. The scaling used in the complementarity term was observed in our numerical experiments to be beneficial in aiding convergence on ill-conditioned SDPs.

The computations of the quantities  $\alpha_p^k, \alpha_d^k$  in Lines 9 and 10 follow the standard techniques described in [43]. The choices of  $\alpha_p^k, \alpha_d^k$  ensures that

$$X^{k+1} \succeq (1 - \tau)X^k \succ 0 \text{ and } z^{k+1} \geq (1 - \tau)z^k > 0.$$

The inversion of  $X^k$  in the computation of  $\alpha_p^k$  is not necessary. Instead, one can equivalently compute the largest eigenvalue of  $(D^k)^{-\frac{1}{2}} \text{mat}(\widetilde{\Delta x})(D^k)^{-\frac{1}{2}}$  where  $\widetilde{\Delta x}$  is defined in (31) and  $D^k$  is the diagonal matrix in the LDL<sup>T</sup> factorization of  $X^k$ . Thus, in our implementations we never invert  $L(X^k)$ .

The convergence analysis of interior point methods typically require the iterates to lie in a neighborhood around the central path [32]. However, computational experiments with SDPT3 [45] have demonstrated that such restrictions are not necessary for good practical performance. Our computational experiments have also shown that the imposition of a neighborhood around the central path was not necessary.

The results are obtained using the following values for the parameters in the algorithm:  $\delta = 0.5, \kappa = 0.5, \tau = 0.95$  and  $\epsilon = 10^{-6}$ . The initial value of barrier parameter was  $\mu^0 = 1$  and the initial iterates were set as  $X^0 = I_n, z^0 = \mu^0 \mathbf{1}_n$ , and  $\lambda^0 = -(\mathcal{A}\mathcal{A}^*)^{-1}\mathcal{A}(C - \text{Diag}(z^0))$ . The choice of  $\lambda^0$  minimizes  $\|C + \mathcal{A}^*(\lambda) - \text{Diag}(z^0)\|_F$  and is also referred to as the least-squares multiplier estimate.

Algorithm 3 was implemented in MATLAB R2014a and executed on a Linux machine with 3.20 GHz Intel(R) Core(TM) i7-3930K processor and 32 GB RAM. We tested the algorithm on 79 of the SPDLIB instances [7] available at <http://euler.nmt.edu/~brian/sdplib/sdplib.html>. To compare the performance of the LDL<sup>T</sup> direction against the standard SDP directions, we also implemented infeasible interior point algorithms based on:

- primal barrier direction and monotone update strategy for barrier parameter (**Barrier**). The primal barrier direction does not use the dual variables  $z$  in the algorithm and the termination condition for the algorithm was modified as

$$\theta(\mathbf{v}; \mu) := \max \left( \|\text{vec}(D^{\frac{1}{2}}L^T(C + \mathcal{A}^*(\lambda))LD^{\frac{1}{2}} - \mu I_n)\|_\infty, \|r_p\|_\infty \right).$$

where  $\mathbf{v} = (X, \lambda)$ .

- HKM direction and monotone update strategy for barrier parameter (**HKM**). The termination condition for the algorithm was

$$\theta(\mathbf{v}; \mu) := \max \left( \|\text{vec}(C + \mathcal{A}^*(\lambda) - S)\|_\infty, \|r_p\|_\infty, |(X \bullet S)/n - \mu| \right).$$

where  $\mathbf{v} = (X, \lambda, S)$ .

- HKM direction and predictor-corrector based update strategy for barrier parameter (**HKMPC**). The predictor step and centering parameter were computed as described in Algorithm NT-PC-QR of [43].

In our implementations, we scaled the constraint matrices  $A_j$  and right hand side  $b_j$  by  $1/\|\text{vec}(A_j)\|_\infty$ . The objective matrix  $C$  was scaled by  $1/\|\text{vec}(C)\|_\infty$ . SDPT3 and SeDuMi also implement similar constraint scaling strategies. We also implemented iterative refinement [29] to address inaccurate solutions of the linear system (31) when computing  $\Delta\lambda$ . Iterative refinement was invoked whenever the error in satisfaction of the linear system was greater than  $10^{-15}$  and was limited to a maximum of 10 refinement steps. SDPT3, for example, also employs iterative refinement.

Table 3 presents the number of problems solved by our implementations of the SDP algorithms at different convergence tolerances. We have also included the performance of the state-of-the-art SDP codes SeDuMi [41] and SDPT3 [45]. From Table 3,



Table 3: Number of problems solved by each algorithm on the test set of 79 SDPLIB problems for different convergence tolerances.

$\epsilon$	Barrier	LDL <sup>T</sup>	HKM	HKMPC	SeDuMi	SDPT3
$10^{-6}$	46	58	62	64	45	65
$10^{-5}$	54	67	66	78	59	71
$10^{-4}$	56	74	73	78	70	74

it is clear that the barrier algorithm performs worse than the other algorithms. At a tolerance of  $\epsilon = 10^{-6}$ , LDL<sup>T</sup> (Algorithm 3) solved 58 instances while HKM and HKMPC solved 62 and 64 instances, respectively. The number of instances solved by HKMPC is comparable to that of SDPT3. On the other hand, SeDuMi solved only 45 of the instances. At a relaxed tolerance of  $\epsilon = 10^{-5}$ , LDL<sup>T</sup> and HKM solved 67 and 66 instances, respectively, while HKMPC solved all but one of the instances. Once again the performance of LDL<sup>T</sup> is better than that of SeDuMi which solved only 59 instances while SDPT3 solved 71 instances.

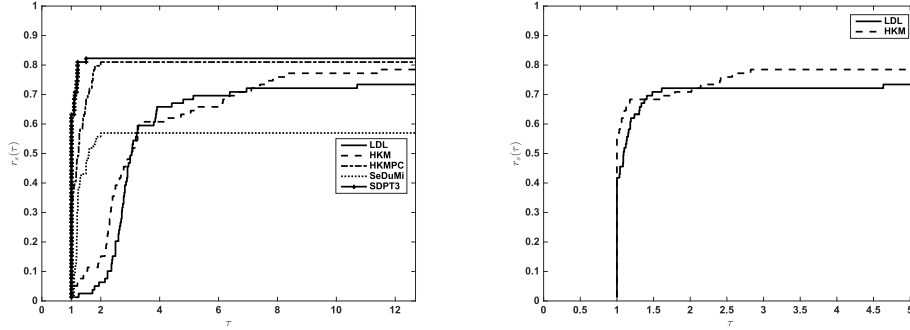
Detailed results on the performance of LDL<sup>T</sup>, HKM, HKMPC and SDPT3 are provided in Table 4 of Appendix H. A careful reading of the results will demonstrate to the reader that HKMPC and SDPT3 require far fewer iterations for convergence than LDL<sup>T</sup> and HKM. To compare the performance of algorithms we employ *performance profiles*, introduced by Dolan and Moré [14]. Figure 1(a) plots the performance profiles of the algorithms - LDL<sup>T</sup>, HKM, HKMPC, and SDPT3. The performance measure  $r_s(\tau)$  in the vertical axis of Figure 1(a) is computed as follows. Let  $it(i, s)$  denote the number of iterations taken by algorithm  $s$  to solve problem instance  $i$ . Then, the quantity  $r_s(\tau)$  is computed as

$$r_s(\tau) = \frac{1}{n_p} \left| \left\{ i \left| \frac{it(i, s)}{\min_{s'} it(i, s')} \leq \tau \right. \right\} \right|$$

where  $n_p$  denotes the number of problem instances. For an algorithm  $s$ ,  $r_s(\tau)$  represents the fraction of problem instances that are solved by algorithm  $s$  within a factor  $\tau$  of the fewest number of iterations taken among all algorithms on that problem. From Figure 1(a) it is clear that HKMPC, and SDPT3 require the fewest number of iterations on most problems. However, it is not possible to draw conclusions on the relative performance of other algorithms. We refer the interested reader to Gould and Scott [20] for a discussion on the limitations of performance profiles.

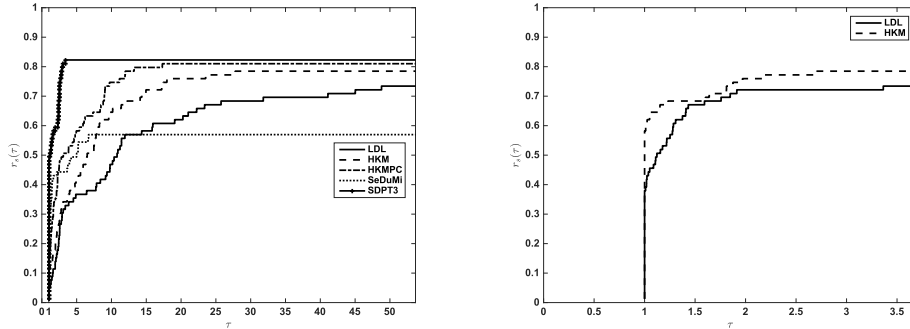
In order to better understand the relative performance of LDL<sup>T</sup> and HKM, Figure 1(b) plots the performance profiles of just these two algorithms. From Table 3 and Figure 1(b), it is clear that the performance of the LDL<sup>T</sup> is comparable to that of the HKM. Figures 2(a) and 2(b) plots performance profiles in terms of time taken by the algorithms. The profiles show a similar trend to that for iterations. The superior performance of HKMPC is attributable entirely to the *adaptive* barrier strategy of the predictor-corrector algorithm. This is consistent with the existing understanding of the practical performance of interior point algorithms. The adaptive barrier strategy of the predictor-corrector algorithm generates search directions that are close to the central path, while also making significant progress towards the solution of the SDP. We believe such a strategy will benefit the LDL<sup>T</sup> algorithm since the stepsize  $\alpha^k$  in the initial iterations is curtailed to ensure that iterates do not violate the inequality constraints  $d_{[i]}(X), z_{[i]} \geq 0$ . However, it is unclear how to extend this to the LDL<sup>T</sup>





(a) Comparing iterations taken by  $LDL^T$ , HKM, HKMPC, SeDuMi, and SDPT3. (b) Comparing iterations taken by  $LDL^T$ , and HKM.

Fig. 1: Performance profiles comparing iterations taken by different algorithms. The convergence tolerance was set to  $10^{-6}$ .



(a) Comparing computational time taken by  $LDL^T$ , HKM, HKMPC, SeDuMi, and SDPT3. (b) Comparing computational time taken by  $LDL^T$ , and HKM.

Fig. 2: Performance profiles comparing computational time taken by different algorithms. The convergence tolerance was set to  $10^{-6}$ .

algorithm given the nonlinearity in the dual stationary conditions.

**8. Conclusions.** We presented an interior point algorithm based on the  $LDL^T$  formulation for SDPs. The contributions of the paper can be summarized as: (i) derivation of the first and second derivative formulae for the SDPs that are efficient to compute; (ii) presentation of the  $LDL^T$  formulation in the context of standard interior point framework with comparable work per iteration; (iii) existence of the central path; and (iv) conditioning of the Schur complement matrix that arises in the step computation. The numerical results on SDPLIB problems clearly motivate the need to develop adaptive barrier strategies to improve the numerical performance. We also believe the dual formulation can exploit sparsity more effectively than the

primal formulation. We will explore these in a subsequent study.

### Appendix A. Proof of Lemma 2.2.

Consider the statement in (a). Following Higham [24, Theorem 10.9(a)], we have there exists an upper triangular matrix such that  $X = R^T R$ . The Cholesky factorization without pivoting described in [24, § 10.3] yields an upper triangular matrix  $R$  in which the  $i$ th row of  $R$  is zero whenever  $R_{[ii]} = 0$ . Defining the lower triangular matrix  $\bar{L} = R^T$ , unit lower triangular matrix  $L$  and diagonal matrix  $D$  as

$$L_{[ij]} = \begin{cases} \bar{L}_{[ij]}/\bar{L}_{[jj]} & \text{for } i > j, \bar{L}_{[jj]} > 0 \\ \bar{L}_{[ij]} & \text{otherwise.} \end{cases} \quad \text{and } D_{[ii]} = \bar{L}_{[ii]}^2$$

$$\implies LDL^T = \sum_{i=1}^n D_{[ii]} L_{[\cdot i]} L_{[\cdot i]}^T = \sum_{i: D_{[ii]} > 0} \bar{L}_{[\cdot i]} \bar{L}_{[\cdot i]}^T = \sum_{i: R_{[ii]} > 0} R_{[\cdot i]}^T R_{[\cdot i]} = X.$$

We will prove uniqueness of  $D$  by contradiction. Suppose  $L', D'$  are also  $LDL^T$  factors of  $X$  with  $D \neq D'$  and that  $X$  is only positive semidefinite. In the following, we will show that  $L_{[\cdot i]} = L'_{[\cdot i]}$  and  $D_{[ii]} = D'_{[ii]}$  sequentially for each  $i$  starting from 1.

- (i) Consider  $i = 1$ . From the definition of the  $LDL^T$  factorization,  $X_{[\cdot 1]} = D_{[11]} L_{[\cdot 1]}$   $= D'_{[11]} L'_{[\cdot 1]}$  and  $L_{[11]} = L'_{[11]} = 1$  we have that  $D_{[11]} = D'_{[11]}$ . Further if  $D_{[11]} > 0$ , then  $L_{[\cdot 1]} = L'_{[\cdot 1]}$ . Thus the claim holds for  $i = 1$ .
- (ii) For  $i = 2$ , consider the matrix  $X^{(2)} = X - D_{[11]} L_{[\cdot 1]} L_{[\cdot 1]}^T = X - D'_{[11]} L'_{[\cdot 1]} (L'_{[\cdot 1]})^T$ . By the definition of the  $LDL^T$  factorization

$$X^{(2)} = \sum_{j=2}^n D_{[jj]} L_{[\cdot j]} L_{[\cdot j]}^T = \sum_{j=2}^n D'_{[jj]} L'_{[\cdot j]} (L'_{[\cdot j]})^T.$$

From (i), we have that first row and column of  $X^{(2)}$  is zero. Among the terms involved in the summations on the right hand side, the term for  $j = 2$  is the only contributor to the second row and column of  $X^{(2)}$ . We can repeat the same argument in (i) for the second column of the matrix  $X^{(2)}$  and show that  $D_{[22]} = D'_{[22]}$  and  $L_{[\cdot 2]} = L'_{[\cdot 2]}$  if  $D_{[22]} > 0$ .

- (iii) For all other  $i \geq 3$ , consider the matrix  $X^{(i)} = X - \sum_{j < i} D_{[jj]} L_{[\cdot j]} L_{[\cdot j]}^T$ . The first  $i - 1$  rows and columns of  $X^{(i)}$  are zero. The same arguments can be repeated for  $X^{(i)}$  to show that the claim holds for  $D_{[ii]}$  and  $L_{[\cdot i]}$ .

This proves the claim on the uniqueness.

Consider (b). From Higham [24, Theorem 10.9(b)] there exists a permutation  $\Pi$  such that

$$\Pi^T X \Pi = R^T R \text{ with } R = \begin{bmatrix} R_{11} & R_{12} \\ \mathbf{0}_{(n-p) \times p} & \mathbf{0}_{(n-p) \times (n-p)} \end{bmatrix}$$

and  $R_{11} \in \mathbb{R}^{p \times p}$ ,  $R_{12} \in \mathbb{R}^{p \times (n-p)}$  unique,  $R_{11}$  upper triangular with positive diagonal elements. Define the matrices  $D_1, L_{11}, L_{21}$  as

$$D_{1[ii]} = (R_{11[ii]})^2, L_{11} = R_{11}^T D_1^{-\frac{1}{2}}, L_{21} = R_{12}^T D_1^{-\frac{1}{2}}$$

where  $D_1$  is diagonal. The claim in (b) can be shown to hold for the above choice of  $D_1, L_{11}, L_{21}$  and any choice of an unit lower triangular matrix for  $L_{22}$ .

### Appendix B. Proof of Lemma 2.3.

Consider the *if* part of the statement. Then,

$$\begin{aligned} XS &= L(X)D(X)L(X)^T L(X)^{-T} D(S)L(X)^{-1} = L(X)D(X)D(S)L(X)^{-1} \\ &= L(X)(\mu I_n)L(X)^{-1} = \mu I_n \end{aligned}$$

which proves the claim. To prove the *only if* part, suppose  $XS = \mu I_n$ . Since  $X \in \mathbb{S}_{++}^n$ , the matrix can be factorized as  $X = L(X)D(X)L(X)^T$  with  $D(X) \succ 0$ . Hence,

$$XS = \mu I_n \implies S = \mu X^{-1} = \mu L(X)^{-T} D(X)^{-1} L(X)^{-1}.$$

Thus,  $D(S) = \mu D(X)^{-1} \succ 0$  which proves the *only if* part of the statement. The uniqueness of  $U(S)$  follows by the same argument as uniqueness of  $L(X)$  (refer Lemma 2.1).

### Appendix C. Proof of Lemma 2.4.

The *if* part of the claim is straightforward. Consider the *only if* part of the claim. We have that there exist factorizations  $X = LD(X)L^T$  and  $S = UD(S)U^T$  satisfying Lemma 2.2(a). We assume without loss of generality that

(43a)

$$L_{[i]} = e_i \text{ for } i \text{ such that } D(X)_{[ii]} = 0 \text{ and } U_{[j]} = e_j \text{ for } j \text{ such that } D(S)_{[jj]} = 0.$$

Consider the statement  $XS = 0$ . Substituting the factorization and left-multiplying by  $L^{-1}$  and right-multiplying by  $U^{-T}$  we obtain

$$XS = 0 \implies \underbrace{D(X)L^T U D(S)}_{=: Y} = 0.$$

Since  $L^T, U$  are both unit upper triangular the matrix  $Y$  is also upper triangular and satisfies

$$(43b) \quad Y_{[ij]} = \begin{cases} D(X)_{[ii]} D(S)_{[ii]} & \text{if } i = j \\ 0 & \text{if } i > j \\ D(X)_{[ii]} D(S)_{[jj]} L_{[i]}^T U_{[j]} & \text{otherwise.} \end{cases}$$

Since

$$(43c) \quad Y_{[ij]} = 0 \implies \begin{cases} D(X)_{[ii]} D(S)_{[ii]} = 0 & \text{if } i = j \\ L_{[i]}^T U_{[j]} = 0 & \text{if } i < j, D(X)_{[ii]} D(S)_{[jj]} > 0 \end{cases}$$

and  $Y = 0$  leads to  $D(X)D(S) = 0$ . The matrix  $L^T U$  is unit upper triangular. If the matrix satisfies  $(L^T U)_{[ij]} = 0$  for all  $i < j$ , then  $L^T U = I_n$  and we have that  $L(X) = U(S)^{-T}$ , proving the claim. In the following, we will construct matrices  $\hat{L}$  (unit lower triangular) and  $\hat{U}$  (unit upper triangular) satisfying the properties

$$(43d) \quad L^T U = \hat{L}^T \hat{U}, \quad X = (L\hat{L}^{-1})D(X)(L\hat{L}^{-1})^T, \quad \text{and} \quad S = (U\hat{U}^{-1})D(S)(U\hat{U}^{-1})^T.$$

This allows to define  $L(X) = L\hat{L}^{-1}$ ,  $U(S) = U\hat{U}^{-1}$  as factors of  $X, S$  satisfying

$$L(X)^T U(S) = (L\hat{L}^{-1})^T (U\hat{U}^{-1}) = \hat{L}^{-T} (L^T U) \hat{U} = \hat{L}^{-T} (\hat{L}^T \hat{U}) \hat{U}^{-1} = I_n$$

which proves the claim.

In the rest of the proof we construct  $\hat{L}, \hat{U}$  satisfying (43d). From (43c), we have that the possible nonzeros in  $L^T U$  correspond to  $i, j$  such that  $D(X)_{[ii]} = 0$  or  $D(S)_{[jj]} = 0$ . Accordingly, define the unit lower triangular matrix  $\hat{L}$  and unit upper triangular matrix  $\hat{U}$  as

$$(43e) \quad \hat{L}_{[\cdot i]} = \begin{cases} e_i & \text{if } D(X)_{[ii]} > 0 \\ (L^T U)_{[\cdot i]}^T & \text{otherwise} \end{cases}, \quad \hat{U}_{[j \cdot]} = \begin{cases} e_j & \text{if } D(S)_{[jj]} > 0 \\ \hat{L}^{-T} (L^T U)_{[j \cdot]} & \text{otherwise.} \end{cases}$$

In other words the matrices  $\hat{L}, \hat{U}$  set the columns that are unique in the factorization of  $X, S$  (refer Lemma 2.2(a)) to unit vectors. Using (43e), it can be verified that

$$(\hat{L}^T \hat{U})_{[ij]} = \begin{cases} 1 & \text{if } i = j \\ 0 & \text{if } i > j \\ 0 & \text{if } i < j, D(X)_{[ii]} D(S)_{[jj]} > 0 \\ (L^T U)_{[ij]} & \text{if } D(X)_{[ii]} = 0 \text{ or } D(S)_{[jj]} = 0. \end{cases} \implies L^T U = \hat{L}^T \hat{U}$$

Therefore, the first part of (43d) is satisfied. Further,  $\hat{L}^{-1}, \hat{U}^{-1}$  satisfy

$$(43f) \quad \begin{aligned} \forall i : D(X)_{[ii]} > 0, (\hat{L}^{-1} \hat{L})_{[\cdot i]} = e_i &\implies \hat{L}^{-1} \hat{L}_{[\cdot i]} = e_i \stackrel{(43e)}{\implies} (\hat{L}^{-1})_{[\cdot i]} = e_i \\ &\implies (L \hat{L}^{-1})_{[\cdot i]} = L_{[\cdot i]} \\ \forall j : D(S)_{[jj]} > 0, (\hat{U}^{-1} \hat{U})_{[j \cdot]} = e_j &\implies \hat{U}^{-1} \hat{U}_{[j \cdot]} = e_j \stackrel{(43e)}{\implies} (\hat{U}^{-1})_{[j \cdot]} = e_j \\ &\implies (U \hat{U}^{-1})_{[j \cdot]} = U_{[j \cdot]} \end{aligned}$$

where in the above we have used the identity  $(A^{-1}A)_{[\cdot i]} = e_i$  for any non-singular matrix  $A$ , which follows from  $A^{-1}A = I_n$ . Thus, the unique columns in  $L, U$  (refer to Lemma 2.2(a)) are retained as such in  $L\hat{L}^{-1}, U\hat{U}^{-1}$ , respectively. Hence,

$$\begin{aligned} (L\hat{L}^{-1})D(X)(L\hat{L}^{-1})^T &= \sum_{i: D(X)_{[ii]} > 0} D(X)_{[ii]} (L\hat{L}^{-1})_{[\cdot i]} (L\hat{L}^{-1})_{[\cdot i]}^T \stackrel{(43f)}{=} X \\ \text{and } (U\hat{U}^{-1})D(S)(U\hat{U}^{-1})^T &= \sum_{j: D(S)_{[jj]} > 0} D(S)_{[jj]} (U\hat{U}^{-1})_{[j \cdot]} (U\hat{U}^{-1})_{[j \cdot]}^T \stackrel{(43f)}{=} S. \end{aligned}$$

Thus, we have constructed  $\hat{L}, \hat{U}$  satisfying (43d) which completes the proof.

#### Appendix D. Discussion on Scale Invariance of $LDL^T$ Direction.

Suppose the data matrices  $C, A_j$  in the SDP (1) are scaled as  $UCU^T, UA_jU^T$  where  $U$  is an unit upper triangular matrix. We represent the scaled problem as

$$(44) \quad \begin{aligned} \min_{W \in \mathbb{S}^n} & (UCU^T) \bullet W \\ \text{s.t. } & (UA_jU^T) \bullet W = b_{[j]} \quad \forall j = 1, \dots, m \\ & W \succeq 0. \end{aligned}$$

The goal of this section is to show: *If  $(\Delta X, \Delta \lambda, \Delta z)$  is the  $LDL^T$  direction for the SDP in (1) at an iterate  $(X, \lambda, z)$ , then  $(U^{-T} \Delta X U^{-1}, \Delta \lambda, \Delta z)$  is the  $LDL^T$  direction for the SDP in (44) at an iterate  $(U^{-T} X U^{-1}, \lambda, z)$ .*



Observe that for every  $X \in \mathbb{S}_{++}^n$  that is feasible to the SDP in (1),  $W = U^{-T}XU^{-1} \in \mathbb{S}_{++}^n$  is feasible to the SDP in (44). Further, if  $L, D$  are the LDL<sup>T</sup> factors for such an  $X$ , then  $U^{-T}L, D$  are the LDL<sup>T</sup> factors for  $W = U^{-T}XU^{-1}$ . Let  $(\Delta X, \Delta \lambda, \Delta z)$  denote the LDL<sup>T</sup> direction for the SDP in (1) at the iterate  $(X, \lambda, z)$ . It can be readily verified that the residuals for the scaled SDP in (44) at an iterate  $(U^{-T}XU^{-1}, \lambda, z)$  have the following relation to those defined in (22b)

$$\begin{aligned} R_d(W, \lambda, z) &= UR_dU^T, \\ r_p(W, \lambda, z) &= r_p, \\ \text{and } r_c(W, \lambda, z) &= r_c. \end{aligned}$$

The linear system for computing the LDL<sup>T</sup> direction  $(\Delta W, \Delta \lambda, \Delta z)$  for the SDP in (44) is

$$(45a) \quad \begin{aligned} \text{mat} \left( \begin{array}{ccc} -\sum_{i=1}^n z_{[i]} \nabla^2 d_{[i]}(W) \text{vec}(\Delta W) & +UA^*(\Delta \lambda)U^T & -\sum_{i=1}^n \Delta z_{[i]} \nabla d_{[i]}(W) \\ & \mathcal{A}(U^T \Delta WU) & \\ z_{[i]} (\nabla d_{[i]}(W) \bullet \Delta W) & & \forall i = 1, \dots, n \end{array} \right) \begin{array}{l} = UR_dU^T \\ = r_p \\ = r_{c[i]} \end{array} \end{aligned}$$

where we have used the identities  $\sum_{j=1}^m \Delta \lambda_{[j]} UA_j U^T = UA^*(\Delta \lambda)U^T$  and  $(UA_j U^T) \bullet \Delta W = A_j \bullet (U^T \Delta WU)$ . The steps outlined in §4.1 can be repeated to compute the LDL<sup>T</sup> direction for (44). The key difference is that the occurrences of  $L$  in §4.1 should be replaced by  $U^{-T}L$ . The step  $\Delta z$  in (45a) can be obtained as

$$(45b) \quad \Delta z_{[i]} = \frac{r_{c[i]}}{d_{[i]}(X)} - \frac{z_{[i]}}{d_{[i]}(X)} (\nabla d_{[i]}(W) \bullet \Delta W).$$

Substituting for  $\Delta z_{[i]}$  from (45b) into the first equation in (45a) results in

$$(45c) \quad \text{mat} \left( -\sum_{i=1}^n z_{[i]} \nabla^2 d_{[i]}(W) \text{vec}(\Delta W) \right) = UR_dU^T + UL^{-T} \text{Diag}(r_c \circ^{-1} d) L^{-1}U^T$$

where the left hand side can be simplified as

$$(45d) \quad \begin{aligned} &\text{mat} \left( -\sum_{i=1}^n z_{[i]} \nabla^2 d_{[i]}(W) \text{vec}(\Delta W) \right) + \sum_{i=1}^n \frac{z_{[i]}}{d_{[i]}(W)} (\nabla d_{[i]}(W) \bullet \Delta W) \nabla d_{[i]}(W) \\ &= UL^{-T} (K \circ (L^{-1}U^T \Delta WU L^{-T})) L^{-1}U^T. \end{aligned}$$

Note again that the simplification follows from replacing occurrences of  $L^{-1}$  and  $L^{-T}$  in (25) with  $L^{-1}U^T$  and  $UL^{-T}$  respectively. Substituting the simplification above into the left hand side of (45c), multiplying on the left and right by  $L^T U^{-1}$  and  $U^{-T}L$  respectively we obtain

$$(45e) \quad K \circ \widetilde{\Delta W} + \widetilde{\mathcal{A}}^*(\Delta \lambda) = L^T R_d L + \text{Diag}(r_c \circ^{-1} d)$$

where  $\widetilde{\Delta W} = L^{-1}U^T \Delta WU L^{-T}$  and  $\widetilde{\mathcal{A}}, \widetilde{\mathcal{A}}^*$  are defined in (27). The reduced system obtained by eliminating  $\Delta z$  can be written using (45e) and the scaled quantities in (27) as

$$(45f) \quad \begin{aligned} K \circ \widetilde{\Delta W} + \widetilde{\mathcal{A}}^*(\Delta \lambda) &= L^T R_d L + \text{Diag}(r_c \circ^{-1} d) \\ \widetilde{\mathcal{A}}(\widetilde{\Delta W}) &= r_p. \end{aligned}$$

A visual comparison of (28) and (45f) shows that  $\widetilde{\Delta X} = \widetilde{\Delta W}$ . In other words,  $\Delta W = U^{-T} \Delta X U$  and substituting in (45b) allows to verify that the step  $\Delta z$  is identical to the one obtained in (23). Thus, scaling the data matrices  $C, A_j$  by an unit upper triangular matrix  $U$  results in the  $LDL^T$  direction being transformed from  $(\Delta X, \Delta \lambda, \Delta z)$  to  $(U^{-T} \Delta X U^{-1}, \Delta \lambda, \Delta z)$ .

### Appendix E. Discussion on Primal Nondegeneracy.

The primal nondegeneracy conditions are presented in terms of tangent spaces of semidefinite manifolds. In particular, we are interested in the following manifold of fixed rank, say  $p$ , symmetric, semidefinite matrices

$$\mathcal{M}_p = \{X' \in \mathbb{S}^n \mid \text{rank}(X') = p\} \text{ and } \mathcal{M}_p^+ = \mathbb{S}_+^n \cap \mathcal{M}_p.$$

where  $p$  for instance represents the rank of the solution to the SDP. Arnold [5] and Shapiro and Fan [39] derive the tangent space at  $X \in \mathcal{M}_p$  in terms of the spectral decomposition of  $X$ . In the context of SDPs, it is sufficient to consider  $X \in \mathcal{M}_p^+$ . Further, we assume without loss of generality that  $X \in \mathcal{M}_p^+$  satisfies

$$(46) \quad d_{[1]}(X) \geq \dots \geq d_{[p]}(X) > d_{[p+1]}(X) = \dots = d_{[n]}(X) = 0.$$

The tangent space to  $\mathcal{M}_p$  at  $X \in \mathcal{M}_p^+$  can be defined by the linear equations [38, Eq. (23)],

$$(47) \quad \mathcal{T}_X = \{X' \in \mathbb{S}^n \mid v_i^T X' v_j = 0 \forall 1 \leq i \leq j \leq n-p\}$$

where  $v_1, \dots, v_{n-p}$  is a basis for the null space of the matrix  $X$ . A basis for the null space of matrix  $X \in \mathcal{M}_p^+$  can be obtained as follows. Let  $L, D$  denote the  $LDL^T$  factors of  $X$ . Then  $XL^{-T}e_j = LDe_j \stackrel{(46)}{=} 0$  for all  $j = (p+1), \dots, n$  where  $e_j \in \mathbb{R}^n$  is the unit vector. Hence, a basis for the null space of  $X$  is

$$(48) \quad [v_1 \ \dots \ v_{n-p}] = L(X)^{-T} [e_{p+1} \ \dots \ e_n].$$

It also follows from (48) that  $v_i^T X' v_j = 0 \implies e_{p+i}^T L^{-1} X' L^{-T} e_{p+j} = 0$ . Thus, at  $X \in \mathcal{M}_p^+$  the tangent space is

$$\mathcal{T}_X = \left\{ L \begin{bmatrix} E & F \\ F^T & \mathbf{0}_{(n-p) \times (n-p)} \end{bmatrix} L^T \mid E \in \mathbb{S}^p, F \in \mathbb{R}^{p \times (n-p)} \right\}.$$

Under the trace inner product for symmetric matrices, the space orthogonal to the tangent space  $\mathcal{T}_X$  is

$$\mathcal{T}_X^\perp = \left\{ L^{-T} \begin{bmatrix} \mathbf{0}_{p \times p} & \mathbf{0}_{p \times (n-p)} \\ \mathbf{0}_{(n-p) \times p} & G \end{bmatrix} L^{-1} \mid G \in \mathbb{S}^{n-p} \right\}$$

Further, define the set  $\mathcal{N}$  as the null space of the equality constraints

$$\mathcal{N} = \{X' \in \mathbb{S}^n \mid A_k \bullet X' = 0 \forall k = 1, \dots, m\}$$

and the space orthogonal to  $\mathcal{N}$  is

$$(49) \quad \mathcal{N}^\perp = \left\{ Y \mid Y = \sum_{k=1}^m \alpha_k A_k \right\}.$$

From [3],  $X$  is said to be *primal nondegenerate* if  $X$  is feasible to SDP (1) and

$$(50) \quad \mathcal{T}_X + \mathcal{N} = \mathbb{S}^n.$$

We will prove a necessary and sufficient condition for primal nondegeneracy in terms of the defined spaces. Note this was shown in [3] using the spectral decomposition of the matrix  $X$ .

LEMMA E.1. *Suppose  $X \in \mathcal{M}_p^+$  satisfying (46) is feasible for SDP (1) and  $\{A_k\}$  are linearly independent. Then,  $X$  is primal nondegenerate if and only if the matrices*

$$(51) \quad \left\{ \begin{bmatrix} (L^T A_k L)_{11} & (L^T A_k L)_{21}^T \\ (L^T A_k L)_{21} & \mathbf{0}_{(n-p) \times (n-p)} \end{bmatrix} \right\}_{k=1}^m$$

are linearly independent, where  $(\cdot)_{11} \in \mathbb{S}^p, (\cdot)_{21} \in \mathbb{R}^{(n-p) \times p}$  refer to the subblocks consistent with the notation in (11).

*Proof.* The condition (50) is equivalent to  $\mathcal{T}_X^\perp \cap \mathcal{N}^\perp = \{\emptyset\}$ . Let  $B_k$  for  $k = 1, \dots, m$  denote the matrices in (51). Suppose  $\{B_k\}$  are linearly dependent. Then

$$\begin{aligned} & \sum_{k=1}^m \beta_k B_k = 0 \text{ for some } \{\beta_k\} \text{ not all zero} \\ \iff & \sum_{k=1}^m \beta_k L^T A_k L - \sum_{i=p+1}^n \sum_{j=p+1}^n \alpha_{ij} e_i e_j^T = 0 \text{ where } \alpha_{ij} = \sum_{k=1}^m \beta_k (L^T A_k L)_{[ij]} \\ \iff & \sum_{k=1}^m \beta_k A_k = L^{-T} \left( \sum_{i=p+1}^n \sum_{j=p+1}^n \alpha_{ij} e_i e_j^T \right) L^{-1} \\ \iff & \mathcal{T}_X^\perp \cap \mathcal{N}^\perp \neq \{\emptyset\} \end{aligned}$$

where the last statement follows by noting that the left hand side of the preceding statement is in  $\mathcal{N}^\perp$  while the right hand side is in  $\mathcal{T}_X^\perp$ . This proves the claim.  $\square$

A consequence of (51) is that the dual solutions are unique as shown next.

LEMMA E.2. *Suppose  $X^*$  is optimal to SDP (1) satisfying (46) and is primal nondegenerate. Then  $(\lambda^*, z^*)$  satisfying (5) for  $\mu = 0$  are unique.*

*Proof.* Suppose the multipliers are not unique. Since  $X^*$  is positive semidefinite, the unit lower triangular factor  $L(X^*)$  is not unique. Accordingly, suppose that

- $(\lambda^*, z^*)$  with  $L(X^*) = L^*$
- $(\lambda^\circ, z^\circ)$  with  $L(X^*) = L^\circ$

both satisfy (5) for  $\mu = 0$ . By Lemma 2.2 and (46)

$$(52a) \quad \begin{aligned} & L^* \neq L^\circ \\ \implies & L_{22}^* \neq L_{22}^\circ \\ \implies & L^* = \begin{bmatrix} L_{11}^* & \mathbf{0}_{p \times (n-p)} \\ L_{21}^* & L_{22}^* \end{bmatrix}, L^\circ = \begin{bmatrix} L_{11}^* & \mathbf{0}_{p \times (n-p)} \\ L_{21}^* & L_{22}^\circ \end{bmatrix} \\ \implies & L^* = \hat{L} \begin{bmatrix} I_p & \mathbf{0}_{p \times (n-p)} \\ \mathbf{0}_{(n-p) \times p} & L_{22}^* \end{bmatrix}, L^\circ = \hat{L} \begin{bmatrix} I_p & \mathbf{0}_{p \times (n-p)} \\ \mathbf{0}_{(n-p) \times p} & L_{22}^\circ \end{bmatrix}. \end{aligned}$$

Note that  $\hat{L}$  is also an unit lower triangular factor of  $X^*$ . By the complementarity condition in (5) and (46),

$$(52b) \quad z_{[1]}^* = \dots = z_{[p]}^* = 0 \text{ and } z_{[1]}^\circ = \dots = z_{[p]}^\circ = 0.$$

Since  $(\lambda^*, z^*)$  and  $(\lambda^\circ, z^\circ)$  satisfy (5) we have by the first equation in (5) that

$$(52c) \quad \begin{aligned} C + \mathcal{A}^*(\lambda^*) - (L^*)^{-T} \begin{bmatrix} \mathbf{0}_{p \times p} & \mathbf{0}_{p \times (n-p)} \\ \mathbf{0}_{(n-p) \times p} & D_2^* \end{bmatrix} (L^*)^{-1} &= 0 \\ C + \mathcal{A}^*(\lambda^\circ) - (L^\circ)^{-T} \begin{bmatrix} \mathbf{0}_{p \times p} & \mathbf{0}_{p \times (n-p)} \\ \mathbf{0}_{(n-p) \times p} & D_2^\circ \end{bmatrix} (L^\circ)^{-1} &= 0 \end{aligned}$$

where  $D_2^* = \text{Diag}(z_{[p+1]}^*, \dots, z_{[n]}^*)$  and  $D_2^\circ = \text{Diag}(z_{[p+1]}^\circ, \dots, z_{[n]}^\circ)$ . We multiply the equations in (52c) on the left and right by  $\hat{L}^T$  and  $\hat{L}$  respectively to obtain

$$(52d) \quad \begin{aligned} \hat{L}^T (C + \mathcal{A}^*(\lambda^*)) \hat{L} - \begin{bmatrix} \mathbf{0}_{p \times p} & \mathbf{0}_{p \times (n-p)} \\ \mathbf{0}_{(n-p) \times p} & (L_{22}^*)^{-T} D_2^* (L_{22}^*)^{-1} \end{bmatrix} &= 0 \\ \hat{L}^T (C + \mathcal{A}^*(\lambda^\circ)) \hat{L} - \begin{bmatrix} \mathbf{0}_{p \times p} & \mathbf{0}_{p \times (n-p)} \\ \mathbf{0}_{(n-p) \times p} & (L_{22}^\circ)^{-T} D_2^\circ (L_{22}^\circ)^{-1} \end{bmatrix} &= 0. \end{aligned}$$

Subtracting the equations in (52d) obtain

$$\begin{aligned} \sum_{k=1}^m (\lambda_{[k]}^* - \lambda_{[k]}^\circ) \hat{L}^T A_k \hat{L} - \sum_{i=p+1}^n \sum_{j=p+1}^n \alpha_{ij} e_i e_j^T &= 0 \\ \text{where } \alpha_{ij} &= ((L_{22}^*)^{-T} D_2^* (L_{22}^*)^{-1} - (L_{22}^\circ)^{-T} D_2^\circ (L_{22}^\circ)^{-1})_{[ij]} \\ \implies \lambda^* - \lambda^\circ &= 0 \end{aligned}$$

where the last implication follows by (51). Since  $\lambda^* = \lambda^\circ$ , (52c) implies that  $(L^*)^{-T}$ ,  $\text{Diag}(z^*)$  and  $(L^\circ)^{-T}$ ,  $\text{Diag}(z^\circ)$  are both  $UDU^T$  factorizations of  $(C + \mathcal{A}^*(\lambda^*))$ . By Lemma 2.2 and Remark 2.1  $z^* = z^\circ$ , proving the claim.  $\square$

#### Appendix F. Discussion on Dual Nondegeneracy.

Let  $(\lambda, S)$  be feasible for the dual SDP problem,

$$(53) \quad \begin{aligned} \max_{\lambda \in \mathbb{R}^m, S \in \mathbb{S}^n} & -b^T \lambda \\ \text{s.t. } & C + \mathcal{A}^*(\lambda) - S = 0 \\ & S \succeq 0. \end{aligned}$$

Let  $q = \text{rank}(S)$  and let  $U$  be the unit upper triangular factor in the  $UDU^T$  factorization of  $S$ . According to Lemma 2.2 we have

$$(54) \quad S = UDU^T \text{ with } U = \begin{bmatrix} U_{11} & U_{12} \\ \mathbf{0}_{q \times (n-q)} & U_{22} \end{bmatrix}, D = \begin{bmatrix} \mathbf{0}_{(n-q) \times (n-q)} & \mathbf{0}_{(n-q) \times q} \\ \mathbf{0}_{q \times (n-q)} & D_2 \end{bmatrix}$$

where  $U_{11} \in \mathbb{R}^{(n-q) \times (n-q)}$ ,  $U_{22} \in \mathbb{R}^{q \times q}$  are unit upper triangular,  $U_{12} \in \mathbb{R}^{(n-q) \times q}$  and  $D_2 \succ 0 \in \mathbb{R}^{q \times q}$  is diagonal. Following the exposition in Appendix E, the tangent space at  $S \in \mathcal{M}_q^+$  satisfying the decomposition in (54) is

$$(55) \quad \mathcal{T}_S = \left\{ U \begin{bmatrix} \mathbf{0}_{(n-q) \times (n-q)} & F \\ F^T & G \end{bmatrix} U^T \mid G \in \mathbb{S}^q, F \in \mathbb{R}^{(n-q) \times q} \right\}.$$

From [3],  $(\lambda, S)$  is said to be *dual nondegenerate* if  $(\lambda, S)$  is feasible to (53) and

$$(56) \quad \mathcal{T}_S + \mathcal{N}^\perp = \mathbb{S}^n.$$

We will prove a necessary and sufficient condition for dual nondegeneracy in terms of the defined spaces. This was shown in [3] using the spectral decomposition of  $S$ .



LEMMA F.1. Suppose  $(\lambda, S) \in \mathbb{R}^m \times \mathcal{M}_q^+$  with  $S$  satisfying (54) is feasible to SDP (53) and  $\{A_k\}$  are linearly independent. Then,  $S$  is dual nondegenerate if and only if the matrices

$$(57) \quad \{(U^{-1}A_kU^{-T})_{11}\}_{k=1}^m \text{ span } \mathbb{S}^{n-q},$$

where  $(\cdot)_{11} \in \mathbb{S}^{(n-q) \times (n-q)}$  refers to the subblock consistent with the notation in (54).

*Proof.* Let  $U$  be an upper triangular factor in the  $UDU^T$  factorization of  $S$ . Let  $W \in \mathbb{S}^n$  be any matrix. Then,

$$\begin{aligned} W &= U(U^{-1}WU^{-T})U^T = U \begin{bmatrix} (U^{-1}WU^{-T})_{11} & (U^{-1}WU^{-T})_{21}^T \\ (U^{-1}WU^{-T})_{21} & (U^{-1}WU^{-T})_{22} \end{bmatrix} U^T \\ &= U \begin{bmatrix} (U^{-1}WU^{-T})_{11} & \mathbf{0}_{(n-q) \times q} \\ \mathbf{0}_{q \times (n-q)} & \mathbf{0}_{q \times q} \end{bmatrix} U^T + U \begin{bmatrix} \mathbf{0}_{(n-q) \times (n-q)} & (U^{-1}WU^{-T})_{21}^T \\ (U^{-1}WU^{-T})_{21} & (U^{-1}WU^{-T})_{22} \end{bmatrix} U^T. \end{aligned}$$

There exist  $\alpha_k$  for  $k = 1, \dots, m$  such that

$$(U^{-1}WU^{-T})_{11} = \sum_{k=1}^m \alpha_k (U^{-1}A_kU^{-T})_{11} \text{ if (57) holds.}$$

Then,

$$\begin{aligned} W &= U \left( \sum_{k=1}^m \alpha_k U^{-1}A_kU^{-T} \right) U^T + U \begin{bmatrix} \mathbf{0}_{(n-q) \times (n-q)} & F \\ F^T & G \end{bmatrix} U^T \text{ if (57) holds} \\ \text{with } F^T &= (U^{-1}WU^{-T})_{21} - \sum_{k=1}^m \alpha_k (U^{-1}A_kU^{-T})_{21} \\ G &= (U^{-1}WU^{-T})_{22} - \sum_{k=1}^m \alpha_k (U^{-1}A_kU^{-T})_{22} \end{aligned}$$

$$\implies \mathbb{S}^n = \mathcal{N}^\perp + \mathcal{T}_S \text{ if (57) holds}$$

where the implication follows from (49) and (55). The *only if* part of the claim follows from the definition of dual nondegeneracy in (55) and the form of  $\mathcal{N}^\perp$  in (49) and  $\mathcal{T}_S$  in (55).  $\square$

A consequence of (57) is that the primal solution is unique as shown next.

LEMMA F.2. Suppose  $(\lambda^*, S^*)$  is optimal to the dual SDP (53) and is dual nondegenerate. Then the optimal solution  $X^*$  is unique.

*Proof.* By the complementarity condition and the decomposition for  $S^*$  in (54), any optimal solution  $X^*$  to SDP (1) must be of the form

$$X^* = (U^*)^{-T} \begin{bmatrix} X_{11} & \mathbf{0}_{(n-q) \times q} \\ \mathbf{0}_{q \times (n-q)} & \mathbf{0}_{q \times q} \end{bmatrix} (U^*)^{-1}$$

where  $X_{11} \succeq 0$ . Substituting for  $X^*$  in the equality constraints obtain

$$((U^*)^{-1}A_k(U^*)^{-T})_{11} \bullet X_{11} = b_{[k]} \quad \forall k = 1, \dots, m \implies X_{11} \text{ is unique}$$

where the implication follows by (57), proving the claim.  $\square$

### Appendix G. AHO Step Transformation.

The Newton step of the stationary conditions (2) corresponding to the AHO direction ( $P = I$ ) is computed by solving the linear system

$$(58) \quad \begin{array}{rcl} \mathcal{A}^*(\Delta\lambda) & -\Delta S & = R_p \quad (=: -(C + \mathcal{A}^*(\lambda) - S)) \\ \mathcal{A}(\Delta X) & & = r_p \quad (=: b - \mathcal{A}(X)) \\ \frac{1}{2}(S\Delta X + \Delta X S) & + \frac{1}{2}(X\Delta S + \Delta S X) & = R_c \quad (=: -\frac{1}{2}(XS + SX)). \end{array}$$

At a solution  $(X^*, \lambda^*, S^*)$  to SDP (1), it is well known that  $X^*, S^*$  share the eigenvectors [25] i.e.

$$(59) \quad X^* = Q^* \text{Diag}(\lambda^*)(Q^*)^T \text{ and } S^* = Q^* \text{Diag}(\omega^*)(Q^*)^T$$

where  $Q^* \in \mathbb{R}^{n \times n}$  is an orthonormal matrix of the eigenvectors of  $X^*$  and  $S^*$ , and  $\lambda^*, \omega^* \in \mathbb{R}^n$  are the eigenvalues of  $X^*, S^*$ , respectively. Further,  $\lambda^* \circ \omega^* = 0$  since  $X^* S^* = 0$ . For the purposes of this section, we will assume that *strict complementarity* holds,  $\lambda^* + \omega^* > 0$  and further

$$(60) \quad \begin{array}{l} \lambda_{[1]}^* \geq \dots \geq \lambda_{[p]}^* > \lambda_{[p+1]}^* = \dots = \lambda_{[n]}^* = 0, \\ \text{and } 0 = \omega_{[1]}^* = \dots = \omega_{[p]}^* < \omega_{[p+1]}^* \leq \dots \leq \omega_{[n]}^*. \end{array}$$

Consider the following transformation of the step  $\Delta X$  and  $\Delta S$

$$\widetilde{\Delta X} := (Q^*)^T \Delta X Q^* \text{ and } \widetilde{\Delta S} := (Q^*)^T \Delta S Q^*.$$

With this transformation of variables the linear system in (58) can be recast as

$$(61) \quad \begin{array}{rcl} \widetilde{\mathcal{A}}^*(\Delta\lambda) & -\widetilde{\Delta S} & = (Q^*)^T R_p Q^* \\ \widetilde{\mathcal{A}}(\widetilde{\Delta X}) & & = r_p \\ \Omega_K^* \circ \widetilde{\Delta X} & + \Lambda_K^* \circ \widetilde{\Delta S} & = (Q^*)^T R_c Q^* \end{array}$$

where  $\widetilde{\mathcal{A}}(\widetilde{\Delta X}) = [((Q^*)^T A_1 Q^*) \bullet \widetilde{\Delta X} \dots ((Q^*)^T A_m Q^*) \bullet \widetilde{\Delta X}]$ ,  $\Omega_K^* = \frac{1}{2}(\mathbf{1}_n(\omega^*)^T + (\omega^*) \mathbf{1}_n^T)$  and  $\Lambda_K^* = \frac{1}{2}(\mathbf{1}_n(\lambda^*)^T + (\lambda^*) \mathbf{1}_n^T)$ . The correspondences listed in Table 2 can be inferred by comparing the linear systems in (61) and (38).

### Appendix H. Detailed Numerical Results.

Table 4 presents the results from solving the SDPLIB [7] instances using different algorithms described in §7. The reported results are for the convergence tolerance of  $10^{-6}$ . We set an iteration limit of 200 for all algorithms. The results reported in Table 4 are for the iterate  $\mathbf{v}^k$  that has the least  $\theta(\mathbf{v}^k; 0)$  defined in §7. We will term this iterate the *best iterate* for short and denote it as  $\mathbf{v}^{\text{best}} = (X^{\text{best}}, \lambda^{\text{best}}, z^{\text{best}})$ . The best iterate is indeed the *optimal solution* when the algorithm achieves the tolerance of  $10^{-6}$ . In Table 4, Primal Obj reported is  $C \bullet X^{\text{best}}$ , Dual Obj corresponds to  $-b^T \lambda^{\text{best}}$ , Opt Tol corresponds to  $\theta(\mathbf{v}^{\text{best}}, 0)$  and # iter - number of iterations taken by the algorithm, Time (s) - computational time in seconds, and Status reports the status of the algorithm at termination. The possible termination status are: OPT - solved to tolerance, ITL - iteration limit reached, NUM - numerical issues encountered. Termination of the algorithm with NUM is typically due to the Schur complement matrix being ill-conditioned with condition numbers on order of  $10^{15}$  or higher.

Table 4: Summary of performance on SDPLIB problems with different implementations. Primal Obj - primal objective, Dual Obj - dual objective, Opt Tol - norm of the error in termination criterion, # iter - number of iterations, Time (s) - computational time in seconds, Status - termination status: OPT - solved to tolerance, ITL - iteration limit reached, NUM - numerical issues encountered.

	LDL <sup>T</sup>			HKM			HKMPC			SDPT3		
	Primal Obj Dual Obj	Opt Tol # Iter Time (s)	Status	Primal Obj Dual Obj	Opt Tol # Iter Time (s)	Status	Primal Obj Dual Obj	Opt Tol # Iter Time (s)	Status	Primal Obj Dual Obj	Opt Tol # Iter Time (s)	Status
arch0	-5.662823e-01 -5.666013e-01	9.5e-07 89	OPT	-5.663487e-01 -5.665567e-01	6.8e-07 98	OPT	-5.664371e-01 -5.665432e-01	3.5e-07 23	OPT	-5.665168e-01 -5.665174e-01	3.1e-07 23	OPT
arch2	-6.711453e-01 -6.710242e-01	9.5e-07 67	OPT	-6.713271e-01 -6.715667e-01	17.6 70	OPT	-6.713343e-01 -6.715645e-01	5.7e-07 22	OPT	-6.715135e-01 -6.715158e-01	9.8e-07 21	OPT
arch4	-9.722463e-01 -9.727251e-01	9.5e-07 108	OPT	-9.724341e-01 -9.726737e-01	13.1 127	OPT	-9.723638e-01 -9.726949e-01	7.0 25	OPT	-9.726271e-01 -9.726275e-01	1.1e-07 21	OPT
arch8	-7.056751e+00 -7.057039e+00	9.5e-07 21.3	OPT	-3.128868e+02 -2.930615e+01	21.8 200	ITL	-7.056916e+00 -7.057004e+00	3.4e-07 31	OPT	-7.056982e+00 -7.056982e+00	3.8e-07 21	OPT
control1	-1.778462e+01 -1.778463e+01	9.5e-07 72	OPT	-1.778462e+01 -1.778463e+01	32.8 173	OPT	-1.778462e+01 -1.778463e+01	7.9e-07 26	OPT	-1.778462e+01 -1.778464e+01	5.0e-07 15	OPT
control2	-8.209980e+00 -8.300008e+00	9.5e-07 53	OPT	-8.209980e+00 -8.300008e+00	9.6e-07 128	OPT	-8.209984e+00 -8.300007e+00	7.7e-07 25	OPT	-8.209997e+00 -8.300001e+00	2.7e-07 18	OPT
control3	-1.363323e+01 -1.363328e+01	9.5e-07 70	OPT	-1.363323e+01 -1.363328e+01	9.6e-07 150	OPT	-1.363323e+01 -1.363327e+01	3.4e-07 28	OPT	-1.363326e+01 -1.363327e+01	4.5e-07 18	OPT
control4	-1.979407e+01 -1.979430e+01	3.7e-06 61	NUM	-1.979419e+01 -1.979425e+01	9.6e-07 129	OPT	-1.979421e+01 -1.979424e+01	4.7e-07 30	OPT	-1.979423e+01 -1.979424e+01	3.4e-07 17	OPT
control5	-1.688354e+01 -1.688362e+01	9.5e-07 55	OPT	-1.688354e+01 -1.688362e+01	9.6e-07 111	OPT	-1.688357e+01 -1.688361e+01	4.1e-07 29	OPT	-1.688359e+01 -1.688360e+01	4.7e-07 21	OPT
control6	-3.730386e+01 -3.730455e+01	7.5e-06 64	NUM	-3.730482e+01 -3.730443e+01	4.8e-07 121	OPT	-3.730554e+01 -3.730443e+01	1.0e-07 91	OPT	-3.730437e+01 -3.730443e+01	8.8e-07 19	OPT
control7	-2.062500e+01 -2.062510e+01	9.5e-07 20.8	OPT	-2.062500e+01 -2.062510e+01	9.6e-07 60.3	OPT	-2.062502e+01 -2.062509e+01	7.4e-07 18.4	OPT	-2.062506e+01 -2.062508e+01	4.4e-07 20	OPT
control8	-2.028599e+01 -2.028644e+01	3.8e-06 65	NUM	-2.028627e+01 -2.028659e+01	9.6e-07 148	OPT	-2.028632e+01 -2.028657e+01	4.2e-07 30.2	OPT	-2.028634e+01 -2.028657e+01	7.7e-07 20	OPT
control9	-1.467552e+01 -1.467545e+01	9.5e-07 77	OPT	-1.467552e+01 -1.467545e+01	101.9 123	OPT	-1.467537e+01 -1.467544e+01	5.1e-07 32	OPT	-1.467541e+01 -1.467543e+01	6.4e-07 20	OPT
control10	-3.853203e+01 -3.853318e+01	7.5e-06 90	NUM	-3.853462e+01 -3.853308e+01	130.3 97	OPT	-3.853292e+01 -3.853306e+01	8.3e-07 35	OPT	-3.853282e+01 -3.853307e+01	3.1e-06 25	NUM
control11	-3.195646e+01 -3.195899e+01	1.5e-05 68	NUM	-3.199843e+01 -3.195870e+01	148.0 255.1	OPT	-3.202913e+01 -3.195870e+01	79.2 38	OPT	-3.195861e+01 -3.195860e+01	1.2e-06 24	NUM
equalG11	-6.288533e+02 -6.291584e+02	4.8e-07 30	OPT	-6.285503e+02 -6.291614e+02	9.5e-07 23	OPT	-6.288236e+02 -6.291601e+02	5.3e-07 10	OPT	-6.291551e+02 -6.291553e+02	1.7e-07 15	OPT

Table 4: Summary of performance on SDPLB problems with different implementations. Primal Obj - primal objective, Dual Obj - dual objective, Opt Tol - norm of the error in termination criterion, # iter - number of iterations, Time (s) - computational time in seconds, Status - termination status: OPT - solved to tolerance, NUM - iteration limit reached, ITL - iteration limit reached, NUM - numerical issues encountered.

	$LDL^T$			HKM			HKMPC			SDPT3		
	Primal Obj Dual Obj	Opt Tol # Iter Time (s)	Status	Primal Obj Dual Obj	Opt Tol # Iter Time (s)	Status	Primal Obj Dual Obj	Opt Tol # Iter Time (s)	Status	Primal Obj Dual Obj	Opt Tol # Iter Time (s)	Status
		395.1			213.8			111.6			12.4	
	+4.494356e+01	9.4e-07		+4.4949699e+01	1.1e-04		+4.494382e+01	1.5e-06		+4.494357e+01	2.8e-07	
gpp100	+4.494344e+01	29	OPT	+4.494279e+01	19	NUM	+4.494354e+01	19	NUM	+4.494355e+01	13	OPT
		0.9			0.7			0.9			0.7	
	+7.342920e+00	9.5e-07	OPT	+7.345445e+00	2.2e-05	NUM	+7.343392e+00	2.2e-06	NUM	+7.343076e+00	6.5e-08	OPT
gpp124-1	+7.342920e+00	36	OPT	+7.343023e+00	28	NUM	+7.343069e+00	19	NUM	+7.343076e+00	15	OPT
		2.4			2.1			1.9			1.1	
	+4.686209e+01	9.5e-07	OPT	+4.688048e+01	5.6e-05	NUM	+4.686274e+01	2.2e-06	NUM	+4.686230e+01	4.9e-08	OPT
gpp124-2	+4.686205e+01	38	OPT	+4.686196e+01	23	NUM	+4.686229e+01	17	NUM	+4.686229e+01	14	OPT
		2.6			1.7			1.8			1.0	
	+1.530144e+02	9.5e-07	OPT	+1.530186e+02	9.1e-06	NUM	+1.530150e+02	1.4e-06	NUM	+1.530143e+02	7.1e-07	OPT
gpp124-3	+1.530140e+02	33	OPT	+1.530139e+02	32	NUM	+1.530141e+02	18	NUM	+1.530141e+02	12	OPT
		2.2			2.5			1.9			0.9	
	+4.189870e+02	9.5e-07	OPT	+4.189980e+02	1.6e-05	NUM	+4.189875e+02	3.8e-06	NUM	+4.189877e+02	1.2e-07	OPT
gpp124-4	+4.189869e+02	29	OPT	+4.189873e+02	36	NUM	+4.189875e+02	17	NUM	+4.189876e+02	13	OPT
		2.0			2.6			1.8			1.0	
	+1.544511e+01	9.6e-07	OPT	+1.564240e+01	4.6e-04	NUM	+1.544579e+01	2.0e-06	NUM	+1.544494e+01	7.1e-07	OPT
gpp250-1	+1.544481e+01	17.0	OPT	+1.544273e+01	27	NUM	+1.544490e+01	17	NUM	+1.544492e+01	14	OPT
		13.6			9.3			7.7			1.5	
	+8.186909e+01	8.8e-07	OPT	+8.187986e+01	1.8e-05	NUM	+8.186907e+01	4.4e-07	OPT	+8.186900e+01	2.9e-07	OPT
gpp250-2	+8.186873e+01	35	OPT	+8.186880e+01	37	NUM	+8.186896e+01	21	NUM	+8.186896e+01	13	OPT
		13.6			12.6			9.6			1.6	
	+3.035390e+02	9.5e-07	OPT	+3.035439e+02	8.1e-06	NUM	+3.035396e+02	1.4e-06	NUM	+3.035395e+02	2.8e-07	OPT
gpp250-3	+3.035386e+02	40	OPT	+3.035392e+02	58	NUM	+3.035393e+02	18	NUM	+3.035393e+02	13	OPT
		16.4			19.4			7.8			1.4	
	+7.473275e+02	9.5e-07	OPT	+7.485761e+02	6.7e-04	NUM	+7.473359e+02	4.2e-06	NUM	+7.473285e+02	1.1e-07	OPT
gpp250-4	+7.473271e+02	39	OPT	+7.473137e+02	21	NUM	+7.473231e+02	21	NUM	+7.473283e+02	13	OPT
		15.1			7.3			9.4			1.4	
	+2.532029e+01	9.5e-07	OPT	+2.635756e+01	9.3e-04	NUM	+2.532664e+01	2.8e-06	NUM	+2.532055e+01	8.3e-08	OPT
gpp500-1	+2.532003e+01	75	OPT	+2.531337e+01	28	NUM	+2.532054e+01	17	NUM	+2.532054e+01	5.1	OPT
		210.2			61.8			46.0			1.7	
	+1.560610e+02	9.9e-07	OPT	+1.561746e+02	8.2e-05	NUM	+1.560634e+02	2.1e-06	NUM	+1.560605e+02	3.1e-07	OPT
gpp500-2	+1.560600e+02	37	OPT	+1.560597e+02	31	NUM	+1.560604e+02	6.1e-06	NUM	+1.560604e+02	4.0	OPT
		104.8			68.0			51.5			4.0	
	+5.130194e+02	9.4e-07	OPT	+5.130797e+02	2.9e-05	NUM	+5.130177e+02	2.2e-07	OPT	+5.130185e+02	9.2e-07	OPT
gpp500-3	+5.130174e+02	31	OPT	+5.130169e+02	29	NUM	+5.130176e+02	19	NUM	+5.130176e+02	13	OPT
		86.5			65.7			51.0			4.3	
	+1.567022e+03	9.5e-07	OPT	+1.570289e+03	7.8e-04	NUM	+1.567023e+03	9.4e-07	OPT	+1.567020e+03	5.3e-07	OPT
gpp500-4	+1.567019e+03	30	OPT	+1.566975e+03	19	NUM	+1.567019e+03	14	NUM	+1.567019e+03	14	OPT
		85.6			41.1			38.0			4.5	
hin1	-2.032597e+00	1.9e-06	NUM	-2.032612e+00	9.5e-07	OPT	-2.032656e+00	4.7e-07	OPT	-2.032750e+00	4.6e-06	NUM
		200			23			21			22	
		0.8			0.2			0.2			0.5	
hin2	-1.096930e+01	9.4e-07	OPT	-1.096710e+01	9.5e-07	OPT	-1.096738e+01	6.5e-07	OPT	-1.096925e+01	3.2e-06	NUM
		139			30			20			15	
		0.7			0.2			0.2			0.5	
hin3	-5.694182e+01	1.9e-06	NUM	-5.694117e+01	9.5e-07	OPT	-5.694735e+01	6.1e-07	OPT	-5.696789e+01	7.5e-06	NUM



Table 4: Summary of performance on SDPLIB problems with different implementations. Primal Obj - primal objective, Dual Obj - dual objective, Opt Tol - norm of the error in termination criterion, # iter - number of iterations, Time (s) - computational time in seconds, Status - termination status: OPT - solved to tolerance, ITL - iteration limit reached, NUM - numerical issues encountered.

	LDL <sup>T</sup>			HKM			HKMPC			SDPT3		
	Primal Obj Dual Obj	# Iter Time (s)	Status	Primal Obj Dual Obj	# Iter Time (s)	Status	Primal Obj Dual Obj	# Iter Time (s)	Status	Primal Obj Dual Obj	# Iter Time (s)	Status
	-5.694125e+01	151		-5.694105e+01	31		-5.694408e+01	20		-5.695433e+01	19	
hin4	-2.747637e+02 -2.747640e+02	9.5e-07 33	OPT	-2.747637e+02 -2.747640e+02	9.5e-07 24	OPT	-2.747650e+02 -2.747644e+02	8.0e-08 18	OPT	-2.747658e+02 -2.747649e+02	2.0e-07 17	OPT
hin5	-3.622435e+02 -3.622430e+02	6.0e-05 42	NUM	-3.622435e+02 -3.622419e+02	9.3e-07 46	OPT	-3.623214e+02 -3.622675e+02	1.4e-06 35	NUM	-3.628875e+02 -3.625511e+02	1.3e-04 19	NUM
hin6	-4.489277e+02 -4.489278e+02	9.5e-07 43	OPT	-4.489327e+02 -4.489303e+02	9.5e-07 77	OPT	-4.489391e+02 -4.489334e+02	2.8e-07 25	OPT	-4.489376e+02 -4.489327e+02	2.0e-06 24	NUM
hin7	-3.908188e+02 -3.908143e+02	7.5e-06 62	NUM	-3.908140e+02 -3.908136e+02	9.6e-07 85	OPT	-3.908137e+02 -3.908131e+02	3.3e-07 20	OPT	-3.908264e+02 -3.908199e+02	3.0e-06 18	OPT
hin8	-1.161501e+02 -1.161476e+02	9.5e-07 33	OPT	-1.161501e+02 -1.161504e+02	9.5e-07 30	OPT	-1.161780e+02 -1.161582e+02	1.2e-06 25	NUM	-1.161897e+02 -1.161678e+02	2.4e-05 21	NUM
hin9	-2.362489e+02 -2.362455e+02	9.5e-07 34	OPT	-2.362489e+02 -2.362495e+02	9.5e-07 96	OPT	-2.362490e+02 -2.362495e+02	7.2e-07 22	OPT	-2.362492e+02 -2.362493e+02	8.2e-07 19	OPT
hin10	-1.087600e+02 -1.087631e+02	8.3e-07 30	OPT	-1.088148e+02 -1.087635e+02	9.5e-07 29	OPT	-1.088759e+02 -1.087939e+02	4.4e-07 24	OPT	-1.088827e+02 -1.088291e+02	5.9e-04 31	NUM
hin11	-6.588234e+01 -6.587447e+01	7.6e-06 36	NUM	-6.590343e+01 -6.588304e+01	9.5e-07 24	OPT	-6.593452e+01 -6.589832e+01	7.6e-08 24	OPT	-6.592015e+01 -6.591309e+01	4.5e-04 23	NUM
hin12	-3.193765e-04 -2.821849e-04	2.2e-06 48	NUM	-1.327426e-04 -2.950287e-04	5.0e-06 200	NUM	-2.249339e-04 -2.551432e-04	8.5e-07 52	OPT	-7.653965e-05 -5.392390e-05	6.3e-05 19	NUM
hin13	-1.196089e+01 -1.227702e+01	9.3e-03 200	ITL	-4.529842e+01 -4.481949e+01	7.6e-06 200	NUM	-4.544674e+01 -4.489303e+01	8.4e-06 31	NUM	-4.435726e+01 -4.435083e+01	1.7e-04 32	NUM
hin14	-1.298975e+01 -1.299083e+01	6.0e-05 200	ITL	-1.299173e+01 -1.299080e+01	9.5e-07 30	OPT	-1.299296e+01 -1.299142e+01	2.9e-07 24	OPT	-1.299022e+01 -1.299064e+01	7.5e-05 27	NUM
hin15	-2.402307e+01 -2.431450e+01	1.1e-02 200	ITL	-2.592174e+01 -2.432157e+01	3.1e-05 200	ITL	-2.533888e+01 -2.463701e+01	7.3e-06 37	NUM	-2.401302e+01 -2.399673e+01	3.7e-03 37	NUM
maxG11	-6.291644e+02 -6.291648e+02	5.4e-07 26	OPT	-6.291640e+02 -6.291648e+02	9.5e-07 189.9	OPT	-6.291646e+02 -6.291648e+02	1.8e-07 11	OPT	-6.291645e+02 -6.291648e+02	2.0e-07 13	OPT
mcp100	-2.261571e+02 -2.261574e+02	9.3e-07 25	OPT	-2.261571e+02 -2.261574e+02	9.5e-07 21	OPT	-2.261573e+02 -2.261574e+02	1.1e-07 9	OPT	-2.261571e+02 -2.261574e+02	6.7e-07 10	OPT
mcp124-1	-1.419903e+02 -1.419905e+02	9.5e-07 26	OPT	-1.419903e+02 -1.419905e+02	9.5e-07 22	OPT	-1.419905e+02 -1.419905e+02	6.4e-08 10	OPT	-1.419905e+02 -1.419905e+02	7.1e-08 11	OPT
		1.8			1.8			1.2			0.7	

Table 4: Summary of performance on SDPLB problems with different implementations. Primal Obj - primal objective, Dual Obj - dual objective, Opt Tol - norm of the error in termination criterion, # iter - number of iterations, Time (s) - computational time in seconds, Status - termination status: OPT - solved to tolerance, ITL - iteration limit reached, NUM - numerical issues encountered.

	<i>LDL<sup>T</sup></i>			HKM			HKMPC			SDPT3		
	Primal Obj Dual Obj	Opt Tol # Iter Time (s)	Status	Primal Obj Dual Obj	Opt Tol # Iter Time (s)	Status	Primal Obj Dual Obj	Opt Tol # Iter Time (s)	Status	Primal Obj Dual Obj	Opt Tol # Iter Time (s)	Status
mcp124-2	-2.698800e+02 -2.698802e+02	5.4e-07 2.8 1.8	OPT	-2.698799e+02 -2.698802e+02	9.5e-07 22 1.7	OPT	-2.698801e+02 -2.698802e+02	2.9e-07 9 1.0	OPT	-2.698801e+02 -2.698802e+02	1.3e-07 11 0.8	OPT
mcp124-3	-4.677496e+02 -4.677501e+02	9.5e-07 25 1.8	OPT	-4.677496e+02 -4.677501e+02	9.5e-07 22 1.8	OPT	-4.677501e+02 -4.677501e+02	5.1e-08 10 1.1	OPT	-4.677499e+02 -4.677501e+02	2.3e-07 11 0.7	OPT
mcp124-4	-8.644110e+02 -8.644119e+02	9.5e-07 24 1.7	OPT	-8.644110e+02 -8.644119e+02	9.5e-07 22 1.9	OPT	-8.644116e+02 -8.644119e+02	2.7e-07 9 0.9	OPT	-8.644105e+02 -8.644119e+02	7.9e-07 10 0.7	OPT
mcp250-1	-3.172640e+02 -3.172644e+02	9.4e-07 28 10.7	OPT	-3.172640e+02 -3.172644e+02	9.5e-07 8.2 2.5	OPT	-3.172641e+02 -3.172644e+02	6.5e-07 10 4.6	OPT	-3.172642e+02 -3.172644e+02	2.4e-07 12 1.0	OPT
mcp250-2	-5.319294e+02 -5.319301e+02	9.5e-07 26 10.2	OPT	-5.319294e+02 -5.319301e+02	9.5e-07 7.9 2.3	OPT	-5.319300e+02 -5.319301e+02	5.6e-08 10 4.8	OPT	-5.319296e+02 -5.319301e+02	4.4e-07 11 1.0	OPT
mcp250-3	-9.811714e+02 -9.811726e+02	9.4e-07 25 9.9	OPT	-9.811714e+02 -9.811726e+02	9.5e-07 23 7.9	OPT	-9.811716e+02 -9.811726e+02	8.1e-07 9 4.1	OPT	-9.811718e+02 -9.811726e+02	4.1e-07 11 1.1	OPT
mcp250-4	-1.681958e+03 -1.681960e+03	9.5e-07 25 9.8	OPT	-1.681958e+03 -1.681960e+03	9.5e-07 23 7.6	OPT	-1.681960e+03 -1.681960e+03	7.5e-08 10 4.4	OPT	-1.681959e+03 -1.681960e+03	3.5e-07 11 1.0	OPT
mcp500-1	-5.981480e+02 -5.981486e+02	5.6e-07 43 119.4	OPT	-5.981480e+02 -5.981486e+02	9.5e-07 29 62.4	OPT	-5.981481e+02 -5.981486e+02	3.9e-07 11 29.6	OPT	-5.981483e+02 -5.981486e+02	2.2e-07 13 2.7	OPT
mcp500-2	-1.070056e+03 -1.070057e+03	5.0e-07 30 82.5	OPT	-1.070055e+03 -1.070057e+03	9.5e-07 26 57.6	OPT	-1.070056e+03 -1.070057e+03	2.1e-07 11 31.1	OPT	-1.070057e+03 -1.070057e+03	1.2e-07 13 3.2	OPT
mcp500-3	-1.847968e+03 -1.847970e+03	9.4e-07 25 70.8	OPT	-1.847968e+03 -1.847970e+03	9.5e-07 21 50.4	OPT	-1.847969e+03 -1.847970e+03	4.4e-07 10 28.2	OPT	-1.847969e+03 -1.847970e+03	1.9e-07 12 3.4	OPT
mcp500-4	-3.566734e+03 -3.566738e+03	9.5e-07 25 69.4	OPT	-3.566734e+03 -3.566738e+03	9.5e-07 24 54.4	OPT	-3.566736e+03 -3.566738e+03	5.9e-07 20 28.0	OPT	-3.566736e+03 -3.566738e+03	2.8e-07 12 3.1	OPT
qap5	+4.360010e+02 +4.359999e+02	9.0e-07 29 0.5	OPT	+4.360010e+02 +4.359999e+02	9.5e-07 25 0.5	OPT	+4.360002e+02 +4.360000e+02	1.9e-07 11 0.3	OPT	+4.360000e+02 +4.360000e+02	6.2e-08 9 0.6	OPT
qap6	+3.814470e+02 +3.814240e+02	2.2e-05 108 3.9	NUM	+3.814388e+02 +3.814372e+02	9.5e-07 26 1.0	OPT	+3.814059e+02 +3.814221e+02	1.9e-07 14 0.7	OPT	+3.813928e+02 +3.814158e+02	6.9e-07 12 1.3	OPT
qap7	+4.248389e+02 +4.248018e+02	1.5e-05 200 16.5	ITL	+4.248209e+02 +4.248183e+02	9.5e-07 26 2.2	OPT	+4.248068e+02 +4.248127e+02	2.6e-07 14 1.8	OPT	+4.247879e+02 +4.248040e+02	3.3e-07 13 2.4	OPT
qap8	+7.571217e+02 +7.568743e+02	6.7e-05 134 23.7	NUM	+7.569569e+02 +7.569535e+02	9.5e-07 27 4.8	OPT	+7.568964e+02 +7.569256e+02	1.3e-07 15 3.9	OPT	+7.568396e+02 +7.568978e+02	6.2e-07 12 1.7	OPT
qap9	+1.435344e+03 +1.395478e+03	6.6e-03 148	NUM	+1.409944e+03 +1.409939e+03	9.5e-07 29	OPT	+1.410908e+03 +1.409651e+03	2.2e-04 11	NUM	+1.409919e+03 +1.409930e+03	1.6e-07 13	OPT

Table 4: Summary of performance on SDPLB problems with different implementations. Primal Obj - primal objective, Dual Obj - dual objective, Opt Tol - norm of the error in termination criterion, # iter - number of iterations, Time (s) - computational time in seconds, Status - termination status: OPT - solved to tolerance, ITL - iteration limit reached, NUM - numerical issues encountered.

	LDL <sup>T</sup>			HKM			HKMPC			SDPT3		
	Primal Obj Dual Obj	Opt Tol # Iter Time (s)	Status	Primal Obj Dual Obj	Opt Tol # Iter Time (s)	Status	Primal Obj Dual Obj	Opt Tol # Iter Time (s)	Status	Primal Obj Dual Obj	Opt Tol # Iter Time (s)	Status
qap10	+1.092738e+03 +1.092546e+03	4.1e-05 88	NUM	+1.092610e+03 +1.092604e+03	9.5e-07 28	OPT	+1.092553e+03 +1.092577e+03	8.4e-07 14	OPT	+1.092540e+03 +1.092574e+03	2.3e-07 12	OPT
ss80	-2.023913e+01 -2.023953e+01	81.5 182	OPT	-2.023913e+01 -2.023953e+01	24.6 113	OPT	-2.023946e+01 -2.023951e+01	19.2 22	OPT	-2.023948e+01 -2.023951e+01	5.0 17	OPT
theta1	-2.299996e+01 -2.300001e+01	59.4 31	OPT	-2.299996e+01 -2.300001e+01	37.3 32	OPT	-2.299996e+01 -2.300000e+01	13.3 14	OPT	-2.300000e+01 -2.300000e+01	5.7 10	OPT
theta2	-3.287909e+01 -3.287918e+01	9.8e-07 31	OPT	-3.287909e+01 -3.287918e+01	0.6 33	OPT	-3.287914e+01 -3.287917e+01	0.4 6.3	OPT	-3.287916e+01 -3.287917e+01	2.4e-07 11	OPT
theta3	-4.216686e+01 -4.216700e+01	10.1 31	OPT	-4.216686e+01 -4.216700e+01	9.9 33	OPT	-4.216691e+01 -4.216699e+01	13 34.8	OPT	-4.216699e+01 -4.216699e+01	4.3e-07 5.1	OPT
theta4	-5.032106e+01 -5.032125e+01	9.6e-07 32	OPT	-5.032106e+01 -5.032125e+01	61.2 35	OPT	-5.032112e+01 -5.032124e+01	55.1 119.5	OPT	-5.032117e+01 -5.032123e+01	5.1e-07 11	OPT
theta5	-5.723211e+01 -5.723235e+01	208.3 32	OPT	-5.723211e+01 -5.723235e+01	604.0 36	OPT	-5.723224e+01 -5.723232e+01	333.7 855.9	OPT	-5.723227e+01 -5.723231e+01	3.3e-07 11	OPT
theta6	-6.347685e+01 -6.347713e+01	9.6e-07 32	OPT	-6.347685e+01 -6.347713e+01	522.4 36	OPT	-6.347689e+01 -6.347713e+01	8.0e-07 1346.8	OPT	-6.347702e+01 -6.347710e+01	6.2e-07 11	OPT
truss1	+9.000006e+00 +8.999993e+00	7.2e-07 26	OPT	+9.000006e+00 +8.999993e+00	1346.8 26	OPT	+9.000006e+00 +8.999993e+00	9.9e-07 26	OPT	+8.999997e+00 +8.999996e+00	187.4 8	OPT
truss2	+1.233804e+02 +1.233803e+02	5.5e-07 38	OPT	+1.233805e+02 +1.233803e+02	0.2 29	OPT	+1.233804e+02 +1.233803e+02	0.2 17	OPT	+1.233805e+02 +1.233803e+02	0.3 10	OPT
truss3	+9.10009e+00 +9.109980e+00	1.0e-06 26	OPT	+9.100010e+00 +9.109980e+00	1.5 23	OPT	+9.109998e+00 +9.109994e+00	1.2e-07 0.2	OPT	+9.109999e+00 +9.109993e+00	3.6e-07 0.3	OPT
truss4	+9.010006e+00 +9.009988e+00	9.5e-07 26	OPT	+9.010006e+00 +9.009988e+00	0.3 26	OPT	+9.009999e+00 +9.009994e+00	0.2 10	OPT	+9.009998e+00 +9.009992e+00	3.6e-07 8	OPT
truss5	+1.326357e+02 +1.326356e+02	5.2e-07 38	OPT	+1.326358e+02 +1.326355e+02	0.2 27	OPT	+1.326357e+02 +1.326356e+02	0.2 4.6	OPT	+1.326357e+02 +1.326357e+02	3.7e-08 14	OPT
truss6	+1.812587e+02 +1.611614e+03	1.2e+01 200	ITL	+9.010015e+02 +9.010013e+02	6.3 200	ITL	+9.010015e+02 +9.010014e+02	5.0e-07 47	OPT	+9.010020e+02 +9.010007e+02	7.4e-07 20	OPT
truss7	+1.255106e+02 +2.425458e+03	1.4e+01 200	ITL	+9.000015e+02 +9.000014e+02	26.6 200	ITL	+9.000015e+02 +9.000014e+02	7.7 52	OPT	+9.000022e+02 +9.000013e+02	5.2e-07 18	OPT
truss8	+1.331146e+02	5.4e-07	OPT	+1.331147e+02	18.0	OPT	+1.331146e+02	5.6	OPT	+1.331146e+02	5.9	OPT
		5.4e-07	OPT		9.5e-07	OPT		8.1e-08	OPT		5.9e-07	OPT



Table 4: Summary of performance on SDPLIB problems with different implementations. Primal Obj - primal objective, Dual Obj - dual objective, Opt Tol - norm of the error in termination criterion, # iter - number of iterations, Time (s) - computational time in seconds, Status - termination status: OPT - solved to tolerance, ITL - iteration limit reached, NUM - numerical issues encountered.

$LDL^T$			HKM			HKMPC			SDPT3		
Primal Obj	Opt Tol	Status	Primal Obj	Opt Tol	Status	Primal Obj	Opt Tol	Status	Primal Obj	Opt Tol	Status
Dual Obj	# Iter		Dual Obj	# Iter		Dual Obj	# Iter		Dual Obj	# Iter	
	Time (s)			Time (s)			Time (s)			Time (s)	
+1.331143e+02	40		+1.331141e+02	30		+1.331145e+02	21		+1.331145e+02	14	
	25.5			18.1			15.7			1.8	

**Acknowledgement.** We thank Ajit Gopalakrishnan for his efforts in the initial development of the ideas. We are grateful to the anonymous referees for their careful reading of the manuscript and suggestions that have significantly improved the presentation.

REFERENCES

- [1] F. ALIZADEH, *Combinatorial optimization with interior point methods and semidefinite matrices*, PhD thesis, University of Minnesota, Minneapolis, MN, 1991.
- [2] F. ALIZADEH, J.-P. A. HAEBERLY, AND M. L. OVERTON, *Primal-dual interior point methods for semidefinite programming: convergence rates, stability and numerical results*, SIAM J. Optimization, 8 (1998), pp. 746–768.
- [3] F. ALIZADEH, J.-P. A. HAEBERLY, AND M. L. OVERTON, *Complementarity and nondegeneracy in semidefinite programming*, Mathematical Programming, 77 (1997), pp. 111–128.
- [4] M. F. ANJOS AND J. B. LASSERRE, eds., *Handbook of Semidefinite, Conic and Polynomial Optimization: Theory, Algorithms, Software and Applications*, vol. 166 of International Series in Operational Research and Management Science, Springer, 2012.
- [5] V. I. ARNOLD, *On matrices depending on parameters*, Russian Mathematical Surveys, 26 (1971), pp. 29–43.
- [6] H. Y. BENSON AND R. J. VANDERBEI, *Solving Problems with Semidefinite and Related Constraints Using Interior-Point Methods for Nonlinear Programming*, Mathematical Programming Ser. B, 95 (2003), pp. 279–302.
- [7] B. BORCHERS, *Sdplib 1.2, a library of semidefinite programming test problems*, Optimization Methods and Software, 11 (1999), pp. 683–690.
- [8] S. BURER, *Semidefinite programming in the space of partial positive semidefinite matrices*, SIAM J. Optimization, 14 (2003), pp. 139–172.
- [9] S. BURER, R. D. C. R. D. C. MONTEIRO, AND Y. ZHANG, *Interior-Point Algorithms for Semidefinite Programming Based on a Nonlinear Formulation*, Computational Optimization and Applications, 22 (2002), pp. 49–79.
- [10] S. BURER, R. D. C. R. D. C. MONTEIRO, AND Y. ZHANG, *Solving a class of semidefinite programs via nonlinear programming*, Mathematical Programming (Series A), 93 (2002), pp. 97–122.
- [11] J. DA CRUZ NETO, O. FERREIRA, AND R. D. C. MONTEIRO, *Asymptotic behavior of the central path for a special class of degenerate sdp problems*, Mathematical Programming, 103 (2005), pp. 487–514.
- [12] J. DAHL, L. VANDENBERGHE, AND V. ROYCHOWDHURY, *Covariance selection for non-chordal graphs via chordal embedding*, Optimization Methods & Software, 23 (2008).
- [13] E. DE KLERK, C. ROOS, AND T. TERLAKY, *Infeasible-start semidefinite programming algorithms via self dual embeddings*, Fields Institute Communications, 18 (1998), pp. 215–236.
- [14] E. D. DOLAN AND J. J. MORÉ, *Benchmarking optimization software with performance profiles*, Mathematical Programming (Series B), 91 (2002), pp. 201–213.
- [15] R. FLETCHER, *Semi-Definite Matrix Constraints in Optimization*, SIAM J. Control Optim., 23 (1985), pp. 493–513.
- [16] K. FUJISAWA, M. KOJIMA, AND K. NAKATA, *Exploiting sparsity in primal-dual interior-point method for semidefinite programming*, Mathematical Programming, 79 (1997), pp. 235–253.
- [17] M. FUKUDA, M. KOJIMA, K. MUROTA, AND K. NAKATA, *Exploiting sparsity in semidefinite programming via matrix completion i: general framework*, SIAM J. Optimization, 11 (2000), pp. 647–674.
- [18] D. GOLDFARB AND K. SCHEINBERG, *Interior point trajectories in semidefinite programming*, SIAM Journal on Optimization, 8 (1998), pp. 871–886.
- [19] G. GOLUB AND C. VAN LOAN, *Matrix Computations*, Johns Hopkins Studies in the Mathematical Sciences, Johns Hopkins University Press, 1996.
- [20] N. I. M. GOULD AND J. SCOTT, *A note on performance profiles for benchmarking software*, ACM Trans. Math. Softw., 43 (2016), pp. 15:1–15:5, <http://doi.acm.org/10.1145/2950048>.
- [21] M. HALICKA, *Analyticity of the central path at the boundary point in semidefinite programming*, European Journal of Operational Research, 143 (2002), pp. 311–324.
- [22] M. HALICKA, E. DE KLERK, AND C. C. ROOS, *Limiting behavior of the central path in semidefinite optimization*, Optimization Methods and Software, 20 (2005), pp. 99–113.
- [23] C. HELMBERG, F. RENDL, R. VANDERBEI, AND H. WOLKOWICZ, *An interior-point method for semidefinite programming*, SIAM J. Optimization, 6 (1996), pp. 342–361.

- [24] N. J. HIGHAM, *Accuracy and Stability of Numerical Algorithms*, SIAM, 1996.
- [25] M. KOJIMA, S. SHINDOH, AND S. HARA, *Interior-point methods for the monotone semidefinite linear complementarity problem in symmetric matrices*, SIAM J. Optimization, 7 (1997), pp. 86–125.
- [26] Z. LU AND R. D. C. MONTEIRO, *Error bounds and limiting behavior of weighted paths associated with the SDP map  $X^{\frac{1}{2}}SX^{\frac{1}{2}}$* , SIAM Journal on Optimization, 15 (2004), pp. 348–374.
- [27] Z. LU AND R. D. C. MONTEIRO, *Limiting behavior of the Alizadeh-Haeberly-Overton weighted paths in semidefinite programming*, Optimization Methods and Software, 22 (2007), pp. 849–870.
- [28] Z. Q. LUO, J. F. STURM, AND S. ZHANG, *Superlinear convergence of a symmetric primal-dual path following algorithm for semidefinite programming*, SIAM Journal on Optimization, 8 (1998), pp. 59–81.
- [29] C. B. MOLER, *Iterative Refinement in Floating Point*, Journal of the ACM, 14 (1967), pp. 316–321.
- [30] R. D. C. MONTEIRO, *Primal-dual path-following algorithms for semidefinite programming*, SIAM J. Optimization, 7 (1997), pp. 663–678.
- [31] R. D. C. MONTEIRO, *First- and second-order methods for semidefinite programming*, Mathematical Programming, 97 (2003), pp. 209–244.
- [32] R. D. C. MONTEIRO AND T. TSUCHIYA, *Polynomial convergence of a new family of primal-dual algorithms for semidefinite programming*, SIAM J. Optimization, 9 (1999), pp. 551–577.
- [33] R. D. C. MONTEIRO AND P. R. ZANJACOMO, *General interior-point maps and existence of weighted paths for nonlinear semidefinite complementarity problems*, Mathematics of Operations Research, 25 (2000), pp. 381–399.
- [34] K. NAKATA, K. FUJISAWA, M. FUKUDA, M. KOJIMA, AND K. MUROTA, *Exploiting sparsity in semidefinite programming via matrix completion ii: implementation and numerical results*, Mathematical Programming, Series B, 95 (2003), pp. 303–327.
- [35] Y. E. NESTEROV AND A. S. NEMIROVSKII, *Interior Point Polynomial Methods in Convex Programming: Theory and Applications*, SIAM, Philadelphia, 1994.
- [36] Y. E. NESTEROV AND M. J. TODD, *Primal-dual interior-point methods for self-scaled cones*, Mathematics of Operations Research, 22 (1997), pp. 1–42.
- [37] J. RENEGAR, *A Mathematical View of Interior-Point Methods in Convex Optimization*, MPS-SIAM Series on Optimization, Society for Industrial and Applied Mathematics, 2001.
- [38] A. SHAPIRO, *First and second order analysis of nonlinear semidefinite programs*, Mathematical Programming, 77 (1997), pp. 301–320.
- [39] A. SHAPIRO AND M. K. H. FAN, *On eigenvalue optimization*, SIAM J. Optimization, 5 (1995), pp. 552–569.
- [40] G. SRIJUNTSIRI AND S. A. VAVASIS, *A fully sparse implementation of a primal-dual interior-point potential reduction method for semidefinite programming*, Tech. Report arXiv:cs/0412009v1, arXiv, 2004, <https://arxiv.org/abs/cs/0412009>.
- [41] J. F. STURM, *Using sedumi 1.02, a matlab toolbox for optimization over symmetric cones*, Optimization Methods and Software, 11 (1999), pp. 625–653.
- [42] M. J. TODD, *A study of search directions in primal-dual interior-point methods for semidefinite programming*, Optimization Methods and Software, 11 (1999), pp. 1–46.
- [43] M. J. TODD, K. C. TOH, AND R. H. TÛTÛNCÛ, *On the Nesterov-Todd direction in semidefinite programming*, SIAM J. Optimization, 8 (1998), pp. 769–796.
- [44] K. C. TOH, *Solving large scale semidefinite programs via an iterative solver on the augmented systems*, SIAM J. Optimization, 14 (2004), pp. 670–698.
- [45] R. H. TÛTÛNCÛ, K. C. TOH, AND M. J. TODD, *Solving semidefinite-quadratic-linear programs using SDPT3*, Mathematical Programming Ser. B, 95 (2003), pp. 189–217.
- [46] H. WOLKOWICZ, R. SAIGAL, AND L. VANDENBERGHE, eds., *Handbook of Semidefinite Programming Theory, Algorithms, and Applications*, vol. 27 of International Series in Operations Research and Management Science, Springer, 2000.
- [47] M. YAMASHITA AND K. NAKATA, *Fast implementation for semidefinite programs with positive matrix completion*, Optimization Methods and Software, 30 (2015), pp. 1030–1049.
- [48] Y. ZHANG, *On extending some primal-dual interior-point algorithms from linear programming to semidefinite programming*, SIAM J. Optimization, 8 (1998), pp. 365–386.