

First- and Second-Order Methods for Semidefinite Programming

Renato. D. C. Monteiro *

December 2002 (revised May 3, 2003)

Abstract

In this paper, we survey the most recent methods that have been developed for the solution of semidefinite programs. We first concentrate on the methods that have been primarily motivated by the interior point (IP) algorithms for linear programming, putting special emphasis in the class of primal-dual path-following algorithms. We also survey methods that have been developed for solving large-scale SDP problems. These include first-order nonlinear programming (NLP) methods and more specialized path-following IP methods which use the (preconditioned) conjugate gradient or residual scheme to compute the Newton direction and the notion of matrix completion to exploit data sparsity.

Key words: Semidefinite programming, interior-point methods, polynomial complexity, path-following methods, primal-dual methods, nonlinear programming, Newton method, first-order methods, bundle method, matrix completion.

AMS subject classification 2000: 65K05, 90C06, 90C22, 90C25, 90C30, 90C51

1 Introduction

One form of describing a (linear) semidefinite programming (SDP) problem is as the problem of minimizing a linear function of a symmetric matrix variable X subject to linear equality constraints on X and the essential constraint that X be positive semidefinite (see (1)). The last constraint is nonlinear and nonsmooth, but convex, so semidefinite programs are convex optimization problems. In this paper, we survey the most recent methods that have been developed for the solution of this important class of problems. We first concentrate on the methods that have been primarily motivated by the interior point (IP) algorithms for linear programming, putting special emphasis in the class of primal-dual path-following algorithms (Sections 3 and 4). Since these methods are based on Newton steps, we refer to them as *second-order* methods. We also survey methods that have been developed for solving large-scale SDP problems (Section 5). These include first-order nonlinear programming (NLP) methods and more specialized second-order methods which use the (preconditioned) conjugate gradient or residual scheme to compute the Newton direction and the notion of matrix completion to exploit data sparsity.

*School of Industrial and Systems Engineering, Georgia Institute of Technology, Atlanta, Georgia 30332, USA (monteiro@isye.gatech.edu). The author's research presented in this survey article has been supported in part by NSF through grants INT-9600343, INT-9910084, CCR-9700448, CCR-9902010, CCR-0203113 and ONR through grants N00014-93-1-0234, N00014-94-1-0340 and N00014-03-1-0401.

Prior to the development of IP methods for SDP, there has been a somewhat scattered and slow development but relatively long history of SDP. Very early works in solution stability of linear differential equations and control theory has demonstrated the modeling power of linear matrix inequalities (LMI) (see Boyd et al. [11] for a detailed historic account on this). Early approaches to graph theory involving SDP problems have been proposed in Cullum et al. [21], Donath and Hoffman [22, 23] and Lovász [54]. Also, SDP was studied very early by the nonlinear programming community [27, 28, 80, 87, 88, 91]. Despite these early developments, SDP has become a central focus and exciting area in the field of optimization only in the last twelve years or so. A major factor behind this explosive interest in SDP was the the development of efficient algorithms for their solution and subsequently the discovery of SDP as a powerful modeling tool. Landmark works by Nesterov and Nemirovskii [72, 73, 74] develop a deep and unified theory of IP methods for solving convex programs based on the notion of self-concordant barrier functions. (See their book [75] for a comprehensive treatment of this subject.) In particular, they and, independently Alizadeh in his breakthrough work [1], showed that primal-only and/or dual-only IP methods for LP can all be extended to SDP. Other primal-only methods have also been proposed by Freund [30] and Jarre [43]

Since (small to medium sized) SDP problems can be efficiently solved by IP methods, several applications in convex constrained optimization, control theory and combinatorial optimization, which could not be easily handled before, can now be routinely solved. Many problems can be cast in the form of an semidefinite program. For example, LP, optimizing a convex quadratic form subject to convex quadratic inequality constraints, minimizing the volume of an ellipsoid that covers a given set of points and ellipsoids, maximizing the volume of an ellipsoid that is contained in a polytope, matrix norm minimization, cluster analysis using ellipsoids, and a variety of maximum and minimum eigenvalue problems can be cast as SDPs (see Vandenberghe and Boyd [103]). In addition, SDP has applications in minimum trace factor analysis [10, 27, 28, 87, 106] and optimal experiment design [84] in the area of statistics, and in engineering fields such as structural design [6, 8] and control theory [11]. Another important application of SDP is to the solution of NP-hard combinatorial optimization problems, for which semidefinite programs serve as tractable relaxations (e.g., see Shor [91], Lovász and Schrijver [55], Goemans and Williamson [34], Alizadeh [1], Kamath and Karmarkar [44]) and also as the basis for polynomial-time approximation algorithms with strong performance guarantees (e.g., [34]). In fact, the work [34] was another major factor in the increased interest on SDP, specially by the “approximation algorithms” community. Finally, a long list of SDP applications can be found in the Handbook of Semidefinite Programming [85], the books by Bental and Nemirovskii [7] and Boyd et al. [11], as well as the survey papers by Vandenberghe and Boyd [103] and Todd [95].

The first algorithms for SDP which are extensions of the well-known primal-dual LP algorithms, such as the long-step path-following algorithm of Kojima, Mizuno and Yoshise [47] and Tanabe [92, 93], the short-step path-following algorithm of Kojima, Mizuno and Yoshise [46] and Monteiro and Adler [60, 61], and the predictor-corrector algorithm of Mizuno, Todd and Ye [57], were only introduced approximately five years after the works of Alizadeh [1] and Nesterov and Nemirovskii [72, 73, 74] on primal-only IP algorithms. In contrast to primal-dual methods for LP, there are many ways one can compute the Newton search directions used in primal-dual algorithms for SDP. For this reason, the polynomial (and superlinear) convergence theory of primal-dual methods for SDP is substantially more difficult than for LP. Several Newton-type SDP directions have been proposed which generalize the primal-dual Newton direction for LP. The first three and most popular of them are: i) the Alizadeh, Haeberly and Overton (AHO) direction proposed in [2], ii) the Nesterov and Todd (NT) direction introduced in [76, 77], iii) a direction independently proposed by Helmberg, Rendl, Vanderbei and Wolkowicz [41] and Kojima, Shindoh and Hara [51], and later rediscovered by Monteiro [58], which we refer to as the HRVW/KSH/M direction, iv) a direction which is the symmetric dual of the HRVW/KSH/M direction, which we simply refer to as the dual HRVW/KSH/M direction.

The first theoretical foundations for primal-dual SDP methods were given in [51, 77], which deal exclusively with short-step algorithms and potential-reduction algorithms. They left open the question of whether a polynomially convergent long-step path-following method existed for SDP, which was affirmatively settled by Monteiro [58] for feasible algorithms and subsequently by Zhang [107] for infeasible algorithms. Using a scaling and symmetrization scheme, these two works also introduced a family of search directions, usually referred to as the Monteiro and Zhang (MZ) family, which contains all four directions mentioned above. Convergence analyses for primal-dual path-following algorithms based on the whole MZ family (or a large subset of it) have been given by Monteiro and Zhang [68] for long-step methods, and Monteiro [59] for short-step methods.

Prior to the introduction of the MZ family, Kojima et al. [51] had already introduced a family of directions, commonly referred to as the KSH family, which contained all the directions mentioned above except for the AHO direction. Polynomial convergence of short-step primal-dual path-following algorithms is established for the two HRVW/KSH/M directions in [51] and for the whole KSH family in Monteiro and Tsuchiya [64]. Kojima et al. [51] also establishes the polynomiality of a potential-reduction algorithm based on the whole KSH family. Other family of primal-dual search directions have also been introduced by Monteiro and Tsuchiya [63], Tseng [100] and Burer and Monteiro [16]. As for the KSH family, both the Monteiro and Tsuchiya (MT) and the Tseng families contain the HRVW/KSH/M directions and the NT direction but not the AHO direction. Also, a number of authors have proposed various primal-dual directions for SDP (see for example Gu [36], Kruk et al. [52], Potra [89], Monteiro and Zanjácomo [67], Todd [94]) and Toh [97]. Unified convergence analysis of short-step and long-step path-following SDP methods are given in Kojima, Shida and Shindoh [50] and Burer and Monteiro [14], respectively. The asymptotic convergence rates of primal-dual path-following algorithms have been studied by several authors (see for example Luo et al. [56], Potra and Sheng [82, 83] and Kojima et al. [49]). We provide a survey of second-order methods, with special emphasis on primal-dual path-following feasible IP methods, in Sections 3 and 4. Many of the approaches discussed in this survey paper in the context of SDP have been generalized to other cone programming problems (see for example Alizadeh and Schmieta [3], Faybusovich [25, 26, 24], Monteiro and Tsuchiya [65], Nesterov and Todd [76, 77], Schmieta and Alizadeh [86], Tsuchiya [101, 102]).

SDPs that arise as relaxations of combinatorial optimization problems are typically large-scale, and it has become evident in the past few years that current second-order IP methods, in its most naive form, are largely unsuitable in practice when applied to such SDPs due to an inherent high demand for computer time and storage. In attempting to address such issues for large-scale SDPs, several researchers have developed less computationally intensive first-order NLP methods and implementations that have made significant progress in this area (see Burer and Monteiro [16, 15], Burer, Monteiro and Zhang [19, 17, 18] and Helmberg and Rendl [40]). Even though these types of methods can find low accurate near optimal solutions relatively fast, they are not recommended for problems in which high accuracy is a necessity. Also, as opposed to the primal-dual IP methods mentioned above, these methods are primal- or dual-only methods and the design of a universal set of termination conditions that works well for all classes of SDP problems is not obvious, and might not be feasible. Other promising approaches for solving large-scale problems are second-order methods that use iterative methods to compute their corresponding search direction and/or efficiently exploit data sparsity in the primal and/or dual spaces (see for example Burer [12], Choi and Ye [20], Fukuda et al. [33], Nakata et al. [69], Toh and Kojima [99] and Toh [98]). We survey the methods mentioned on this paragraph in Section 5.

The organization of this chapter is as follows. In Section 2, we introduce the pair of dual SDP problems which will be the subject of our study and discuss some of the duality results that hold for it. We also describe the associated central path which plays an important role on the algorithms presented in the subsequent sections. Section 3 describes the main Newton-type directions that have

been proposed in the context of primal-dual interior-point algorithms for SDP. Section 4 describes the convergence properties of primal-dual path-following (feasible only) IP algorithms based on three families of search directions, namely: the MZ, MT and KSH families. Section 5 describes several methods which are particularly suitable for the solution of large-scale SDP problems. Finally, Section 6 discusses the computational complexity of a single iteration of some of the methods presented in this paper.

1.1 Notation

In this subsection, we introduce some notation that will be used throughout our presentation.

\mathcal{S}^n denotes the space of $n \times n$ symmetric matrices, and $X \succeq 0$ indicates that X is positive semidefinite. We also write \mathcal{S}_+^n for $\{X \in \mathcal{S}^n : X \succeq 0\}$, and \mathcal{S}_{++}^n for its interior, the set of positive definite matrices in \mathcal{S}^n . For $X \in \mathcal{S}_+^n$, $X^{1/2}$ denotes its positive semidefinite square root. If $X \in \mathcal{S}_{++}^n$, we write $X^{-1/2}$ for the inverse of $X^{1/2}$, or equivalently the positive semidefinite square root of X^{-1} . We use $A \bullet B$ to denote the inner product $\text{trace } A^T B = \sum_{i,j} a_{ij} b_{ij}$ of two $m \times n$ matrices $A = (a_{ij})$ and $B = (b_{ij})$. We write \mathcal{S}_\perp^n for the space of $n \times n$ skew-symmetric matrices; this is the orthogonal complement, with respect to the inner product above, of \mathcal{S}^n in $\mathfrak{R}^{n \times n}$.

Given a linear operator $\mathcal{F} : E \rightarrow F$ between two finite dimensional inner product spaces $(E, \langle \cdot, \cdot \rangle_E)$ and $(F, \langle \cdot, \cdot \rangle_F)$, its *adjoint* is the unique operator $\mathcal{F}^* : F \rightarrow E$ satisfying $\langle \mathcal{F}(u), v \rangle_F = \langle u, \mathcal{F}^*(v) \rangle_E$ for all $u \in E$ and $v \in F$. A linear operator $\mathcal{G} : E \rightarrow E$ is called *self-adjoint* if $\mathcal{G} = \mathcal{G}^*$. Moreover, \mathcal{G} is said to be *positive semidefinite* (res. *positive definite*) if $\langle \mathcal{G}(u), u \rangle_E \geq 0$ (resp., $\langle \mathcal{G}(u), u \rangle_E > 0$) for all $u \in E$.

2 The SDP problem, duality and the central path

In this section, we introduce the pair of dual SDP problems which will be the subject of our study and discuss some of the duality results that hold for it. We also describe the associated central path which plays an important role in the algorithms presented in Sections 4 and 5.

We consider the SDP given in the following standard form:

$$(P) \quad \begin{aligned} \min_X \quad & C \bullet X \\ & A_i \bullet X = b_i, \quad i = 1, \dots, m, \\ & X \succeq 0, \end{aligned} \tag{1}$$

where each $A_i \in \mathcal{S}^n$, $b \in \mathfrak{R}^m$, and $C \in \mathcal{S}^n$ are given, and $X \in \mathcal{S}^n$. Throughout this article, we assume that the set of matrices $\{A_i\}$ is linearly independent. The dual problem associated with (P) is:

$$(D) \quad \begin{aligned} \max_{y,S} \quad & b^T y \\ & \sum_{i=1}^m y_i A_i + S = C, \\ & S \succeq 0, \end{aligned} \tag{2}$$

where $y \in \mathfrak{R}^m$ and $S \in \mathcal{S}^n$. We write $F(P)$ and $F(D)$ for the sets of feasible solutions to (P) and (D) respectively, and correspondingly $F^0(P)$ and $F^0(D)$ for the sets of strictly feasible solutions to (P) and (D) respectively; here “strictly” means that X or S is required to be positive definite. Hence

$$\begin{aligned} F(P) &:= \{X \in \mathcal{S}_+^n : A_i \bullet X = b_i, \quad i = 1, \dots, m\}, \\ F^0(P) &:= \{X \in F(P) : X \in \mathcal{S}_{++}^n\}, \end{aligned}$$

$$\begin{aligned}
F(D) &:= \{(y, S) \in \mathbb{R}^m \times \mathcal{S}_+^n : \sum_{i=1}^m y_i A_i + S = C\}, \\
F^0(D) &:= \{(y, S) \in F(D) : S \in \mathcal{S}_{++}^n\}.
\end{aligned}$$

The optimal values for (P) and (D) will be denoted by $\text{val}(P)$ and $\text{val}(D)$, respectively.

We start by giving the following simple result commonly referred to as the “weak duality lemma”.

Proposition 2.1 *If X and (y, S) are feasible in (P) and (D) respectively, then*

$$C \bullet X - b^T y = X \bullet S \geq 0.$$

Thus, the quantity $X \bullet S$ is the “excess” of the primal objective function value $C \bullet X$ over the dual value $b^T y$. It is commonly referred to as the *duality gap* at (X, y, S) .

Corollary 2.1 *Suppose that X and (y, S) are feasible solutions for (P) and (D) respectively, satisfying $X \bullet S = 0$, or equivalently $XS = 0$. Then X and (y, S) are optimal in their respective problems.*

We say that (P) or (D) satisfies strong duality if there exist X and (y, S) satisfying the assumptions of Corollary 2.1. Unfortunately, the optimal values of (P) and (D) are not necessarily equal for every SDP problem, and even if they are, strong duality does not necessarily have to hold. The following result gives sufficient conditions for the duality gap between (P) and (D) to be zero and/or for strong duality to hold.

Proposition 2.2 *The following statements hold:*

- a) *If $\text{val}(P) > -\infty$ and $F^0(P) \neq \emptyset$ then the set of optimal solutions for (D) is nonempty and bounded and $\text{val}(P) = \text{val}(D)$;*
- b) *If $\text{val}(D) < \infty$ and $F^0(D) \neq \emptyset$ then the set of optimal solutions for (P) is nonempty and bounded and $\text{val}(P) = \text{val}(D)$;*
- c) *If $F^0(P) \neq \emptyset$ and $F^0(D) \neq \emptyset$ then the set of optimal solutions of (P) and (D) are nonempty and bounded and $\text{val}(P) = \text{val}(D)$. (Hence, strong duality is satisfied.)*

A proof of Proposition 2.2 can be found for example in Section 4.2 of [75]. By Corollary 2.1, it is clear that the conditions below (together with X and S belonging to \mathcal{S}_+^n) are sufficient for X and (y, S) to be optimal solutions:

$$\begin{aligned}
& \sum_{i=1}^m y_i A_i + S = C, \\
A_i \bullet X & = b_i, \quad \text{for } i = 1, \dots, m, \\
XS & = 0.
\end{aligned} \tag{3}$$

To simplify the notation slightly, we introduced the operator $\mathcal{A} : \mathcal{S}^n \rightarrow \mathbb{R}^m$ defined as $(\mathcal{A}X)_i := A_i \bullet X$ for all $X \in \mathcal{S}^n$ and $i = 1, \dots, m$. Then the adjoint $\mathcal{A}^* : \mathbb{R}^m \rightarrow \mathcal{S}^n$ of this operator is given by $\mathcal{A}^*y = \sum_{i=1}^m y_i A_i$ for every $y \in \mathbb{R}^m$. Using this notation, we can rewrite the equations above as

$$\begin{aligned}
& \mathcal{A}^*y + S = C, \\
\mathcal{A}X & = b, \\
XS & = 0.
\end{aligned} \tag{4}$$

In both (3) and (4) we could replace the last equation equivalently by $X^{1/2}SX^{1/2} = 0$, or by $S^{1/2}XS^{1/2} = 0$.

The central path is defined as the set of solutions $(X, y, S) = (X(\nu), y(\nu), S(\nu)) \in \mathcal{S}_+^n \times \mathbb{R}^m \times \mathcal{S}_+^n$ to

$$\begin{aligned} \mathcal{A}^*y + S &= C, \\ \mathcal{A}X &= b, \\ XS &= \nu I, \end{aligned} \tag{5}$$

for all $\nu > 0$. Clearly any solution to these equations gives strictly feasible solutions to both (P) and (D), since the last condition implies that X and S are nonsingular, hence positive definite. It turns out that the existence of strictly feasible solutions for both (P) and (D) is sufficient for the existence and uniqueness of solutions to (5) for every positive ν .

The key to the proof of the above result is the analysis of a certain barrier problem associated with (P) whose set of optimality conditions is exactly (5). Consider the following barrier function for the cone of positive semidefinite matrices \mathcal{S}_+^n :

$$f(X) := -\ln \det X \tag{6}$$

(by convention, we call this a barrier function for \mathcal{S}_+^n , even though it is defined only for points in \mathcal{S}_{++}^n ; it clearly tends to $+\infty$ as X in \mathcal{S}_{++}^n converges to a point on the boundary of \mathcal{S}_+^n). We need to deal with the derivatives of f . The first derivative of f at $X \in \mathcal{S}_{++}^n$ is $f'(X) = -X^{-1}$, in the usual sense that

$$f(X + H) = f(X) + [-X^{-1}] \bullet H + o(\|H\|).$$

For the second derivative, we introduce the useful notation $P \odot Q$ for $n \times n$ matrices P and Q (usually P and Q are symmetric). This is an operator defined from \mathcal{S}^n to \mathcal{S}^n by

$$(P \odot Q)U := \frac{1}{2}(PUQ^T + QUP^T). \tag{7}$$

Then it is not too hard to show that the second derivative of f is $f''(X) = X^{-1} \odot X^{-1}$, in the usual sense that

$$f(X + H) = f(X) + f'(X) \bullet H + \frac{1}{2}[(X^{-1} \odot X^{-1})H] \bullet H + o(\|H\|^2).$$

Note that $f''(X)$ is a self-adjoint and positive definite operator:

$$f''(X)U \bullet V = [X^{-1}UX^{-1}] \bullet V = \text{trace } X^{-1}UX^{-1}V = \text{trace } X^{-1}VX^{-1}U = f''(X)V \bullet U$$

and

$$f''(X)U \bullet U = \text{trace } X^{-1}UX^{-1}U = \text{trace } (X^{-1/2}UX^{-1/2})^2 = \|X^{-1/2}UX^{-1/2}\|_F^2,$$

where $\|\cdot\|_F$ denotes the Frobenius norm, the norm induced by the inner product we are using; this quantity is nonnegative, and positive unless $X^{-1/2}UX^{-1/2}$ (and hence U) is zero.

Since f'' is positive definite everywhere, f is strictly convex. Now we consider the primal barrier problem:

$$\begin{aligned} (P_\nu) \quad \min_X \quad & C \bullet X + \nu f(X) \\ & \mathcal{A}X = b, \\ & X \in \mathcal{S}_{++}^n, \end{aligned}$$

for positive ν . Note that the KKT (or Lagrange) conditions for this problem are necessary and sufficient, since the objective function is convex and the constraints linear (apart from the open set constraint). These optimality conditions can be written as

$$\mathcal{A}X = b, \quad C - \nu X^{-1} - \mathcal{A}^*y = 0,$$

for $X \in \mathcal{S}_{++}^n$, which can alternatively be expressed as (5) by setting $S := \nu X^{-1} \in \mathcal{S}_{++}^n$.

Let us note that (5) also gives the optimality conditions for the dual barrier problem:

$$(D_\nu) \quad \begin{aligned} \max_{y,S} \quad & b^T y - \nu f(S) \\ & \mathcal{A}^* y + S = C, \\ & S \in \mathcal{S}_{++}^n. \end{aligned}$$

The next result shows that the central path is well-defined and is strongly related to the optimal solutions of (P_ν) and (D_ν) .

Theorem 2.1 *Suppose that both (P) and (D) have strictly feasible solutions. Then, for every $\nu > 0$, there is a unique solution $(X(\nu), y(\nu), S(\nu))$ in $\mathcal{S}_{++}^n \times \mathbb{R}^m \times \mathcal{S}_{++}^n$ to the central path equations (5). Moreover, for every $\nu > 0$, $X(\nu)$ and $(y(\nu), S(\nu))$ are the unique optimal solutions of the barrier problems (P_ν) and (D_ν) , respectively.*

A proof of Theorem 2.1 can be found for example in Section 2 of [71]. The above result guarantees the existence and uniqueness of points on the central path, but it does not justify calling it a path. This fact will follow if we show that the equations defining it are differentiable, with a derivative that is square and nonsingular at points on the path. Unfortunately, while the equations of (5) are certainly differentiable, the derivative is not even square since the left-hand side maps $(X, y, S) \in \mathcal{S}_{++}^n \times \mathbb{R}^m \times \mathcal{S}_{++}^n \subseteq \mathcal{S}^n \times \mathbb{R}^m \times \mathcal{S}^n$ to a point in $\mathcal{S}^n \times \mathbb{R}^m \times \mathbb{R}^{n \times n}$; XS is usually not symmetric even if X and S are. We therefore need to change the equations defining the central path. There are many possible approaches, which as we shall see lead to different search directions for our algorithms, but for now we choose a simple one: we replace $XS = \nu I$ by $-\nu X^{-1} + S = 0$. As in our discussion of the barrier function f , the function $X \rightarrow -\nu X^{-1}$ is differentiable, with derivative $\nu(X^{-1} \odot X^{-1})$. So the central path is defined by the equations

$$\Phi_P(X, y, S) := \begin{pmatrix} \mathcal{A}^* y + S \\ \mathcal{A} X \\ -\nu X^{-1} + S \end{pmatrix} = \begin{pmatrix} C \\ b \\ 0 \end{pmatrix}, \quad (8)$$

whose derivative is

$$\Phi'_P(X, y, S) := \begin{pmatrix} 0 & \mathcal{A}^* & \mathcal{I} \\ \mathcal{A} & 0 & 0 \\ \nu(X^{-1} \odot X^{-1}) & 0 & \mathcal{I} \end{pmatrix}, \quad (9)$$

where \mathcal{I} denotes the identity operator on \mathcal{S}^n . We have been rather loose in writing this in matrix form, since the blocks are operators rather than matrices, but the meaning is clear. We want to show that this derivative is nonsingular, and for this it suffices to prove that its null-space is trivial. Since similar equations will occur frequently, let us derive this from a more general result.

Theorem 2.2 *Suppose the operators \mathcal{E} and \mathcal{F} map \mathcal{S}^n to itself, and that \mathcal{E} is nonsingular and $\mathcal{E}^{-1}\mathcal{F}$ is positive definite. Then the solution to*

$$\begin{aligned} \mathcal{A}^* \Delta y + \Delta S &= R_d, \\ \mathcal{A} \Delta X &= r_p, \\ \mathcal{E} \Delta X + \mathcal{F} \Delta S &= R_{EF} \end{aligned} \quad (10)$$

is uniquely given by

$$\begin{aligned} \Delta y &= (\mathcal{A}\mathcal{E}^{-1}\mathcal{F}\mathcal{A}^*)^{-1}(r_p - \mathcal{A}\mathcal{E}^{-1}(R_{EF} - \mathcal{F}R_d)), \\ \Delta S &= R_d - \mathcal{A}^* \Delta y, \\ \Delta X &= \mathcal{E}^{-1}(R_{EF} - \mathcal{F}\Delta S). \end{aligned} \quad (11)$$

Proof: The formulae for ΔS and ΔX follow directly from the first and third equations. Now substituting for ΔS in the formula for ΔX , and inserting this in the second equation, we obtain after some manipulation

$$(\mathcal{A}\mathcal{E}^{-1}\mathcal{F}\mathcal{A}^*)\Delta y = r_p - \mathcal{A}\mathcal{E}^{-1}(R_{EF} - \mathcal{F}R_d).$$

Since $\mathcal{E}^{-1}\mathcal{F}$ is positive definite and the A_i 's are linearly independent, the $m \times m$ matrix on the left is positive definite and hence nonsingular. This verifies that Δy is uniquely determined as given, and then so are ΔS and ΔX . Moreover, these values solve the equations. \square

In our case, \mathcal{F} is the identity, while \mathcal{E} is $\nu(X^{-1} \odot X^{-1})$ with inverse $\nu^{-1}(X \odot X)$. This is easily seen to be positive definite, in the same way we showed that $f''(X)$ was. Hence the theorem applies, and so the derivative of the function Φ_P is nonsingular on the central path (and throughout $\mathcal{S}_{++}^n \times \mathfrak{R}^m \times \mathcal{S}_{++}^n$); thus the central path is indeed a differentiable path.

By taking the trace of the last equation of (5), we obtain the last part of the following theorem, which summarizes what we have observed.

Theorem 2.3 *Assume that both (P) and (D) have strictly feasible solutions. Then the set of solutions to (5) for all positive ν forms a nonempty differentiable path, called the central path. If $(X(\nu), y(\nu), S(\nu))$ solve these equations for a particular positive ν , then $X(\nu)$ is a strictly feasible solution to (P) and $(y(\nu), S(\nu))$ a strictly feasible solution to (D), with duality gap*

$$C \bullet X(\nu) - b^T y(\nu) = X(\nu) \bullet S(\nu) = n\nu. \quad (12)$$

A rigorous proof of Theorem 2.3 can be found for example in Section 2 of [71]. It has been shown that as ν tends to 0, the central path $(X(\nu), y(\nu), S(\nu))$ converges to a specific primal-dual optimal solution of (P) and (D) (see Goldfarb and Scheinberg [35], Kojima et al. [51] and Halicka et al. [37, 38]). Given that we start with strictly feasible solutions to (P) and (D), it is simple to maintain equality in the first two equations of (5), but satisfying the last nonlinear equation exactly is hard. Hence *path-following methods* require that the iterates satisfy these last equations in some approximate sense. The requirement can be stated in terms of certain *neighborhoods* of the central path, defined either in terms of norms or semi-norms of $X^{1/2}SX^{1/2} - \nu I$ or in terms of the eigenvalues of this matrix. In fact, we will replace ν by $\mu := \mu(X, S) := (X \bullet S)/n$ (cf. (12)). We then define the following centrality measures for $X, S \in \mathcal{S}_+^n$:

$$\begin{aligned} d_F(X, S) &:= \|X^{1/2}SX^{1/2} - \mu I\|_F &= [\sum_{i=1}^n (\lambda_i(XS) - \mu)^2]^{1/2}, \\ d_\infty(X, S) &:= \|X^{1/2}SX^{1/2} - \mu I\| &= \max_{1 \leq i \leq n} |\lambda_i(XS) - \mu|, \\ d_{-\infty}(X, S) &:= \|X^{1/2}SX^{1/2} - \mu I\|_{-\infty} &:= \max_{1 \leq i \leq n} (\mu - \lambda_i(XS)), \end{aligned} \quad (13)$$

where $\lambda_1(XS) \leq \dots \leq \lambda_n(XS)$ denote the eigenvalues of XS , $\mu := \mu(X, S)$ and $\|\cdot\|$ for a matrix denotes the operator norm with respect to the Euclidean norm on vectors. The last equation basically defines $\|M\|_{-\infty}$ for symmetric M as the negative of the smallest eigenvalue of M ; it is easy to check that this is a semi-norm on \mathcal{S}^n . The equations hold because $X^{1/2}SX^{1/2}$ and XS , being similar, have the same eigenvalues. It is easy to see that these measures are zero, for strictly feasible iterates, only for points on the central path. Also, for $X, S \in \mathcal{S}_+^n$,

$$d_{-\infty}(X, S) \leq d_\infty(X, S) \leq d_F(X, S),$$

and $d_{-\infty}(X, S) < \mu(X, S)$ holds if and only if X and S are positive definite.

For a given $0 < \gamma < 1$, we then define the neighborhoods

$$\begin{aligned}\mathcal{N}_F(\gamma) &:= \{(X, y, S) \in F^0(P) \times F^0(D) : d_F(X, S) \leq \gamma\mu(X, S)\}, \\ \mathcal{N}_\infty(\gamma) &:= \{(X, y, S) \in F^0(P) \times F^0(D) : d_\infty(X, S) \leq \gamma\mu(X, S)\}, \\ \mathcal{N}_{-\infty}(\gamma) &:= \{(X, y, S) \in F^0(P) \times F^0(D) : d_{-\infty}(X, S) \leq \gamma\mu(X, S)\}.\end{aligned}$$

The first and last are usually called the *narrow* and *wide* neighborhoods of the central path.

3 Search directions for the second-order methods

In this section, we describe the main Newton-type directions that have been proposed in the context of primal-dual interior-point algorithms for SDP.

At each iteration, a path-following algorithm has available an iterate (X, y, S) in one of the neighborhoods above, and obtains another such iterate by taking a step in some search direction $(\Delta X, \Delta y, \Delta S)$. Usually this direction is chosen so that $(X + \Delta X, y + \Delta y, S + \Delta S)$ should be a good approximation to $(X(\sigma\mu), y(\sigma\mu), Z(\sigma\mu))$, with $\mu := \mu(X, S)$ and $\sigma \in [0, 1]$, and then the step size is chosen so that the next iterate lies in the appropriate neighborhood. Since the central path is defined by a system of nonlinear equations, the direction is frequently taken as the Newton step for some such system of equations, but choosing the appropriate system is not straightforward.

For simplicity, we are assuming that all our iterates are feasible. Thus our generic feasible-interior-point path-following method can be described as follows:

Generic feasible-interior-point path-following method:

Choose $L > 1$, $0 < \gamma < 1$, and an associated neighborhood $\mathcal{N}(\gamma)$.

Let $(X^0, y^0, S^0) \in \mathcal{N}(\gamma)$ be given and set $\mu_0 := (X^0 \bullet S^0)/n$.

Repeat until $\mu_k \leq 2^{-L}\mu_0$, **do**

- (1) Choose a direction $(\Delta X^k, \Delta y^k, \Delta S^k)$;
- (2) Set $(X^{k+1}, y^{k+1}, S^{k+1}) := (X^k, y^k, S^k) + \alpha_k(\Delta X^k, \Delta y^k, \Delta S^k)$
for some $\alpha_k > 0$ such that $(X^{k+1}, y^{k+1}, S^{k+1}) \in \mathcal{N}(\gamma)$;
- (3) Set $\mu_{k+1} := (X^{k+1} \bullet S^{k+1})/n$ and increment k by 1.

End do

End

In this section, we are interested in the choice of the search direction. This would appear to be easy: merely take a Newton step for the central path equations (5) for $\nu := \sigma\mu$. However, as we observed above (8), these equations map a point in $\mathcal{S}^n \times \mathfrak{R}^m \times \mathcal{S}^n$ to a point in $\mathcal{S}^n \times \mathfrak{R}^m \times \mathfrak{R}^{n \times n}$, and thus a Newton step is not defined. Instead we can use the equations (8), and the Newton step for these defines the so-called *primal* direction for SDP. Indeed, it is easy to see that the resulting ΔX is exactly the Newton step for the primal barrier problem (P_ν) from the current iterate X . Moreover, Theorem 2.2 and the discussion below it implies that this Newton step is well-defined for all $(X, y, S) \in \mathcal{N}(\gamma)$.

Such a primal Newton barrier method was proposed by Nesterov and Nemirovskii [75] as an extension of the corresponding method for linear programming. However, it does not seem to be as attractive, either theoretically or practically, as the primal-dual methods (where ΔX , Δy , and ΔS all depend on both the primal iterate X and the dual iterate (y, S)) to be described below. Theoretical results have only been obtained for short-step methods using the narrow neighborhood $\mathcal{N}_F(\gamma)$, and practically, these methods do not seem to allow as aggressive step sizes as primal-dual methods. The reason is not too hard to see. These primal methods are based on taking Newton steps for systems of equations involving the inverse map $X \rightarrow X^{-1}$, and this map is not as well-behaved (particularly as the solution is approached, or after a long step) as the map $(X, S) \rightarrow XS$ that appears in (5). Thus

we would like to modify the equations (5) to allow a Newton step to be taken without losing this nice behavior.

Because we are assuming our current iterates are feasible, the search directions we consider below can all be obtained as solutions of (10) with r_p and R_d both zero, and suitable \mathcal{E} , \mathcal{F} , and R_{EF} . Computing the corresponding directions follows the scheme of (11), but the details vary by method, and we refer to the original papers for elaboration.

The first method we consider is due to Alizadeh, Haerberly, and Overton [2], and is motivated by symmetrizing the last equation of (5) directly; thus we replace $XS = \nu I$ by $XS + SX = \nu I$. Then the equations do indeed map $\mathcal{S}^n \times \mathcal{R}^m \times \mathcal{S}^n$ to itself. Linearizing this gives the following equation for the direction (in addition to the feasibility equations):

$$\frac{1}{2}(\Delta X S + S \Delta X + X \Delta S + \Delta S X) = \nu I - \frac{1}{2}(XS + SX).$$

Thus the resulting Newton step $(\Delta X, \Delta y, \Delta S)$ solves (10) for

$$\mathcal{E} = S \odot I, \quad \mathcal{F} = X \odot I, \quad R_{EF} = \nu I - \frac{1}{2}(XS + SX).$$

We call this the AHO direction. Note that the last equation is symmetric between the primal and dual: it is unchanged if we interchange X and S and correspondingly ΔX and ΔS . We call directions obtained from such systems *primal-dual symmetric*, and if a direction is not primal-dual symmetric, we call the direction resulting from such an interchange its *dual* direction.

The AHO direction appears to be very natural, and indeed it does perform very well in practice, obtaining highly accurate solutions in a small number of iterations in almost all cases. However, it does have some drawbacks. From a theoretical point of view, it does not have the attractive property of scale invariance (see Todd, Toh, and Tütüncü [96] and [94]); polynomial-time convergence results seem much harder to obtain; and the search directions may not be well-defined if the iterates do not lie in the neighborhood $\mathcal{N}_\infty(1/\sqrt{2})$ (see [96], Monteiro and Zanjácomo [66] and Monteiro and Todd [62]). From a practical point of view, obtaining $\mathcal{E}^{-1}U$ for a symmetric matrix U as needed to solve the system requires the solution of a Lyapunov system; while this is not too onerous, the $m \times m$ matrix $\mathcal{A}\mathcal{E}^{-1}\mathcal{F}\mathcal{A}^*$ that arises in the solution is not in general symmetric, and this leads to operation counts (and times) that are up to twice as high per iteration as for other methods.

The next method we describe, instead of symmetrizing the last equation, allows non-symmetric search directions to make the dimensions match. Thus we initially allow ΔX to be an arbitrary matrix (note that the dual feasibility equations imply that ΔS must be symmetric), so that the central path equations map a point in $\mathcal{R}^{n \times n} \times \mathcal{R}^m \times \mathcal{S}^n$ to one in $\mathcal{S}^n \times \mathcal{R}^m \times \mathcal{R}^{n \times n}$. Then the resulting value for ΔX is symmetrized. Thus the first linearization gives (the feasibility equations and)

$$\Delta X S + X \Delta S = \nu I - XS,$$

or equivalently

$$\Delta X + X \Delta S S^{-1} = \nu S^{-1} - X.$$

To symmetrize ΔX it suffices to symmetrize the second term, since the others are already symmetric. Thus the resulting direction solves (10) for

$$\mathcal{E} = I \odot I, \quad \mathcal{F} = X \odot S^{-1}, \quad R_{EF} = \nu S^{-1} - X.$$

This direction was obtained, by roughly the argument above, independently by Helmberg, Rendl, Vanderbei, and Wolkowicz [41] and Kojima, Shindoh, and Hara [51]. Kojima et al. also described its dual direction with

$$\mathcal{E} = S \odot X^{-1}, \quad \mathcal{F} = I \odot I, \quad R_{EF} = \nu X^{-1} - S.$$

Later, Monteiro [58] gave another derivation of these directions. Each can be viewed in a way very similar to the AHO approach as the Newton step for a symmetrized system of equations, but before symmetrizing XS , a similarity transformation is applied. More precisely, consider the following operator $\mathcal{J}_P(\cdot)$ mapping $\mathfrak{R}^{n \times n}$ into \mathcal{S}^n defined for every $U \in \mathfrak{R}^{n \times n}$ by

$$\mathcal{J}_P(U) = \frac{1}{2} \left[PUP^{-1} + (PUP^{-1})^T \right],$$

where P is a given nonsingular matrix. (If P is the identity, \mathcal{J}_P is the usual projection operator onto \mathcal{S}^n .) The directions above are the Newton steps for the system of equations (5) where the last equation is replaced by

$$\mathcal{J}_P(XS) = \nu I, \tag{14}$$

where $P = S^{1/2}$ for the first and $P = X^{-1/2}$ for the second direction. Because of these three contributions, the first direction is often called the HRVW/KSH/M direction. Zhang [107] then generalized Monteiro's work and analyzed the resulting directions for arbitrary nonsingular P ; the resulting set of directions is called the Monteiro-Zhang family. Clearly this family also contains the AHO direction, corresponding to $P = I$.

Now let us consider the so-called Nesterov-Todd direction, which is the specialization to semidefinite programming of a symmetric primal-dual direction for a certain class of convex programming problems studied in [76, 77]. We will derive this direction as a way to make the primal direction primal-dual symmetric. Recall that the primal direction is defined by the feasibility equations and

$$\nu X^{-1} \Delta X X^{-1} + \Delta S = \nu X^{-1} - S.$$

This last equation can be written equivalently as

$$\Delta X + (\nu^{-1/2} X) \Delta S (\nu^{-1/2} X) = X - \nu^{-1} X S X. \tag{15}$$

On the other hand, its dual direction, which is the Newton step for the dual barrier problem, solves the feasibility equations and

$$\Delta X + (\nu^{1/2} S^{-1}) \Delta S (\nu^{1/2} S^{-1}) = \nu S^{-1} - X, \tag{16}$$

which coincides with the equation displayed above if the current iterate is exactly the point on the central path corresponding to the parameter ν . We will therefore replace the terms $\nu^{-1/2} X$ and $\nu^{1/2} S^{-1}$ by a matrix that is in some respects half way between them (or halfway between X and S^{-1}). Suppose we substitute for these two expressions $W \in \mathcal{S}_{++}^n$, so that (15) and (16) become

$$\begin{aligned} W^{-1} \Delta X W^{-1} + \Delta S &= \nu X^{-1} - S \\ \Delta X + W \Delta S W &= \nu S^{-1} - X. \end{aligned} \tag{17}$$

These two equations are equivalent as long as $WSW = X$, which implies that $WX^{-1}W = S^{-1}$ also. By pre- and post-multiplying by $S^{1/2}$, we find that

$$W = W_{nt} := S^{-1/2} (S^{1/2} X S^{1/2})^{1/2} S^{-1/2}; \tag{18}$$

this matrix is called the *metric geometric mean* of X and S^{-1} [4]; it is similarly the metric geometric mean of $\nu^{-1/2} X$ and $\nu^{1/2} S^{-1}$. In interior-point methods, it is called the *scaling* matrix corresponding to the primal and dual iterates X and S .

The direction defined by the feasibility equations and (17), or equivalently by (10) for

$$\mathcal{E} = I \odot I, \quad \mathcal{F} = W \odot W, \quad R_{EF} = \nu S^{-1} - X,$$

where W is the scaling matrix defined by (18), is called the Nesterov-Todd (NT) direction. Todd, Toh, and Tütüncü [96] also showed that the NT direction is a member of the Monteiro-Zhang family, corresponding to $P = W^{-1/2}$ in (14).

The AHO, HRVW/KSH/M, and NT directions are those most used in practice, as well as being the ones most analyzed theoretically. Before ending this section, however, we briefly mention several other directions which have been addressed in the literature. First, in addition to the HRVW/KSH/M direction and its dual, Kojima, Shindoh, and Hara [51] described a whole family of directions (including also the NT direction, but not the AHO direction). Next, Monteiro and Tsuchiya [63] introduced and analyzed a family of directions called the Monteiro-Tsuchiya family, basically defined by taking a Newton step for the central path equations (5) where the final equation is replaced by

$$X^{1/2}SX^{1/2} = \nu I.$$

A whole family arises if this approach is applied to a scaled problem, involving a nonsingular matrix P as in the Monteiro-Zhang family. We refer to the original paper and to [94] for more details. Tseng [100] introduced yet another family of search directions. While the Tseng and Monteiro-Tsuchiya families differ from the Kojima-Shindoh-Hara family, and from each other, all contain the HRVW/KSH/M direction, its dual, and the NT direction, but exclude the AHO direction. Finally, Gu [36] and Toh [97] have recently proposed two other members of the Monteiro-Zhang family which seem promising. Kojima, Shida, and Shindoh [50] describe most of these directions and families and show the relationships between them; Todd [94] describes twenty particular search directions and analyzes their theoretical properties.

4 Primal-dual path-following methods

In this section we discuss the main convergence results that have been obtained for feasible primal-dual path-following algorithms based on three families of search directions, namely: the MZ, MT, and KSH families.

4.1 Algorithms based on the MZ family

In this subsection we discuss primal-dual path-following algorithms based on the MZ family of search directions which was introduced by Monteiro and Zhang [58, 68, 107].

Given $(X, y, S) \in \mathcal{S}_{++}^n \times \mathfrak{R}^m \times \mathcal{S}_{++}^n$ and a nonsingular matrix $P \in \mathfrak{R}^{n \times n}$, consider the linear system of equations

$$\begin{aligned} \mathcal{A}^* \Delta y + \Delta S &= C - S - \mathcal{A}^* y, \\ \mathcal{A} \Delta X &= b - \mathcal{A} X, \\ \mathcal{J}_P(\Delta X S + X \Delta S) &= \sigma \mu I - \mathcal{J}_P(X S), \end{aligned} \tag{19a}$$

where $(\Delta X, \Delta y, \Delta S) \in \mathcal{S}^n \times \mathfrak{R}^m \times \mathcal{S}^n$, $\sigma \in [0, 1]$ is the centering parameter and $\mu = \mu(X, S) := (X \bullet S)/n$ is the normalized duality gap corresponding to (X, y, S) . This is the Newton system for the feasibility equations and the central path (14). Under certain conditions (see Proposition 4.1) on (X, y, S) , it can be shown that system (19) has a unique solution. As P varies over the set of nonsingular $n \times n$ -matrices, the set of resulting directions gives rise to the MZ family of directions with parameter σ at the point (X, y, S) .

It can be shown that set of solutions to system (19) remains invariant as long as the matrix $V = P^T P$ does not change. Hence, for fixed $V \in \mathcal{S}_{++}^n$, it is enough to consider only the matrix $V^{1/2}$ among all those scaling matrices P such that $P^T P = V$. Hence, there is no loss of generality in

restricting our attention to those scaling matrices P that are in \mathcal{S}_{++}^n , since they yield all the possible directions that can be generated by system (19) as P varies over the set of nonsingular matrices.

We now describe a scaling procedure which allows us to view the MZ direction as being the AHO direction in a certain scaled space. For a given $P \in \mathcal{S}_{++}^n$, consider the following change of variables

$$\tilde{X} := PXP, \quad (\tilde{y}, \tilde{S}) := (y, P^{-1}SP^{-1}). \quad (20)$$

Letting

$$\tilde{C} := P^{-1}CP^{-1}, \quad (\tilde{A}_i, \tilde{b}_i) := (P^{-1}A_iP^{-1}, b_i), \quad \text{for } i = 1, \dots, m,$$

we easily see that problems (P) and (D) are equivalent to the pair of problems

$$\begin{aligned} (\tilde{P}) \quad & \min \left\{ \tilde{C} \bullet \tilde{X} : \tilde{A}\tilde{X} = \tilde{b}, \tilde{X} \succeq 0 \right\}, \\ (\tilde{D}) \quad & \max \left\{ \tilde{b}^T \tilde{y} : \tilde{A}^* \tilde{y} + \tilde{S} = \tilde{C}, \tilde{S} \succeq 0 \right\}, \end{aligned}$$

where the operator $\tilde{A} : \mathcal{S}^n \rightarrow \mathfrak{R}^m$ is defined as $(\tilde{A}X)_i = \tilde{A}_i \bullet X$ for every $X \in \mathcal{S}^n$ and $i = 1, \dots, m$. It can be easily verified that if (X, y, S) and $(\tilde{X}, \tilde{y}, \tilde{S})$ in $\mathcal{S}_{++}^n \times \mathfrak{R}^m \times \mathcal{S}_{++}^n$ are related according to (20), then $(\tilde{X}, \tilde{y}, \tilde{S})$ is feasible in (\tilde{P}) and (\tilde{D}) if and only if (X, y, S) is feasible in (P) and (D) ; moreover, we have $d_F(\tilde{X}, \tilde{S}) = d_F(X, S)$, $d_\infty(\tilde{X}, \tilde{S}) = d_\infty(X, S)$, $d_{-\infty}(\tilde{X}, \tilde{S}) = d_{-\infty}(X, S)$, and $\mu = \tilde{\mu}$, where $\tilde{\mu} := (\tilde{X} \bullet \tilde{S})/n$. Also, if $(\tilde{X}_\nu, \tilde{y}_\nu, \tilde{S}_\nu)$ denotes the point on the central path with parameter $\nu > 0$ for the pair of problems (\tilde{P}) and (\tilde{D}) , then $(\tilde{X}_\nu, \tilde{y}_\nu, \tilde{S}_\nu) = (PX_\nu P, y_\nu, P^{-1}S_\nu P^{-1})$.

A solution $(\Delta X, \Delta y, \Delta S)$ of the MZ system (19) corresponds in the scaled space to the scaled direction $(\widetilde{\Delta X}, \widetilde{\Delta y}, \widetilde{\Delta S})$ defined as

$$\widetilde{\Delta X} := P\Delta XP, \quad \widetilde{\Delta S} := P^{-1}\Delta SP^{-1}, \quad \widetilde{\Delta y} = \Delta y. \quad (21)$$

The direction $(\widetilde{\Delta X}, \widetilde{\Delta y}, \widetilde{\Delta S})$ is easily seen to be a solution of the AHO system:

$$\begin{aligned} \tilde{A}^* \widetilde{\Delta y} + \widetilde{\Delta S} &= \tilde{C} - \tilde{S} - \tilde{A}^* \tilde{y}, \\ \tilde{A} \widetilde{\Delta X} &= \tilde{b} - \tilde{A} \tilde{X}, \\ \mathcal{J}_I(\widetilde{\Delta X} \tilde{S}) + \tilde{X} \widetilde{\Delta S} &= \mathcal{J}_I(\sigma \tilde{\mu} I - \tilde{X} \tilde{S}). \end{aligned}$$

In other words, $(\widetilde{\Delta X}, \widetilde{\Delta y}, \widetilde{\Delta S})$ is the AHO direction (with parameter σ) at the point $(\tilde{X}, \tilde{y}, \tilde{S})$ for the pair of problems (\tilde{P}) and (\tilde{D}) .

The result stated next gives two sufficient conditions for the MZ direction to be well-defined.

Proposition 4.1 *Suppose that $(X, y, S) \in \mathcal{S}_{++}^n \times \mathfrak{R}^m \times \mathcal{S}_{++}^n$, $\sigma \in \mathfrak{R}$ and $P \in \mathcal{S}_{++}^n$. Then, system (19) has a unique solution if either one of the following two conditions hold:*

- a) $d_\infty(X, S) = d_\infty(\tilde{X}, \tilde{S}) < \mu/\sqrt{2}$;
- b) $\mathcal{J}_P(XS) = \mathcal{J}_I(\tilde{X}\tilde{S}) \succeq 0$.

Monteiro and Zanjácomo [66] established Proposition 4.1(a) with the factor $1/\sqrt{2}$ replaced by $1/2$ and Monteiro and Todd (Proposition 4.2 of [62]) subsequently extended it for the factor $1/\sqrt{2}$. Proposition 4.1(b) was established by Shida, Shindoh and Kojima [90] for the case in which $P = I$ and was subsequently extended for a general $P \in \mathcal{S}_{++}^n$ by Todd, Toh and Tütüncü (Theorem 3.1 of [96]).

The algorithms considered in this subsection are all special cases of the following framework.

Framework MZ-PF:

Same as the Generic Algorithm of Section 3 but with step (1) replaced by

- (1') Choose a matrix $P^k \in \mathcal{S}_{++}^n$ and a centrality parameter $\sigma_k \in [0, 1]$;
compute the solution $(\Delta X^k, \Delta y^k, \Delta S^k)$ of system (19) with $P = P^k$,
 $\sigma = \sigma_k$ and $(X, y, S) = (X^k, y^k, S^k)$.

End

In the remaining part of this subsection, we will state polynomial convergence results for two special classes of algorithms from Framework MZ-PF, namely: short-step methods and long-step methods.

We start with short-step methods. These methods generate their sequence of iterates within a neighborhood $\mathcal{N}_F(\gamma)$ with $\gamma < 1/\sqrt{2}$, and hence the corresponding MZ directions are all well-defined in view of Proposition 4.1(a). Short-step methods (regardless of the family of directions used) choose the sequences $\{\sigma_k\}$ and $\{\alpha_k\}$ according to the following strategy.

Short-step strategy: for all $k \geq 0$, let $\alpha_k = 1$ and σ_k be the smallest $\sigma \in [0, 1]$ such that $(X^k, y^k, S^k) + (\Delta X^k(\sigma), \Delta y^k(\sigma), \Delta S^k(\sigma)) \in \mathcal{N}_F(\gamma)$, where $(\Delta X^k(\sigma), \Delta y^k(\sigma), \Delta S^k(\sigma))$ denotes the search direction at (X^k, y^k, S^k) as an affine function of σ

The following convergence result for the short-step method based on the MZ family is due to Monteiro [59].

Theorem 4.1 (Corollary 4.2 of [59]) *Let $\gamma \in (0, 1/\sqrt{2})$ and $(X^0, y^0, S^0) \in \mathcal{N}_F(\gamma)$ be given. Then, the sequence of iterates $\{(X^k, y^k, S^k)\} \subset \mathcal{N}_F(\gamma)$ generated by Framework MZ-PF with the short-step strategy satisfies $X^k \bullet S^k \leq (1 - \delta/\sqrt{n})^k (X^0 \bullet S^0)$, for every $\delta > 0$ satisfying*

$$\frac{2(\gamma^2 + \delta^2)}{(1 - \sqrt{2}\gamma)^2} \left(1 - \frac{\delta}{\sqrt{n}}\right)^{-1} \leq \gamma.$$

As a consequence, the method terminates in at most $\mathcal{O}(\sqrt{n}L)$ iterations.

Monteiro [59] has also established an $\mathcal{O}(\sqrt{n}L)$ iteration-complexity bound for a Mizuno-Todd-Ye predictor-corrector algorithm [57] based on the whole MZ family of directions. Due to space constraints, we omit the details of this algorithm and instead refer the reader to Section 4.2 of [59]. Convergence of the short-step methods based on the NT direction and the two HRVW/KSH/M directions has been established earlier by Nesterov and Todd [76, 77] and Kojima, Shindoh and Hara [51], respectively. Theorem 4.1 generalizes these two results to the whole MZ family of directions. Since the MZ family contains the AHO direction, the above result also implies that the short-step method based on the AHO direction has an $\mathcal{O}(\sqrt{n}L)$ iteration-complexity bound.

We now describe the convergence results for long-step methods based on a subclass of the MZ family of directions. Short-step methods have the best known theoretical iteration-complexity bounds although in practice they are not effective due to the restrictive condition that the iterates have to remain in a relatively small central path neighborhood $\mathcal{N}_F(\cdot)$, and hence generate small steps. A practical advantage of the long-step methods is that they are based on the larger central path neighborhood $\mathcal{N}_{-\infty}(\cdot)$, which allows for more freedom in the choice of the next iterate and hence larger steps (hence, the name long-step methods). However, long-step methods have worse theoretical iteration-complexity bounds than short-step methods.

Since long-step methods are based on the central-path neighborhood $\mathcal{N}_{-\infty}(\gamma)$ which is generally wider than the neighborhood $\mathcal{N}_{\infty}(\gamma)$ where the MZ direction is known to be well-defined (if $\gamma < 1/\sqrt{2}$) regardless of the choice of the scaling matrix P (see Theorem 4.1(a)), it is important to restrict the

choice of P so as to guarantee that the MZ direction is well-defined at any $(X, y, S) \in \mathcal{N}_{-\infty}(\gamma)$. Indeed, consider the following class of scaling matrices:

$$\mathcal{P}(X, S) := \{P \in \mathcal{S}_{++}^n : PXS P^{-1} = P^{-1}SXP\} = \{P \in \mathcal{S}_{++}^n : \tilde{X}\tilde{S} = \tilde{S}\tilde{X}\}. \quad (23)$$

Since the scaling matrices in $\mathcal{P}(X, S)$ make \tilde{X} and \tilde{S} commute, we will refer to $\mathcal{P}(X, S)$ as the *commutative class* of scaling matrices at (X, S) . Clearly, by Proposition 4.1(b), the MZ direction is well-defined in the sense that (19) has a unique solution whenever $P \in \mathcal{P}(X, S)$. Indeed, if \tilde{X} and \tilde{S} commute, then $\mathcal{J}_T(\tilde{X}\tilde{S}) = \tilde{X}\tilde{S} = \tilde{X}^{1/2}\tilde{S}\tilde{X}^{1/2}$, which is positive definite.

Observe that the scaling matrices $P = W_{nt}^{1/2}$, $P = S^{1/2}$ and $P = X^{-1/2}$, corresponding to the NT direction, the HRVW/KSH/M direction and the dual HRVW/KSH/M direction, are contained in $\mathcal{P}(X, S)$, and hence our analysis applies to these directions. However, $P = I$ does not belong to $\mathcal{P}(X, S)$, except for the unlikely case when X and S commute; hence, our analysis does not apply to the AHO direction.

Long-step methods (based on any family of directions) choose the sequences $\{\sigma_k\}$ and $\{\alpha_k\}$ according to the following strategy.

Long-step strategy: *Let $\gamma, \sigma \in (0, 1)$ be given. For every $k \geq 0$, let $\sigma_k = \sigma$ and α_k be the largest $\alpha \geq 0$ such that $(X^k, y^k, S^k) + \alpha(\Delta X^k, \Delta y^k, \Delta S^k) \in \mathcal{N}_{-\infty}(\gamma)$.*

The main convergence result for long-step methods based on the MZ family of directions establishes an iteration-complexity bound in terms of the quantity κ_∞ defined as

$$\kappa_\infty := \sup \left\{ \mathbf{cond} \left[(\tilde{X}^k)^{-1} \tilde{S}^k \right] : k = 0, 1, 2, \dots \right\}, \quad (24)$$

where $\tilde{X}^k := PX^kP$ and $\tilde{S}^k := P^{-1}S^kP^{-1}$. Here, for a matrix $W \in \mathbb{R}^{n \times n}$ whose eigenvalues are all positive real numbers, we denote by $\mathbf{cond}(W)$ the ratio between the largest and the smallest eigenvalues of W . Clearly, $\kappa_\infty \geq 1$. The following convergence result is due to Monteiro and Zhang [68].

Theorem 4.2 (Theorem 3.1 of [68]) *Let $\gamma, \sigma \in (0, 1)$ and $(X^0, y^0, S^0) \in \mathcal{N}_{-\infty}(\gamma)$ be given. Assume that the sequence of scaling matrices $\{P^k\} \subset \mathcal{S}_{++}^n$ satisfies $P^k \in \mathcal{P}(X^k, S^k)$ for all $k \geq 0$ and $\kappa_\infty < \infty$. Then the sequence of iterates $\{(X^k, y^k, S^k)\}$ generated by Framework MZ-PF with the long-step strategy satisfies $X^k \bullet S^k \leq (1 - (1 - \sigma)\bar{\alpha})^k (X^0 \bullet S^0)$, where*

$$\bar{\alpha} := \min \left(1, \frac{\sigma\gamma}{1 - 2\sigma + \sigma^2/(1 - \gamma)} \frac{1}{\sqrt{\kappa_\infty n}} \right).$$

Consequently, the long-step algorithm terminates in at most $O(\sqrt{\kappa_\infty} nL)$ iterations.

The next result states the consequences of Theorem 4.2 for three special choices of the sequence $\{P^k\}$ which lead to the NT direction, the HRVW/KSH/M direction and the dual HRVW/KSH/M direction, respectively.

Corollary 4.1 (Theorem 3.2 of [68]) *The long-step algorithms based on the NT direction, the HRVW/KSH/M direction and the dual HRVW/KSH/M direction have iteration-complexity bounds equal to $\mathcal{O}(nL)$, $\mathcal{O}(n^{3/2}L)$ and $\mathcal{O}(n^{3/2}L)$, respectively.*

We note that the iteration-complexity bounds obtained in Theorem 4.1 for the long-step methods based on the HRVW/KSH/M and the dual HRVW/KSH/M directions were first obtained by Monteiro [58]. Among all long-step algorithms based on the MZ family, the one based on the Nesterov-Todd direction achieves the best possible iteration-complexity bound, namely $\mathcal{O}(nL)$, that can be derived from Theorem 4.2. In this case, we have $\tilde{X}^k = \tilde{S}^k$ for all $k \geq 0$, and hence $\kappa_\infty = 1$ in view of (24).

4.2 Algorithms based on the MT family

In this subsection we discuss primal-dual path-following algorithms based on the MT family of search directions which was introduced by Monteiro and Tsuchiya [63]. All the results stated below for the MT family are due to Monteiro and Tsuchiya [63] and proofs can be found there.

Recall that each direction of the MZ family can be transformed into the AHO direction in a certain scaled space obtained by a scaling transformation that depends on P (see (20)). This property of the MZ family could in fact be used to describe it. More precisely, given a scaling matrix $P \in \mathcal{S}_{++}^n$, a point $(X, y, S) \in \mathcal{S}_{++}^n \times \mathfrak{R}^m \times \mathcal{S}_{++}^n$ and a parameter σ , the corresponding MZ direction $(\Delta X, \Delta y, \Delta S)$ can be obtained by performing the following steps:

MZ construction:

- i) map (X, y, S) into the point $(\tilde{X}, \tilde{y}, \tilde{S}) \in \mathcal{S}_{++}^n \times \mathfrak{R}^m \times \mathcal{S}_{++}^n$ in the scaled space computed by using (20);
- ii) compute the Newton direction $(\widetilde{\Delta X}, \widetilde{\Delta y}, \widetilde{\Delta S})$ for the system $\mathcal{J}_I(\tilde{X}\tilde{S}) - \sigma\tilde{\mu}I = 0$, $\tilde{A}\tilde{X} - \tilde{b} = 0$ and $\tilde{A}^*\tilde{y} + \tilde{S} - \tilde{C} = 0$; and
- iii) map the direction $(\widetilde{\Delta X}, \widetilde{\Delta y}, \widetilde{\Delta S})$ back into the original space using (21) to obtain $(\Delta X, \Delta y, \Delta S)$.

A similar type of construction can be used to define a general direction of the MT family. The only difference with respect to the above construction is that in ii) the equation $\mathcal{J}_I(\tilde{X}\tilde{S}) = \sigma\tilde{\mu}I$ is replaced by the alternative central path equation $\tilde{X}^{1/2}\tilde{S}\tilde{X}^{1/2} - \sigma\tilde{\mu}I = 0$. (In [63], arbitrary nonsingular scaling matrices P can be chosen instead of just symmetric positive definite ones as here; however, as with the MZ family, it can be shown that the direction depends only on PP^T , so without loss of generality we restrict attention to \mathcal{S}_{++}^n .) To define the procedure more precisely, given a scaling matrix $P \in \mathcal{S}_{++}^n$, a point $(X, y, S) \in \mathcal{S}_{++}^n \times \mathfrak{R}^m \times \mathcal{S}_{++}^n$ and a parameter σ , the corresponding MT direction $(\Delta X, \Delta y, \Delta S)$ can be obtained by performing the following steps:

MT construction:

- i') same as i) as above;
- ii') compute the Newton direction $(\widetilde{\Delta X}, \widetilde{\Delta y}, \widetilde{\Delta S})$ for the system $\tilde{X}^{1/2}\tilde{S}\tilde{X}^{1/2} - \sigma\tilde{\mu}I = 0$, $\tilde{A}\tilde{X} - \tilde{b} = 0$ and $\tilde{A}^*\tilde{y} + \tilde{S} - \tilde{C} = 0$; and
- iii') same as iii) above.

We now discuss how to compute the Newton direction at step ii'). For any $A \in \mathcal{S}_{++}^n$, it can be shown that the operator $\Xi_A := A \odot I = I \odot A$ is positive definite, and hence has an inverse. The following result expresses the directional derivative of the square root function $X \rightarrow X^{1/2}$ in terms of the inverse of $\Xi_{X^{1/2}}$. (For its proof, see Lemma 2.2 of [63] for example).

Lemma 4.1 *Let $\theta : \mathcal{S}_{++}^n \rightarrow \mathcal{S}_{++}^n$ denote the square root function $\theta(X) = X^{1/2}$. Then, θ is an analytic function, and*

$$\theta'(X)H = \frac{1}{2} (\Xi_{X^{1/2}})^{-1} H, \quad \text{for every } X \in \mathcal{S}_{++}^n \text{ and } H \in \mathcal{S}^n,$$

where $\theta'(X)$ is the derivative of θ at X and $\theta'(X)H$ is the linear map $\theta'(X)$ evaluated at H .

Using Lemma 4.1, it is now easy to see that the Newton direction $(\widetilde{\Delta X}, \widetilde{\Delta y}, \widetilde{\Delta S})$ for the system in ii') is the solution of the following system of linear equations:

$$\tilde{A}^*\widetilde{\Delta y} + \widetilde{\Delta S} = \tilde{C} - \tilde{A}^*\tilde{y} - \tilde{S}, \quad (25a)$$

$$\tilde{A}\widetilde{\Delta X} = \tilde{b} - \tilde{A}\tilde{X}, \quad (25b)$$

$$\mathcal{J}_I \left(\Xi_{X^{1/2}}^{-1}(\widetilde{\Delta X})\tilde{S}\tilde{X}^{1/2} \right) + \tilde{X}^{1/2}\widetilde{\Delta S}\tilde{X}^{1/2} = \sigma\tilde{\mu}I - \tilde{X}^{1/2}\tilde{S}\tilde{X}^{1/2}. \quad (25c)$$

As with step ii) of the MZ construction, step ii') of the MT construction is not always well-defined in that the coefficient matrix of system (25) may be singular. The following result gives two conditions which guarantee that this coefficient matrix is nonsingular.

Proposition 4.2 (Theorems 2.4 and 6.2 of [63]) *Let $(X, y, S) \in \mathcal{S}_{++}^n \times \mathbb{R}^m \times \mathcal{S}_{++}^n$, $\sigma \in \mathbb{R}$ and $P \in \mathcal{S}_{++}^n$ be given and assume that one of the following conditions hold:*

- a) $d_\infty(X, S) = d_\infty(\tilde{X}, \tilde{S}) < \mu/\sqrt{2}$;
- b) $\tilde{X}^{1/2}\tilde{S} + \tilde{S}\tilde{X}^{1/2} = (PXP)^{1/2}(P^{-1}SP^{-1}) + (P^{-1}SP^{-1})(PXP)^{1/2} \succ 0$.

Then, the corresponding direction obtained from the MT construction is well-defined in the sense that system (25) has exactly one solution.

The algorithms considered in this subsection are all special cases of the following framework.

Framework MT-PD:

Same as Framework MZ-PD but with step (2) replaced by

- (2') Compute the MT direction $(\Delta X^k, \Delta y^k, \Delta S^k)$ according to the MT construction with $P = P^k$, $\sigma = \sigma_k$ and $(X, y, S) = (X^k, y^k, S^k)$.

End

Condition a) of Proposition 4.2 gives the foundation for the first two classes of algorithms that we consider, namely: short-step methods and semilong-step methods based on the MT family of directions. These two methods, while not imposing any condition on the choice of the scaling matrix P , require that the iterates be generated inside central-path neighborhoods which automatically enforce condition a) of Proposition 4.2, namely: $\mathcal{N}_F(\gamma)$ and $\mathcal{N}_\infty(\gamma)$, respectively, where $\gamma \in (0, 1/\sqrt{2})$.

The main results regarding the polynomial convergence of short-step methods and semilong-step methods based on the MT family are stated in the next two theorems. Their proofs can be found in Monteiro and Tsuchiya [63].

Theorem 4.3 (Corollary 4.2 of [63]) *Let $\gamma \in (0, 1/\sqrt{2})$ and $(X^0, y^0, S^0) \in \mathcal{N}_F(\gamma)$ be given. Then, the sequence of iterates $\{(X^k, y^k, S^k)\} \subset \mathcal{N}_F(\gamma)$ generated by Framework MT-PF with the short-step strategy satisfies $X^k \bullet S^k \leq (1 - \delta/\sqrt{n})^k (X^0 \bullet S^0)$, for every $\delta > 0$ satisfying*

$$\frac{40(\gamma^2 + \delta^2)}{(1 - \sqrt{2}\gamma)^2} \left(1 - \frac{\delta}{\sqrt{n}}\right)^{-1} \leq \gamma.$$

As a consequence, short-step methods based on the whole MT family of directions terminate in at most $\mathcal{O}(\sqrt{n}L)$ iterations.

Theorem 4.4 (Corollary 4.4 of [63]) *Let $\gamma \in (0, 1/\sqrt{2})$, $\bar{\sigma} \in (0, 1)$ and $(X^0, y^0, S^0) \in \mathcal{N}_\infty(\gamma)$ be given. Consider the class of semilong-step methods from the Framework MT-PD where the sequences $\{\sigma_k\}$ and $\{\alpha_k\}$ are chosen according to*

$$\begin{aligned} \sigma_k &:= \bar{\sigma}, \\ \alpha_k &:= \max \left\{ \alpha \in [0, 1] : (X^k, y^k, S^k) + \alpha(\Delta X^k, \Delta y^k, \Delta S^k) \in \mathcal{N}_\infty(\gamma) \right\}, \end{aligned}$$

Then, the sequence of iterates $\{(X^k, y^k, S^k)\} \subset \mathcal{N}_\infty(\gamma)$ generated by any algorithm of this class satisfies $X^k \bullet S^k \leq (1 - \bar{\eta})^k (X^0 \bullet S^0)$ for all $k \geq 0$, where

$$\bar{\eta} := \frac{\bar{\sigma}(1 - \bar{\sigma})\gamma(1 - \sqrt{2}\gamma)^2}{40n[\gamma^2 + (1 - \bar{\sigma})^2]}.$$

As a consequence, semilong-step methods based on the whole MT family of directions terminate in at most $\mathcal{O}(nL)$ iterations.

Note that the choice of the sequence $\{\alpha_k\}$ for the semilong-step method described in Theorem 4.4 is similar to that made for the long-step method except that $\mathcal{N}_\infty(\gamma)$ replaces the larger neighborhood $\mathcal{N}_{-\infty}(\gamma)$. An interesting open question is whether a polynomial iteration-complexity bound can be established for semilong-step methods based on the whole MZ family of directions.

Condition b) of Proposition 4.2 forms the foundation for the long-step algorithms. Given $(X, S) \in \mathcal{S}_{++}^n \times \mathcal{S}_{++}^n$, denote by $\mathcal{P}_0(X, S)$ the class of scaling matrices satisfying condition (b) of Proposition 4.2. We refer to the family of directions corresponding to these scaling matrices as the MT* family. The following convergence result applies to long-step methods based on the MT* family of directions.

Theorem 4.5 (Corollary 6.7 of [63]) *Let $\gamma, \sigma \in (0, 1)$ and $(X^0, y^0, S^0) \in \mathcal{N}_{-\infty}(\gamma)$ be given. Assume that the sequence of scaling matrices $\{P^k\} \subset \mathcal{S}_{++}^n$ satisfies $P^k \in \mathcal{P}_0(X^k, S^k)$ for all $k \geq 0$. Then the sequence of iterates $\{(X^k, y^k, S^k)\} \subset \mathcal{N}_{-\infty}(\gamma)$ generated by Framework MT-PF with the long-step strategy satisfies $X^k \bullet S^k \leq (1 - \bar{\eta})^k (X^0 \bullet S^0)$ for all $k \geq 0$, where*

$$\bar{\eta} := \frac{\bar{\sigma}(1 - \bar{\sigma})\gamma(1 - \gamma)^{1/2}}{30n^{3/2}[1 - 2\bar{\sigma} + \bar{\sigma}^2/(1 - \gamma)]}.$$

As a consequence, semilong-step methods based on the MT* family of directions terminate in at most $\mathcal{O}(n^{3/2}L)$ iterations.

We end this subsection by giving a few remarks. It is possible to derive a symmetric MT family based on the alternative central path equation $\tilde{S}^{1/2}\tilde{X}\tilde{S}^{1/2} - \sigma\tilde{\mu}I = 0$, obtained from the one used in this subsection by interchanging the role of \tilde{X} and \tilde{S} . It is easy to see that the symmetric MT family obtained by applying the Newton method to this equation plus the feasibility equations (in the scaled space) has similar properties to the one studied in this subsection and that it contains the NT direction and the HRVW/KSH/M direction.

We refer to the MT direction corresponding to $P = I$ as the $X^{1/2}SX^{1/2}$ -Newton direction. Like the AHO direction, this is a pure Newton direction in the sense that it is a Newton direction for a system of the form $\Phi(X, S) = \sigma\mu I$ where the map $\Phi(\cdot, \cdot)$ is independent of the current iterate or any parameter such as the scaling matrix P . On the basis of the results obtained so far, the $X^{1/2}SX^{1/2}$ -Newton direction has clear theoretical advantages over the AHO direction in the sense that polynomial convergence of the semilong-step path-following algorithm is only known for the former direction. So far this is the only pure Newton path-following algorithm which is polynomially convergent and is based on a central-path neighborhood wider than $\mathcal{N}_F(\cdot)$.

The MT* family also has theoretical advantages over the MZ* family based on the results so far. While for the MZ* family the iteration-complexity bound depends on a certain condition number associated with the sequence $\{P^k\}$ of scaling matrices, the corresponding bound for the MT* family does not depend on this sequence.

Finally, Monteiro and Zanjácomo [67] have reported promising computational results for algorithms based on the $X^{1/2}SX^{1/2}$ -Newton direction and two other pure Newton directions based on alternative central path equations, namely: $S^{1/2}XS^{1/2} = \sigma\mu I$ and $L_S^T X L_S = \sigma\mu I$, where L_S denotes the Cholesky lower triangular factor of S , that is $S = L_S L_S^T$ with L_S lower triangular.

4.3 Algorithms based on the KSH family

In this section we describe primal-dual path-following algorithms based on the KSH family of search directions which was introduced in Kojima, Shindoh and Hara [51].

In contrast to the MT and MZ families, whose directions are parametrized by a nonsingular scaling matrix P (in addition to a centrality parameter σ), the directions of the KSH family are parametrized by subspaces $H \subset \mathcal{S}_\perp^n \times \mathcal{S}_\perp^n$ having the property that $\dim(H) = n(n-1)/2$ and H is monotone, that is $U \bullet V \geq 0$ for all $(U, V) \in H$. In what follows, we denote this set of subspaces by \mathcal{H} . Given $(X, y, S) \in \mathcal{S}_{++}^n \times \mathfrak{R}^m \times \mathcal{S}_{++}^n$ and $H \in \mathcal{H}$, consider the following Newton-type system of linear equations

$$\mathcal{A}\Delta X = b - \mathcal{A}X, \quad \mathcal{A}^*\Delta y + \Delta S = C - \mathcal{A}^*y - S, \quad (26a)$$

$$X(\Delta S + \Delta S') + (\Delta X + \Delta X')S = \sigma\mu I - XS, \quad (26b)$$

$$(\Delta X, \Delta S) \in \mathcal{S}^n \times \mathcal{S}^n, \quad (\Delta X', \Delta S') \in H. \quad (26c)$$

It has been shown in Theorem 4.2 of Kojima et al. [51] that, for every $H \in \mathcal{H}$, system (26) always has a unique solution $(\Delta X, \Delta y, \Delta S, \Delta X', \Delta S')$ (see also Theorem 2.2 of Monteiro and Tsuchiya [64] for a simpler proof). The component $(\Delta X, \Delta y, \Delta S)$ of this solution is then the KSH search direction at (X, y, S) corresponding to H and σ .

Kojima [45] (see Lemma 2.1 of Monteiro [58]) has shown that the KSH search directions corresponding to the subspaces $H = \mathcal{S}_\perp^n \times \{0\}$ and $H = \{0\} \times \mathcal{S}_\perp^n$ are equal to the MZ directions with $P = S^{1/2}$ and $P = X^{-1/2}$, respectively (that is, the HRVW/KSH/M direction and its dual direction, respectively). Also, it is shown in Kojima, Shida and Shindoh [48] that the NT direction is a member of the KSH family (the corresponding H is not so simple to describe). On the other hand, the AHO direction is not in the KSH family since, as opposed to the directions of the KSH family, the AHO direction does not necessarily exist at every point $(X, y, S) \in \mathcal{S}_{++}^n \times \mathfrak{R}^m \times \mathcal{S}_{++}^n$ (see the discussion before Theorem 3.2 of Todd et al. [96]).

The HRVW/KSH/M direction can be equivalently obtained by first solving the Newton system

$$\mathcal{A}^*\widehat{\Delta y} + \widehat{\Delta S} = C - \mathcal{A}^*y - S, \quad \mathcal{A}\widehat{\Delta X} = b - \mathcal{A}X, \quad (27)$$

$$X\widehat{\Delta S} + \widehat{\Delta X}S = \sigma\mu I - XS, \quad (28)$$

with the restriction that $(\widehat{\Delta X}, \widehat{\Delta y}, \widehat{\Delta S}) \in \mathfrak{R}^{n \times n} \times \mathfrak{R}^m \times \mathcal{S}^n$. It is easily seen that this system has a unique solution and that $(\Delta X, \Delta y, \Delta S) := ((\widehat{\Delta X} + \widehat{\Delta X}^T)/2, \widehat{\Delta y}, \widehat{\Delta S})$ is the direction of the KSH family corresponding to $H = \mathcal{S}_\perp^n \times \{0\}$, and hence the HRVW/KSH/M direction according to the discussion in the previous paragraph.

We are now ready to state a general framework based on the KSH family.

Framework KSH-PD:

Same as Framework MZ-PD but with steps (1) and (2) replaced by

- (1') Choose a linear subspace $H_k \in \mathcal{H}$ and a centrality parameter $\sigma_k \in [0, 1]$;
- (2') Compute the KSK direction $(\Delta X^k, \Delta y^k, \Delta S^k)$ at (X^k, y^k, S^k) corresponding to $H = H_k$ and $\sigma = \sigma_k$.

End

Polynomial convergence results have been obtained for some variants of the above framework. By taking $H_k = \mathcal{S}_\perp^n \times \{0\}$ or $H_k = \{0\} \times \mathcal{S}_\perp^n$ for all k , that is by selecting the search directions to be either the HRVW/KSH/M direction or its dual direction at every iteration, Kojima et al. [51] establish that the short-step path-following method converges in $\mathcal{O}(\sqrt{n}L)$ iterations. (An alternative derivation of this result was also given in Monteiro [58] by viewing these directions as members of the MZ family.) This result was then generalized by Monteiro and Tsuchiya [64], who establish the same iteration-complexity bound for a short-step path-following method and a predictor-corrector method with no

restriction on the subspaces H_k 's. Kojima et al. [51] have also established the polynomial convergence of a primal-dual potential-reduction algorithm with no restriction on the subspaces H_k 's.

Finally, we observe that it is an open question whether it is possible to establish the polynomial convergence of path-following algorithms based on neighborhoods other than the Frobenius neighborhood $\mathcal{N}_F(\cdot)$ and whose search directions are from the KSH family with no restriction on the H_k 's.

5 First- and second-order methods for large-scale SDPs

In this section, we present methods which are particularly suitable for the solution of large-scale SDP problems.

The methods of Subsections 5.1 and 5.2 are based on replacing the matrix variable of one of the problems (P) or (D), i.e. either the variables X or S , by a factorization of the form VV^T where V is a (possibly specially structured) matrix with n rows. In both cases, the resulting nonlinear optimization problems are then solved by means of first-order nonlinear programming methods that can suitably handle problems with large number of variables (e.g., limited memory quasi-Newton methods).

The method of Subsection 5.1 applies the above change of variables to the primal SDP problem (P) and restricts V to the set of matrices $\mathfrak{R}^{n \times r}$ where r is a positive integer number much smaller than n . The resulting formulation is then a nonlinear programming with much smaller number of variables than the original problem (P). The method of Subsection 5.2 applies the above change of variables to (in fact, a special version of) the dual SDP problem (D) and restricts V to the set of lower triangular matrices with all positive diagonal elements. In addition, it considerably reduces the number of variables by eliminating variables, i.e. expressing some variables as functions of others. The resulting formulation takes a particularly simple form, namely that of minimizing a nonlinear function over an orthant of the form $\mathfrak{R}^m \times \mathfrak{R}_{++}^n$, where m and n are as before.

Finally, Subsection 5.3 briefly describes the spectral bundle method for SDP while Subsection 5.4 describes second-order methods that have been proposed to take advantage of data sparsity and hence handle large-scale SDPs more effectively.

5.1 Low-rank factorization methods

In this subsection, we briefly discuss first-order SDP methods based on solving a reformulation of the SDP problem (P) that results after performing the change of variables $X = RR^T$ where $R \in \mathfrak{R}^{n \times r}$ and $r \in \{1, \dots, n\}$. Since the resulting reformulation is nonconvex, no global convergence analysis is currently available for these methods. In practice, however, these first-order methods are able to handle several classes of SDP problems extremely fast and always generate a sequence of iterates which converge to the optimal solution set of problem (P).

The straightforward idea underlying the method in this subsection and the next one is that a matrix X is in \mathcal{S}_+^n if and only if $X = VV^T$ for some matrix $V \in \mathfrak{R}^{n \times n}$. Applying this observation to problem (P) yields the following equivalent nonconvex programming formulation

$$\min \{ C \bullet (VV^T) : \mathcal{A}(VV^T) = b, V \in \mathfrak{R}^{n \times n} \}, \quad (29)$$

which can be solved by a variety of nonlinear optimization methods. Even though (29) is nonconvex, it can be shown that its set of local minima coincide with its set of global minima, although its set of stationary points is generally larger than its set of global minima. A first-order algorithm based on the above formulation was developed by Homer and Peinado in [42]. Another formulation related to (29) is obtained by assuming that the matrix V is lower triangular, and a first-order algorithm based on this second formulation was developed by Burer and Monteiro [15]. Both algorithms of [42, 15] have

been implemented only to solve the SDP relaxation of the maximum cut problem, for which feasibility of the sequence of iterates is easy to enforce.

Another variation based on the ideas above has been pursued by Burer and Monteiro [16]. Instead of the above factorization, they consider factorizations of X of the form $X = RR^T$, where $R \in \mathfrak{R}^{n \times r}$ and $r < n$. Restricting X in (1) to matrices of the above form, i.e., of rank less than or equal to r , yields the following formulation

$$\min \{ C \bullet (RR^T) : \mathcal{A}(RR^T) = b, R \in \mathfrak{R}^{n \times r} \}. \quad (30)$$

The advantage of the resulting formulation is that it has relatively few variables when r is small. The choice of r must be carefully made so as to guarantee that optimal solutions R^* of (30) yield optimal solutions $R^*(R^*)^T$ of (1). In this respect, the following result due to Barvinok [5] and Pataki [81] is useful.

Proposition 5.1 *If the primal SDP problem (P) has an optimal solution then it has an optimal solution X^* whose rank r^* satisfies the inequality $r^*(r^* + 1) \leq 2m$.*

Using this result, it is easy to see that if $r \geq \lfloor \sqrt{2m} \rfloor$ and R^* is an optimal solution of (30) then $R^*(R^*)^T$ is an optimal solution of (1). Burer and Monteiro [16] have then used a slightly modified version of the augmented Lagrangian method (see Section 12.2 of [29]) to solve (30) with r as indicated and have obtained very good computational results. Despite the fact that, like (29), the new formulation is nonconvex, extensive computational experiments have shown that the method of [16] always converges to the exact optimal value. However, in contrast to (29), it is still an open problem to establish under the assumption that $r \geq \lfloor \sqrt{2m} \rfloor$ whether every local minimum of (30) is actually a global minimum of (30).

Despite the lack of a global convergence proof, computational experiments have shown that the above method substantially outperforms other available algorithms (e.g., second-order IP methods [31, 9] and the spectral bundle method [40]) when solving certain classes of large-scale SDP problems. For example, the method can solve maximum cut semidefinite relaxations of graphs containing 20,000 nodes in less than 10 minutes. On the other hand, the algorithms of Subsections 5.2 and 5.3 take more than 60 hours to solve these problems, and second-order IP methods are not able to perform one iteration due to their large time and memory requirements on such problems. The above method has been implemented for several structured subclasses of SDPs by Burer and Monteiro in a code called SDP-LR, which can be download freely from the website <http://www.isye.gatech.edu/~monteiro/software>.

Finally, Vanderbei and Y. Benson [104] have developed a method in which the primal variable X is factored as $L_X \text{Diag}(d(X))L_X$, where L_X is unit lower triangular and $d(X) \in \mathfrak{R}^n$, and the constraint $X \succeq 0$ is enforced by adding the nonlinear constraint $d(X) \geq 0$. They show that this last constraint is a convex type constraint in the sense that $d(X)$ is a concave function.

5.2 A dual logarithmic barrier method

In this subsection, we discuss a log-barrier type method based on solving a sequence of reformulations of problem (D_ν) . These reformulations are obtained by performing the changing of variables $S = LL^T$ in problem (D_ν) , where L is the Cholesky lower triangular factor of S , and some variables are subsequently expressed as functions of the others.

Although the method can be described for the general dual problem (2), its most simple description is obtained for the following special form of (2):

$$(D') \quad \max \left\{ d^T z + b^T y : \text{Diag}(z) + \mathcal{A}^* y + S = C, S \succeq 0 \right\}, \quad (31)$$

where $d \in \Re^n$ and $\text{Diag}(z)$ is the diagonal matrix with the components of z along its diagonal. Then, strict feasibility for this problem can be expressed as

$$C - \text{Diag}(z) - \mathcal{A}^*y = \left(\text{Diag}(w) + \tilde{L} \right) \left(\text{Diag}(w) + \tilde{L} \right)^T, \quad (32)$$

where $w \in \Re_{++}^n$ and $\tilde{L} \in \Re^{n \times n}$ is strictly lower triangular. Note that we have expressed S as the product LL^T where $L = \text{Diag}(w) + \tilde{L}$ is its Cholesky lower triangular factor. A key idea of the first-order method is then to use (32) to express (z, \tilde{L}) in terms of (w, y) . In this respect, we have the following result.

Theorem 5.1 (Theorem 4 of [18]) *The following statements hold:*

- a) for each $(w, y) \in \Re_{++}^n \times \Re^m$, there exists a unique pair (\tilde{L}, z) , where \tilde{L} is a strictly lower triangular $n \times n$ matrix and $z \in \Re^n$, satisfying (32);
- b) the functions $\tilde{L}(w, y)$ and $z(w, y)$ defined according to (32) are each infinitely differentiable on their domain $\Re_{++}^n \times \Re^m$;
- c) $\Re_{++}^n \times \Re^m$ and the set of strictly feasible solutions of (31) are in bijective correspondence according to the assignment $(w, y) \mapsto (z, y, S)$ where $z \equiv z(w, y)$, $S \equiv LL^T$ and $L \equiv \text{Diag}(w) + \tilde{L}(w, y)$.

Hence, the problem of maximizing $d^T z + b^T y$ subject to the strict feasibility of (z, y, S) is equivalent to the simple problem

$$(NLP) \quad \inf \left\{ f(w, y) \equiv -d^T z(w, y) - b^T y : w > 0 \right\}. \quad (33)$$

The derivatives of the objective function $f(w, y)$ can be effectively computed by means of the following result which also introduces a primal estimate associated with the point $(w, y) \in \Re_{++}^n \times \Re^m$.

Proposition 5.2 (Proposition 7 of [18]) *Let $(w, y) \in \Re_{++}^n \times \Re^m$ be given and define $L \equiv \text{Diag}(w) + \tilde{L}(w, y)$. Then, the system of linear equations*

$$\text{diag}(X) = d, \quad (XL)_{ij} = 0, \quad i > j, \quad X \in \mathcal{S}^n \quad (34)$$

has a unique solution in \mathcal{S}^n , which we will denote by $X(w, y)$. Moreover:

- a) $\nabla_w f(w, y) = 2 \text{diag}(XL)$;
- b) $\nabla_y f(w, y) = \mathcal{A}(X) - b$.

Note that the dual of (31) is the SDP

$$(P') \quad \min \{ C \bullet X : \text{diag}(X) = d; \mathcal{A}(X) = b; X \succeq 0 \}. \quad (35)$$

By (34), the *primal estimate* $X = X(w, y)$ always satisfies the first constraint of (35). Moreover, by Proposition 5.2(b), $X(w, y)$ satisfies the second constraint if and only if $\nabla_y f(w, y) = 0$. The next result gives condition under which the third constraint of (35) is satisfied.

Proposition 5.3 (Lemma 5 of [18]) *Let $(w, y) \in \Re_{++}^n \times \Re^m$ be given and define $L \equiv \text{Diag}(w) + \tilde{L}(w, y)$, $S = LL^T$, $X \equiv X(w, y)$, and $\nabla_w f \equiv \nabla_w f(w, y)$. Then,*

- a) $X \succeq 0$ if and only if $\nabla_w f \geq 0$; in addition, $X \succ 0$ if and only if $\nabla_w f > 0$;

b) $w \circ \nabla_w f(w, y) = 2 \operatorname{diag}(L^T X L)$, hence $w^T \nabla_w f(w, y) = 2 \operatorname{trace}(L^T X L) = 2 X \bullet S$, where “ \circ ” denotes the Hadamard (component-wise) product.

We now discuss how problem (NLP) indirectly yields a solution for the pair of SDPs (P') and (D'). Given a sequence of points $\{(w^k, y^k)\}_{k \geq 0} \subset \mathfrak{R}_{++}^n \times \mathfrak{R}^m$, we define $L^k \equiv \operatorname{Diag}(w^k) + \tilde{L}(w^k, y^k)$, $S^k \equiv L^k (L^k)^T$, $X^k \equiv X(w^k, y^k)$, $z^k \equiv z(w^k, y^k)$, $\nabla_w f^k \equiv \nabla_w f(w^k, y^k)$ and $\nabla_y f^k \equiv \nabla_y f(w^k, y^k)$ for all $k \geq 0$. The following result gives sufficient conditions for the accumulation points of the sequences $\{X^k\}$ and $\{(z^k, y^k, S^k)\}$ to be optimal solutions of problems (P') and (D'), respectively.

Theorem 5.2 (Theorem 6 of [18]) *Let $\{(w^k, y^k)\}_{k \geq 0} \subset \mathfrak{R}_{++}^n \times \mathfrak{R}^m$ be a sequence of points such that $\nabla_w f^k \geq 0$ for all $k \geq 0$, the sequence $\{(w^k)^T \nabla_w f^k\}$ is bounded, and $\lim_{k \rightarrow \infty} \nabla_y f^k = 0$. Then,*

- (a) *the sequences $\{X^k\}$, $\{(z^k, y^k, S^k)\}$, $\{L^k\}$, $\{w^k\}$ and $\{\nabla_w f^k\}$ are all bounded;*
- (b) *if in addition $\lim_{k \rightarrow \infty} (w^k)^T \nabla_w f^k = 0$, then any accumulation points of $\{X^k\}$ and $\{(z^k, y^k, S^k)\}$ are optimal solutions of (P') and (D'), respectively.*

The first-order algorithm of this subsection generates a sequence satisfying the conditions of the above theorem and therefore it asymptotically approaches a solution of the pair of SDPs (P') and (D'). The basis of this algorithm is the following log-barrier subproblem given by

$$(NLP_\nu) \quad \min \left\{ f(w, y) - 2\nu \sum_{i=1}^n \log w_i : (w, y) \in \mathfrak{R}_{++}^n \times \mathfrak{R}^m \right\},$$

where $\nu > 0$ is a positive parameter. This subproblem will be successively solved for a sequence of parameters $\{\nu_k\}$ converging to zero. The following result establishes the connection of this log-barrier subproblem with the central path of the SDP problems (P') and (D').

Proposition 5.4 (Theorem 7 of [18]) *For each $\nu > 0$, (NLP_ν) has a unique minimum point (w_ν, y_ν) , which is also its unique stationary point. Moreover, if we define $z_\nu \equiv z(w_\nu, y_\nu)$, $L_\nu \equiv \operatorname{Diag}(w_\nu) + \tilde{L}(w_\nu, y_\nu)$, $S_\nu \equiv L_\nu L_\nu^T$, $X_\nu \equiv X(w_\nu, y_\nu)$, then the point $(X_\nu, z_\nu, y_\nu, S_\nu)$ is the point on the central path corresponding to the parameter ν for the pair of SDPs (P') and (D').*

Since (w_ν, y_ν) is a stationary point of (NLP_ν), it satisfies the conditions

$$\nabla_w f \circ w = 2\nu e, \quad \nabla_y f = 0, \quad (36)$$

where $e \in \mathfrak{R}^n$ is the vector of all ones. Thus, the sequence $\{(w^k, y^k)\}$ defined as $(w^k, y^k) \equiv (w_{\nu_k}, y_{\nu_k})$ for all k , where $\{\nu_k\} \subset \mathfrak{R}_{++}$ is a sequence converging to zero, can be easily seen to satisfy the hypothesis of Theorem 5.2, and hence its accumulation points yield optimal solutions for the pair of SDPs (P') and (D').

It turns out that it suffices to solve (NLP_ν) only approximately. Indeed, let constants $\gamma_1 \in [0, 1)$, $\gamma_2 > 1$, and $\Gamma > 0$ be given, and, for each $\nu > 0$, define

$$\mathcal{N}(\nu) \equiv \{(w, y) \in \mathfrak{R}_{++}^n \times \mathfrak{R}^m : 2\gamma_1 \nu e \leq w \circ \nabla_w f(w, y) \leq 2\gamma_2 \nu e, \|\nabla_y f(w, y)\| \leq \Gamma \nu\}. \quad (37)$$

Note that each $(w, y) \in \mathcal{N}(\nu)$ satisfies $\nabla_w f(w, y) > 0$ and that the unique minimizer (w_ν, y_ν) of (NLP_ν) is in $\mathcal{N}(\nu)$, as can be seen from equation (36). We are now ready to state the log-barrier algorithm for solving the pair of SDPs (P') and (D').

Log-Barrier Algorithm:

Let $\sigma \in (0, 1)$ and $\nu_0 > 0$ be given, and set $k = 0$.

For $k = 0, 1, 2, \dots$

1. Use a minimization method to solve (NLP_{ν_k}) approximately, obtaining a point $(w^k, y^k) \in \mathcal{N}(\nu_k)$.
2. Set $\nu_{k+1} = \sigma\nu_k$, increment k by 1, and return to step 1.

End

We stress that since (NLP_{ν_k}) has a unique stationary point (w_{ν_k}, y_{ν_k}) for all $\nu_k > 0$ which is also the unique minimum, step 1 of the algorithm will succeed using any reasonable unconstrained minimization method. Specifically, any convergent, gradient-based method will eventually produce a point in the set $\mathcal{N}(\nu_k)$.

As shown in [17], slight modifications to the above log-barrier method allow one to solve the general pair of SDP problems (1) and (2). In particular, under the assumption that $d < 0$, it can be shown that any optimal solution (z^*, y^*, S^*) of the SDP problem obtained by adding the constraint $z \geq 0$ to (31) has the property that $z^* = 0$ and (y^*, S^*) is an optimal solution of (2). This modified SDP problem can be transformed, according to the ideas of the previous paragraph, to a problem like (33) with the additional constraint that $z(w, y) > 0$, which can easily be handled by log-barrier type methods. One disadvantage of this modified approach, however, is that knowledge of a strictly feasible solution for (2) is needed in order to have an immediately available initial feasible solution for the transformed problem.

In [19], Burer et al. discuss an efficient implementation of the above log-barrier algorithm and report computational results on semidefinite relaxations of three types of combinatorial optimization problems, namely the maximum cut, the maximum stable set, and the frequency assignment SDP relaxations. The computational results demonstrate that the algorithm is very efficient for solving large-scale SDPs and is particularly effective for problems with a large number of primal constraints, such as the maximum stable set and the frequency assignment SDP relaxations.

The nonlinear programming approach described in this subsection and the previous might at first seem a little strange since it takes a convex programming problem, namely the SDP problem (31), and transforms it into the nonconvex problem (33). However, the approach described in this subsection preserves a lot of the nice properties inherent in a convex programming problem. Indeed, Proposition 5.4 claims that even though the subproblem (NLP_{ν}) is nonconvex, it has a unique stationary point which is also its unique global minimum. Hence, this global minimum can be easily approached by using standard first-order methods for unconstrained nonlinear programming. Moreover, it can be proved that the Hessian of $f(w, y)$ is positive definite at any point $(w, y) \in \mathfrak{R}_{++}^n \times \mathfrak{R}^m$ such that $\nabla_w f(w, y) \succ 0$, and hence at any point in the neighborhood $\cup_{\nu > 0} \mathcal{N}(\nu)$ of the central path of (33), that is the set of minimizers $\{(w_{\nu}, y_{\nu}) : \nu > 0\}$ (see Proposition 5.3(a) above and Theorem 2.4 of Burer et al. [17]). Motivated by this last observation, Burer et al. (see Section 4 of [17]) develop a second-order potential-reduction algorithm for the formulation (33).

5.3 The spectral bundle method for SDP

In this subsection, we briefly discuss a method for solving large-scale SDPs based on the bundle method for nonsmooth convex programming that have been proposed by Helmberg and Rendl [40].

Consider first the following special form of (1):

$$(P'') \quad \min \{C \bullet X : \mathcal{A}(X) = b, \text{trace}(X) = 1, X \succeq 0\}.$$

Note that, as opposed to (1), (P'') requires that the trace of X be equal to one, and hence it is usually

referred to as a constant trace problem. Its dual is the SDP:

$$(D'') \quad \max \left\{ b^T y + \lambda : C \succeq \mathcal{A}^* y + \lambda I \right\}.$$

Noting that (y, λ) is feasible to (D'') if and only if $-\lambda \geq \lambda_{\max}(\mathcal{A}^* y - C)$ and switching the minimization to maximization by multiplying the objective function by minus one, we see that (D'') is equivalent to

$$\min \left\{ g(y) := -b^T y + \lambda_{\max}(\mathcal{A}^* y - C) : y \in \mathbb{R}^m \right\}.$$

This is a convex but nonsmooth unconstrained problem, to which standard nonsmooth methods for convex programming can be applied. Helmberg and Rendl [40] develops a variant of the bundle method, which they referred to as the spectral bundle method, specially tailored to take advantage of the special form of the objective function. The basic idea behind the bundle method is to build a cutting plane model of the objective function by computing subgradients of the objective function at several iterates of the algorithm. We refer the reader to [40] for further details on this algorithm.

Rendl and Helmberg [40] also shows how an SDP problem of the form of (1) with a bounded feasible region can be put into the form (P'') by a sequence of elementary transformations. They have also obtained very good computational results with their spectral bundle in the solution of large-scale SDP problems which can not be handled by the methods of Section 4 due to their size. Finally, other variants of the above bundle method which incorporates second-order information have been developed by Oustry [78, 79] and Helmberg and Oustry [39].

5.4 Second-order methods for large scale SDPs

In this subsection, we briefly survey second-order methods that have been proposed in the literature for handling large-scale SDP problems.

The first second-order methods proposed for solving large-scale SDPs work directly in the space of y of the dual problem (2) in order to exploit sparsity in the data. More precisely, consider the following barrier subproblem

$$\max_y \left\{ h(y) := b^T y + \nu \log \det(C - \mathcal{A}^* y) : C - \mathcal{A}^* y \succ 0 \right\},$$

which is essentially problem (D_ν) without the dual slack S . The basic idea exploited by these methods is that if the matrices C and A_i 's are all sparse, then the dual slack $C - \mathcal{A}^* y$ will be sparse and hence evaluation of the function $h(y)$ and its first and second order derivatives will be relatively fast. Even though first derivatives of $h(y)$ are relatively easy to compute (see for example Vavasis [105]), at present there is no successful implementation of a first order method based on the above subproblem. On the other hand, a second order method based on the above subproblem has been developed by Benson, Ye and Zhang [9]. The search direction of their algorithm is the Newton direction corresponding to equation (16). Hence, their method is closer to the ones described in the previous section, the main difference being that their algorithm is a dual-only instead of a primal-dual one. Even though their method can handle larger problems than the the primal-dual algorithms discussed in Section 4, it is still not able to handle many of the large-scale problems that the algorithms of Subsections 5.1, 5.2 and 5.3 can solve. (As a side remark, we mention that Fukuda and Kojima [32] have proposed a first-order interior-point method which also works in the space of y only in order to take advantage of sparsity of the dual slack $C - \mathcal{A}^* y$.)

The design of the first primal-dual IP algorithm which effectively exploits data sparsity in both the primal and dual space is due to Fukuda et al. [33]. The key idea used in their algorithm is that of a matrix completion. Note that the primal variable X is always dense, but that only the entries

of X corresponding to the nonzero entries of $S = C - \mathcal{A}^*y$ are really needed to evaluate the primal objective function and primal constraints. The other entries of X are only needed to ensure that X is positive definite. In view of these observations, Fukuda et al. noted that it is sufficient to work with only a partial symmetric matrix \hat{X} , i.e., a symmetric matrix in which only a part of the entries are specified. Let $F \subset \{1, \dots, n\} \times \{1, \dots, n\}$ be the entries in which the matrix is specified, and assume that F is symmetric in the sense that $(i, j) \in F$ if and only if $(j, i) \in F$. A matrix completion X of \hat{X} is a matrix $X \in \mathcal{S}^n$ such that $X_{ij} = \hat{X}_{ij}$ for all $(i, j) \in F$. A positive definite matrix completion of \hat{X} is a matrix completion X of \hat{X} such that X is positive definite. The maximum determinant matrix completion of \hat{X} is the unique optimal solution of the following optimization problem:

$$\max\{\det X : X \text{ is a positive definite matrix completion of } \hat{X}\}.$$

Let us denote this solution by $X(\hat{X})$, that is, as a function of \hat{X} .

A key observation is that $X = X(\hat{X})$ is characterized by the property that $(X^{-1})_{ij} = 0$ for all $(i, j) \notin F$. Hence, the inverse of $X(\hat{X})$ is a sparse matrix if the size of F is relatively small. In their algorithm, the set F is chosen to be the chordal extension of the aggregate sparsity E of the SDP data, which is defined as follows: $(i, j) \in E$ if and only if the (i, j) -th entry of at least one of the matrices C and A_j , $j = 1, \dots, m$, is nonzero. Without giving the formal definition of the chordal extension F of E (see instead [33]), we mention that it has the property that the inverse of $X = X(\hat{X})$ admits a Cholesky factorization $X^{-1} = LDL^T$, where L is a unit lower triangular matrix whose nonzero entries all lie in F . An efficient scheme to compute this factorization is developed in [33]. This paper also describes a potential-reduction algorithm based on the potential function

$$\rho \log(\hat{X} \bullet S) - \log \det X(\hat{X}) - \log \det S,$$

where $\rho > n$ is a fixed constant. An iterate of this algorithm consists of a triple (\hat{X}, y, S) where \hat{X} is a partial symmetric matrix with all partial entries in F . The HRVW/KSH/M direction is then computed at the point $(X(\hat{X}), y, S)$, but only the entries of ΔX lying in F are computed to obtain a partial primal search direction $\widehat{\Delta X}$, in addition to the usual dual component $(\Delta y, \Delta S)$. In this process, the dense matrix $X = X(\hat{X})$ is never computed. Instead, the factorization $X^{-1} = LDL^T$ is used whenever one needs to operate with the matrix X . The next point is then set to $(\hat{X}, y, S) + \alpha(\widehat{\Delta X}, \Delta y, \Delta S)$, where $\alpha > 0$ is a suitable stepsize which guarantees that the above potential function is sufficiently decreased. Polynomial convergence of this primal-dual potential-reduction algorithm is established in [33] and computational results of their algorithm are reported in [69].

Burer [12] on the other hand describes a primal-dual path-following IP algorithm based on the following central path equation

$$L_S = \sqrt{\nu} \left[L(\hat{X}) \right]^{-1},$$

where L_S is the Cholesky lower triangular factor of S and $L(\hat{X})$ is the Cholesky lower triangular factor of the inverse of $X(\hat{X})$. Since this central path system is already square (i.e., its number of equations is equal to its number of variables), the corresponding Newton direction can be computed to yield a direction $(\widehat{\Delta X}, \Delta y, \Delta S)$, where $\widehat{\Delta X}$ is a partial direction with all partial entries lying in F . Using the fact that the Newton direction is well-defined for every well-centered point, Burer [12] then develops a primal-dual short-step algorithm with an $\mathcal{O}(nL)$ iteration-complexity bound.

All the second-order methods discussed in Sections 3 and 4 as well as the ones above can be implemented by computing the Newton direction using the (preconditioned) conjugate gradient or residual method. From now on, we will refer to the resulting methods as *iterative second-order methods*. Implementations of iterative second-order methods can be found in Burer [13], Choi and Ye [20], Lin and Saigal [53], Nakata et al. [70], Toh [98] and Toh and Kojima [99]. In particular, Burer [13] and

Toh [98] have been able to solve some large-scale SDP problems very successfully. The development of efficient iterative second-order methods is a research area that certainly deserves much more attention since they have the potential to supersede first-order methods in the solution of large-scale SDPs. However, at present, it is too early to make definitive comparisons about these two classes of algorithms.

6 Computational complexity per iteration

In this section, we describe the number of flops per iteration that the following two classes of methods need to perform:

- i) direct second-order methods, i.e. the ones based on direct matrix factorizations;
- ii) first-order methods (more specifically, the methods of Subsections 5.1 and 5.2);
- iii) iterative second-order methods.

Our goal is to give a rough idea of how these methods compare to one another for both dense and sparse problems.

We first consider the methods of class i). Assume first that the matrices C and A_i , $i = 1, \dots, m$, are dense. These methods all have computational complexity per iteration that can be expressed as

$$\beta m^2 n^2 + \gamma m n^3 + \xi m^3 + \mathcal{O}(m n^2 + m^2 n + n^3), \quad (38)$$

where $\beta, \gamma, \xi > 0$ are (small) constants depending on the search direction used. Section 3.7 of Monteiro and Zanjácomo [67] lists the best known values of these constants for six primal-dual search directions, including the AHO, HRVW/KSH/M and NT directions. Note that the first two terms in (38) have degree four while the other ones all have degree 3. Note also that among all the third degree terms, only one was written outside the $\mathcal{O}(\cdot)$ function, the reason being that this term, namely ξm^3 , can become of the same order of magnitude as the first two terms in (38) when the number m of constraints of the SDP is $\Theta(n^2)$. (This situation occurs often for SDPs that arise as relaxations of combinatorial and/or graph problems.) Note that for problems in which $m = \Theta(n^2)$, the overall complexity becomes $\Theta(n^6)$, which shows that the methods of class i) become prohibitively expensive for such problems.

We will now roughly describe the parts of a general second-order method that contribute to the first three terms in (38). The search direction $(\Delta X, \Delta y, \Delta S)$ of an arbitrary second-order method is the solution of a system of the form (10), where the operators \mathcal{E} and \mathcal{F} depend on the search direction used. Theorem 2.2 claims that Δy satisfies $(\mathcal{A}\mathcal{E}^{-1}\mathcal{F}\mathcal{A}^*)\Delta y = u$ for some $u \in \Re^m$. In terms of the matrices A_i 's, this means that Δy is a solution of the linear system $B\Delta y = u$, where the (i, j) -th entry of the matrix $B \in \Re^{m \times m}$ is given by

$$B_{ij} = A_i \bullet \mathcal{E}^{-1}\mathcal{F}(A_j). \quad (39)$$

The second term in (38) is due to the computation of the m matrices $V_j \equiv \mathcal{E}^{-1}\mathcal{F}(A_j)$, $j = 1, \dots, m$, while the first one is due to the calculation of the m^2 (or possibly, only $m(m+1)/2$ if B is symmetric) inner products $A_i \bullet V_j$ in (39). The solution of $B\Delta y = u$ via direct factorization methods then contributes with the third term to the bound (38).

When the SDP data is sparse, the first and second terms in (38) can be replaced by (much) smaller terms but the third term can not be reduced in view of the fact that B always results in a dense matrix. In conclusion, second-order methods based on direct factorization are also prohibitively expensive for sparse SDP problems for which m is large. However, these methods can be quite effective in the solution of SDP problems for which both m and n are relatively small.

We now consider the methods of class ii). For a given matrix P , let $\text{nz}(P)$ denote the number of nonzero entries of P . An iteration of the low-rank method of Subsection 5.1 has a computational complexity bound of $\mathcal{O}(r \text{nz}(S) + \text{nz}(C) + \sum_{i=1}^m \text{nz}(A_i))$ flops, and hence of $\mathcal{O}((r+m)n^2)$ flops when the SDP data is dense (see Propositions 3.2 and 3.3 of [16]). On the other hand, an iteration of the log-barrier method of Subsection 5.2 has a computational complexity bound of $\mathcal{O}(n \text{nz}(L) + \text{nz}(C) + \sum_{i=1}^m \text{nz}(A_i))$ flops (and hence, $\mathcal{O}((n+m)n^2)$ flops when the SDP data is dense), where L is the Cholesky lower triangular factor of S (see Subsection 5.3 of [18]). Comparing the bounds for the sparse and dense cases, one can immediately see how the presence of sparsity in the SDP data dramatically affects the complexity bounds for both the low-rank and log-barrier methods. Observe that the bound for the low-rank method is generally much better than the one for the log-barrier method since $\text{nz}(S)$ (resp., r) is usually much smaller than $\text{nz}(S)$ (resp., n). Moreover, since the bound for the log-barrier algorithm depends on $\text{nz}(L)$, a symmetric permutation of the rows and columns of S (determined once at the start of the algorithm) is performed so as to keep the “fill-in” of L , that is the quantity $\text{nz}(L) - \text{nz}(S)$, relatively small.

In conclusion, the number of flops per iteration performed by first-order methods is generally much smaller than that of second-order methods. However, first-order methods generally take many (from few hundreds to few thousands) iterations to converge and tend to have difficulty to obtain highly accurate nearly optimal solutions. On the other hand, second-order methods converge quickly (in a few tens iterations) to highly accurate solutions. Overall, first-order methods are much superior to second-order methods based on direct factorization in the solution of large-scale SDP problems. The major drawback of the latter methods is that finding the solution of the linear system $B\Delta y = u$ may be prohibitively expensive (sometimes impossible) not only due to the high number of flops required but also to the high amount of memory space needed to store B .

Finally, we briefly discuss the computational complexity of iterative second-order methods. First, they do not need to compute and/or store the matrix B . Second, the amount of work and storage requirement of an inner iteration (i.e., an iteration of the iterative method used to solve the system $B\Delta y = u$) of these methods are very comparable to that of an iteration of the log-barrier method. Third, the total number of inner iterations performed by an iterative second-order method depends crucially on the choice of preconditioner used in the iterative algorithm for solving the linear systems $B\Delta y = u$. In conclusion, from the “inner iteration point of view”, these methods behave quite similar to first-order methods. The development of efficient preconditioners for solving the systems $B\Delta y = u$ is certainly the crucial step towards making iterative second-order methods become superior to first-order methods.

References

- [1] F. Alizadeh. Interior point methods in semidefinite programming with applications to combinatorial optimization. *SIAM Journal on Optimization*, 5(1):13–51, 1995.
- [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 Journal on Optimization*, 8:746–768, 1998.
- [3] F. Alizadeh and S. Schmieta. Symmetric cones, potential-reduction methods and word-by-word extensions. In R. Saigal, L. Vandenberghe, and H. Wolkowicz, editors, *Handbook of Semidefinite Programming*, pages 195–233. Kluwer Academic Publishers, Boston-Dordrecht-London, 2000.
- [4] T. Ando. Concavity of certain maps and positive definite matrices and applications to hadamard products. *Linear Algebra and its Applications*, 26:203–241, 1979.
- [5] A. Barvinok. Problems of distance geometry and convex properties of quadratic maps. *Discrete Computational Geometry*, 13:189–202, 1995.
- [6] A. Ben-Tal and M.P. Bendsøe. A new method for optimal truss topology design. *SIAM Journal on Optimization*, 3:322–358, 1993.

- [7] A. Ben-Tal and A. Nemirovskii. *Lectures on Modern Convex Optimization: Analysis, Algorithms, Engineering Applications*. MPS-SIAM Series on Optimization. SIAM, Philadelphia, 2000.
- [8] A. Ben-Tal and A. Nemirovskii. Potential reduction polynomial time method for truss topology design. *SIAM Journal on Optimization*, 4:596–612, 1994.
- [9] S. Benson, Y. Ye, and X. Zhang. Solving large-scale sparse semidefinite programs for combinatorial optimization. *SIAM Journal on Optimization*, 10:443–461, 2000.
- [10] P.M. Bentler and J.A. Woodward. Inequalities among lower bounds to reliability: with applications to test construction and factor analysis. *Psychometrika*, 45:249–267, 1980.
- [11] S. Boyd, E. Ghaoui, E. Feron, and V. Balakrishnan. *Linear matrix inequalities in system and control theory*, volume 15 of *Studies in Applied Mathematics*. Society for Industrial and Applied Mathematics, Philadelphia, PA, 1994.
- [12] S. Burer. Semidefinite programming in the space of partial positive semidefinite matrices. manuscript, Department of Management Sciences, University of Iowa, Iowa City, IA 52242-1000, USA, May 2002.
- [13] S. Burer. Semidefinite programming in the space of partial positive semidefinite matrices, November 2002. Talk given at the Jan Jose INFORMS meeting.
- [14] S. Burer and R.D.C. Monteiro. A general framework for establishing polynomial convergence of long-step methods for semidefinite programming. manuscript, School of ISyE, Georgia Tech, Atlanta, GA, 30332, USA, August 1999. Conditionally accepted in *Optimization Methods and Software*.
- [15] S. Burer and R.D.C. Monteiro. A projected gradient algorithm for solving the Maxcut SDP relaxation. *Optimization Methods and Software*, 15:175–200, 2001.
- [16] S. Burer and R.D.C. Monteiro. A nonlinear programming algorithm for solving semidefinite programs via low-rank factorization. *Mathematical Programming, Series B*, 95:329–357, 2003.
- [17] S. Burer, R.D.C. Monteiro, and Y. Zhang. Interior-point algorithms for semidefinite programming based on a nonlinear programming formulation. *Computational Optimization and Applications*, 22:49–79, 2002.
- [18] S. Burer, R.D.C. Monteiro, and Y. Zhang. Solving a class of semidefinite programs via nonlinear programming. *Mathematical Programming*, 93:97–123, 2002.
- [19] S. Burer, R.D.C. Monteiro, and Y. Zhang. A computational study of a gradient-based log-barrier algorithm for a class of large-scale SDPs. *Mathematical Programming, Series B*, 95:359–379, 2003.
- [20] C. Choi and Y. Ye. Solving sparse semidefinite programs using the dual scaling algorithm with an iterative solver. Manuscript, Department of Management Sciences, University of Iowa, Iowa City, IA 52242, USA, 2000.
- [21] J. Cullum, W.E. Donath, and P. Wolfe. The minimization of certain nondifferentiable sums of eigenvalues of symmetric matrices. *Mathematical Programming Study*, 3, 1975.
- [22] W. E. Donath and A. J. Hoffman. Algorithms for partitioning graphs and computer logic based on eigenvectors of connection matrices. *IBM Technical Disclosure Bulletin*, 15, 1972.
- [23] W. E. Donath and A. J. Hoffman. Lower bounds for the partitioning of graphs. *IBM J. Res. and Devel.*, 17, 1973.
- [24] L. Faybusovich. Euclidean jordan algebras and interior-point algorithms. *Positivity*, 1:331–357, 1997.
- [25] L. Faybusovich. Linear systems in jordan algebras and primal-dual interior point algorithms. *Journal of Computational and Applied Mathematics*, 86:149–175, 1997.
- [26] L. Faybusovich. A Jordan-algebraic approach to potential-reduction algorithms. Technical report, Dept. of Mathematics, University of Notre Dame, Room 370, CCMB, Notre Dame, IN 46556-5683, April 1998.
- [27] R. Fletcher. A nonlinear programming in statistics (educational testing). *SIAM Journal on Scientific and Statistical Computing*, 2:257–267, 1981.
- [28] R. Fletcher. Semidefinite matrix constraints in optimization. *SIAM Journal on Control and Optimization*, 23:493–513, 1985.
- [29] R. Fletcher. *Practical Methods of Optimization*. John Wiley & Sons, New York, second edition, 1987.
- [30] R. M. Freund. Complexity of an algorithm for finding an approximate solution of a semidefinite program with no regularity condition. Working paper OR 302-94, Operations Research Center, Massachusetts Institute of Technology, Cambridge, December 1994.

- [31] K. Fujisawa, M. Kojima, and K. Nakata. Exploiting sparsity in primal-dual interior-point methods for semidefinite programming. *Mathematical Programming*, 79:235–253, 1997.
- [32] M. Fukuda and M. Kojima. Interior-point methods for lagrangian dual of semidefinite programs. manuscript, Department of Mathematical and Computing Sciences, Tokyo Institute of Technology, 2-12-1 Oh-Okayama, Meguro-ku, Tokyo 152, Japan, December 2000.
- [33] M. Fukuda, M. Kojima, K. Murota, and K. Nakata. Exploiting sparsity in semidefinite programming via matrix completion I: General framework. *SIAM Journal on Optimization*, 11:647–674, 2001.
- [34] M.X. Goemans and D.P. Williamson. .878-approximation algorithms for MAX CUT and MAX 2SAT. *Proceedings of the Symposium of Theoretical Computer Science*, pages 422–431, 1994.
- [35] D. Goldfarb and K. Scheinberg. Interior point trajectories in semidefinite programming. *SIAM Journal on Optimization*, 8:871–886, 1998.
- [36] M. Gu. On primal-dual interior point methods for semidefinite programming. Technical Report CAM report97-12, Department of Mathematics, University of California, Los Angeles, CA 90095-1555, 1997.
- [37] M. Halicka, E. de Klerk, and C. Roos. On the convergence of the central path in semidefinite optimization. Technical Report, Faculty of Technical Mathematics and Informatics, TU Delft, NL-2628 CD Delft, The Netherlands, June 2001.
- [38] M. Halicka, E. de Klerk, and C. Roos. Limiting behavior of the central path in semidefinite optimization. preprint, Faculty of Technical Mathematics and Informatics, TU Delft, NL-2628 CD Delft, The Netherlands, June 2002.
- [39] C. Helmberg and F. Oustry. Bundle methods to minimize the maximum eigenvalue function. In R. Saigal, L. Vandenbergh, and H. Wolkowicz, editors, *Handbook of Semidefinite Programming*, pages 307–337. Kluwer Academic Publishers, Boston-Dordrecht-London, 2000.
- [40] C. Helmberg and F. Rendl. A spectral bundle method for semidefinite programming. *SIAM Journal on Optimization*, 10:673–696, 2000.
- [41] C. Helmberg, F. Rendl, R. J. Vanderbei, and H. Wolkowicz. An interior-point method for semidefinite programming. *SIAM Journal on Optimization*, 6:342–361, 1996.
- [42] S. Homer and M. Peinado. Design and performance of parallel and distributed approximation algorithms for maxcut. *Journal of Parallel and Distributed Computing*, 46:48–61, 1997.
- [43] F. Jarre. An interior-point method for minimizing the maximum eigenvalue of a linear combination of matrices. *SIAM Journal on Control and Optimization*, 31:1360–1377, 1993.
- [44] A.P. Kamath and N.K. Karmarkar. A continuous approach to compute upper bounds in quadratic maximization problems with integer constraints. In C. Floudas and P. Pardalos, editors, *Recent Advances in Global Optimization*, pages 125–140. Princeton University Press, Oxford, UK, 1992.
- [45] M. KOJIMA. 1995. private communication.
- [46] M. Kojima, S. Mizuno, and A. Yoshise. A polynomial-time algorithm for a class of linear complementarity problems. *Mathematical Programming*, 44:1–26, 1989.
- [47] M. Kojima, S. Mizuno, and A. Yoshise. A primal-dual interior point algorithm for linear programming. In N. Megiddo, editor, *Progress in Mathematical Programming : Interior Point and Related Methods*, pages 29–47. Springer Verlag, New York, 1989.
- [48] M. Kojima, M. Shida, and S. Shindoh. A note on the Nesterov-Todd and the Kojima-Shindoh-Hara search directions in semidefinite programming. *Mathematical Programming*, 80:129–160, 1998.
- [49] M. Kojima, M. Shida, and S. Shindoh. A predictor-corrector interior-point algorithm for the semidefinite linear complementarity problem using the Alizadeh-Haeberly-Overton search direction. *SIAM Journal on Optimization*, 9:444–465, 1999.
- [50] M. Kojima, M. Shida, and S. Shindoh. Search directions in the SDP and the monotone SDLCP: generalization and inexact computation. *Mathematical Programming*, 85:51–80, 1999.
- [51] M. Kojima, S. Shindoh, and S. Hara. Interior-point methods for the monotone semidefinite linear complementarity problem in symmetric matrices. *SIAM Journal on Optimization*, 7:86–125, 1997.
- [52] S. Kruk, M. Muramatsu, F. Rendl, R.J. Vanderbei, and H. Wolkowicz. The Gauss-Newton direction in linear and semidefinite programming. *Optimization Methods and Software*, 15:1–28, 2001.

- [53] C.-J. Lin and R. Saigal. On solving large scale semidefinite programming problems - a case study of quadratic assignment problem. Technical Report, Dept. of Industrial and Operations Engineering, The University of Michigan, Ann Arbor, MI 48109-2177, 1997.
- [54] L. Lovász. On the Shannon Capacity of a graph. *IEEE Transactions of Information Theory*, IT-25(1):1–7, January 1979.
- [55] L. Lovász and A. Schrijver. Cones of matrices and setfunctions, and 0-1 optimization. *SIAM Journal on Optimization*, 1:166–190, 1991.
- [56] 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*, 8:59–81, 1998.
- [57] S. Mizuno, M. J. Todd, and Y. Ye. On adaptive step primal–dual interior–point algorithms for linear programming. *Mathematics of Operations Research*, 18:945–981, 1993.
- [58] R.D.C. Monteiro. Primal-dual path following algorithms for semidefinite programming. *SIAM Journal on Optimization*, 7:663–678, 1997.
- [59] R.D.C. Monteiro. Polynomial convergence of primal-dual algorithms for semidefinite programming based on Monteiro and Zhang family of directions. *SIAM Journal on Optimization*, 8:797–812, 1998.
- [60] R.D.C. Monteiro and I. Adler. Interior path-following primal-dual algorithms. Part I: Linear programming. *Mathematical Programming*, 44:27–41, 1989.
- [61] R.D.C. Monteiro and I. Adler. Interior path-following primal-dual algorithms. Part II: Convex quadratic programming. *Mathematical Programming*, 44:43–66, 1989.
- [62] R.D.C. Monteiro and M. J. Todd. Path-following methods for semidefinite programming. In R. Saigal, L. Vandenberghe, and H. Wolkowicz, editors, *Handbook of Semidefinite Programming*. Kluwer Academic Publishers, Boston-Dordrecht-London, 2000.
- [63] R.D.C. Monteiro and T. Tsuchiya. Polynomial convergence of a new family of primal-dual algorithms for semidefinite programming. *SIAM Journal on Optimization*, 9:551–577, 1999.
- [64] R.D.C. Monteiro and T. Tsuchiya. Polynomiality of primal-dual algorithms for semidefinite linear complementarity problems based on the Kojima-Shindoh-Hara family of directions. *Mathematical Programming*, 84:39–53, 1999.
- [65] R.D.C. Monteiro and T. Tsuchiya. Polynomial convergence of primal-dual algorithms for the second-order cone program based on the MZ-family of directions. *Mathematical Programming*, 88:61–83, 2000.
- [66] R.D.C. Monteiro and P.R. Zanjácomo. A note on the existence of the Alizadeh-Haeberly-Overton direction for semidefinite programming. *Mathematical Programming*, 78:393–396, 1997.
- [67] R.D.C. Monteiro and P.R. Zanjácomo. Implementation of primal-dual methods for semidefinite programming based on Monteiro and Tsuchiya Newton directions and their variants. *Optimization Methods and Software*, 11/12:91–140, 1999.
- [68] R.D.C. Monteiro and Y. Zhang. A unified analysis for a class of path-following primal-dual interior-point algorithms for semidefinite programming. *Mathematical Programming*, 81:281–299, 1998.
- [69] K. Nakata, K. Fujisawa, M. Fukuda, M. Kojima, and K. Murota. Exploiting sparsity in semidefinite programming via matrix completion II: Implementation and numerical results. manuscript, Department of Mathematical and Computing Sciences, Tokyo Institute of Technology, 2-12-1 Oh-Okayama, Meguro-ku, Tokyo 152-8552, Japan, February 2001.
- [70] K. Nakata, K. Fujisawa, and M. Kojima. Using the conjugate gradient method in interior-point methods for semidefinite programs. pages 297–316. 1998. (in Japanese).
- [71] Y. E. Nesterov. Long-step strategies in interior-point primal-dual methods. *Mathematical Programming*, 76:47–94, 1996.
- [72] Y. E. Nesterov and A. S. Nemirovskii. Polynomial barrier methods in convex programming. *Ekonomika i Matem. Metody*, 24:1084–1091, 1988. (In Russian; English transl. Matekon: Translations of Russian and East European Math. Economics.).
- [73] Y. E. Nesterov and A. S. Nemirovskii. Self-concordant functions and polynomial time methods in convex programming. Book–Preprint, Central Economic & Mathematical Institute, USSR Acad. Sci. Moscow, USSR, 1989. Published in Nesterov and Nemirovsky [75].
- [74] Y. E. Nesterov and A. S. Nemirovskii. Optimization over positive semidefinite matrices: Mathematical background and user’s manual. Technical report, Central Economic & Mathematical Institute, USSR Acad. Sci. Moscow, USSR, 1990.
- [75] Y. E. Nesterov and A. S. Nemirovskii. *Interior Point Polynomial Algorithms in Convex Programming: Theory and Applications*. Society for Industrial and Applied Mathematics, Philadelphia, 1994.

- [76] Y. E. Nesterov and M. J. Todd. Self-scaled barriers and interior-point methods for convex programming. *Mathematics of Operations Research*, 22:1–42, 1997.
- [77] Y. E. Nesterov and M. J. Todd. Primal-dual interior-point methods for self-scaled cones. *SIAM Journal on Optimization*, 8:324–364, 1998.
- [78] F. Oustry. The \square -Lagrangian of the maximum eigenvalue function. *SIAM Journal on Optimization*, 9:526–549, 1999.
- [79] F. Oustry. A second-order bundle method to minimize the maximum eigenvalue function. *Mathematical Programming*, 89:1–33, 2000.
- [80] M.L. Overton and R.S. Womersley. Optimality conditions and duality theory for minimizing sums of the largest eigenvalues of symmetric matrices. *Mathematical Programming*, 62:321–357, 1993.
- [81] G. Pataki. On the rank of extreme matrices in semidefinite programs and the multiplicity of optimal eigenvalues. *Mathematics of Operations Research*, 23:339–358, 1998.
- [82] F. A. Potra and R. Sheng. Superlinear convergence of interior-point algorithms for semidefinite programming. Reports on Computational Mathematics 86, Department of Mathematics, The University of Iowa, Iowa City, Iowa, April 1996.
- [83] F. A. Potra and R. Sheng. A superlinearly convergent primal-dual infeasible-interior-point algorithm for semidefinite programming. *SIAM Journal on Optimization*, 8:1007–1028, 1998.
- [84] F. Pukelsheim. *Optimal design of experiments*. Wiley, New York, 1993.
- [85] R. Saigal, L. Vandenberghe, and H. Wolkowicz. *Handbook of Semidefinite Programming*. Kluwer Academic Publishers, Boston-Dordrecht-London, 2000.
- [86] S. Schmieta and F. Alizadeh. Associative algebras, symmetric cones and polynomial time interior point algorithms. Report RRR 17-98, RUTCOR, Rutgers University, 640 Bartholomew Road, Piscataway, NJ 08854-8003, 1998.
- [87] A. Shapiro. Weighted minimum trace factor analysis. *Psychometrika*, 47:243–264, 1982.
- [88] A. Shapiro. Minimum trace factor analysis. In S. Kotz and N.L. Johnson, editors, *Encyclopedia of Statistical Sciences*, volume 5. John Wiley and Sons, New York, 1985.
- [89] R. Sheng and F. A. Potra. Nonsymmetric search directions for semidefinite programming. Research Report ANL/MCS-P692-0997, Mathematics and Computer Science Division, Argonne National Laboratory, Argonne, IL 60439, USA, 1997.
- [90] M. Shida, S. Shindoh, and M. Kojima. Existence and uniqueness of search directions in interior-point algorithms for the SDP and monotone SDLCP. *SIAM Journal on Optimization*, 8:387–396, 1998.
- [91] N.Z. Shor. Dual quadratic estimates in polynomial and Boolean programming. *Annals of Operations Research*, 25:163–168, 1990.
- [92] K. Tanabe. Complementarity-enforcing centered Newton method for mathematical programming: Global methods. In K. Tone, editor, *New method for linear programming*, volume 5 of *Cooperative Research Report*, pages 118–144. Institute of Statistical Mathematics, Tokyo, Japan, 1987.
- [93] K. Tanabe. Centered Newton method for mathematical programming. In M. Iri and K. Yajima, editors, *System Modelling and Optimization*, pages 197–206. Springer-Verlag, Tokyo, Japan, 1988.
- [94] M. J. Todd. A study of search directions in primal-dual interior-point methods for semidefinite programming. *Optimization Methods and Software*, 11:1–46, 1999.
- [95] M. J. Todd. Semidefinite optimization. manuscript, School of Operations Research and Industrial Engineering, Cornell University, Ithaca, NY 14853, USA, August 2001.
- [96] M. J. Todd, K. C. Toh, and R. H. Tütüncü. On the Nesterov-Todd direction in semidefinite programming. *SIAM Journal on Optimization*, 8:769–796, 1998.
- [97] K. C. Toh. Some new search directions for primal-dual interior point methods in semidefinite programming. *SIAM Journal on Optimization*, 11:223–242, 2000.
- [98] K. C. Toh. Solving large scale semidefinite programs via an iterative solver on the augmented systems. Research Report B-388, Department of Mathematical and Computing Sciences, Tokyo Institute of Technology, Tokyo, Japan, December 2002.
- [99] K.C. Toh and M. Kojima. Solving some large scale semidefinite programs via the conjugate residual method. Working paper, Department of Mathematics, National University of Singapore, 10 Kent Ridge Crescent, 119620 Singapore, August 2000.

- [100] P. Tseng. Search directions and convergence analysis of some infeasible path-following methods for the monotone semi-definite LCP. *Optimization Methods and Software*, 9:245–268, 1998.
- [101] T. Tsuchiya. A polynomial primal-dual path-following algorithm for second-order cone programming. Research Memorandum 649, The Institute of Statistical Mathematics, Tokyo, Japan, October 1997. Revised: December 1997.
- [102] T. Tsuchiya. A convergence analysis of the scaling-invariant primal-dual path-following algorithms for second-order cone programming. Research Memorandum 664, The Institute of Statistical Mathematics, Tokyo, Japan, February 1998. Revised: May 1998.
- [103] L. Vandenberghe and S. Boyd. Semidefinite programming. *SIAM Review*, 38:49–95, 1996.
- [104] R. J. Vanderbei and Y. Benson. On formulating semidefinite programming problems as smooth convex nonlinear optimization problems. Report orfe 99-01, Operations Research and Financial Engineering, Princeton, NJ, 1999.
- [105] S. A. Vavasis. A note on efficient computation of the gradient in semidefinite programming. Technical report, Department of Computer Science, Cornell University, Ithaca, NY, 1999.
- [106] G.A. Watson. Algorithms for minimum trace factor analysis. *SIAM Journal on Matrix Analysis and its Applications*, 13:1039–1053, 1992.
- [107] Y. Zhang. On extending some primal–dual interior–point algorithms from linear programming to semidefinite programming. *SIAM Journal on Optimization*, 8:365–386, 1998.