Konrad-Zuse-Zentrum
für Informationstechnik Berlin

Takustraße 7
D-14195 Berlin-Dahlem
Germany

C. HELMBERG[1]

Semidefinite Programming

[1] Konrad-Zuse-Zentrum für Informationstechnik Berlin, Takustraße 7, D-14195 Berlin, Germany, helmberg@zib.de, http://www.zib.de/helmberg

Preprint SC 99-49 (October 1999)

Semidefinite Programming

C. Helmberg*

October 1999

Abstract

Due to its many applications in control theory, robust optimization, combinatorial optimization and eigenvalue optimization, semidefinite programming had been in wide spread use even before the development of efficient algorithms brought it into the realm of tractability. Today it is one of the basic modeling and optimization tools along with linear and quadratic programming. Our survey is an introduction to semidefinite programming, its duality and complexity theory, its applications and algorithms.

Key Words. Semidefinite programming.
MSC 1991. 90C25.

1 Introduction

Semidefinite programming is linear programming over positive semidefinite matrices. The beautiful properties of symmetric positive semidefinite matrices and their associated convex quadratic forms have fascinated mathematicians since the discovery of conic sections. Many properties of nonlinear objects have therefore been related to the behavior of convex quadratic functions (see, *e.g.*, the references in [7] for early applications in control theory or [12] for applications in combinatorial optimization). The development of interior point methods for semidefinite programming in the late eighties [30] made it possible to optimize over this set. That aroused much interest and lead to heavy activity in this field. Since then, many new algorithms have been proposed. For some of them, high quality implementations are now available. They allow to use semidefinite programming as a standard optimization tool, even though large scale problems in the style of linear programming are still out of reach. With the availability of optimization software, more and more problems are modeled as semidefinite programs. In fact, semidefinite programming has become one of the basic modeling and optimization tools along with linear and quadratic programming.

This survey wants to provide an easy access to the fundamental concepts of semidefinite programming, to illustrate the wide variety of its applications, and to give a basic understanding of its currently most successful algorithms. In order to compensate for unavoidable omissions we point to other surveys wherever possible. Current standard references for semidefinite programming are [30, 43]; both contain a large collection of applications. A comprehensive book, dealing with all aspects of semidefinite programming, is currently in preparation [40] and should be available soon.

The paper is organized as follows. In Section 2, we review some basic notions from linear algebra and fundamental properties of the cone of positive semidefinite matrices. Semidefinite programs and their duals are introduced in Section 3. We also discuss their relation to linear programs and some nonlinear convex optimization problems. The semidefinite cone is not finitely generated and thus not polyhedral. This makes duality theory slightly more involved than in linear programming; this is the topic of Section 4. In Section 5, we briefly investigate the geometry of semidefinite sets and survey some complexity results. The wide range of applications is one of the

*Konrad-Zuse-Zentrum für Informationstechnik Berlin, Takustraße 7, 14195 Berlin, Germany. `helmberg@zib.de`, `http://www.zib.de/helmberg`

main attractions of semidefinite programming. We present a small selection in Section 6. In Section 7, we sketch the algorithms which, in our opinion, dominate in current implementational efforts, namely, primal-dual interior point algorithms [19, 26, 32, 1], a potential reduction algorithm[5], and the spectral bundle method[18]. We also provide some guidelines on what classes of problems they might best be employed. We conclude by giving a short outlook in Section 8.

2 The Cone of Positive Semidefinite Matrices

We first review some basic notions form linear algebra. Unless stated otherwise, we refer to [20, 21] for proofs. The set $M_{m,n}$ of $m \times n$ real matrices can be interpreted as a vector space in $\mathbb{R}^{mn}$. The set of square matrices of dimension n is denoted by M_n. The operator $\mathrm{vec}() : M_{m,n} \to \mathbb{R}^{mn}$ will be used to explicitly transform matrices into vectors by stacking the columns on top of each other. A natural inner product between two elements $A, B \in M_{m,n}$ is

$$\langle A, B\rangle = \mathrm{vec}(B)^T \mathrm{vec}(A) = \sum_{i=1}^{m}\sum_{j=1}^{n} a_{ij}b_{ij} = \mathrm{tr}(B^T A). \tag{1}$$

The *trace* $\mathrm{tr}(\cdot)$ is the sum of the diagonal elements of a square matrix. It is a linear function. The trace of $A \in M_n$ equals the sum of the eigenvalues of A. The norm associated with inner product (1) is the *Frobenius norm*, $\|A\|_{\mathrm{F}} = \sqrt{\langle A, A\rangle}$. If the argument of the trace is a product of matrices, the matrices may be "rotated" without affecting the result,

$$\langle AB, C\rangle = \mathrm{tr}(C^T AB) = \mathrm{tr}(BC^T A) = \langle A, CB^T\rangle .$$

We will usually work with the set of symmetric matrices S_n, a vector space in $\mathbb{R}^{\binom{n+1}{2}}$. All results on the positive semidefinite cone will be stated with respect to this $\binom{n+1}{2}$-dimensional space. It is convenient to use the inner product of $M_{m,n}$ for S_n, as well. For $A, B \in S_n$, $\langle A, B\rangle = \mathrm{tr}(B^T A) = \mathrm{tr}(AB)$. The Schur Theorem implies that all eigenvalues $\lambda_i(A)$ of $A \in S_n$ are real. Furthermore, there is an orthonormal matrix $P \in M_n$ that diagonalizes A, $P^T AP = \Lambda_A$.

We will use positive semidefiniteness exclusively in connection with symmetric matrices.

Definition 2.1
$A \in S_n$ *is* positive semidefinite *(*$A \in S_n^+$, $A \succeq 0$*) if* $x^T Ax \geq 0 \quad \forall x \in \mathbb{R}^n$.
$A \in S_n$ *is* positive definite *(*$A \in S_n^{++}$, $A \succ 0$*) if* $x^T Ax > 0 \quad \forall x \in \mathbb{R}^n \setminus \{0\}$.

The next proposition follows directly from the definition and will used throughout this text.

Proposition 2.2 *Let* $B \in M_n$ *be a nonsingular matrix. Then* $A \in S_n^+$ *if and only if* $B^T AB \in S_n^+$ *and* $A \in S_n^{++}$ *if and only if* $B^T AB \in S_n^{++}$.

There are several equivalent characterizations for positive definite matrices.

Theorem 2.3 *(Characterizations of positive definite matrices)*
For $A \in S_n$ *the following statements are equivalent:*
1. A *is positive definite.*
2. $\lambda_i(A) > 0 \quad i = 1, \dots, n$.
3. There exists $C \in M_n$ *with* $\mathrm{rank}(C) = n$ *so that* $A = C^T C$.
4. $\det(A_i) > 0$ *for a nested sequence*[1] A_i, $i = 1, \dots, n$, *of principal submatrices of* A.

Because the eigenvalues of the inverse A^{-1} of A are $1/\lambda_i(A)$, the inverse A^{-1} is positive definite if and only if A is positive definite.

For positive semidefinite matrices almost the same characterizations are valid.

[1] A nested sequence is determined by a sequence of proper subsets $J_1 \subset J_2 \subset \ldots \subset J_n = \{1, \dots, n\}$ of indices with $|J_i| = i$ for $i = 1, \dots, n$

Theorem 2.4 *(Characterizations of positive semidefinite matrices)*
For $A \in S_n$ the following statements are equivalent:
1. A is positive semidefinite.
2. $\lambda_i(A) \geq 0 \quad i = 1, \dots, n$
3. There exists $C \in M_{m,n}$ so that $A = C^T C$. For any such C, $\operatorname{rank}(C) = \operatorname{rank}(A)$.
4. $\langle A, B \rangle \geq 0$ for all $B \in S_n^+$.

An intriguing interpretation of a factorization $A = C^T C \succeq 0$ is obtained by viewing the columns of C as vectors v_i. The elements a_{ij} are the scalar products $\langle v_i, v_j \rangle$ of the vectors v_i and v_j. A is referred to as the *Gram matrix* of the vectors $v_1, \dots, v_n$. Factorizations can be determined algorithmically, *e.g.*, by Cholesky-, LDL^T-, or eigenvalue factorizations (see [14]).

The set of positive semidefinite matrices S_n^+ is a full dimensional, closed pointed cone in $\mathbb{R}^{\binom{n+1}{2}}$. Since the eigenvalues are the roots of the characteristic polynomial, they depend continuously on the matrix elements. Therefore, the set of positive definite matrices S_n^{++} forms the interior of the cone S_n^+. The boundary of S_n^+ consists of the positive semidefinite matrices having at least one zero eigenvalue.

The cone of positive semidefinite matrices S_n^+ is *self-polar* or *self-dual*, *i.e.*, the *polar cone* $(S_n^+)^* = \{A \in S_n : \langle A, B \rangle \geq 0 \text{ for all } B \in S_n^+\}$ satisfies $(S_n^+)^* = S_n^+$. This is equivalent to Theorem 2.4(4). Since the polar cone may be interpreted as the set of tight valid linear inequalities for S_n^+ or, equivalently, as the set of tangent planes to S_n^+, we speak of $(S_n^+)^*$ as the *dual cone* to S_n^+.

A convex set $F \subseteq C$ is called a *face* of a convex set C if for any two elements $x, y \in C$ with $\alpha x + (1-\alpha) y \in F$ for some $\alpha \in (0,1)$ we have $x, y \in F$. The following theorem gives a characterization of the faces of the semidefinite cone.

Theorem 2.5 *([2]) F is a face of S_n^+ if and only if $F = \{0_{n\times n}\}$ or*

$$F = \{X : X = PWP^T,\ W \in S_k^+\}$$

for some $k \in \{1, \dots, n\}$, $P \in M_{n,k}$ with $\operatorname{rank}(P) = k$.

In the theorem, the columns of P span, within $\mathbb{R}^n$, the subspace of all eigenvectors to nonzero eigenvalues of matrices in the face. The faces of S_n^+ have dimension $\binom{k+1}{2}$, so there are considerable jumps in dimension. The extremal rays of S_n^+ are the faces $\{\lambda x x^T : \lambda \geq 0\}$ for $\|x\| = 1$, $x \in \mathbb{R}^n$. By eigenvalue decomposition, these rays form a minimal generating system for S_n^+. In contrast to polyhedral cones, S_n^+ cannot be generated by a finite set.

The cone of semidefinite matrices induces a partial order on the set of symmetric matrices by defining, for $A, B \in S_n$, $A \succeq B$ ($A \succ B$) if $A - B \in S_n^+$ ($A - B \in S_n^{++}$). This is the origin of the notation $A \succeq 0$ ($A \succ 0$) in Definition 2.1.

3 Semidefinite Programs

Semidefinite programming is linear programming over the cone of positive semidefinite matrices. In comparison to standard linear programming, the vector $x \in \mathbb{R}_+^n$ of variables (the nonnegative orthant, a polyhedral cone) is replaced by a matrix variable $X \in S_n^+$ (the nonpolyhedral cone of semidefinite matrices). In order to pronounce this similarity, we first formulate the problem with respect to the vector representation $\operatorname{vec}(X)$ of X,

$$\min \quad c^T \operatorname{vec}(X) \ \text{ s.t. } \ A \operatorname{vec}(X) = b,\ X \succeq 0,$$

for given vectors $c \in \mathbb{R}^{n^2}$, $b \in \mathbb{R}^m$, and a constraint matrix $A \in M_{m,n^2}$.

Usually, semidefinite programs arise in a natural way from problems whose data is given by matrices. The use of the vec-operator tends to hide the obvious and complicates the formulation. It pays to use a more agreeable notation by interpreting the vector c and the rows of A as matrices.

Let $C \in M_n$ denote the matrix corresponding to c, *i.e.*, $c = \operatorname{vec}(C)$. Then the inner product $c^T \operatorname{vec}(X)$ in vector space can equivalently be written as the inner product $\langle C, X \rangle$ in matrix space.

Since X is a symmetric matrix, the skew-symmetric part of C is of no influence in this inner product. Without loss of generality we thus require C to be a symmetric matrix.

Likewise, we interpret row $A_{i,\cdot}$ as a symmetric matrix $A_i \in S_n$, rewrite the i-th constraint $A_{i,\cdot}\operatorname{vec}(X)$ as $\langle A_i, X\rangle$, and collect the constraints in a linear operator $\mathcal{A} : S_n \to \mathbb{R}^m$,

$$\mathcal{A}X = \begin{pmatrix} \langle A_1, X\rangle \\ \vdots \\ \langle A_m, X\rangle \end{pmatrix}.$$

With this notation we arrive at our standard formulation of a semidefinite program,

$$\text{(PSDP)} \qquad \begin{array}{ll} \min & \langle C, X\rangle \\ \text{s.t.} & \mathcal{A}X = b \\ & X \succeq 0. \end{array}$$

In order to derive the dual of this program we need the adjoint operator to $\mathcal{A}$, which we denote by $\mathcal{A}^T : \mathbb{R}^m \to S_n$. Since $\langle \mathcal{A}X, y\rangle = \sum_{i=1}^m y_i \langle A_i, X\rangle = \langle \sum_{i=1}^m y_i A_i, X\rangle$ for all $X \in S_n$ and $y \in \mathbb{R}^m$, it has the form

$$\mathcal{A}^T y = \sum_{i=1}^m y_i A_i.$$

With respect to the initial vector formulation, $\mathcal{A}^T y$ is simply a different representation of $A^T y$, emphasizing the fact that we are working with matrices.

The dual is obtained via a Lagrange approach by interchanging inf and sup,

$$\inf_{X\succeq 0} \sup_{y\in\mathbb{R}^m} \langle C, X\rangle + \langle b - \mathcal{A}X, y\rangle \geq \sup_{y\in\mathbb{R}^m} \inf_{X\succeq 0} \langle b, y\rangle + \left\langle X, C - \mathcal{A}^T y\right\rangle. \tag{2}$$

The supremum on the right hand side is bounded from below if and only if the inner minimization over $X \succeq 0$ remains finite for some $\hat{y} \in \mathbb{R}^m$. This requires $C - \mathcal{A}^T\hat{y}$ to be in the dual cone to S_n^+, *i.e.*, $C - \mathcal{A}^T y \in S_n^+$ (Theorem 2.4(4)). We write this condition by introducing a slack matrix Z,

$$\text{(DSDP)} \qquad \begin{array}{ll} \max & \langle b, y\rangle \\ \text{s.t.} & \mathcal{A}^T y + Z = C \\ & y \in \mathbb{R}^m,\ Z \succeq 0. \end{array}$$

This is the standard formulation of the dual semidefinite program to (PSDP).

The use of the free variables y in (DSDP) may raise doubts whether (DSDP) is indeed a semidefinite program. To remove these doubts, we give a slightly different representation of (PSDP) and (DSDP) that highlights their common structure [30]. To this end we assume that the system $\mathcal{A}X = b$ is consistent, *i.e.*, there exists an $\hat{X} \in S_n$ satisfying $\mathcal{A}\hat{X} = b$. In this case, all y variables in (DSDP) can be eliminated. We first express the cost function in terms of $Z = C - \mathcal{A}^T y$,

$$\langle b, y\rangle = \langle \mathcal{A}\hat{X}, y\rangle = \langle \hat{X}, \mathcal{A}^T y\rangle = \langle \hat{X}, C - Z\rangle.$$

Now variables y only serve to span the feasible set of Z-values. Let $\mathcal{R}(\mathcal{A}^T)$ denote the range space of $\mathcal{A}^T$, let $\mathcal{N}(\mathcal{A})$ denote the null space of $\mathcal{A}$, and observe that these two subspaces are orthogonal complements, $\mathcal{R}(\mathcal{A}^T) = \mathcal{N}(\mathcal{A})^\perp$. The dual (primal) equality constraints require a feasible Z (X) to be contained in the affine subspace $\{C + \mathcal{N}(\mathcal{A})^\perp\}$ ($\{\hat{X} + \mathcal{N}(\mathcal{A})\}$). In these terms the primal-dual pair of problems takes the following form,

$$\begin{array}{ll} \min & \langle C, X\rangle \\ \text{s.t.} & X \in (S_n^+ \cap \{\hat{X} + \mathcal{N}(\mathcal{A})\}) \end{array} \qquad \begin{array}{ll} \max & \langle \hat{X}, C - Z\rangle \\ \text{s.t.} & Z \in (S_n^+ \cap \{C + \mathcal{N}(\mathcal{A})^\perp\}). \end{array} \tag{3}$$

Thus, (PSDP) and (DSDP) have indeed the same structure: Optimize a linear cost function over a convex set defined by the intersection of an affine subspace with the semidefinite cone. Any property holding for the primal formulation has its analogue in the dual formulation.

So far we have been considering just one semidefinite variable. We may also formulate problems that contain several semidefinite variables,

$$\begin{array}{llll} \min & \sum_{i=1}^{k} \langle C_i, X_i \rangle & \max & \langle b, y \rangle \\ \text{s.t.} & \sum_{i=1}^{k} \mathcal{A}_i X_i = b & \text{s.t.} & \mathcal{A}_i^T y + Z_i = C_i \quad i = 1, \ldots, k \\ & X_1 \in S_{n_1}^+, \ldots, X_k \in S_{n_k}^+ & & y \in \mathbb{R}^m, Z_1 \in S_{n_1}^+, \ldots, Z_k \in S_{n_k}^+. \end{array} \tag{4}$$

In many practical applications such a structure arises naturally and it is important to exploit it to obtain efficient implementations. For theoretical purposes, however, the standard primal-dual pair of problems is sufficient. In fact, any semidefinite program in several semidefinite variables of varying dimensions can be formulated equivalently as a standard (PSDP), because

$$X_1 \succeq 0,\ X_2 \succeq 0,\ \ldots,\ X_k \succeq 0 \quad \Longleftrightarrow \quad \begin{bmatrix} X_1 & 0 & \cdots & 0 \\ 0 & X_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & X_k \end{bmatrix} \succeq 0.$$

It is now easy to see that linear programming is a special case of semidefinite programming: Interpret each single component $x_i \geq 0$ as a 1×1 positive semidefinite matrix in (4).

Several other convex optimization problems may be formulated as semidefinite programs. A standard tool in such reformulations is the following theorem. It gives a characterization of the positive definiteness of a matrix via the positive definiteness of the Schur complement with respect to a block partitioning of the matrix.

Theorem 3.1 *(Schur Complement) Let $A \in S_r^{++}$, $B \in S_q$, and $C \in M_{r,q}$. Then*

$$\begin{bmatrix} A & C \\ C^T & B \end{bmatrix} \succeq 0 \quad \Longleftrightarrow \quad B \succeq C^T A^{-1} C.$$

We illustrate its use for a convex quadratic constraint:

$$x^T Q x \leq q^T x + c. \tag{5}$$

Here, $x \in \mathbb{R}^n$ is the vector of variables and $Q \in S_n^+$, $q \in \mathbb{R}^n$, $c \in \mathbb{R}$ are given constants. Since Q may be singular, we factorize it, $Q = C^T C$, and use the identity I in $x^T C^T I C x$ as the positive definite matrix A in the Schur complement. The theorem allows to reformulate the quadratic constraint as

$$\begin{bmatrix} I & Cx \\ x^T C^T & q^T x + c \end{bmatrix} \succeq 0. \tag{6}$$

The nonlinear constraint on x is transformed into a linear constraint over the cone of positive semidefinite matrices. This proves that quadratically constrained convex quadratic programming problems can be formulated as semidefinite programs.

Monotone linear complementarity problems are a popular generalization of linear programming. In semidefinite programming, linear complementarity problems over the semidefinite cone are covered by standard semidefinite programs and do not add further generality, see [25]. In consequence, convex quadratic semidefinite programming is included in linear semidefinite programming.

Sometimes it is useful, for numerical or structural reasons, to transform one representation of a semidefinite program into another. This is achieved by transformations of the type $W = QXQ^T$ for nonsingular $Q \in M_n$ (see Proposition 2.2). These transformations are referred to as *scaling* and belong to the automorphism group of the semidefinite cone, *i.e.*, they are bijective linear maps on the set of symmetric matrices, leaving the semidefinite cone invariant. Since $X = Q^{-1} W Q^{-T}$ and, for arbitrary $A \in S_n$,

$$\langle A, X \rangle = \left\langle A, Q^{-1} W Q^{-T} \right\rangle = \left\langle Q^{-T} A Q^{-1}, W \right\rangle,$$

the correct transformation of a coefficient matrix A is $Q^{-T}AQ^{-1}$, which is the adjoint to the inverse transformation of QXQ^T. With

$$\bar{C} = Q^{-T}CQ^{-1}, \quad \bar{A}_i = Q^{-T}A_iQ^{-1} \quad i = 1, \dots, m,$$

and the corresponding linear operators $\bar{\mathcal{A}}W$ and $\bar{\mathcal{A}}^T\bar{y}$ we obtain the transformed primal-dual pair

$$(\mathrm{P}_Q) \quad \begin{array}{rl} \min & \langle \bar{C}, W \rangle \\ \text{s.t.} & \bar{\mathcal{A}}W = b \\ & W \succeq 0 \end{array} \qquad\qquad (\mathrm{D}_Q) \quad \begin{array}{rl} \max & \langle b, \bar{y} \rangle \\ \text{s.t.} & \bar{\mathcal{A}}^T\bar{y} + \bar{Z} = \bar{C} \\ & \bar{Z} \succeq 0. \end{array}$$

Proposition 3.2 *X is a feasible solution of (PSDP) if and only if the associated $W = QXQ^T$ is a feasible solution of (P_Q). Furthermore, X and W satisfy $\langle C, X \rangle = \langle \bar{C}, W \rangle$. (y, Z) is a feasible solution of (DSDP) if and only if the associated $(\bar{y}, \bar{Z}) = (y, Q^{-T}ZQ^{-1})$ is a feasible solution of (D_Q). Trivially, $\langle b, y \rangle = \langle b, \bar{y} \rangle$.*

4 Duality

We derived the dual to (PSDP) by a Lagrange approach in (2). The construction implies that the dual objective value cannot exceed the value of the primal. This is called *weak duality*. The gap between a dual feasible solution (y, Z) and a primal feasible solution X is

$$\langle C, X \rangle - \langle b, y \rangle = \langle \mathcal{A}^T y + Z, X \rangle - \langle \mathcal{A}X, y \rangle = \langle Z, X \rangle \geq 0. \tag{7}$$

If $\langle Z, X \rangle$ turns out to be zero, then this primal-dual pair is an optimal solution. In contrast to linear programming, it is no longer true that optimality implies $\langle Z, X \rangle = 0$. We illustrate the basic difficulty by analyzing an example of [43] in detail.

Example 4.1 *Consider the following primal semidefinite program,*

$$\min\ x_{12} \ \text{ s.t. } \begin{bmatrix} 0 & x_{12} & 0 \\ x_{12} & x_{22} & 0 \\ 0 & 0 & 1 + x_{12} \end{bmatrix} \succeq 0.$$

In order to set up the dual program we write cost and constraint coefficients in matrix form,

$$C = \begin{bmatrix} 0 & \frac{1}{2} & 0 \\ \frac{1}{2} & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}, A_1 = \begin{bmatrix} 0 & -\frac{1}{2} & 0 \\ -\frac{1}{2} & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}, A_2 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}, A_3 = \begin{bmatrix} 0 & 0 & 1 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \end{bmatrix}, A_4 = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix}.$$

The right hand side vector reads $b = (1, 0, 0, 0)^T$. Dualizing by the standard procedure yields

$$\max\ y_1 \ \text{ s.t. } Z = C - y_1A_1 - y_2A_2 - y_3A_3 - y_4A_4 \succeq 0.$$

The dual program can be written in the form

$$\max\ y_1 \ \text{ s.t. } Z = \begin{bmatrix} -y_2 & \frac{1+y_1}{2} & -y_3 \\ \frac{1+y_1}{2} & 0 & -y_4 \\ -y_3 & -y_4 & -y_1 \end{bmatrix} \succeq 0.$$

A necessary condition for the primal matrix to be positive semidefinite is that x_{12} is zero, because $x_{11} = 0$. Likewise, we obtain from $z_{22} = 0$ that $z_{12} = 0$ and hence $y_1 = -1$ in the dual program. The gap between any pair of primal and dual optimal solutions is one.

This insufficiency of the primal-dual pair is due to the dualization procedure (2) which is purely algebraic and does not take into account the actual geometry of the feasible sets. In this particular example, the primal equality constraints imply that any feasible $X \succeq 0$ has a zero eigenvalue *with eigenvector* $(1,0,0)^T$*. By Theorem 2.5, the primal feasible set is contained in a face of the semidefinite cone that has the following form,*

$$F = \left\{PWP^T : W \succeq 0\right\} \quad \textit{with} \quad P = \begin{bmatrix} 0 & 0 \\ 1 & 0 \\ 0 & 1 \end{bmatrix}.$$

The particular choice of P *is convenient, but any* P *whose columns form a basis of the space orthogonal to the nullspace of the feasible set will do as well. If we replace the condition* $X \succeq 0$ *by* $X \in F$ *in the primal problem, the primal problem remains unchanged. Constructing the dual in analogy to (2) yields that the corresponding* Z *must be positive semidefinite with respect to the subspace spanned by* P*. Applying this to the current example we obtain*

$$\begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} -y_2 & \frac{1+y_1}{2} & -y_3 \\ \frac{1+y_1}{2} & 0 & -y_4 \\ -y_3 & -y_4 & -y_1 \end{bmatrix} \begin{bmatrix} 0 & 0 \\ 1 & 0 \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} 0 & -y_4 \\ -y_4 & -y_1 \end{bmatrix} \succeq 0.$$

For this specialized dual problem the optimal solution is attained for $y_1 = 0$ *and the gap between the optimal values of the primal problem and the specialized dual problem has disappeared.*

The gap between optimal primal and dual objective value is guaranteed to be zero (*strong duality* holds) if at least one of (PSDP) and (DSDP) has a *strictly feasible* point.

Definition 4.2
A point X *is* strictly feasible *for (PSDP) if it is feasible for (PSDP) and satisfies* $X \succ 0$*.*
A pair (y, Z) *is* strictly feasible *for (DSDP) if it is feasible for (DSDP) and satisfies* $Z \succ 0$*.*
A semidefinite program is strictly feasible *if it has strictly feasible solutions.*

The assumption of the existence of such a point is a Slater-type regularity condition that form sufficient conditions for strong duality in general convex programming (see, *e.g.*, [39]).

Theorem 4.3 *(Strong Duality)*
Let $p^* = \inf\left\{\langle C, X\rangle : \mathcal{A}X = b, X \succeq 0\right\}$ *and* $d^* = \sup\left\{\langle b, y\rangle : \mathcal{A}^T y + Z = C, Z \succeq 0\right\}$*.*
(i) If (PSDP) is strictly feasible with p^* *finite, then* $p^* = d^*$ *and this value is attained for (DSDP).*
(ii) If (DSDP) is strictly feasible with d^* *finite, then* $p^* = d^*$ *is attained for (PSDP).*
(iii) If (PSDP) and (DSDP) are both strictly feasible, then $p^* = d^*$ *is attained for both problems.*

The following folklore example illustrates that the primal optimal solution may not be attained if there is no strictly feasible dual solution.

Example 4.4

$$\min\ x_{11}\ \text{ s.t. } \begin{bmatrix} x_{11} & 1 \\ 1 & x_{22} \end{bmatrix} \succeq 0 \qquad\qquad \max\ 2y_1\ \text{ s.t. } \begin{bmatrix} 1 & -y_1 \\ -y_1 & 0 \end{bmatrix} \succeq 0.$$

The primal problem has a strictly feasible solution ($x_{11} = 2$, $x_{22} = 2$)*, the dual optimal solution* 0 *is attained for* $y_1 = 0$ *which is also the only feasible solution. Because of the semidefiniteness constraint, the determinants of the primal principal submatrices must be nonnegative:* $x_{11} \geq 0$*,* $x_{22} \geq 0$*,* $x_{11}x_{22} - 1 \geq 0$*. This yields the lower bound* $x_{11} \geq \frac{1}{x_{22}}$ *which is zero for* $x_{22} \to \infty$*. The primal optimal value is not attained. Note, that the dual does not have a strictly feasible solution and that the primal feasible set is not polyhedral.*

What can we do if the semidefinite program at hand is not strictly feasible? If we know the minimal face of the positive semidefinite cone that contains the feasible set, then we can project the problem onto this face as described in Example 4.1 and obtain a well posed problem [44]. The minimal cone can be constructed explicitly if a point in the relative interior of the feasible set is known.

Proposition 4.5 *Let $X \in \mathcal{X} = \{X \succeq 0 : \mathcal{A}X = b\}$ with eigenvalue decomposition $X = P\Lambda P^T$, $P^T P = I$ and $\Lambda \succ 0$ diagonal. Denote by $S_P = \{PVP^T : V \succeq 0\}$ the face of S_n^+ spanned by P. Then X is in the relative interior of $\mathcal{X}$ if and only if S_P is the smallest face of S_n^+ containing $\mathcal{X}$.*

If we do not know this minimal face, then there is no obvious way to arrive at an equivalent well posed problem. In theory it is possible to construct, via an algebraic description of the minimal face, an extended dual semidefinite program. This dual, called the *extended Slater dual*, guarantees that the gap between primal optimal value and dual optimal value is zero [38]. The construction is quite involved and requires the introduction of several additional semidefinite variables. Although the extended Slater dual can be obtained in polynomial time, the computational burden is too high for practical applications.

5 Geometry and Complexity

Next, we study the facial structure of feasible sets. A feasible set of a semidefinite program is the intersection of an affine subspace with the semidefinite cone. The faces of intersections of convex sets are the intersections of the faces of the convex sets. Consequently, the facial structure of the semidefinite cone has a strong influence on the facial structure of the feasible set. One such consequence is that in general feasible sets are not polyhedral. Another one is that optimal solutions are likely to have small rank, because the low dimensional faces of the semidefinite cone have small rank. The following lemma gives a mathematically precise formulation of this phenomenon.

Theorem 5.1 *([35])*
(i) Let F be a face of dimension k of the feasible set of (PSDP). For $X \in F$ the rank $r = \operatorname{rank}(X)$ is bounded by $\binom{r+1}{2} \le m + k$.
(ii) Let F be a face of dimension k of the set $\{Z \succeq 0 : \exists y \in \mathbb{R}^m : Z + \mathcal{A}^T y = C\}$ of feasible Z-values of (DSDP). For $Z \in F$ the rank $r = \operatorname{rank}(Z)$ is bounded by $\binom{r+1}{2} \le \binom{n+1}{2} - m + k$.

Sensitivity analysis in semidefinite programming is considerably more involved than in linear programming and out of the scope of this survey. We refer the interested reader to [6] and references therein.

It is well known that "under reasonable assumptions" convex programming is of polynomial complexity. In particular, if a full dimensional compact convex set is given by a weak violation oracle with its "interesting region" contained in a ball centered at the origin with radius R, then there exists an oracle polynomial time algorithm that solves the weak optimization problem [15]. In the case of semidefinite programming, a polynomial violation oracle is Gaussian elimination pivoting on diagonal elements. The next two examples illustrate why the radius R and the "weakness" of the optimization problem are needed for semidefinite programming.

Example 5.2 *([38])*

$$\min\ x_m \ \text{ s.t. } (x_1 - 4) \succeq 0, \begin{bmatrix} 1 & x_1 \\ x_1 & x_2 \end{bmatrix} \succeq 0, \begin{bmatrix} 1 & x_2 \\ x_2 & x_3 \end{bmatrix} \succeq 0, \dots, \begin{bmatrix} 1 & x_{m-1} \\ x_{m-1} & x_m \end{bmatrix} \succeq 0.$$

The encoding length of this program is $O(m)$. The primal feasible set can also be described by $x_1 \ge 2^2$, $x_2 \ge x_1^2 \ge (2^2)^2 = 2^{(2^2)}$, $x_3 \ge x_2^2 \ge 2^{(2^3)}$, $\dots$, $x_m \ge x_{m-1}^2 \ge 2^{(2^m)}$. A strictly feasible solution exists and the optimal solution is obtained by setting all variables to their respective lower bounds. But the optimal solution is doubly exponential in m and the feasible region is doubly exponentially far away from the origin, i.e., *R grows doubly exponentially in m. Thus, the encoding length of any feasible solution is $\Omega(2^m)$ if binary encoding is used.*

Example 5.3

$$\min\ x_{12} \ \text{ s.t. } \begin{bmatrix} 1 & x_{12} \\ x_{12} & 2 \end{bmatrix} \succeq 0.$$

Although all coefficients are integers, the optimal solution is $x_{12} = -\sqrt{2}$*. In contrast to linear programming we cannot expect solutions to be rational numbers when coefficients are restricted to integers.*

Strong bounds on the complexity of semidefinite programming were obtained by employing complexity results of the first order theory over the reals, where solutions may be described as the roots of polynomials with integral coefficients.

Theorem 5.4 ([37]) *For integral* $\mathcal{A} : S_n \to \mathbb{R}^m$ *and* $b \in \mathbb{Z}^m$ *let* $\mathcal{X} = \{X \succeq 0 : \mathcal{A}X \leq b\}$*, and let* l *denote the maximum length of a binary encoding of the coefficients in* $\mathcal{A}$ *and* b*.*

1. *If* $\mathcal{X} \neq \emptyset$*, then there exists an* $X \in \mathcal{X}$ *satisfying* $\|X\| \leq R$ *with* $\log R = l \cdot n^{O(\min\{m,n^2\})}$*. If, in addition,* $\mathcal{X}$ *is bounded, then* $\|X\| \leq R$ *for all* $X \in \mathcal{X}$*.*
2. *It can be tested in* $mn^{O(\min\{m,n^2\})}$ *arithmetic operations over* $l \cdot n^{O(\min\{m,n^2\})}$ *bit numbers whether* $\mathcal{X}$ *is empty.*

Thus, for fixed m, the feasibility problem can be solved in polynomial time.

Another important result, due to the same authors, states that integer semidefinite programming is polynomially solvable in fixed dimensions. In this case, the coefficients of $\mathcal{A}$ and b may be algebraic numbers, *i.e.*, roots of polynomials.

Theorem 5.5 ([23]) *For fixed* n *there exists a polynomial time algorithm that finds an integral* $X \in S_n^+$ *satisfying* $\mathcal{A}X \leq b$ *or decides that no such matrix exists.*

Note that the result does not depend on the number of constraints m, but only on the dimension of X.

In the Turing model of computation, semidefinite programming is either in *NP*∩*co-NP* or outside *NP*∪*co-NP*. In the real number model of computation it is, in fact, in *NP*∩*co-NP*. It is not known, however, whether in the Turing model semidefinite programming is in *NP* or not [38].

6 Applications

Extensive lists of applications from various areas have been compiled in [30] and [43]. We pick out four basic examples.

6.1 Control Theory

The book [7] contains a wealth of applications in control theory, the following example among others.

A *polytopic linear differential inclusion* is described by

$$\text{(PLDI)} \quad \dot{x} = A(t)x \quad \text{with } A(t) \in \operatorname{conv}\{A_1, \ldots, A_k\} \subset M_n.$$

The symbol $\dot{x}$ denotes $\frac{d}{dt}x(t)$, as usual; the function $x(t)$ can be interpreted as the state of the system over time. A (PLDI) allows to model linear systems where the matrix $A(t)$ is uncertain over time, but is known to remain in the convex hull of matrices A_1 to A_k. A (PLDI) is called *stable*, if all trajectories of (PLDI) converge to zero as $t \to \infty$. This is certainly satisfied if there is a norm $\|x\|_H = \sqrt{x^T H x}$ with $H \succ 0$ so that $\frac{d}{dt}\|x(t)\|_H^2 < 0$ on all trajectories. The system is then called *quadratically stable*, and the function $x^T H x$ is called a *quadratic Lyapunov function*. Since $\frac{d}{dt}x^T H x = x^T(A(t)^T H + HA(t))x$, the condition on the derivative is equivalent to the existence of a feasible solution in the open semidefinite set

$$H \succ 0, \qquad A_i^T H + HA_i \prec 0 \quad \text{for } i = 1, \ldots, k. \tag{8}$$

Interior point methods (see Section 7.1) allow to solve these problems directly. Alternatively, a corresponding semidefinite program (in fact, an eigenvalue optimization problem, see Section 6.3) could read

$$\max \ \lambda \ \text{ s.t. } H \succeq \lambda I, \quad A_i^T H + H A_i \preceq -\lambda I \quad \text{for } i = 1, \ldots, k.$$

Now consider the case, that one would like to force the state of the system to zero by linear state-feedback control, *i.e.*, by a control term $B(t)u$ where the control variables $u \in \mathbb{R}^m$ depend linearly on the state x, $u = Kx$. Then we are dealing with a *closed-loop* PLDI,

$$\dot{x} = (A(t) + B(t)K)x \quad \text{with } A(t) \in \operatorname{conv}\{[A_1, B_1], \ldots, [A_k, B_k]\} \subset M_{n,n+m}. \tag{9}$$

The system $\dot{x} = A(t)x + B(t)u$ is called *quadratically stabilizable* via linear state-feedback if there exists a K so that (9) is quadratically stable. Scaling (8) for matrices $A_i + B_i K$ by $Q = H^{-1}$ gives rise to the necessary and sufficient condition

$$Q \succ 0, \qquad Q(A_i + B_i K)^T + (A_i + B_i K)Q \prec 0 \quad \text{for } i = 1, \ldots, k.$$

In order to remove the bilinear terms in Q and K we substitute $KQ = Y$,

$$Q \succ 0, \qquad QA_i^T + A_i Q + YB_i^T + B_i Y \prec 0 \quad \text{for } i = 1, \ldots, k.$$

If this system is feasible, appropriate Q and Y can be determined algorithmically. The matrix K is then reconstructed by $K = YQ^{-1}$.

6.2 Combinatorial Optimization

A comprehensive survey on combinatorial applications is given in [12]. Probably the most important is quadratic $\{-1, 1\}$ programming,

$$\max_{x \in \{-1,1\}^n} x^T C x \quad \Longleftrightarrow \quad \max_{x \in \{-1,1\}^n} \left\langle C, xx^T \right\rangle.$$

For $x \in \{-1, 1\}^n$ the matrix xx^T is positive semidefinite of rank one. All its diagonal elements are equal to one, $\operatorname{diag}(xx^T) = e$ (e denoting the vector of all ones). In fact, these properties describe the discrete set of matrices that are dyadic products of $\{-1, 1\}$ vectors, $\{X \succeq 0 : \operatorname{diag} X = e, \operatorname{rank}(X) = 1\} = \{xx^T : x \in \{-1, 1\}^n\}$. By dropping the rank one constraint, one enlarges the discrete set in order to obtain a polynomially solvable convex relaxation,

$$\max_{x \in \{-1,1\}^n} \left\langle C, xx^T \right\rangle \quad \leq \quad \max \ \langle C, X \rangle \ \text{ s.t. } \operatorname{diag}(X) = e, X \succeq 0.$$

The semidefinite relaxation on the right first appears in [41, 10] and forms the basis of an intriguing approximation algorithm, due to Goemans and Williamson [13]: A feasible solution $\hat{X}$ of the semidefinite program is interpreted as the Gram matrix $\hat{X} = V^T V$ of the columns v_i of V. A random vector h is drawn according to the standard normal distribution in $\mathbb{R}^n$. Using h, the columns of V are rounded to $\{-1, 1\}$-values, $\bar{x}_i = \operatorname{sgn}(h^T v_i)$.

The probability that two vectors v_i and v_j give rise to different values is proportional to the angle $\arccos(v_i^T v_j)$ between v_i and v_j. Notice, $\|v_i\| = 1$ because $\operatorname{diag}(\hat{X}) = e$. This leads to an expected value of $E(\bar{x}^T C \bar{x}) = \frac{2}{\pi} \sum_{i,j} c_{ij} \arcsin(v_i^T v_j) = \frac{2}{\pi} \sum_{i,j} c_{ij} \arcsin(\hat{x}_{ij})$. Even though $\arcsin(\hat{x}_{ij})$ is easily upper and lower bounded by affine functions in x_{ij}, the dependence on the signs of the c_{ij} makes it difficult to bound $E(\bar{x}^T C \bar{x})$ in terms of the objective value $\langle C, \hat{X} \rangle$. For this, special properties of the cost matrices have to be exploited. In the case of cost matrices corresponding to the max-cut problem on graphs with nonnegative edge weights, Goemans and Williamson showed that the expected value is at least 0.878 times the objective value, $E(\bar{x}^T C \bar{x}) > 0.878 \langle C, \hat{X} \rangle$. Applying this rounding procedure to the optimal solution X_* of the semidefinite relaxation, yields an approximation algorithm for max-cut with performance guarantee of 0.878. The approach was extended to max-k-cut and max-bisection [11], coloring [22], and inspired much further work.

The semidefinite relaxation is equivalent, via scaling [16], to the semidefinite relaxation of quadratic $\{0, 1\}$-programming of Lovász and Schrijver [28] that forms the basis of a generic procedure for deriving semidefinite relaxations of constrained quadratic $\{0, 1\}$-programming problems.

6.3 Eigenvalue Optimization

Eigenvalue optimization is a field in its own right and has many practical applications, see [27] for a survey. Several basic problems in eigenvalue optimization may be formulated as semidefinite programs. For example, minimizing the maximum eigenvalue over an affine matrix function is a semidefinite programming problem,

$$\min_{y\in\mathbb{R}^m} \lambda_{\max}(C-\mathcal{A}^Ty)+b^Ty \quad \Longleftrightarrow \quad \min_{\lambda\in\mathbb{R},y\in\mathbb{R}^m} \lambda+b^Ty \text{ s.t. } \lambda I \succeq C-\mathcal{A}^Ty. \tag{10}$$

Conversely, the dual of any semidefinite program with constant trace $\operatorname{tr} X = \langle I, X\rangle = a > 0$ on the feasible set $\{X \succeq 0 : \mathcal{A}X = b\}$ can be formulated as an eigenvalue optimization problem, because (w.l.o.g. assume $a = 1$)

$$\max_{\substack{\mathcal{A}X=b\\ X\succeq 0}} \langle C, X\rangle = \max_{\substack{\mathcal{A}X=b\\ \langle I,X\rangle=1\\ X\succeq 0}} \langle C, X\rangle = \inf_{\substack{C-\lambda I-\mathcal{A}^Ty\preceq 0\\ \lambda\in\mathbb{R},y\in\mathbb{R}^m}} \lambda+b^Ty = \inf_{y\in\mathbb{R}^m} \lambda_{\max}(C-\mathcal{A}^Ty)+b^Ty.$$

The second equality follows from strong duality (Theorem 4.3), because the dual has a strictly feasible point (choose λ large enough). It can be worked out that any semidefinite programming problem with bounded feasible set can be scaled to a program with constant trace on the feasible set. The construction of this scaling requires some knowledge about the problem and the scaling will in general destroy sparsity of the coefficient matrices.

6.4 Robust Optimization

The optimal solutions of real world optimization problems may be quite sensitive to changes in the data. If such a solution is computed on basis of slightly perturbed data or if the solution cannot be implemented with the necessary precision in practice, then the solution may be worthless. In robust optimization a solution is sought that is optimal among all solutions that are feasible for all reasonable perturbations of the data. In fact, the approach described for polytopic linear differential inclusions above may also be interpreted in this vein and is often referred to as *robust control*. In [4], Ben-Tal and Nemirovski investigate robust formulations of several standard optimization problems for data with ellipsoidal uncertainties. For example, in the case of quadratically constrained convex quadratic programming they show that the robust version

$$\begin{array}{ll} \min & c^Tx \\ \text{s.t.} & x^TA^TAx+2b^Tx+\gamma\le 0 \qquad \forall (A,b,\gamma)\in \bigcup_{i=1}^m \mathcal{U}_i \end{array}$$

with

$$\mathcal{U}_i = \left\{ (A,b,\gamma) = (A_{i0},b_{i0},\gamma_{i0}) + \sum_{j=1}^k u_i(A_{ik},b_{ik},\gamma_{ik}) :\ u^Tu\le 1 \right\} \qquad \text{for } i=1,\ldots,m$$

can be solved via semidefinite programming. Similarly, a particular case of robust semidefinite programming is itself a semidefinite program, namely the case of "rank 2" ellipsoidal uncertainties of the form

$$\begin{array}{ll} \min & \langle b,y\rangle \\ \text{s.t.} & C+\mathcal{A}^Ty+h(y)d^T+d^Th(y)\succeq 0 \qquad \forall h(\cdot)\in\left\{\sum_{i=1}^k u_ih_i(\cdot) :\ u^Tu\le 1\right\} \end{array}$$

where $h_i(\cdot):\mathbb{R}^m\to\mathbb{R}^n$ is an affine function of y for $i=1,\ldots,k$. This special class of problems arises in robust truss topology design, where the truss should not only be as stiff as possible with respect to one particular load, but should also withstand small forces in arbitrary directions acting on the nodes (see [3]). The transformations of the robust formulations to semidefinite programs are outside the scope of this survey and we refer the reader to [4] for details.

7 Algorithms

7.1 Interior Point Methods

Interior point methods are currently the most efficient approach for solving general semidefinite programs. The basic idea is to compute a Newton step in each iteration with respect to an auxiliary barrier problem that keeps the iterates inside the positive semidefinite cone. Since Newton's method exploits second order information and works particularly well on this class of barrier problems, the algorithms converge very fast. An approximately optimal solution is obtained within a polynomial number of iterations. On the other hand, the computation of a single step is computationally rather expensive and structural properties of the constraint matrices are difficult to exploit. Within current technology this limits these methods to problems of moderate size, of about 10000 constraints, say.

Throughout this section we work under the following assumption.

Assumption 7.1 *There exists a strictly feasible X^0 for (PSDP) and a strictly feasible pair (y^0, Z^0) for (DSDP).*

A point satisfying Assumption 7.1 can always be constructed for a so called *skew-symmetric embedding* of the original problem, we refer to [9] for details.

Interior point algorithms start off at an interior point of the cone of positive semidefinite matrices. The iterates are determined as approximate minimizers of a sequence of auxiliary problems that contain an additional barrier term $-\mu \log \det(X)$ in the cost function, where $\mu > 0$ is the so called *barrier parameter* and $-\log \det(X)$ is the *barrier function.* The auxiliary problem reads

$$\min \ \langle C, X \rangle - \mu \log \det(X) \ \text{ s.t. } \ \mathcal{A}X = b, \ X \succ 0.$$

Since $\det(X) = \prod_{i=1}^{n} \lambda_i(X)$, we have $-\log \det(X) = -\sum_{i=1}^{n} \log \lambda_i(X)$ (in linear programming, the matrix is diagonal, $X = \text{Diag}(x)$ with $x \in \mathbb{R}^n_+$, and the barrier function reads $-\sum_{i=1}^{n} \log x_i$). The barrier function grows to infinity if an eigenvalue of X tends to zero, *i.e.*, if X approaches the boundary of the semidefinite cone. Hence, the barrier parameter μ controls the distance of the optimal solution of the auxiliary problem (if it exists it is unique by the strict convexity of the barrier function) to the boundary. For a sequence of barrier problems with $\mu \to 0$, the original cost function eventually dominates on the interior of the feasible set, except within an ε distance of the boundary, and the sequence of minimizers of the barrier problems converges to an optimizer of the original problem.

Remark 7.2 *The barrier function $-\log \det(X)$ belongs to the class of* strongly self concordant functions *(see [30]) which harmonize with Newton's method. Intuitively, Newton's method approximates a function by a quadratic model and solves the minimization problem exactly within this model. The resulting descent direction is the better the less the function deviates from this quadratic model. For strongly self concordant functions the change of the second derivative is bounded by a Lipschitz condition, therefore the quadratic model is of good quality for comparatively large regions.*

Remark 7.3 *If the feasible set is bounded and $C = 0$, then the optimum is independent of the choice of $\mu > 0$ and is called the* analytic center *of the feasible set. In contrast to a geometric definition of center (*e.g.*, the center of gravity) the analytic center depends on the description of the feasible set. To see this, consider the case where an inequality is given twice. The barrier terms will then push the analytic center farther away from this inequality.*

The barrier problem is transformed into an unconstrained problem for $X \succ 0$ by introducing a Lagrange multiplier y for the the equality constraints,

$$\mathcal{L}_\mu(X, y) = \langle C, X \rangle - \mu \log \det(X) + \langle y, b - \mathcal{A}X \rangle \,.$$

An optimal solution of the barrier problem is a saddle point of the Lagrange function $\mathcal{L}_\mu$. Saddle points satisfy the KKT-conditions, or first order necessary conditions, for $\mathcal{L}_\mu$. By matrix calculus the gradient of $\log\det(X)$ is $\nabla_X \log\det(X) = X^{-1}$, so we arrive at the system

$$\begin{aligned} \nabla_X \mathcal{L}_\mu &= C - \mu X^{-1} - \mathcal{A}^T y = 0 \\ \nabla_y \mathcal{L}_\mu &= b - \mathcal{A}X = 0. \end{aligned}$$

In a primal-dual formulation we set $Z = \mu X^{-1}$ and rewrite the KKT-conditions in the following form (starting from the dual barrier problem one would arrive at the same system),

$$\begin{aligned} \mathcal{A}X &= b, \quad X \succ 0 \\ \mathcal{A}^T y + Z &= C, \quad Z \succ 0 \\ XZ &= \mu I. \end{aligned} \tag{11}$$

The first line requires primal feasibility, the second dual feasibility, and the third is a perturbed complementarity condition. In linear programming, X and Z are both diagonal matrices, and the perturbed complementarity condition reduces to $x_i z_i = \mu$ for $i = 1 \ldots, n$. For semidefinite programming the full matrix product XZ is needed. The solution of the system is unique with respect to X and Z, but not necessarily for y. By Assumption 7.1, however, the system $\mathcal{A}X = b$ is consistent and it is possible to eliminate all y-variables as explained in (3). The sole purpose of y is to span the feasible set of Z. Therefore we will concentrate on the variables X and Z, and use y only if it is convenient.

We denote the solution of system (11) for some fixed μ by (X_μ, Z_μ). X_μ and Z_μ are the unique optimal solutions of the respective barrier problems. They are feasible points of the original problems with a gap of $\langle Z, X\rangle = n\mu$ between the objective values, cf. (7). The set of solutions (X_μ, Z_μ) for $\mu > 0$ forms the so called *central path* which is a smooth curve. For $\mu \to 0$ it converges to a point (X^*, Z^*) with X^* an optimal solution of the original primal and Z^* an optimal solution of the original dual problem.

In linear programming the central path converges to a strictly complementary solution, *i.e.*, a primal-dual pair of optimal vectors $x, z \in \mathbb{R}^n$ with either $x_i = 0$ or $z_i = 0$ (but not both) for $i = 1 \ldots n$. In order to compare this to semidefinite programming we have to clarify what complementarity means in this setting. For this, we have to introduce the following concept. Two matrices $A, B \in M_n$ are *simultaneously diagonalizable* if A and B share a common basis $S \in M_n$ so that $S^{-1}AS$ and $S^{-1}BS$ are both diagonal.

Theorem 7.4 *(see [20]) $A, B \in M_n(\mathbb{C})$ are simultaneously diagonalizable if and only if A and B commute (*i.e.*, if $AB = BA$). In particular, for $A, B \in S_n$, the product $AB \in S_n$ if and only if there is an orthonormal matrix P that diagonalizes A and B.*

For any primal-dual pair of optimal solutions $(\hat{X}, \hat{Z})$, the inner product $\langle \hat{X}, \hat{Z}\rangle$ is zero. Via eigenvalue decomposition this implies that $\hat{X}\hat{Z} = 0$ and so $\hat{X}$ and $\hat{Z}$ are simultaneously diagonalizable. The non-zero eigenvectors of any optimal primal solution $\hat{X}$ are in the null space of any optimal dual solution $\hat{Z}$ and vice versa. In other words, the two minimal faces of the semidefinite cone containing the respective convex sets of primal and dual optimal solutions are spanned by orthogonal subspaces of $\mathbb{R}^n$. This motivates the following definition.

Definition 7.5 *A pair of optimal solutions (X^*, Z^*) is* maximally complementary *if X^* and Z^* have maximal rank among all optimal solutions.*
An optimal pair (X^*, Z^*) is strictly complementary *if* $\operatorname{rank}(X^*) + \operatorname{rank}(Z^*) = n$.

In contrast to linear programming, the existence of strictly complementary solutions is not guaranteed. However, the central path gets as close to strict complementarity as possible. The point to which the central path converges for $\mu \to 0$ is maximally complementary.

Theorem 7.6 *([9]) For a sequence $\mu_k > 0$, $k \in \mathbb{N}$, with $\mu_k \to 0$, the corresponding solutions (X_{μ_k}, Z_{μ_k}) of (11) converge to a maximally complementary, optimal pair (X^*, Z^*) of (PSDP) and (DSDP).*

In other words, the central path converges to a point in the relative interior of the optimal face.

We would like to compute an approximate solution of (11), *i.e.*, a solution to

$$F_\mu(X,y,Z) = \begin{pmatrix} \mathcal{A}X - b \\ \mathcal{A}^T y + Z - C \\ XZ - \mu e \end{pmatrix} = 0.$$

Newton's method computes a step direction $(\Delta X, \Delta y, \Delta Z)$ by solving $F_\mu + \nabla F_\mu \cdot (\Delta X, \Delta y, \Delta Z)^T = 0$. Here, the step direction can be determined by the linearized system

$$\begin{aligned} \mathcal{A}(\Delta X) &= -(\mathcal{A}X - b) && (12) \\ \mathcal{A}^T(\Delta y) + \Delta Z &= -(\mathcal{A}^T y + Z - C) && (13) \\ \Delta X Z + X \Delta Z &= \mu I - XZ. && (14) \end{aligned}$$

Unfortunately, X and Z do not commute, $XZ \neq ZX$, and the same is true in the linearization of the complementarity condition (quite in contrast to the case of linear programming, where all matrix variables are diagonal). In general we cannot expect that there exist symmetric ΔX and ΔZ that solve (12) to (14). Solving the system for square matrices $\Delta X, \Delta Z \in M_n$ yields a symmetric ΔZ because of (13) but (in general) an unsymmetric ΔX. Since the next iterate $X + \alpha \Delta X$ has to be a symmetric positive definite matrix this is a serious problem. A number of approaches have been proposed to get around this difficulty. We present only three (see [42] for a survey on search directions).

The first approach [19, 26] allows ΔX to be unsymmetric in order to guarantee that the system is solvable. The skew-symmetric part of ΔX is then ignored and the symmetric part constitutes the new step direction,

$$\Delta \hat{X} Z + X \Delta Z = \mu I - XZ, \qquad \Delta X = \frac{\Delta \hat{X} + \Delta \hat{X}^T}{2}. \tag{15}$$

The second approach is based on the concept of *self-scaled barrier functions* for self-scaled cones [32, 31]. A special scaling matrix $W = X^{\frac{1}{2}}(X^{\frac{1}{2}} Z X^{\frac{1}{2}})^{-\frac{1}{2}} X^{\frac{1}{2}} \in S_n^+$ satisfying $W^{-\frac{1}{2}} X W^{-\frac{1}{2}} = W^{\frac{1}{2}} Z W^{\frac{1}{2}}$ is used to reformulate the complementarity condition,

$$W^{-1} \Delta X W^{-1} + \Delta Z = \mu X^{-1} - Z. \tag{16}$$

Any solution to this system is guaranteed to be symmetric.

In a third approach [1], the complementarity condition (14) is modified so as to allow for symmetric updates only. Consider the linearization of $XZ + ZX - \mu I$,

$$\Delta X Z + X \Delta Z + Z \Delta X + \Delta Z X = 2\mu I - XZ - ZX. \tag{17}$$

Symmetrization is implicit, the existence of symmetric solutions ΔX and ΔZ is guaranteed if X and Z are "close" to the central path.

The three search directions presented can be generalized by introducing a symmetrization operator [45]

$$H_P(M) = \frac{1}{2}(PMP^{-1} + (PMP^{-1})^T) \tag{18}$$

with a given nonsingular matrix $P \in S_n$. Symmetrizing the complementarity condition by this operator yields

$$H_P(XZ + \Delta X Z + X \Delta Z) = \mu I. \tag{19}$$

The choice $P = I$ corresponds to search direction (17). For $P = Z^{\frac{1}{2}}$ it is equivalent to (15). It can be worked out that (16) is obtained by choosing P so that $P^T P = W$.

The search directions (15), (16), and (17) are currently the most popular in practical implementations. Other interesting choices may show up in the future. The algorithmic framework for all these methods or even hybrid methods is the same.

Algorithm 7.7

Input: *$\mathcal{A}$, b, C, and some starting point (X^0, y^0, Z^0) with $X^0 \succ 0$ and $Z^0 \succ 0$ (usually this starting point will have to satisfy some additional conditions).*

1. *Choose μ.*
2. *Compute $(\Delta X, \Delta y, \Delta Z)$ by solving (12), (13) together with a variant of (14).*
3. *Choose some $\alpha \in (0, 1]$ so that $X + \alpha\Delta X$ and $Z + \alpha\Delta Z$ remain positive definite.*
4. *Set $(X, y, Z) := (X + \alpha\Delta X, y + \alpha\Delta y, Z + \alpha\Delta Z)$.*
5. *If $\|\mathcal{A}X - b\|$ and $\|\mathcal{A}^T y + Z - C\|_{\mathrm{F}}$ and $\langle X, Z\rangle$ are small enough then **stop**, else **goto** 1.*

In order to prove polynomial iteration complexity for a particular scheme of search directions, rather strong restrictions must be imposed on the starting point and the specific choices of μ and α. The best complexity results are obtained for feasible methods that closely trace the central path. For these methods, a primal-dual feasible pair (X, y, Z) with $\langle X, Z\rangle < \varepsilon$ can be found within $O(\sqrt{n}\log(\langle X^0, Z^0\rangle/\varepsilon))$ iterations, where (X_0, y_0, Z_0) is a feasible starting point close to the central path (see, *e.g.*, [29]).

The amount of work per iteration depends on the search direction. We explain the main steps of method (15). System (12)-(14) is solved by expressing ΔX in terms of ΔZ in (14) and ΔZ in terms of Δy in (13). Now it remains to solve (12) for Δy,

$$\mathcal{A}(X\mathcal{A}^T(\Delta y)Z^{-1}) = b - \mathcal{A}(\mu Z^{-1} + X(C - \mathcal{A}^T y - Z)Z^{-1}). \tag{20}$$

The matrix $M = \mathcal{A}(X\mathcal{A}^T(\cdot)Z^{-1}) \in S_m$ is positive definite if $X \succ 0$, $Z \succ 0$, and $\mathcal{A}$ has full row rank. Its elements are determined by $M_{ij} = \mathrm{tr}(XA_iZ^{-1}A_j)$. In general, the construction of M takes $O(mn^3 + m^2n^2)$ operations, but structural properties of the A_i can be exploited to speed up this step for particular inputs. Since either X or Z^{-1} is dense, M is typically dense even in the case of sparse data; The factorization of M needs $m^3/3$ operations and is usually the most expensive step in each iteration. If n is of the same order of magnitude as m, then the line search in step 3 of Algorithm 7.7 is as important as the factorization of M.

For a large but structured Z, Benson, Ye, and Zhang [5] pointed out that a dual *potential reduction method*, *i.e.*, a pure dual method, may help to save work. Linearizing $X = \mu Z^{-1}$ (instead of $XZ = \mu I$) yields

$$X + \Delta X = \mu Z^{-1} - \mu Z^{-1}\Delta Z Z^{-1} \tag{21}$$

in (14). For this choice and strictly feasible $Z = C - \mathcal{A}^T(y) \succ 0$, (20) transforms to

$$\mathcal{A}(Z^{-1}\mathcal{A}^T(\Delta y)Z^{-1}) = b/\mu - \mathcal{A}(Z^{-1}). \tag{22}$$

The resulting Δy is the Newton step for the dual barrier problem. The primal matrix X does not appear and it is not necessary to compute Z^{-1} explicitly. Indeed, any product $W = Z^{-1}A_i$ can be computed by solving $ZW = A_i$ and this allows to exploit the structure of a sparse factorization of Z. In the line search, sparse factorizations are also sufficient to test for positive semidefiniteness of Z. By (21), a feasible $X + \Delta X$ is obtained if $Z^{-\frac{1}{2}}\Delta Z Z^{-\frac{1}{2}} \preceq I$. A sufficient condition for this is that

$$1 \geq \|Z^{-\frac{1}{2}}\Delta Z Z^{-\frac{1}{2}}\|_{\mathrm{F}}^2 = \|Z^{-\frac{1}{2}}\mathcal{A}^T(\Delta y)Z^{-\frac{1}{2}}\|_{\mathrm{F}}^2 = \mathcal{A}(Z^{-1}\mathcal{A}^T(\Delta y)Z^{-1})^T\Delta y = [b/\mu - \mathcal{A}(Z^{-1})]^T\Delta y.$$

If this is the case, then the primal objective value can be computed by $\langle C, X + \Delta X\rangle = b^Ty + \langle X + \Delta X, Z\rangle = b^Ty + \mu n - \mu\langle\mathcal{A}^T(\Delta y), Z^{-1}\rangle = b^Ty + \mu n - \mu\Delta y^T\mathcal{A}(Z^{-1})$ (use (7), (21), and $\Delta Z = \mathcal{A}^T\Delta y$). This yields a reliable stopping criterion at no additional cost.

Unfortunately, $\bar{M} = \mathcal{A}(Z^{-1}\mathcal{A}^T(\cdot)Z^{-1})$ of (22) with $\bar{M}_{ij} = \mathrm{tr}(Z^{-1}A_iZ^{-1}A_j)$ is still dense, so the approach is again limited to a moderate number of constraints, say m not larger than 10000 on current workstations.

7.2 The Spectral Bundle Method

The spectral bundle method of [18] is a specialization of the proximal bundle method of [24] to eigenvalue optimization problems of the form (cf. Section 6.3)

$$\text{(E)} \qquad \min_{y \in \mathbb{R}^m} f(y) \text{ with } f(y) = \lambda_{\max}(C - \mathcal{A}^T y) + b^T y.$$

The main purpose of formulating certain semidefinite programs as eigenvalue optimization problems is to open possibilities for exploiting structure. If the matrix $C - \mathcal{A}^T y$ is well structured in the sense that multiplication of this matrix with a vector is inexpensive, then Lanczos methods allow to compute the maximum eigenvalue and a corresponding eigenvector efficiently. Eigenvectors to the maximal eigenvalue give rise to subgradients of f. The subgradients are used to form a simplified model of f that is easier to solve. The solutions of the model yield new candidates at which f is again evaluated. In the case of a better objective value the algorithm moves on to the new point, otherwise the new subgradient is used to improve the model. Subgradient methods are an attractive choice for large scale problems (with n and m above 1000 and 10000, respectively) whenever the value of f and a corresponding eigenvector can be determined quickly. They are first order methods and require little memory. They show fast improvement in the beginning but slow down significantly as the optimal solution is approached. Therefore they should best be employed if a rough estimate of the optimal solution is sufficient.

We start by investigating f. Since $\lambda_{\max}(A) = \max_{v^T v=1} v^T A v$, $v^T A v = \langle A, vv^T \rangle$, and $\operatorname{conv}\{vv^T : v^T v = \operatorname{tr}(vv^T) = 1\} = \{W \succeq 0 : \operatorname{tr} W = 1\} =: \mathcal{W}$, the maximum eigenvalue function may equivalently be formulated as a semidefinite program,

$$\lambda_{\max}(A) = \max\{\langle A, W \rangle : W \in \mathcal{W}\}. \tag{23}$$

Therefore, any subset $\widehat{\mathcal{W}} \subset \mathcal{W}$ gives rise to a model function $f_{\widehat{\mathcal{W}}}$ minorizing f,

$$f_{\widehat{\mathcal{W}}}(y) := \max_{W \in \widehat{\mathcal{W}}} \langle C - \mathcal{A}^T y, W \rangle + b^T y \quad \text{ with } \quad f_{\widehat{\mathcal{W}}}(y) \le f_{\mathcal{W}}(y) = f(y).$$

Since f is the maximum of the linear functions $\langle b - \mathcal{A}W, y \rangle + \langle C, W \rangle$ over $W \in \mathcal{W}$, the function f is convex and the subdifferential of f at y is generated by all maximizers W at y,

$$\partial f(y) = \left\{ b - \mathcal{A}W : W \in \mathcal{W}, \langle C - \mathcal{A}^T y, W \rangle = \lambda_{\max}(C - \mathcal{A}^T y) \right\} \tag{24}$$

In particular, any eigenvector v of $\lambda_{\max}(C - \mathcal{A}^T y)$ gives rise to a subgradient $b - \mathcal{A}(vv^T)$ of f at y. A point y is optimal if and only if $0 \in \partial f(y)$, *i.e.*, there is a $W_* \in \operatorname{Argmax}_{W \in \mathcal{W}} \langle C - \mathcal{A}^T y, W \rangle$ with $\mathcal{A}W = b$.

We now explain the proximal bundle approach within our specialized setting. At iteration k, a new candidate y^{k+1} is determined as the minimizer of $f_{\widehat{\mathcal{W}}^k}(y) + \frac{u}{2}\|y - \hat{y}^k\|^2$, where $\widehat{\mathcal{W}}^k \subset \mathcal{W}$ is formed from accumulated subgradient information. The quadratic term $\|y - \hat{y}^k\|^2$ keeps the new candidate close to the last successful iterate $\hat{y}^k$ (sometimes called the *stability center*). The *weight* u provides some indirect control on this distance. In the evaluation step, $\lambda_{\max}(C - \mathcal{A}^T y^{k+1})$ and a corresponding eigenvector v (or some $W_s^{k+1} \in \operatorname{Argmax}_{W \in \mathcal{W}} \langle C - \mathcal{A}^T y^{k+1}, W \rangle$) giving rise to a subgradient are computed. If $f(y^{k+1})$ satisfies a sufficient decrease criterion then the algorithm moves to y^{k+1} by setting $\hat{y}^{k+1} = y^{k+1}$. This is a *descent step*. Otherwise y^{k+1} is ignored, $\hat{y}^{k+1} = \hat{y}^k$, but the subgradient information is used to improve the model at y^{k+1} in $\widehat{\mathcal{W}}^{k+1}$. This is a *null step*.

In the spectral bundle method of [18], the set $\widehat{\mathcal{W}}^k$ is restricted to the form

$$\widehat{\mathcal{W}}^k = \left\{ P_k V P_k^T + \alpha \overline{W}_k : \operatorname{tr} V + \alpha = 1, V \in S_{r_k}^+, \alpha \ge 0 \right\}, \tag{25}$$

where $P_k \in M_{n,r_k}$ is an orthonormal matrix and $\overline{W}_k \in \mathcal{W}$. For this $\widehat{\mathcal{W}}^k$,

$$f_{\widehat{\mathcal{W}}^k}(y) = \max\left\{ \lambda_{\max}(P_k^T (C - \mathcal{A}^T y) P_k), \langle C - \mathcal{A}^T y, \overline{W}_k \rangle \right\} + b^T y \le f(y). \tag{26}$$

Thus, for small r_k, the value of the cutting plane model can be determined efficiently. In order to obtain a large value of $f_{\widehat{\mathcal{W}}^k}$ in the vicinity of the current minimizer y, the matrix P_k should span the eigenspaces of the largest eigenvalues of $C - \mathcal{A}^T y$.

Without $\overline{W}_k$ the set $\widehat{\mathcal{W}}^k$ corresponds to a $\binom{r_k+1}{2}$ dimensional face of the semidefinite cone (Theorem 2.5), which might be too small to contain an optimal W_*. The matrix $\overline{W}_k$ allows $\widehat{\mathcal{W}}^k$ to reach into the interior of S_n^+ without significantly increasing the cost of computing the next trial point.

Using the strong duality theorem, it can be worked out that solving the augmented model $f_{\widehat{\mathcal{W}}^k}(y) + \frac{u}{2}\|y - \hat{y}^k\|^2$ is equivalent to solving its dual,

$$\begin{aligned} &\min_{y\in\mathbb{R}^m} \max_{W\in\widehat{\mathcal{W}}^k} \left\langle C - \mathcal{A}^T y, W\right\rangle + b^T y + \frac{u}{2}\|y - \hat{y}^k\|^2 = \\ &= \max_{W\in\widehat{\mathcal{W}}^k} \min_{y\in\mathbb{R}^m} \langle C, W\rangle + \langle b - \mathcal{A}W, y\rangle + \frac{u}{2}\|y - \hat{y}^k\|^2. \end{aligned} \tag{27}$$

The inner minimization over y is unconstrained and can be solved explicitly,

$$y^k_{\min}(W) = \hat{y}^k + \frac{\mathcal{A}W - b}{u}.$$

By substituting $y^k_{\min}(W)$ into the dual, the dual reduces to a quadratic semidefinite programming problem in small dimension with variables $V \in S_{r_k}$ and $\alpha \in \mathbb{R}_+$,

$$\text{(QSP)} \qquad \begin{array}{ll} \min & \frac{1}{2u}\|b - \mathcal{A}W\|^2 - \left\langle W, C - \mathcal{A}^T\hat{y}^k\right\rangle - b^T\hat{y}^k \\ \text{s.t.} & W = P_k V P_k^T + \alpha\overline{W}_k \\ & \operatorname{tr} V + \alpha = 1 \\ & V \succeq 0, \alpha \geq 0. \end{array}$$

The dimension of the original problem (the size of n and m) has no influence on subproblem (QSP) except in the computation of the cost coefficients. An optimal (not necessarily unique) solution W^{k+1} to (QSP) can be computed by interior point methods and gives rise to the candidate $y^{k+1} = y^k_{\min}(W^{k+1})$.

Algorithm 7.8

Input: *$y^0 \in \mathbb{R}^m$, $\varepsilon \geq 0$, $\kappa \in (0,1)$, a weight $u > 0$.*

1. *Set $k = 0$, $\hat{y}^0 = y^0$, compute $f(y^0)$ and $\widehat{\mathcal{W}}^0$.*
2. *(Trial point finding). Compute W^{k+1} and $y^{k+1} = y^k_{\min}(W^{k+1})$.*
3. *(Stopping criterion). If $f(\hat{y}^k) - f_{W^{k+1}}(y^{k+1}) \leq \varepsilon(|f(\hat{y}^k)| + 1)$ then **stop**.*
4. *(Evaluation). Find $W_S^{k+1} \in \operatorname{Argmax}_{W\in\mathcal{W}} \left\langle C - \mathcal{A}^T y^{k+1}, \cdot\right\rangle$ and determine $f(y^{k+1})$.*
5. *(Descent test). If $f(\hat{y}^k) - f(y^{k+1}) \geq \kappa[f(\hat{y}^k) - f_{W^{k+1}}(y^{k+1})]$ then set $\hat{y}^{k+1} = y^{k+1}$ (descent step); otherwise set $\hat{y}^{k+1} = \hat{y}^k$ (null step).*
6. *(Model updating). Choose a $\widehat{\mathcal{W}}^{k+1} \supset \left\{W^{k+1}, W_S^{k+1}\right\}$ of the form (25).*
7. *Increase k by one and **goto** 2.*

The proof of convergence of the algorithm relies heavily on $W^{k+1}, W_S^{k+1} \in \widehat{\mathcal{W}}^{k+1}$ (see step 6). In the case of a null step, the presence of W^{k+1} in $\widehat{\mathcal{W}}^{k+1}$ ensures that the value of the augmented model has to increase for $y \neq y^{k+1}$ by (27), whereas W_S^{k+1} guarantees that the value increases in y^{k+1} itself. Since the minimal value of the augmented model is at most $f(\hat{y}^k)$, one can work out that a descent step has to occur after finitely many null steps if $\hat{y}^k$ is not optimal. This is one of the main ingredients to the proof of the following theorem.

Theorem 7.9 *([24]) Let $\varepsilon = 0$. Either $\hat{y}^k \to \bar{y} \in \operatorname{Argmin}_{y\in Y} f(y)$, or $\operatorname{Argmin}_{y\in Y} f(y) = \emptyset$ and $\|\hat{y}^k\| \to \infty$. In both cases $f(\hat{y}^k) \downarrow \inf_{y\in Y} f$.*

The minimal choice of the form (25) for $\widehat{\mathcal{W}}^{k+1}$ in step 6 is to set $\overline{W}_{k+1} = W^{k+1}$ and $P_{k+1} = v$, where v is a normalized eigenvector to $\lambda_{\max}(C - \mathcal{A}^T y^{k+1})$. Hence, $r_{k+1} = 1$ already ensures convergence. A more realistic update strategy is given in [18]. It is possible to incorporate bounds on y in the spectral bundle method without significant loss in efficiency [17].

Another subgradient approach to problems of the form (E) is investigated in [33]. The local second order method of Overton [34] is combined with a global ε-descent method. The descent method determines a step direction on basis of $f_{\widehat{\mathcal{W}}_\delta}(y)$, where $\widehat{\mathcal{W}}_\delta$ is the face spanned by the eigenvectors to eigenvalues within some $\delta > 0$ of the maximal eigenvalue (cf. [8, 36]). This choice guarantees a certain minimal descent in the subsequent line search. It is not yet clear whether this approach will be efficient in practice, because each iteration requires the computation of a large fraction of the spectrum of $C - \mathcal{A}^T y$ and the structure of the system for local quadratic convergence resembles that of (20).

8 Conclusion and Outlook

Semidefinite programming is rich in theory and in applications. Research is still very active in the development and study of algorithms but several implementable options are already available. Demand for efficient solvers is currently highest within control and signal processing, but interest is also increasing in other areas, such as truss topology and material design, statistics, and combinatorial optimization. Available primal-dual interior point codes solve problems with matrix variables of order $n \le 200$ and $m \le 3000$ constraints within reasonable time. For large scale problems, subgradient methods like the spectral bundle method are, so far, the only choice; unfortunately, a general purpose code is still missing. For pointers to recent papers and software on the world wide web we refer to

`http://www.zib.de/helmberg/semidef.html`

Acknowlegdements. We thank Andreas Eisenblätter and Sven Krumke for thoughtful comments that helped to improve the presentation of the paper.

References

[1] F. Alizadeh, J.-P. A. Haeberly, and M. L. Overton. Primal-dual interior-point methods for semidefinite programming: Convergence rates, stability and numerical results. *SIAM J. Optim.*, 8(3):746–768, Aug. 1998.

[2] G. P. Barker and D. Carlson. Cones of diagonally dominant matrices. *Pacific Journal of Mathematics*, 57(1):15–32, 1975.

[3] A. Ben-Tal and A. Nemirovski. Robust truss topology design via semidefinite programming. *SIAM J. Optim.*, 7(4):991–1016, Nov. 1997.

[4] A. Ben-Tal and A. Nemirovski. Robust convex optimization. *Math. Oper. Res.*, 23(4):769–805, Nov. 1998.

[5] S. Benson, Y. Ye, and X. Zhang. Solving large-scale sparse semidefinite programs for combinatorial optimization. Working paper, Department of Management Science, University of Iowa, IA, 52242, USA, Sept. 1997. To appear in SIAM J. Opt.

[6] F. Bonnans, R. Cominetti, and A. Shapiro. Sensitivity analysis of optimization problems under second order regular constraints. *Math. Oper. Res.*, 23(4):806–831, Nov. 1998.

[7] S. Boyd, L. El Ghaoui, E. Feron, and V. Balakrishnan. *Linear Matrix Inequalities in System and Control Theory*, volume 15 of *SIAM Studies in Applied Mathematics*. SIAM, Philadelphia, 1994.

[8] J. Cullum, W. E. Donath, and P. Wolfe. The minimization of certain nondifferentiable sums of eigenvalues of symmetric matrices. *Math. Programming Study*, 3:35–55, 1975.

[9] E. de Klerk, C. Roos, and T. Terlaky. Initialization in semidefinite programming via a self-dual skew-symmetric embedding. *Operations Research Letters*, 20:213–221, 1997.

[10] C. Delorme and S. Poljak. Combinatorial properties and the complexity of a max-cut approximation. Technical Report 91687-OR, LRI University Paris-Sud, F-91405 ORSAY Cedex, France, 1991.

[11] A. Frieze and M. Jerrum. Improved approximation algorithms for MAX k-CUT and MAX BISECTION. In E. Balas and J. Clausen, editors, *Integer Programming and Combinatorial Optimization*, volume 920 of *Lecture Notes in Computer Science*, pages 1–13. Springer, May 1995.

[12] M. X. Goemans. Semidefinite programming in combinatorial optimization. *Math. Programming*, 79:143–161, Oct. 1997.

[13] M. X. Goemans and D. P. Williamson. Improved approximation algorithms for maximum cut and satisfiability problems using semidefinite programming. *J. ACM*, 42:1115–1145, 1995.

[14] G. H. Golub and C. F. van Loan. *Matrix Computations.* The Johns Hopkins University Press, 2^{nd} edition, 1989.

[15] M. Grötschel, L. Lovász, and A. Schrijver. *Geometric Algorithms and Combinatorial Optimization*, volume 2 of *Algorithms and Combinatorics.* Springer, 2^{nd} edition, 1988.

[16] C. Helmberg. Fixing variables in semidefinite relaxations. In R. Burkard and G. Woeginger, editors, *Algorithms – ESA'97*, volume 1284 of *Lecture Notes in Computer Science*, pages 259–270. Springer, Sept. 1997. Full version to appear in SIAM J. Matrix Anal. Appl.

[17] C. Helmberg, K. C. Kiwiel, and F. Rendl. Incorporating inequality constraints in the spectral bundle method. In R. E. Bixby, E. A. Boyd, and R. Z. Ríos-Mercado, editors, *Integer Programming and Combinatorial Optimization*, volume 1412 of *Lecture Notes in Computer Science*, pages 423–435. Springer, June 1998.

[18] C. Helmberg and F. Rendl. A spectral bundle method for semidefinite programming. ZIB Preprint SC-97-37, Konrad-Zuse-Zentrum für Informationstechnik Berlin, Aug. 1997. Revised October 1997. To appear in SIAM J. Optim.

[19] C. Helmberg, F. Rendl, R. J. Vanderbei, and H. Wolkowicz. An interior–point method for semidefinite programming. *SIAM J. Optim.*, 6(2):342–361, May 1996.

[20] R. A. Horn and C. R. Johnson. *Matrix Analysis.* Cambridge University Press, 1985.

[21] R. A. Horn and C. R. Johnson. *Topics in Matrix Analysis.* Cambridge University Press, 1991.

[22] D. Karger, R. Motwani, and M. Sudan. Approximate graph coloring by semidefinite programming. In *FOCS 94*, pages 2–13, 1994.

[23] L. Khachiyan and L. Porkolab. Computing integral points in convex semi-algebraic sets. In *FOCS 97*, pages 162–171, 1997.

[24] K. C. Kiwiel. Proximity control in bundle methods for convex nondifferentiable minimization. *Math. Programming*, 46:105–122, 1990.

[25] M. Kojima, M. Shida, and S. Shindoh. Reduction of monotone linear complementarity problems over cones to linear programs over cones. *Acta Mathematica Vietnamica*, 22:147–157, 1997.

[26] M. Kojima, S. Shindoh, and S. Hara. Interior–point methods for the monotone linear complementarity problem in symmetric matrices. *SIAM J. Optim.*, 7(1):86–125, Feb. 1997.

[27] A. S. Lewis and M. L. Overton. Eigenvalue optimization. *Acta Numerica*, pages 149–190, 1996.

[28] L. Lovász and A. Schrijver. Cones of matrices and set-functions and 0-1 optimization. *SIAM J. Optim.*, 1(2):166–190, May 1991.

[29] R. D. C. Monteiro. Polynomial convergence of primal-dual algorithms for semidefinite programming based on Monteiro and Zhang family of directions. *SIAM J. Optim.*, 8(3):797–812, Aug. 1998.

[30] Y. Nesterov and A. Nemirovskii. *Interior–Point Polynomial Algorithms in Convex Programming.* SIAM Studies in Applied Mathematics, Philadelphia, 1994.

[31] Y. Nesterov and M. J. Todd. Primal-dual interior-point methods for self-scaled cones. Technical Report TR 1125, School of Operations Research and Industrial Engineering, Cornell University, Ithaca, New York 14853, Aug. 1995. To appear in SIAM J. Optim.

[32] Y. Nesterov and M. J. Todd. Self-scaled barriers and interior-point methods for convex programming. *Math. Oper. Res.*, 22:1–42, 1997.

[33] F. Oustry. A second-order bundle method to minimize the maximum eigenvalue function. Technical Report RR-3738, INRIA Rhone-Alpes, ZIRST - 655 avenue de l'Europe, F-38330 Montbonnot Saint-Martin, July 1999. Submitted to Math. Programming.

[34] M. L. Overton. Large–scale optimization of eigenvalues. *SIAM J. Optim.*, 2(1):88–120, Feb. 1992.

[35] G. Pataki. On the rank of extreme matrices in semidefinite programming and the multiplicity of optimal eigenvalues. *Math. Oper. Res.*, 23(2):339–358, May 1998.

[36] E. Polak and Y. Wardi. Nondifferentiable optimization algorithm for designing control systems having singular value inequalities. *Automatica*, 18:267–283, 1982.

[37] L. Porkolab and L. Khachiyan. On the complexity of semidefinite programs. *Journal of Global Optimization*, 10(4):351–365, June 1997.

[38] M. Ramana. An exact duality theory for semidefinite programming and its complexity implications. *Math. Programming*, 77(2):129–162, 1997.

[39] R. T. Rockafellar. *Convex Analysis.* Princeton University Press, Princeton, New Jersey, 1970.

[40] R. Saigal, L. Vandenberghe, and H. Wolkowicz, editors. *Handbook of Semidefinite Programming.* Kluwer Academic Publishers, Boston/Dordrecht/London, 1999. To appear.

[41] N. Z. Shor. Quadratic optimization problems. *Soviet Journal of Computer and Systems Sciences*, 25:1–11, 1987. Originally published in Tekhnicheskaya Kibernetika, No. 1, 1987, pp. 128-139.

[42] M. Todd. On search directions in interior-point methods for semidefinite programming. Technical Report TR1205, School of Operations Research and Industrial Engineering, Cornell University, Ithaca, NY 14853-3801, Oct. 1997.

[43] L. Vandenberghe and S. Boyd. Semidefinite programming. *SIAM Review*, 38(1):49–95, Mar. 1996.

[44] H. Wolkowicz. Some applications of optimization in matrix theory. *Linear Algebra and its Applications*, 40:101–118, 1981.

[45] Y. Zhang. On extending some primal–dual interior-point algorithms from linear programming to semidefinite programming. *SIAM J. Optim.*, 8(2):365–386, 1998.