Chapter 2

Semidefinite Optimization

Pablo A. Parrilo

In this chapter we introduce one of the core theoretical and computational techniques in convex algebraic geometry, namely, *semidefinite optimization*. We begin by reviewing linear programming and proceed to define and discuss semidefinite programs from the algebraic, geometric, and computational perspectives. We define *spectrahedra* as the feasible sets of semidefinite programs, study their properties, and discuss numerous examples. Despite the many parallels, the duality theory of semidefinite optimization is more complicated than in the case of linear programming, and we elaborate on the similarities and differences. We also showcase a number of applications of semidefinite optimization in several areas of applied mathematics and engineering and give a short discussion of algorithmic and software aspects. For the convenience of the reader, we present additional background material on convex geometry and optimization in Appendix A.

2.1 From Linear to Semidefinite Optimization

Semidefinite optimization is a branch of convex optimization that is of great theoretical and practical interest. Informally, the main idea is to generalize linear programming and the associated feasible sets (polyhedra) to the case where the decision variables are symmetric matrices, and the inequalities are to be understood as matrices being positive semidefinite. Formal definitions and examples will be presented shortly in Subsection 2.1.2, preceded by a review of the familiar case of linear programming. A few selected standard references for linear programming and their applications are the books [5, 12, 29, 42].

2.1.1 Linear Programming

Linear programming is the problem of minimizing a linear function subject to linear constraints. A linear programming problem (LP) in standard form is usually written as

$$\begin{array}{ll} \text{minimize} & c^T x \\ \text{subject to} & Ax = b, \\ & x \geq 0, \end{array} \tag{LP-P}$$

where $A \in \mathbb{R}^{m\times n}$, $b \in \mathbb{R}^m$, and we are minimizing over the decision variable $x \in \mathbb{R}^n$. The inequality $x \geq 0$ is interpreted componentwise, i.e., $x_i \geq 0$ for $i = 1, \ldots, n$.

Geometrically, an LP problem has a nice and natural interpretation. Its feasible set is the intersection of an affine subspace (defined by the equations $Ax = b$), and the nonnegative orthant. Since it is the intersection of two convex sets, the feasible set of (LP-P) is always convex. In general, a set defined by finitely many linear inequalities or equations is called a *polyhedron*, and it is always convex. Thus, linear programming corresponds exactly to the minimization of a linear function over a polyhedron. If a polyhedron is bounded, it is called a *polytope*.

Perhaps one of the most remarkable and useful features of linear programming is that to every LP problem we can associate a corresponding *dual* problem. This is another LP problem ("its dual LP"), which for the case of (LP-P) is

$$\begin{array}{ll} \text{maximize} & b^T y \\ \text{subject to} & A^T y \leq c. \end{array} \tag{LP-D}$$

Notice that here we are again optimizing a linear function over a polyhedron. As we will see, there are very natural and direct algebraic relationships between the primal problem (LP-P) and its dual problem (LP-D).

Remark 2.1. *In practice, LP problems may not naturally present themselves in the form* (LP-P)*, where all the decision variables are nonnegative and only equality constraints are present, or the form* (LP-D)*, where there are no sign restrictions on the variables and only inequalities appear. However, they can always be put in either form, by introducing additional slack variables and/or splitting variables if necessary. The details can be found in any textbook on linear programming.*

Example 2.2. Consider the following LP problem:

$$\text{minimize} \;\; x_1 - 8x_2 \qquad \text{subject to} \;\; \left\{ \begin{array}{rcl} -x_1 + 3x_2 + x_3 & = & 4, \\ 4x_1 - x_2 + x_4 & = & 6, \\ x_1, x_2, x_3, x_4 & \geq & 0. \end{array} \right. \tag{2.1}$$

The feasible region is a two-dimensional polyhedron. Its projection into the (x_1, x_2)-plane is drawn in Figure 2.1. Notice that the optimal solution is achieved at a vertex, namely, $x^\star = (2, 2, 0, 0)$, with optimal cost $p^\star = -14$.

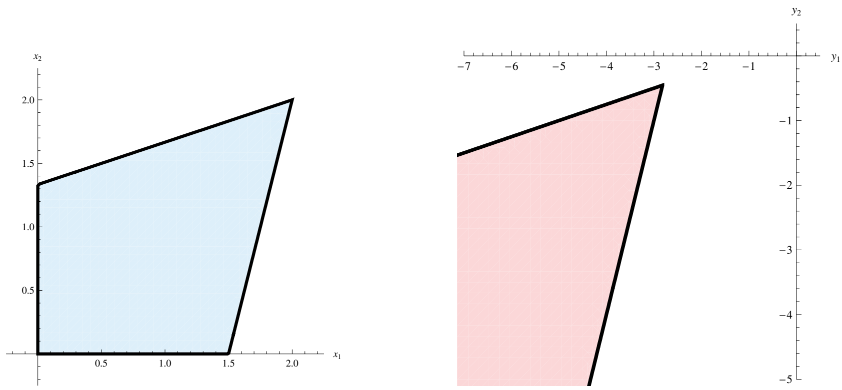

Figure 2.1. *Feasible sets of the primal and dual LP problems* (2.1) *and* (2.2).

The corresponding dual LP is

$$\text{maximize} \;\; 4y_1 + 6y_2 \qquad \text{subject to} \quad \begin{cases} -y_1 + 4y_2 & \leq \;\; 1, \\ 3y_1 - y_2 & \leq \;\; -8, \\ y_1 & \leq \;\; 0, \\ y_2 & \leq \;\; 0. \end{cases} \tag{2.2}$$

The dual feasible set (y_1, y_2) is presented in the same figure, with optimal solution $y^\star = (-\frac{31}{11}, -\frac{5}{11})$ and optimal cost $d^\star = -14$. For this example we have

$$p^\star = d^\star = -14,$$

and thus the optimal values of the primal and dual problems are the same. ∎

Even in this simple example, we can observe many of the important features of linear programming. The following facts are well known.

Geometry of the feasible set: The feasible sets of linear programs are *polyhedra*. The geometry of polyhedra is quite well understood. In particular, the Minkowski-Weyl theorem (e.g., Appendix A, [5], or [48, Section 1.1]) states that every polyhedron P is finitely generated, i.e., it can be written as

$$P = \operatorname{conv}(u_1, \dots, u_r) + \operatorname{cone}(v_1, \dots, v_s),$$

where u_i, v_i are the *vertices* and *extreme rays* of P, respectively, and the *convex hull* and *conical hull* are defined by

$$\operatorname{conv}(u_1, \dots, u_r) = \left\{ \sum_{i=1}^{r} \lambda_i u_i \;\middle|\; \sum_{i=1}^{r} \lambda_i = 1, \quad \lambda_i \geq 0, \quad i = 1, \dots, r \right\}$$

and

$$\operatorname{cone}(v_1, \dots, v_s) = \left\{ \sum_{i=1}^{s} \lambda_i v_i \;\middle|\; \lambda_i \geq 0, \quad i = 1, \dots, s \right\}.$$

Rational solutions: Unless the problem is unbounded, the optimal solution of a linear programming problem is always achieved at extreme points of the feasible set. Since these correspond to vertices of a polyhedron, the solution can be characterized in terms of a system of linear equations, corresponding to the equations and inequalities that are active at the optimal point. Thus, if the problem description (i.e., the matrices A, b, c) is given by rational numbers, there are always extreme points that are rational and achieve the optimal cost.

Weak duality: For *any* feasible solutions x, y of (LP-P) and (LP-D), respectively, it always holds that

$$c^T x - b^T y = x^T c - (Ax)^T y = x^T (c - A^T y) \geq 0, \tag{2.3}$$

where the last inequality follows from the feasibility conditions $x \geq 0$ and $A^T y \leq c$. Thus, from any feasible dual solution one can obtain a lower bound on the value of the primal. Conversely, primal feasible solutions give upper bounds on the value of the dual.

Strong duality: If both primal and dual problems are feasible, then they achieve exactly the same optimal value, and there exist optimal feasible solutions $x^\star, y^\star$ such that $c^T x^\star = b^T y^\star$. This is a consequence of the separation theorems for convex sets; see, e.g., Section A.3.3 in Appendix A.

Complementary slackness: Strong duality, combined with (2.3), implies that at optimality we must have

$$x_i^\star (c - A^T y^\star)_i = 0, \qquad i = 1, \dots, n.$$

In other words, there is a correspondence between primal variables and dual inequalities that says that whenever a primal variable is nonzero, the corresponding dual inequality must be tight.

In the linear programming case, these properties are well known and relatively easy to prove. Interestingly, as we will see in the next section, some of these properties will break down as soon as we leave linear programming and go to the more general case of semidefinite programming. These technical aspects will cause some minor difficulties, although with the right assumptions in place, the resulting theory will closely parallel the linear programming case.

Exercise 2.3. Consider a finite set of points $S = \{a_1, a_2, \dots, a_n\}$ in $\mathbb{R}^d$, where $n > d$. Prove using linear programming duality that exactly one of the following statements must hold:

- The origin is in the convex hull of S.
- There exists a hyperplane passing through the origin, such that all points a_i are strictly on one side of the hyperplane.

Exercise 2.4. Consider the set of $n \times n$ matrices with nonnegative entries that have all row and column sums equal to 1 (i.e., the *doubly stochastic* matrices).

1. Write explicitly the equations and inequalities describing this set for $n = 2, 3, 4$.

2. Compute (using `CDD`, `lrs`, or other software; see Section 2.3.2) all the extreme points of these polytopes.

3. How many extreme points did you find? What is the structure of the extreme points? Can you conjecture what happens for arbitrary values of n?

4. Google "Birkhoff–Von Neumann theorem," and check your guess.

2.1.2 Semidefinite Programming

Semidefinite programming is a broad generalization of linear programming, where the decision variables are symmetric matrices. A semidefinite programming problem (SDP) corresponds to the optimization of a linear function subject to *linear matrix inequality* (LMI) constraints. Semidefinite programs are convex optimization problems and have very appealing numerical properties (e.g., [7, 44, 45]).

Our notation is as follows: the set of real symmetric $n \times n$ matrices is denoted by $\mathcal{S}^n$. A matrix $A \in \mathcal{S}^n$ is *positive semidefinite* if $x^T Ax \geq 0$ for all $x \in \mathbb{R}^n$ and is *positive definite* if $x^T Ax > 0$ for all nonzero $x \in \mathbb{R}^n$. Equivalently, A is positive semidefinite if its eigenvalues $\lambda_i(A)$ satisfy $\lambda_i(A) \geq 0, i = 1, \ldots, n$, and is positive definite if $\lambda_i(A) > 0$, $i = 1, \ldots, n$. The set of $n \times n$ positive semidefinite matrices is denoted $\mathcal{S}^n_+$, and the set of positive definite matrices is denoted $\mathcal{S}^n_{++}$. As we will prove soon, $\mathcal{S}^n_+$ is a proper cone (i.e., closed, convex, pointed, and solid). We use the inequality signs "$\preceq$" and "$\succeq$" to denote the partial order induced by $\mathcal{S}^n_+$ (usually called the *Löwner* partial order); i.e., we write $A \succeq B$ if and only if $A - B$ is positive semidefinite. For a square matrix A, its *trace* is defined as $\mathrm{Tr}(A) = \sum_i A_{ii}$. See Section A.1 for further characterizations and general properties of positive semidefinite matrices.

Spectrahedra. Recall that a polyhedron is a set defined by finitely many linear inequalities and that feasible sets of LPs are polyhedra. Similarly, we define *spectrahedra* as sets defined by finitely many LMIs. These sets will correspond exactly to feasible sets of semidefinite programming problems.

Definition 2.5. *A* linear matrix inequality *(LMI) has the form*

$$A_0 + \sum_{i=1}^{m} A_i x_i \succeq 0,$$

where $A_i \in \mathcal{S}^n$ are given symmetric matrices.

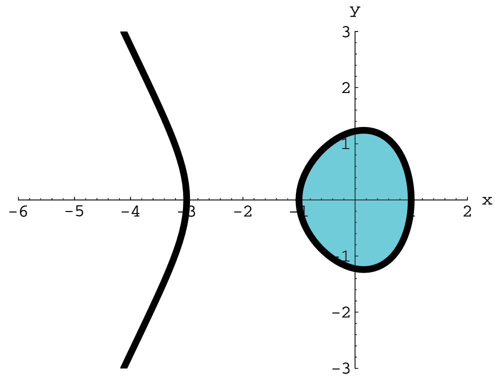

Figure 2.2. *The shaded set is a spectrahedron, with a semidefinite representation given by* (2.4).

Definition 2.6. *A set* $S \subset \mathbb{R}^m$ *is a* spectrahedron *if it has the form*

$$S = \left\{ (x_1, \dots, x_m) \in \mathbb{R}^m \; : \; A_0 + \sum_{i=1}^{m} A_i x_i \succeq 0 \right\},$$

for some given symmetric matrices $A_0, A_1, \dots, A_m \in \mathcal{S}^n$.

Geometrically, a spectrahedron is defined by intersecting the positive semidefinite cone and an affine subspace (the span of $A_1, \dots, A_m$, translated to A_0). Spectrahedra are closed convex sets, since a matrix inequality is equivalent to infinitely many scalar inequalities of the form $v^T(A_0 + \sum_{i=1}^m A_i x_i)v \geq 0$, one for each value of $v \in \mathbb{R}^n$. Since it is always possible to "bundle" several matrix inequalities into a single LMI (by choosing the matrices A_i to be block-diagonal), there is no loss of generality in defining spectrahedra in terms of a single matrix inequality. In particular, this shows that polyhedra are a particular case of spectrahedra, corresponding to all matrices A_i being diagonal.

Recall that the positive semidefiniteness of a matrix can be characterized in terms of scalar inequalities on the coefficients of its characteristic polynomial or its principal minors (see Proposition A.1). Thus, one can obtain an explicit description of a spectrahedron in terms of a finite collection of unquantified scalar polynomial inequalities in the variables x_i. In other words, spectrahedra are *basic semialgebraic sets*, that are convex.

Example 2.7 (elliptic curve). Consider the spectrahedron in $\mathbb{R}^2$ given by

$$\left\{ (x, y) \in \mathbb{R}^2 \; : \; A(x,y) := \begin{bmatrix} x+1 & 0 & y \\ 0 & 2 & -x-1 \\ y & -x-1 & 2 \end{bmatrix} \succeq 0 \right\}. \tag{2.4}$$

This set is shown in Figure 2.2. To obtain scalar inequalities defining the set, let $p_A(t) = \det(tI - A(x,y)) = t^3 + p_2 t^2 + p_1 t + p_0$ be the characteristic polynomial of

$A(x, y)$. Positive semidefiniteness of $A(x, y)$ is then equivalent to the conditions

$$
\begin{aligned}
-p_2 &= x + 5 \geq 0, \\
p_1 &= -x^2 + 2x - y^2 + 7 \geq 0, \\
-p_0 &= 3 + x - x^3 - 3x^2 - 2y^2 \geq 0.
\end{aligned}
$$

It can be seen that this spectrahedron corresponds to the "oval" of the elliptic curve $3 + x - x^3 - 3x^2 - 2y^2 = 0$. Notice that the boundary of the set is given by the determinant of the matrix inequality (why?), and the role of the other inequalities is to cut down and isolate the relevant component. ∎

As defined above, a spectrahedron S is a closed convex subset of the affine space $\mathbb{R}^m$. Following standard usage, we will also use "spectrahedron" to denote the set $\{A_0 + \sum_{i=1}^m A_i x_i \mid x \in \mathbb{R}^m\} \cap \mathcal{S}_+^n$. Notice that this is a convex set of *matrices* instead of a subset of $\mathbb{R}^m$, but if the matrices A_i are linearly independent, these two convex sets are affinely equivalent.

Projected spectrahedra. Also of interest are the linear projections of spectrahedra, which we will call *projected spectrahedra*:

Definition 2.8. *A set $S \subset \mathbb{R}^m$ is a* projected spectrahedron *if it has the form*

$$
S = \left\{ (x_1, \ldots, x_m) \in \mathbb{R}^m \,:\, \exists (y_1, \ldots, y_p) \in \mathbb{R}^p, \quad A_0 + \sum_{i=1}^m A_i x_i + \sum_{j=1}^p B_j y_j \succeq 0 \right\}, \tag{2.5}
$$

where $A_0, A_1, \ldots, A_m, B_1, \ldots, B_p$ are given symmetric matrices.

As the name indicates, geometrically this corresponds to a spectrahedron in $\mathbb{R}^{m+p}$ that is projected under the linear map $\pi : \mathbb{R}^{m+p} \to \mathbb{R}^m$, $(x, y) \mapsto x$. Since spectrahedra are semialgebraic sets, by the Tarski–Seidenberg theorem (Section A.4.4 in Appendix A) projected spectrahedra are also semialgebraic. Thus, they can be defined in terms of finite unions of sets defined by polynomial inequalities involving only the variables x_i, although in practice it is not always easy or convenient to do so.

Example 2.9. Consider the projected spectrahedron in $\mathbb{R}^2$ given by

$$
\left\{ (x, y) \in \mathbb{R}^2 \,:\, \exists z \in \mathbb{R}, \quad \begin{bmatrix} z + y & 2z - x \\ 2z - x & z - y \end{bmatrix} \succeq 0, \quad z \leq 1 \right\}. \tag{2.6}
$$

This set is shown in Figure 2.3. It corresponds to the projection on $\mathbb{R}^2$ of the spectrahedron in $\mathbb{R}^3$ defined by the intersection of a quadratic cone and a halfspace (see Figure 2.4).

For any fixed value of z, the set described by the 2×2 matrix inequality is a disk of radius z centered at $(2z, 0)$. Thus, this spectrahedron is the convex hull of the disk of unit radius centered at $(2, 0)$ and the origin. ∎

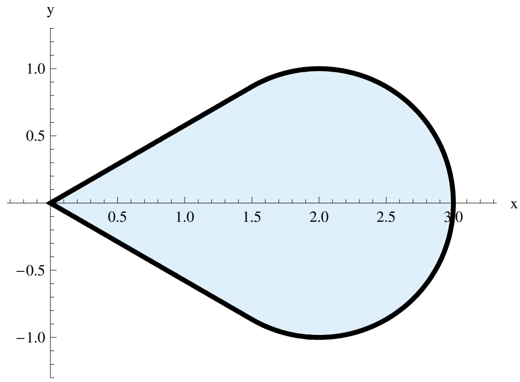

Figure 2.3. *A projected spectrahedron defined by* (2.6).

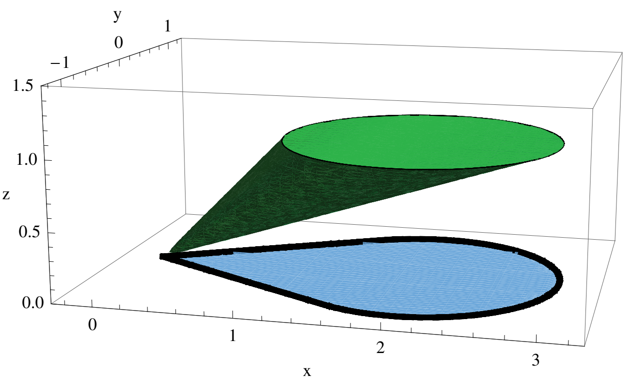

Figure 2.4. *A spectrahedron and its projection.*

As we will see later in much more detail in Chapters 3 and 6, there are simple examples of projected spectrahedra that are *not* spectrahedra (in fact, the set in Example 2.9 is one such case). This is in strong contrast with the case of polyhedra, for which we know (e.g., via Fourier–Motzkin elimination) that the linear projection of a polyhedron is always a polyhedron. Thus, this is a key distinguishing feature of semidefinite programming, since by adding additional *slack* or *lifting* variables, we can significantly expand the expressibility of our class of sets.

Projected spectrahedra are very important for optimization. Indeed, by including the additional "lifting" variables y_i, we will see that it is possible to reduce a linear optimization problem over a projected spectrahedron to the solution of a standard semidefinite program. Furthermore, projected spectrahedra have very high expressive power, in the sense that many convex sets of interest can be represented in this form. Although in general it may be hard to explicitly represent projected spectrahedra in terms of their defining inequalities in their ambient space (see Section 5.6 in Chapter 5), having a representation of the form (2.5) will often be enough for optimization purposes.

Exercise 2.10. Both spectrahedra and projected spectrahedra are convex sets. Show that spectrahedra are always closed sets. What about projected spectrahedra?

Primal SDP formulation. Semidefinite programs are linear optimization problems over spectrahedra. An SDP problem in standard primal form is written as

$$\begin{array}{lll} \text{minimize} & \langle C, X\rangle & \\ \text{subject to} & \langle A_i, X\rangle = b_i, & i = 1,\dots,m, \\ & X \succeq 0, & \end{array} \tag{SDP-P}$$

where $C, A_i \in \mathcal{S}^n$, and $\langle X, Y\rangle := \mathrm{Tr}(X^T Y) = \sum_{ij} X_{ij} Y_{ij}$. The matrix $X \in \mathcal{S}^n$ is the variable over which the minimization is performed. The inequality in the third line means that the matrix X must be positive semidefinite. Notice the strong formal similarities to the LP formulation (LP-P). As we will see in Section 2.1.4, this formal analogy can be pushed even further to *conic optimization problems*.

Let us make a few quick comments before presenting examples of semidefinite programs. The set of feasible solutions of (SDP-P), i.e., the set of matrices X that satisfy the constraints, is a spectrahedron, and thus it is always convex. This follows directly from the fact that the feasible set is the intersection of an affine subspace and the positive semidefinite cone $\mathcal{S}^n_+$, both of which are convex sets. However, unlike the linear programming case, in general the set of feasible solutions will not be polyhedral.

Example 2.11. Consider the semidefinite optimization problem

$$\begin{array}{ll} \text{minimize} & 2x_{11} + 2x_{12} \\ \text{subject to} & x_{11} + x_{22} = 1, \\ & \begin{bmatrix} x_{11} & x_{12} \\ x_{12} & x_{22} \end{bmatrix} \succeq 0. \end{array} \tag{2.7}$$

Clearly, this has the form (SDP-P), with $m = 1$ and

$$C = \begin{bmatrix} 2 & 1 \\ 1 & 0 \end{bmatrix}, \qquad A_1 = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \qquad b_1 = 1.$$

The constraints are satisfied if and only if $x_{11}(1 - x_{11}) \geq x_{12}^2$, and thus the feasible set is a closed disk, which is *not* polyhedral. Figure 2.5 shows the feasible set, parametrized by the variables (x_{11}, x_{12}). The optimal solution is equal to

$$X^\star = \begin{bmatrix} \frac{2-\sqrt{2}}{4} & -\frac{1}{2\sqrt{2}} \\ -\frac{1}{2\sqrt{2}} & \frac{2+\sqrt{2}}{4} \end{bmatrix},$$

with optimal cost $1 - \sqrt{2}$, which is clearly *not* rational. ∎

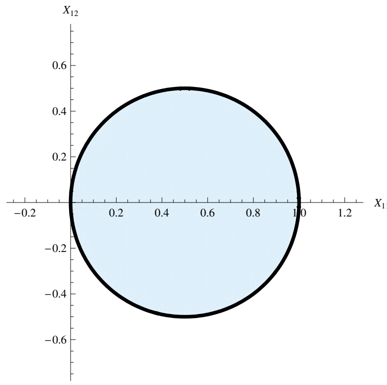

Figure 2.5. *Feasible set of the primal SDP problem* (2.7).

As we have seen from this simple example, SDP problems with rational data do not necessarily have rational optimal solutions. Since the solutions are nevertheless algebraic numbers, a natural question is to analyze their *algebraic degree*, i.e., the minimum degree of a polynomial with integer coefficients needed to specify the solution. The algebraic degree of semidefinite programming is studied in Chapter 5, Section 5.5.

In the particular case when $C = 0$ in (SDP-P), the problem reduces to whether or not the constraints can be satisfied for some matrix X. This is referred to as a *feasibility problem*. As described later, the algebraic nature and convexity of semidefinite programming has made it possible to develop sophisticated and reliable analytical and numerical methods to solve them.

Duality. A very important feature of semidefinite programming, from both the theoretical and applied viewpoints, is the associated *duality theory*. For every semidefinite program of the form (SDP-P) (usually called the *primal problem*), there is another associated SDP, called the *dual problem*, that can be stated as

$$\begin{aligned} \text{maximize} \quad & b^T y \\ \text{subject to} \quad & \sum_{i=1}^m A_i y_i \preceq C, \end{aligned} \tag{SDP-D}$$

where $b = (b_1, \ldots, b_m)$, and $y = (y_1, \ldots, y_m)$ are the dual decision variables.

As in the linear programming case, the key relationship between the primal and the dual problems is that feasible solutions of one problem can be used to bound the values of the other. Indeed, let X and y be any two feasible solutions of the primal and dual problems, respectively. We then have the following inequality:

$$\langle C, X\rangle - b^T y = \langle C, X\rangle - \sum_{i=1}^m y_i \langle A_i, X\rangle = \left\langle C - \sum_{i=1}^m A_i y_i, X \right\rangle \geq 0, \tag{2.8}$$

where the last inequality follows from the fact that the inner product of two positive semidefinite matrices is nonnegative. From (SDP-P) and (SDP-D) we can see that the left-hand side of (2.8) is the difference between the primal and dual objective functions. The inequality in (2.8) tells us that the value of the primal objective function evaluated at any feasible matrix X is always greater than or equal to the dual objective function at any dual feasible y. This is known as *weak duality*. Thus, we can use any X for which (SDP-P) is feasible to compute an upper bound for the value of $b^T y$ in (SDP-D), and we can also use any feasible y of (SDP-D) to compute a lower bound for the value of $\langle C, X\rangle$ in (SDP-P). Furthermore, in the case of feasibility problems (i.e., $C = 0$), the dual problem can be used to certify nonexistence of solutions to the primal problem. This property will be crucial in our later developments.

If X and Y are positive semidefinite matrices, then $\langle X, Y\rangle = 0$ if and only if $XY = YX = 0$ (e.g., Corollary A.24). Thus, the expression (2.8) allows us to give a simple sufficient characterization of optimality.

Lemma 2.12 (optimality conditions for SDP). *Assume* (X, y) *are primal and dual feasible solutions of* (SDP-P) *and* (SDP-D), *respectively, that satisfy the* complementary slackness *condition*

$$\left(C - \sum_{i=1}^{m} A_i y_i\right) X = 0 \tag{2.9}$$

(and thus achieve the same cost $\langle C, X\rangle = b^T y$*). Then,* (X, y) *are primal and dual* optimal *solutions of the SDP problem.*

In general, the converse statement may require some additional assumptions, to be discussed shortly.

Example 2.13. Here we continue Example 2.11. The SDP dual to (2.7) is

$$\begin{aligned} &\text{maximize } y \\ &\text{subject to } \begin{bmatrix} 2-y & 1 \\ 1 & -y \end{bmatrix} \succeq 0. \end{aligned}$$

The optimal solution is $y^\star = 1 - \sqrt{2}$, with optimal cost $1 - \sqrt{2}$. Notice that in this example, the optimal values of the primal and dual problems are equal. Furthermore, complementary slackness holds:

$$\left(C - \sum_{i=1}^{m} A_i y_i^\star\right) X^\star = \begin{bmatrix} 1+\sqrt{2} & 1 \\ 1 & \sqrt{2}-1 \end{bmatrix} \begin{bmatrix} \frac{2-\sqrt{2}}{4} & -\frac{1}{2\sqrt{2}} \\ -\frac{1}{2\sqrt{2}} & \frac{2+\sqrt{2}}{4} \end{bmatrix} = 0. \qquad \blacksquare$$

As opposed to the linear programming case, *strong* duality may fail in general semidefinite programming. We present below a simple example (from [36]), for which both the primal and dual problems are feasible, but their optimal values are different (i.e., there is a nonzero finite duality gap). Further examples and a detailed discussion will be presented in Section 2.1.5.

Example 2.14. Let $\alpha \geq 0$, and consider the primal-dual pair

$$\begin{array}{rl} \text{minimize} & \alpha X_{11} \\ \text{subject to} & X_{22} = 0, \\ & X_{11} + 2X_{23} = 1, \\ & X \succeq 0, \end{array} \qquad \begin{array}{rl} \text{maximize} & y_2 \\ \text{subject to} & \begin{bmatrix} y_2 & 0 & 0 \\ 0 & y_1 & y_2 \\ 0 & y_2 & 0 \end{bmatrix} \preceq \begin{bmatrix} \alpha & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}. \end{array}$$

For a primal feasible point, X being positive semidefinite and $X_{22} = 0$ imply $X_{23} = 0$, and thus $X_{11} = 1$. The primal optimal cost $p^\star$ is then equal to α (and is achieved). On the dual side, the vanishing of the $(3, 3)$ entry implies that y_2 must be zero, and thus $d^\star = 0$. The duality gap $p^\star - d^\star$ is then equal to α. ∎

The example above (and others like it), are somewhat "pathological." We will see in Section 2.1.5 that under relatively mild conditions, usually called *constraint qualifications*, strong duality will also hold in semidefinite programming. The simplest and most useful case corresponds to the so-called *Slater conditions*, where the primal and/or dual problems are required to be *strictly feasible*. On the primal side, this means that there exists $X \succ 0$ that satisfies the linear constraints, and on the dual side, there exists y such that $C - \sum_i A_i y_i \succ 0$ (notice that the inequalities are strict). In this case, the situation is as nice as in the linear programming case.

Theorem 2.15. *Assume that both the primal* (SDP-P) *and dual* (SDP-D) *semidefinite programs are* strictly *feasible. Then, both problems have optimal solutions, and the corresponding optimal costs are equal; i.e., there is no duality gap.*

This statement will reappear, in a more general setting, in Section 2.1.5. For many problems (for instance, the ones discussed in the next section), these assumptions hold and are relatively straightforward to verify. In full generality, however, they may be restrictive, and thus we investigate in Section 2.1.5 the geometric reasons why strong duality may fail in semidefinite optimization, as well as possible workarounds.

Exercise 2.16. Consider the following SDP problem:

$$\text{minimize } x \qquad \text{subject to} \quad \begin{bmatrix} x & 1 \\ 1 & y \end{bmatrix} \succeq 0.$$

1. Draw the feasible set. Is it convex?

2. Is the primal strictly feasible? Is the dual strictly feasible?

3. What can you say about strong duality? Are the results consistent with Theorem 2.15?

Exercise 2.17. Do the assumptions of Theorem 2.15 hold for Example 2.14?

2.1.3 Spectrahedra and Their Properties

Before proceeding further, we present several interesting examples of sets that are expressible in terms of semidefinite programming. We will revisit several of these throughout the different chapters in this book.

Spectraplex: The *spectraplex* or *free spectrahedron* $\mathcal{O}_n$ is the set of $n \times n$ positive semidefinite matrices of trace one, i.e.,

$$\mathcal{O}_n = \{X \in \mathcal{S}^n \mid \quad X \succeq 0, \quad \operatorname{Tr} X = 1\}.$$

The hyperplane $\operatorname{Tr} X{=}1$ intersects $\mathcal{S}^n_+$ on a compact set and thus defines a base for this cone. The extreme points of $\mathcal{O}_n$ are exactly the rank one matrices of the form $X = xx^T$, where $x \in \mathbb{R}^n$ and $\|x\| = 1$. The two-dimensional spectraplex $\mathcal{O}_2$ is affinely isomorphic to the unit disk in the plane and has already appeared in Example 2.11.

Elliptope and dual elliptope: Let $\bar{\mathcal{E}}_n$ be the set of positive semidefinite matrices with unit diagonal, i.e.,

$$\bar{\mathcal{E}}_n = \{X \in \mathcal{S}^n \mid \quad X \succeq 0, \quad X_{ii} = 1, \quad i = 1, \dots, n\}.$$

The convex set $\bar{\mathcal{E}}_n$ is contained in a subspace of $\mathcal{S}^n$ of codimension n, defined by the constraints $X_{ii} = 1$. It is often useful to consider it instead as a full-dimensional convex body in $\mathbb{R}^{\binom{n}{2}}$. For this, define an orthogonal projection $\pi : \mathcal{S}^n \to \mathbb{R}^{\binom{n}{2}}$ that projects a matrix X onto its off-diagonal entries X_{ij} for $i < j$.

The *elliptope* $\mathcal{E}_n$ is defined as $\mathcal{E}_n = \pi(\bar{\mathcal{E}}_n)$ and is a full-dimensional compact convex set in $\mathbb{R}^{\binom{n}{2}}$. As we will see in Section 2.2.2, this set is of great importance when studying semidefinite relaxations of combinatorial problems. Many geometric aspects of elliptopes have been extensively studied, e.g., in [26].

The elliptope $\mathcal{E}_n$ is a convex body containing the origin in its interior. Thus, we can define its polar dual set $\mathcal{E}_n^\circ = \{y \in \mathbb{R}^{\binom{n}{2}} : y^T x \le 1 \quad \forall x \in \mathcal{E}_n\}$, known as the *dual elliptope*. It follows from the expressions above that $\mathcal{E}_n^\circ$ is a (scaled) *projection* of the spectraplex onto the off-diagonal entries:

$$\mathcal{E}_n^\circ = -2\pi(\mathcal{O}_n). \tag{2.10}$$

For nice pictures of the 3×3 elliptope and its dual body, see Figure 5.8 in Chapter 5.

Operator and nuclear norms: Let $A \in \mathbb{R}^{n_1 \times n_2}$ be a matrix. The *spectral* or *operator* norm of A is given by its maximum norm gain, i.e.,

$$\|A\| = \max_{v \in \mathbb{R}^{n_2}, \|v\|=1} \|Av\| = \sigma_1(A),$$

where $\sigma_1(A)$ is the largest singular value of A.

The *nuclear norm* of a matrix is equal to the sum of its singular values, i.e.,

$$\|A\|_* := \sum_{i=1}^{r} \sigma_i(A), \tag{2.11}$$

where r is the rank of A. The nuclear norm is alternatively known by several other names including the Schatten 1-norm, the Ky Fan r-norm, and the trace class norm. As we will see in Section 2.2.6, the nuclear norm is particularly useful in optimization problems involving ranks of matrices.

The operator norm and the nuclear norm are *dual norms* in the sense that their unit balls are convex bodies that are polar duals, i.e.,

$$\{A \in \mathbb{R}^{n_1 \times n_2} \, : \, \|A\| \leq 1\}^\circ = \{B \in \mathbb{R}^{n_1 \times n_2} \, : \, \|B\|_* \leq 1\}.$$

Therefore, any two matrices A and B satisfy

$$\langle A, B \rangle \leq \|A\| \|B\|_*.$$

Furthermore, the following inequalities hold for any matrix A of rank at most r:

$$\|A\| \leq \|A\|_F \leq \|A\|_* \leq \sqrt{r}\|A\|_F \leq r\|A\|, \tag{2.12}$$

where $\|A\|_F$ is the Frobenius norm, defined as $\|A\|_F := (\mathrm{Tr} A^T A)^{\frac{1}{2}} = (\sum_{ij} a_{ij}^2)^{\frac{1}{2}}$.

Both the operator norm and the nuclear norm have nice characterizations in terms of semidefinite programming. In particular, the operator norm $\|A\|$ is the optimal solution of the primal-dual pair of semidefinite programs

$$\begin{array}{ll} \text{maximize} & \mathrm{Tr}\ 2A^T X_{12} \\ \text{subject to} & \mathrm{Tr} \begin{bmatrix} X_{11} & X_{12} \\ X_{12}^T & X_{22} \end{bmatrix} = 1, \\ & X \succeq 0, \end{array} \qquad \begin{array}{ll} \text{minimize} & t \\ \text{subject to} & \begin{bmatrix} tI_{n_1} & A \\ A^T & tI_{n_2} \end{bmatrix} \succeq 0. \end{array} \tag{2.13}$$

To see the exact correspondence between the standard form (SDP-P)-(SDP-D) and this formulation, notice that we can take $m = 1$, X is a block $(n_1 + n_2) \times (n_1 + n_2)$ matrix, A_1 is the $(n_1 + n_2) \times (n_1 + n_2)$ identity matrix, $b_1 = 1$, and the cost matrix C is the block matrix $\left(\begin{smallmatrix} 0 & -A \\ -A^T & 0 \end{smallmatrix} \right)$. Notice that we have the factor of 2 here because $\mathrm{Tr}\, CX = \mathrm{Tr}\, 2A^T X_{12}$, and we have "maximize" in (2.13) instead of "minimize" in (SDP-P) due to change of sign in the objective function.

Similarly (or "dually"), the nuclear norm $\|A\|_*$ corresponds to the optimal value of the primal-dual pair

$$\begin{array}{llcll} \text{maximize} & \operatorname{Tr} A^T Y & \qquad & \text{minimize} & \frac{1}{2}(\operatorname{Tr} W_1 + \operatorname{Tr} W_2) \\ \text{subject to} & \begin{bmatrix} I_{n_1} & Y \\ Y^T & I_{n_2} \end{bmatrix} \succeq 0, & & \text{subject to} & \begin{bmatrix} W_1 & A \\ A^T & W_2 \end{bmatrix} \succeq 0. \end{array} \tag{2.14}$$

Since the operator norm and the nuclear norm are dual norms, their unit balls are dual polar convex bodies. In Figure 2.6 we illustrate these convex sets for the case of a 2×2 symmetric matrix given by

$$A = \begin{bmatrix} x & y \\ y & z \end{bmatrix}. \tag{2.15}$$

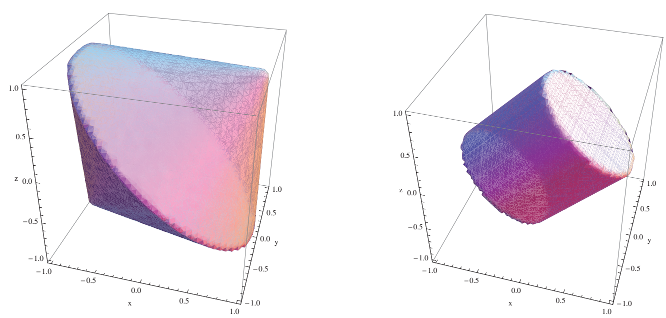

Figure 2.6. *Unit balls of the spectral norm and the nuclear norm, for the space of* 2×2 *symmetric matrices.*

***k*-ellipse:** We consider a class of planar convex sets defined by the algebraic curves known as *k-ellipses* [33]. Recall that the standard ellipse in $\mathbb{R}^2$ is defined as the locus of points with the sum of distances to two fixed points (the *foci*) a fixed constant. Extending this definition to k foci, one can define the k-ellipse as the algebraic curve in $\mathbb{R}^2$ consisting of all points whose sum of distances from k given points is a fixed number. More formally, fix a positive real number d, and fix k distinct points $(u_1, v_1), (u_2, v_2), \ldots, (u_k, v_k)$ in $\mathbb{R}^2$. The *k-ellipse* with *foci* (u_i, v_i) and *radius* d is the following curve in the plane:

$$\left\{ (x, y) \in \mathbb{R}^2 \;\middle|\; \sum_{i=1}^{k} \sqrt{(x - u_i)^2 + (y - v_i)^2} \; = \; d \right\}. \tag{2.16}$$

In Figure 2.7, we present a few k-ellipses with different numbers of foci. In contrast to the classical circle (corresponding to $k = 1$) and ellipse ($k = 2$), a k-ellipse does not necessarily contain all the foci in its interior. We define the closed convex set $\mathcal{C}_k$ to be the region whose boundary is the k-ellipse, and it is a sublevel set of the

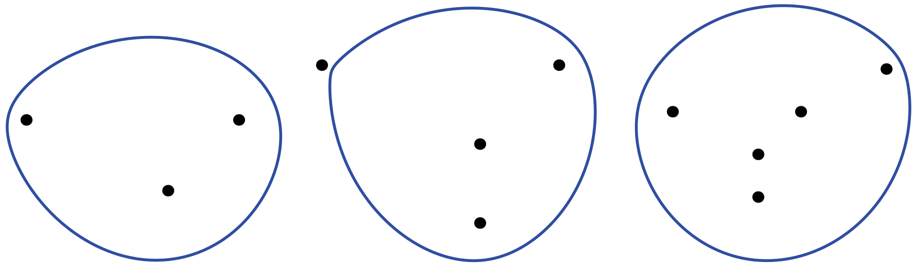

Figure 2.7. *A* 3*-ellipse, a* 4*-ellipse, and a* 5*-ellipse, each with its foci.*

convex function

$$(x, y) \quad \mapsto \quad \sum_{i=1}^{k} \sqrt{(x-u_i)^2 + (y-v_i)^2}. \tag{2.17}$$

In order for $\mathcal{C}_k$ to be nonempty, it is necessary and sufficient that the radius d be greater than or equal to the global minimum $d^\star$ of the convex function (2.17).

The set $\mathcal{C}_k$ is a projected spectrahedron, since it admits a semidefinite representation. This can be easily obtained by adding slack variables d_i and rewriting the function (2.17) in terms of 2×2 matrices. The region $\mathcal{C}_k$ is given by the points (x, y) for which there exist $(d_1, \dots, d_k)$ satisfying

$$\sum_{i=1}^{k} d_i \leq d, \qquad \begin{bmatrix} d_i + x - u_i & y - v_i \\ y - v_i & d_i - x + u_i \end{bmatrix} \succeq 0, \qquad i = 1, \dots, k.$$

To see this, notice that the 2×2 matrix above is positive semidefinite if and only if $(x-u_i)^2 + (y-v_i)^2 \leq d_i^2$ and $d_i \geq 0$.

In a less obvious fashion, the k-ellipse can also be represented without additional slack variables, so it is also a spectrahedron. However, in this case the size of the matrices is much bigger. Below we present a concrete statement; see [33] for a sharper result and an explicit construction of this representation.

Theorem 2.18. *The convex set $\mathcal{C}_k$ whose boundary is the k-ellipse of foci (u_i, v_i) and radius d is defined by the LMI*

$$x \cdot A_k + y \cdot B_k + C_k \succeq 0, \tag{2.18}$$

where A_k, B_k, C_k are symmetric $2^k \times 2^k$ matrices. The entries of A_k and B_k are integer numbers, and the entries of C_k are linear forms in the parameters $d, u_1, v_1, \dots, u_k, v_k$.

For illustration, we present the case $k = 3$ of the theorem. A spectrahedral representation of the 3-ellipse is obtained by requiring the following 8×8 matrix to be positive semidefinite:

$$
\begin{bmatrix}
d+3x-u_1-u_2-u_3 & y-v_1 & y-v_2 & 0 & y-v_3 & 0 & 0 & 0 \\
y-v_1 & d+x+u_1-u_2-u_3 & 0 & y-v_2 & 0 & y-v_3 & 0 & 0 \\
y-v_2 & 0 & d+x-u_1+u_2-u_3 & y-v_1 & 0 & 0 & y-v_3 & 0 \\
0 & y-v_2 & y-v_1 & d-x+u_1+u_2-u_3 & 0 & 0 & 0 & y-v_3 \\
y-v_3 & 0 & 0 & 0 & d+x-u_1-u_2+u_3 & y-v_1 & y-v_2 & 0 \\
0 & y-v_3 & 0 & 0 & y-v_1 & d-x+u_1-u_2+u_3 & 0 & y-v_2 \\
0 & 0 & y-v_3 & 0 & y-v_2 & 0 & d-x-u_1+u_2+u_3 & y-v_1 \\
0 & 0 & 0 & y-v_3 & 0 & y-v_2 & y-v_1 & d-3x+u_1+u_2+u_3
\end{bmatrix}.
$$

Exercise 2.19. Prove the relation (2.10) between the elliptope and the spectraplex.

Exercise 2.20. Show that the two semidefinite programs in (2.14) are indeed a primal-dual pair.

Exercise 2.21. Prove the correctness of the semidefinite characterizations of the operator and nuclear norms given in (2.13) and (2.14).

Exercise 2.22. Show that for the symmetric matrix in (2.15), the inequalities that define the boundary of the unit balls of the operator and spectral norms shown in Figure 2.6 are

$$y^2 + (x+z) - xz \le 1, \qquad y^2 - (x+z) - xz \le 1$$

and

$$(x-z)^2 + 4y^2 \le 1, \qquad x+z \le 1, \qquad -(x+z) \le 1,$$

respectively.

Exercise 2.23. Analyze the structure of the convex sets in Figure 2.6. What are the matrices associated with the flat facets (or the vertices)? How can you interpret the rotational symmetries of these convex bodies?

2.1.4 Conic Programming

The strong formal similarities between linear programming and semidefinite programming (equations (LP-P)-(LP-D) vs. (SDP-P)-(SDP-D)) suggest that a more

minimize $c^T x$ subject to $Ax = b$ $x \geq 0$	maximize $b^T y$ subject to $A^T y \leq c$	(LP)
minimize $\langle C, X \rangle$ subject to $\langle A_i, X \rangle = b_i$, $X \succeq 0$	maximize $b^T y$ subject to $\sum_i A_i y_i \preceq C$	(SDP)
minimize $\langle c, x \rangle_S$ subject to $\mathcal{A}x = b$, $x \in \mathcal{K}$	maximize $\langle y, b \rangle_T$ subject to $c - \mathcal{A}^* y \in \mathcal{K}^*$	(CP)

Table 2.1. *Primal-dual formulations of linear programming (LP), semidefinite programming (SDP), and general conic programming (CP).*

general formulation, encompassing both cases, may be possible. Indeed, a general class of optimization problems that unifies linear and semidefinite optimization (as well as a few other additional cases) is *conic programming.* We describe the conic framework next, explaining first the key idea, followed by the mathematical formulation.

The starting point is the geometric interpretation of linear and semidefinite programming. The feasible set of an LP problem in standard form (LP-P) is the intersection of an affine subspace (described by the equations $Ax = b$) and the nonnegative orthant $\mathbb{R}^n_+$. Similarly, the feasible set of a semidefinite program (SDP-P) is the intersection of an affine subspace (described by $\langle A_i, X \rangle = b_i$) with the set of positive semidefinite matrices $\mathcal{S}^n_+$. Since both $\mathbb{R}^n_+$ and $\mathcal{S}^n_+$ are closed convex cones (in fact, they are *proper* cones—see below), one can define a general class of optimization problems where the feasible set is the intersection of a proper cone and an affine subspace. This is exactly what conic optimization will do!

We present a formal description next. We will be a bit more careful than usual here in the definition of the respective spaces and mappings. It does not make much of a difference if we are working in $\mathbb{R}^n$ (since we can identify a space and its dual through the inner product), but it is "good hygiene" to keep these distinctions in mind and will prove useful when dealing with more complicated spaces. We consider two real vector spaces, S and T, and a linear mapping $\mathcal{A} : S \to T$. Recall that every real vector space has an associated dual space, which is the vector space of real-valued linear functionals. We denote these dual spaces by S^* and T^*, respectively, and the pairing between an element of a vector space and one of the dual as $\langle \cdot, \cdot \rangle$

(i.e., $f(x) = \langle f, x\rangle$). Recall that the *adjoint mapping* of $\mathcal{A}$ is the unique linear map $\mathcal{A}^* : T^* \to S^*$ defined by

$$\langle \mathcal{A}^* y, x\rangle_S = \langle y, \mathcal{A}x\rangle_T \qquad \forall x \in S,\, y \in T^*.$$

Notice here that the brackets on the left-hand side of the equation represent the pairing in S, and those on the right-hand side correspond to the pairing in T.

A cone $\mathcal{K} \subset S$ is *pointed* if $\mathcal{K} \cap (-\mathcal{K}) = \{0\}$ and is *solid* if it is full-dimensional (i.e., $\dim \mathcal{K} = \dim S$). A cone that is convex, closed, pointed, and solid is called a *proper* cone. Given a cone $\mathcal{K}$, its *dual cone* is $\mathcal{K}^* := \{z \in S^* : \langle z, x\rangle_S \geq 0 \quad \forall x \in \mathcal{K}\}$. The dual of a proper cone is also a proper cone; see Exercise 2.24. An element x is in the interior of the proper cone $\mathcal{K}$ if and only if $\langle x, z\rangle > 0\ \forall z \in \mathcal{K}^*, z \neq 0$.

Standard conic programs. Given a linear map $\mathcal{A} : S \to T$ and a proper cone $\mathcal{K} \subset S$, we define the primal-dual pair of (conic) optimization problems

$$\begin{array}{rlcrl} \text{minimize} & \langle c, x\rangle_S & \qquad & \text{maximize} & \langle y, b\rangle_T \\ \text{subject to} & \mathcal{A}x = b, & & \text{subject to} & c - \mathcal{A}^* y \in \mathcal{K}^*, \\ & x \in \mathcal{K}, & & & \end{array}$$

where $b \in T$, $c \in S^*$. Notice that exactly the same proof presented earlier works here to show weak duality:

$$\begin{aligned} \langle c, x\rangle_S - \langle y, b\rangle_T &= \langle c, x\rangle_S - \langle y, \mathcal{A}x\rangle_T \\ &= \langle c, x\rangle_S - \langle \mathcal{A}^* y, x\rangle_S \\ &= \langle c - \mathcal{A}^* y, x\rangle_S \\ &\geq 0. \end{aligned} \tag{2.19}$$

In the usual cases (e.g., LP and SDP), all vector spaces are finite-dimensional and thus isomorphic to their duals. The specific correspondence between these is given through whatever inner product we use.

Among the classes of problems that can be interpreted as particular cases of the general conic formulation we have linear programs, second-order cone programs (SOCP), and semidefinite programs, when we take the cone $\mathcal{K}$ to be the nonnegative orthant $\mathbb{R}^n_+$, the second-order cone $\mathcal{L}^n_+$ (Exercise 2.25), or the positive semidefinite cone $\mathcal{S}^n_+$, respectively. Two other important cases are when $\mathcal{K}$ is the *hyperbolicity cone* associated with a given hyperbolic polynomial [22, 40] and the cone $\Sigma_{n,2d}$ of multivariate polynomials that are *sums of squares*. We discuss this latter example in much more detail in Chapter 3.

Despite the formal similarities, there are a number of differences between linear programming and general conic programming. We have already seen in (2.19) that weak duality always holds for conic programming. However, recall from Example 2.14 that in semidefinite programming (and thus, in general conic programming) there may be a nonzero duality gap. In the next section, we explore the geometric reasons for the possible failure of strong duality in conic programming.

Exercise 2.24. Let $\mathcal{K} \subset S$ be a proper cone. Show that its dual cone $\mathcal{K}^* \subset S^*$ is also a proper cone, and $\mathcal{K}^{**} = \mathcal{K}$.

Exercise 2.25. The *second-order* (or Lorentz) cone is defined as

$$\mathcal{L}_+^n = \left\{ (x_0, x_1, \ldots, x_n) \in \mathbb{R}^{n+1} : \left(\sum_{i=1}^n x_i^2 \right)^{\frac{1}{2}} \le x_0 \right\}.$$

Show that $\mathcal{L}_+^n$ is a proper cone and is isomorphic to its dual cone.

Exercise 2.26. Classify the following statements as true or false. A proof or counterexample is required.

Let $\mathcal{A} : \mathbb{R}^n \to \mathbb{R}^m$ be a linear mapping and $K \subset \mathbb{R}^n$ a cone.

1. If K is convex, then $\mathcal{A}(K)$ is convex.
2. If K is solid, then $\mathcal{A}(K)$ is solid.
3. If K is pointed, then $\mathcal{A}(K)$ is pointed.
4. If K is closed, then $\mathcal{A}(K)$ is closed.

Do the answers change if $\mathcal{A}$ is injective and/or surjective? How?

2.1.5 Strong Duality

As we have indicated earlier, strong duality in semidefinite programming is a bit more delicate than in the linear programming case. Most of the time (and particularly, in applications) this will not be a source of too many difficulties. However, it is important to understand the geometry behind this, as well as what conditions we can impose to ensure that strong duality will hold.

As we showed in (2.19), weak duality always holds in conic programming (and thus, also for semidefinite programming (2.8)). However, it is possible to have finite duality gaps (as in Example 2.14), or other "anomalies," as the following simple example illustrates.

Example 2.27. Consider the primal-dual SDP pair

$$\begin{array}{ll} \text{minimize} & x_{11} \\ \text{subject to} & 2x_{12} = 1, \\ & \begin{bmatrix} x_{11} & x_{12} \\ x_{12} & x_{22} \end{bmatrix} \succeq 0, \end{array} \qquad \begin{array}{ll} \text{maximize} & y \\ \text{subject to} & \begin{bmatrix} 0 & y \\ y & 0 \end{bmatrix} \preceq \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix}. \end{array}$$

For the dual problem, $y = 0$ provides an optimal solution, with optimal value $d^\star = 0$. On the primal side, however, we cannot have $x_{11} = 0$, since this would violate the positive semidefiniteness constraint. However, by choosing $x_{11} = \epsilon$, $x_{22} = 1/\epsilon$, we obtain a cost p^* that is arbitrarily small but always strictly positive. ∎

The example above shows that, in contrast with the case of linear programming, in semidefinite or conic programming optimal solutions *may not be attained*, even if there is zero duality gap.

There are several geometric interpretations of what causes the failure of strong duality for general conic problems. Perhaps the most natural one is based on the fact that the image of a proper cone under a linear map may not be closed, and thus it is *not* necessarily a proper cone. This fact may seem a bit surprising (or perhaps wrong!) the first time one encounters it, but after a while it becomes quite reasonable. (If this is the first time you have heard about this, we strongly encourage you to stop reading and think of a counterexample right now! Or, see Exercise 2.30.)

Strong duality and infeasibility certificates. To better understand strong duality, we begin with a simple geometric interpretation in the conic setting, in terms of the *separating hyperplane theorem*. Recall that this theorem (see Section A.3.3 in Appendix A for several versions of this important result) establishes that if we have two disjoint convex sets, where one of them is closed and the other compact, there always exists a hyperplane that separates the two sets. For simplicity, we concentrate only on the case of conic feasibility, i.e., where we are interested in deciding the existence of a solution x to the equations

$$\mathcal{A}x = b, \qquad x \in \mathcal{K}, \tag{2.20}$$

where as before $\mathcal{K}$ is a proper cone in the vector space S. We want to understand when this problem is feasible and how to certify its infeasibility whenever there are no solutions.

To do this, consider the image $\mathcal{A}(\mathcal{K})$ of the cone under the linear mapping. Notice that feasibility of (2.20) is equivalent to the point b being contained in $\mathcal{A}(\mathcal{K})$. We have now two convex sets in T, namely, $\mathcal{A}(\mathcal{K})$ and the singleton $\{b\}$, and we want to know whether these sets intersect or not. *If* these sets satisfy certain properties (for instance, closedness and compactness), then we could go on to apply the (strict) separating hyperplane theorem and produce a linear functional y that will be positive on one set and negative on the other. In particular, nonnegativity of y on $\mathcal{A}(\mathcal{K})$ implies

$$\langle y, \mathcal{A}x\rangle \geq 0 \;\; \forall x \in \mathcal{K} \quad \Longleftrightarrow \quad \langle \mathcal{A}^* y, x\rangle \geq 0 \;\; \forall x \in \mathcal{K} \quad \Longleftrightarrow \quad \mathcal{A}^* y \in \mathcal{K}^*.$$

Thus, if (2.20) is infeasible, and provided the hypotheses of the separating hyperplane theorem apply, there exists a (suitably normalized) linear functional y which satisfies

$$\langle y, b\rangle = -1, \qquad \mathcal{A}^* y \in \mathcal{K}^*. \tag{2.21}$$

This yields a *certificate* of the infeasibility of the conic system (2.20).

When can we actually do this? The set $\{b\}$ is certainly compact, so a natural condition is that $\mathcal{A}(\mathcal{K})$ be a closed set. However, as we have mentioned, the image of a proper cone is not necessarily closed, so we cannot automatically conclude this. However, under certain conditions, we can ensure that this set will be closed. Well-known sufficient conditions for this are the following.

Theorem 2.28. *Let $\mathcal{K} \subset S$ be a proper cone and $\mathcal{A} : S \to T$ be a linear map. The following two conditions are equivalent:*

(i) $\mathcal{K} \cap \ker\mathcal{A} = \{0\}$.

(ii) *There exists $y \in T^*$ such that $\mathcal{A}^* y \in \operatorname{int}(\mathcal{K}^*)$.*

Furthermore, if these conditions hold, then $\mathcal{A}(\mathcal{K})$ is a closed cone.

The first condition, while intuitive, has the drawback that it is not directly verifiable. The second condition is often more convenient, since it can be certified by exhibiting such a y, and can be interpreted as the range of $\mathcal{A}^*$ properly intersecting $\mathcal{K}^*$.

Proof. The equivalence of (i) and (ii) follows from Exercise 2.32, taking $L = \ker \mathcal{A}$, and thus $L^\perp = \operatorname{range} \mathcal{A}^*$.

Assume now that (ii) holds, and define $C = \{x \in \mathcal{K} : \langle \mathcal{A}^* y, x\rangle = 1\}$. We claim that the set C is compact. Indeed, C is closed (being the intersection of two closed sets), and it is also bounded, since if there is a sequence $x_k \in C$ with $\|x_k\|$ going to infinity, then defining $z = \lim_{k\to\infty} x_k/\|x_k\|$ (passing to a subsequence if necessary) gives an element of $\mathcal{K}$ (by closedness of $\mathcal{K}$), for which $\langle \mathcal{A}^* y, z\rangle = \lim_{k\to\infty}\langle \mathcal{A}^* y, x_k\rangle/\|x_k\| = \lim_{k\to\infty} 1/\|x_k\| = 0$, contradicting $\mathcal{A}^* y \in \operatorname{int}(\mathcal{K}^*)$.

The set $\mathcal{A}(C)$ is also compact (being the linear image of a compact set) and does not include the origin, since for all $x \in C$ we have $\langle y, \mathcal{A}x\rangle = \langle \mathcal{A}^* y, x\rangle = 1$. Thus, since $\mathcal{A}(\mathcal{K}) = \operatorname{cone}(\mathcal{A}(C))$, it follows from Exercise 4.17 in Chapter 4 that $\mathcal{A}(\mathcal{K})$ is closed. $\square$

To recap, having strictly feasible solutions in $(\operatorname{range} \mathcal{A}^*) \cap \operatorname{int} \mathcal{K}^*$ is a natural condition for the existence of infeasibility certificates of the form (2.21).

For the case of a general conic optimization problem (not just feasibility), similar conditions can be used to ensure that there will be no duality gap between the primal and dual conic programs. The basic idea is to reduce the optimization problem to a pure feasibility question by adjoining a new inequality corresponding to the cost function. In this case, imposing a Slater-type condition will guarantee that optimal solutions for both problems are achieved, with no gap (compare with the semidefinite programming case, Theorem 2.15).

Theorem 2.29. *Consider a conic optimization problem* (CP)*, where both the primal and dual problems are* strictly *feasible. Then, both problems have nonempty, compact sets of optimal solutions, and there is no duality gap.*

Besides Theorem 2.28, many other conditions are known that ensure the closedness of $\mathcal{A}(\mathcal{K})$. In particular, when $\mathcal{K}$ is polyhedral this image is always closed, with no interior-point requirements needed. This corresponds to the case of linear programming and is the reason why strong duality always holds in the LP case.

In Section 3.4.2 of Chapter 3 we will explore in much more detail general infeasibility certificates for different kinds of systems of equations and inequalities.

Exercise 2.30. Consider the set $\mathcal{K} = \{(x, y, z) : y^2 \leq xz,\ z \geq 0\}$. Show that $\mathcal{K}$ is a proper cone. Show that its projection onto the (x, y) plane is not a proper cone.

Exercise 2.31. Let $\mathcal{K}_1$, $\mathcal{K}_2$ be closed convex cones. Show, via a counterexample, that the Minkowski sum $\mathcal{K}_1 + \mathcal{K}_2$ does not have to be closed.

Exercise 2.32. Let $L \subset S$ be a subspace, and $\mathcal{K} \subset S$ be a proper cone. Show that the following two propositions are equivalent:

(i) $L \cap \mathcal{K} = \{0\}$.

(ii) There exists $z \in L^{\perp} \cap \mathrm{int}(\mathcal{K}^*)$.

Hint: For the "difficult" direction (i) $\Rightarrow$ (ii), argue by contradiction, and use homogeneity and the separation theorem for convex sets.

Although as we have seen, "standard" duality may fail in semidefinite (or conic) programming, it is nevertheless possible to formulate a more complicated semidefinite dual program (called the "Extended Lagrange–Slater Dual" in [36]) for which strong duality always holds, regardless of interior-point assumptions. For details, as well as a comparison with the more general "minimal cone" approach, we refer the reader to [36, 37].

2.2 Applications of Semidefinite Optimization

There have been *many* applications of semidefinite optimization in a variety of areas of applied mathematics and engineering. We present here just a few, to give a flavor of what is possible; many others will follow in other chapters. The subsections corresponding to the different examples presented here can be read independently and are not essential for the remainder of the developments in the book.

2.2.1 Lyapunov Stability and Control of Dynamical Systems

One of the earliest and most important applications of semidefinite optimization is in the context of dynamical systems and control theory. The main reason is that it is possible to characterize dynamical properties (e.g., stability) in terms of algebraic statements such as the feasibility of specific systems of inequalities. We describe below a relatively simple example of these ideas that captures many of the features of more complicated problems.

Stability of linear systems. Consider a linear difference equation given by

$$x[k+1] = A\,x[k], \qquad x[0] = x_0. \tag{2.22}$$

This kind of linear recurrence equation is a simple example of a discrete-time dynamical system, where the *state* $x[k]$ evolves over time, starting from an initial condition x_0. The difference equation (2.22), or its continuous-time analogue (the linear differential equation $\frac{d}{dt}x(t) = Ax(t)$), is often used to model the time evolution of quantities such as temperature of objects, size of a population, voltage of electrical circuits, and concentration of chemical mixtures.

A natural and important question about (2.22) is the long-term behavior of the state. In particular, as $k \to \infty$, under what conditions can we guarantee that the state $x[k]$ remains bounded, or converges to zero? It is well known (and easy to prove; see Exercise 2.35) that $x[k]$ converges to zero for all initial conditions x_0 if and only if the spectral radius of the matrix A is smaller than one, i.e., all the eigenvalues λ_i satisfy $|\lambda_i(A)| < 1$ for $i = 1, \dots, n$. In this case we say that the system (2.22), or the matrix A, is *stable* (or *Schur stable*, if the discrete-time aspect is not clear from the context).

While this spectral characterization is very useful, an alternative viewpoint is sometimes even more convenient. The basic idea is to consider a generalization and abstraction of the notion of *energy*, usually known as a *Lyapunov function*. These are functions of the state $x[k]$, with the property that they decrease monotonically along trajectories of the system (2.22). It turns out that for linear systems there is a simple characterization of stability in terms of a *quadratic* Lyapunov function $V(x[k]) = x[k]^T P x[k]$. Notice first that the monotonicity condition $V(x[k+1]) \leq V(x[k])$ (for all states $x[k]$) can be equivalently expressed in terms of the matrix inequality $A^T PA - P \preceq 0$. We then have the following result.

Theorem 2.33. *Given a matrix $A \in \mathbb{R}^{n\times n}$, the following conditions are equivalent:*

1. *All eigenvalues of A are inside the unit circle; i.e., $|\lambda_i(A)| < 1$ for $i = 1, \dots, n$.*
2. *There exists a matrix $P \in \mathcal{S}^n$ such that*
$$P \succ 0, \qquad A^T PA - P \prec 0.$$

Proof. (2) $\Rightarrow$ (1): Let $Av = \lambda v$, where $v \neq 0$. Then
$$0 > v^*(A^T PA - P)v = (|\lambda|^2 - 1)\underbrace{v^* P v}_{>0},$$
and therefore $|\lambda| < 1$.

(1) $\Rightarrow$ (2): Let $P := \sum_{k=0}^{\infty} (A^k)^T A^k$. The sum converges by the eigenvalue assumption. Then
$$A^T PA - P = \sum_{k=1}^{\infty} (A^k)^T A^k - \sum_{k=0}^{\infty} (A^k)^T A^k = -I \prec 0. \qquad \square$$

Thus, the characterization given above enables the study of the stability properties of the linear difference equation (2.22) in terms of a semidefinite programming problem, whose feasible solutions correspond to Lyapunov functions. In Section 3.6.2 we will explore extensions of these ideas to more complicated dynamics, not necessarily linear.

Control design. Consider now the case of a linear system, where there is a *control input* $u[k]$:
$$x[k+1] = A\,x[k] + B\,u[k], \qquad x[0] = x_0, \tag{2.23}$$

where $B \in \mathbb{R}^{n\times m}$. The idea here is that by properly choosing the control input $u[k] \in \mathbb{R}^m$ at each time instant, we may be able (under certain conditions), to affect or steer the behavior of $x[k]$ toward some desired goal. We are interested in the case where the matrix A is not stable, but we can use linear state feedback to set $u[k] = Kx[k]$ for some fixed matrix K (to be chosen appropriately). It is easy to see that after this substitution, the system is described by (2.22), where the matrix A is replaced by $A(K) = A + BK$. Thus, our goal is "stabilization"; i.e., we want to find a matrix K such that $A + BK$ is stable (all eigenvalues have absolute value smaller than one).

Although this problem seems (and is!) fairly complicated due to the nonlinear dependence of the eigenvalues of $A + BK$ on the unknown matrix K, it turns out that it can be nicely solved using semidefinite optimization and the Lyapunov characterization given earlier. Indeed, we can use Schur complements (see Appendix A) to rewrite the condition

$$(A + BK)^T P(A + BK) - P \prec 0, \qquad P \succ 0,$$

as

$$\begin{bmatrix} P & (A + BK)^T P \\ P(A + BK) & P \end{bmatrix} \succ 0.$$

Although nicer, this condition is not quite an SDP yet, since it is bilinear in (P, K) (and, thus, not jointly convex). However, defining $Q := P^{-1}$, and left- and right-multiplying the equation above with the matrix $\mathrm{BlockDiag}(Q, Q)$, we obtain

$$\begin{bmatrix} Q & Q(A + BK)^T \\ (A + BK)Q & Q \end{bmatrix} \succ 0.$$

Notice that this expression contains both Q and KQ, but there is no single appearance of the variable K. Thus, we can define a new variable $Y := KQ$, to obtain

$$\begin{bmatrix} Q & QA^T + Y^T B^T \\ AQ + BY & Q \end{bmatrix} \succ 0. \tag{2.24}$$

This problem is now linear in the new variables (Q, Y). In fact, it is a semidefinite programming problem! After solving it, we can recover the controller K via $K = Q^{-1}Y$. We summarize our discussion in the following result.

Theorem 2.34. *Given two matrices A and B, there exists a matrix K such that $A + BK$ is stable if and only if the spectrahedron described by* (2.24) *is nonempty, i.e., there exist matrices (Q, Y) satisfying this (strict) linear matrix inequality.*

Hence our control design problem is equivalent to solving a semidefinite programming feasibility problem.

Semidefinite programming techniques have become quite central in the analysis and design of control systems. The example above describes only the tip of the iceberg in terms of the many design problems that can be attacked with these techniques; we refer the reader to the works [6, 47] and the references therein.

We remark that the formulas in this example (e.g., (2.24)) do not explicitly depend on the dimensions of the matrices A, B, K, Y, Q. Hence, these kinds of problems are sometimes called *dimension-free*. This dimension-free feature applies to many classical problems in linear systems and has strong implications. Linear control theory problems can often be reduced to polynomials in matrix variables where the feasible set is defined by these polynomials being positive semidefinite. Analyzing this situation requires a theory of inequalities for free noncommutative polynomials extending classical real geometry for commutative polynomials. The convexity aspects of this new area, noncommutative real algebraic geometry, is the subject of Chapter 8.

Exercise 2.35. Show that for the linear difference equation (2.22), the state $x[k]$ converges to zero for all initial conditions x_0 if and only if $|\lambda_i(A)| < 1$ for $i = 1, \ldots, n$. Hint: show that $x[k] = A^k x_0$, and consider first the case where the matrix A is diagonalizable.

Exercise 2.36. The system (2.23) has a *nonstabilizable mode* if the matrix A has a left eigenvector w such that $w^T A = \lambda w^T$, $w^T B = 0$, and $|\lambda| \geq 1$. Show that if this is the case, then the SDP (2.24) cannot be feasible. Interpret this statement in terms of the eigenvalues of $A + BK$. What does this say about the dual SDP?

2.2.2 Binary Quadratic Optimization

Binary (or Boolean) quadratic optimization is a classical combinatorial optimization problem. In the version we consider, we want to minimize a quadratic function, where the decision variables can take only the values ± 1. In other words, we are minimizing an (indefinite) quadratic form over the vertices of an n-dimensional hypercube. The problem is formally expressed as

$$\begin{array}{rl} \text{minimize} & x^T Q x \\ \text{subject to} & x_i \in \{-1, 1\}, \end{array} \tag{2.25}$$

where $Q \in \mathcal{S}^n$. There are many well-known problems that can be naturally written in the form above. Among these, we mention the maximum cut (MAXCUT) problem, 0-1 knapsack, etc.

Notice that the Boolean constraints can be modeled using quadratic equations, i.e.,

$$x_i \in \{-1, 1\} \quad \Leftrightarrow \quad x_i^2 = 1.$$

These n quadratic equations define a finite set, with an exponential number of elements, namely, all the n-tuples with entries in $\{-1, 1\}$. There are exactly 2^n points in this set, so a direct enumeration approach to (2.25) is computationally prohibitive when n is large (already for $n = 30$ we have $2^n \approx 10^9$).

We write the equivalent polynomial formulation

$$\begin{aligned} \text{minimize} \quad & x^T Q x \\ \text{subject to} \quad & x_i^2 = 1, \end{aligned} \tag{2.26}$$

and we denote the optimal value and optimal solution of this problem as $f_\star$ and $x_\star$, respectively. It is well known that the decision version of this problem is *NP-complete* (e.g., [18]). Notice that this is true even if the objective function is convex (i.e., the matrix Q is positive definite), since we can always assume $Q \succeq 0$ by adding to it a large constant multiple of the identity (this only shifts the objective by a constant).

Computing "good" solutions to the binary optimization problem (2.26) is a quite difficult task, so it is of interest to produce accurate bounds on its optimal value. As in all minimization problems, *upper bounds* can be directly obtained from feasible points. In other words, if $x_0 \in \mathbb{R}^n$ has entries equal to ± 1, it always holds that $f_\star \leq x_0^T Q x_0$ (of course, for a poorly chosen x_0, this upper bound may be very loose).

To prove *lower bounds*, we need a different technique. There are several approaches to doing this, but many of them will turn out to be exactly equivalent in the end. In particular, we can provide a lower bound in terms of the following primal-dual pair of semidefinite programming problems:

$$\boxed{\begin{array}{llll} \text{minimize} & \operatorname{Tr} QX & \text{maximize} & \operatorname{Tr} \Lambda \\ \text{subject to} & X_{ii} = 1, & \text{subject to} & Q \succeq \Lambda, \\ & X \succeq 0, & & \Lambda \ \text{diagonal}. \end{array}} \tag{2.27}$$

These semidefinite programs can be interpreted in a number of ways. For instance, it is clear that the optimal solution $X^\star$ of the primal formulation in (2.27) yields a lower bound, since for every x in (2.26), the matrix $X = xx^T$ gives a feasible solution of (2.27) with the same cost: $\operatorname{Tr} QX = \operatorname{Tr} Qxx^T = x^T Q x$. Similarly, for every feasible solution $\Lambda = \operatorname{Diag}(\lambda_1, \ldots, \lambda_n)$ of the dual SDP, we have

$$x^T Q x \geq x^T \Lambda x = \sum_{i=1}^{n} \lambda_i x_i^2 = \operatorname{Tr}\Lambda,$$

thus yielding a lower bound on (2.26).

In certain cases, these SDP-based bounds are *provably* good. Well-known cases are when $(-Q)$ is diagonally dominant or positive semidefinite or has a bipartite structure, in which case results due to Goemans–Williamson [20], Nesterov [31], or Grothendieck/Krivine [30, 2, 25], respectively, have shown that there is at most a small constant factor between the "true" solutions and the SDP relaxations. We discuss these bounds next.

Rounding. As described, the optimal value of the SDP relaxation (2.27) provides a lower bound on the optimal value of the binary minimization problem (2.26). Two natural questions arise:

1. Feasible solutions: can we use the SDP relaxations to provide feasible points that yield good (or optimal) values of the objective?

2. Approximation guarantees: is it possible to quantify the quality of the bounds obtained by SDP?

By suitably "rounding" in an appropriate manner the optimal solution of the SDP relaxation, both questions can be answered in the affirmative. The basic idea is to produce a binary vector x from the SDP solution matrix X, using the following "hyperplane rounding" method [20]:

- Factorize the SDP solution X as $X = V^T V$, where $V = [v_1 \dots v_n] \in \mathbb{R}^{r \times n}$ and r is the rank of X.

- Since $X_{ij} = v_i^T v_j$ and $X_{ii} = 1$, this factorization gives n vectors v_i on the unit sphere in $\mathbb{R}^r$. Thus, instead of assigning either 1 or -1 to each variable, so far we have assigned to each x_i a point on the unit sphere in $\mathbb{R}^r$.

- Now, choose a uniformly distributed random hyperplane in $\mathbb{R}^r$ (passing through the origin), and assign to each variable x_i either a $+1$ or a -1, depending on which side of the hyperplane the point v_i lies.

Since the last step involves a random choice, this is a *randomized rounding method.* By a simple geometric argument, it is possible to quantify the expected value of the objective function.

Lemma 2.37. *Let $x = \operatorname{sign}(V^T r)$, where $X = V^T V$ and r is a standard random Gaussian vector. Then, $\mathbf{E}[x_i x_j] = \frac{2}{\pi} \arcsin X_{ij}$.*

By linearity of expectations, we have the following relationship between the lower bound given by the optimal value of the SDP, the "true" optimal value $f_\star$, and the expected value of the rounded solution x:

$$\operatorname{Tr} QX \leq f_\star \leq \mathbf{E}[x^T Q x] = \frac{2}{\pi} \operatorname{Tr} Q \arcsin[X]. \tag{2.28}$$

The notation $\arcsin[\cdot]$ indicates that the arcsine function is applied componentwise, i.e., $(\arcsin[X])_{ij} = \arcsin X_{ij}$.

Exercise 2.38. Prove Lemma 2.37, and verify that it implements the hyperplane rounding scheme.

Approximation ratios. In many problems, we want to understand how far these upper and lower bounds are from each other. Depending on the specific assumptions on the cost function, the hyperplane rounding method (or slight variations) will give solutions with different guaranteed approximation ratios. Since the approximation algorithms literature often considers *maximization* problems (instead of the minimization version (2.26)), in this section we use

$$\begin{array}{ll} \text{maximize} & x^T A x \\ \text{subject to} & x_i^2 = 1 \end{array} \tag{2.29}$$

and state below our assumptions in terms of the matrix A (or, equivalently, the matrix $-Q$ in the minimization formulation (2.25)).

We describe next three well-known cases where *constant* approximation ratios can be obtained.

Diagonally dominant: A symmetric matrix A is *diagonally dominant* if $a_{ii} \geq \sum_{j \neq i} |a_{ij}|$ for all i. This is an important case that corresponds, for instance, to the MAXCUT problem, where the cost function to be maximized is the Laplacian of a graph (V, E), given by $\frac{1}{4} \sum_{(i,j) \in E} (x_i - x_j)^2$. Every diagonally dominant quadratic form can be written as a nonnegative linear combination of terms of the form x_i^2 and $(x_i \pm x_j)^2$ [4]. Thus, to analyze the performance of hyperplane rounding when A is diagonally dominant, it is enough to consider the inequality

$$\mathbf{E}[(x_i \pm x_j)^2/2] = \mathbf{E}[1 \pm x_i x_j] = 1 \pm \frac{2}{\pi} \arcsin X_{ij} \geq \alpha_{GW} \cdot (1 \pm X_{ij}),$$

where $\alpha_{GW} = \min_{t \in [-1,1]} (1 - \frac{2}{\pi} \arcsin t)/(1 - t) \approx 0.878$. Combining this with (2.28), and taking into account the change of signs (since $A = -Q$), it follows that

$$\alpha_{GW} \cdot \operatorname{Tr} AX \leq \mathbf{E}[x^T A x] \leq f_\star \leq \operatorname{Tr} AX;$$

i.e., the vector x obtained by randomly rounding the SDP solution matrix X is at most 13% suboptimal in expectation. This analysis is due to Goemans and Williamson [20] and yields the best currently known approximation ratio for the MAXCUT problem.

Positive semidefinite: Nesterov [31] first analyzed the case of maximizing a convex quadratic function, i.e., when the matrix A is positive semidefinite. Notice that here we do not have any information on the sign of the individual entries a_{ij}, and thus a "global" analysis is needed instead of the term-by-term analysis of the previous case. The key idea is to use the following result.

Lemma 2.39. *Let $f : \mathbb{R} \to \mathbb{R}$ be a function whose Taylor expansion has only nonnegative coefficients. Given a symmetric matrix X, define a matrix Y as $Y_{ij} = f(X_{ij})$ (equivalently, $Y = f[X]$). Then $X \succeq 0$ implies $Y \succeq 0$.*

This lemma is a rather direct consequence of the Schur product theorem; see Exercise 2.42. Since the scalar function $f(t) = \arcsin(t) - t$ has only nonnegative Taylor coefficients, if $X \succeq 0$, we have $\arcsin[X] \succeq X$, and thus

$$\mathbf{E}[x^T A x] = \frac{2}{\pi} \operatorname{Tr} A \arcsin[X] \geq \frac{2}{\pi} \cdot \operatorname{Tr} AX.$$

Thus, in this case we have

$$\frac{2}{\pi} \cdot \operatorname{Tr} AX \leq \mathbf{E}[x^T A x] \leq f_\star \leq \operatorname{Tr} AX.$$

Notice that $\frac{2}{\pi} \approx 0.636$, so the approximation ratio in this case is slightly worse than for the diagonally dominant case.

Bipartite: This case corresponds to the cost function being bilinear and has been analyzed in [2, 30]. We assume that the matrix A has a structure

$$A = \frac{1}{2} \begin{bmatrix} 0 & S \\ S^T & 0 \end{bmatrix}.$$

Letting $x = [p; q]$, an equivalent formulation is in terms of a *bilinear* optimization problem

$$\text{maximize } p^T S q,$$

where $S \in \mathbb{R}^{n \times m}$ and p, q are in $\{+1, -1\}^n$ and $\{+1, -1\}^m$, respectively.

This problem has a long history in operator theory and functional analysis and was first analyzed (in a quite different form) by Grothendieck. For this class of problems, it follows from his results that a *constant ratio* approximation is possible. In fact, the worst-case ratio (over all instances) between the values of the semidefinite relaxation and the bilinear binary optimization problem is called the *Grothendieck constant* and is usually denoted K_G,

$$K_G := \sup_A \frac{\operatorname{Tr} AX}{f_\star},$$

where X is, as before, the optimal solution of the SDP relaxation. The exact value is this constant is unknown at this time. The argument below is essentially due to Krivine [25] and provides an upper bound to the Grothendieck constant.

Since there are no assumptions about the sign of the entries of the matrix S, we cannot directly apply the techniques discussed earlier to prove a bound on the quality of hyperplane rounding. The basic strategy in Krivine's approach is the following: instead of using hyperplane rounding directly on the solution X of the SDP relaxation, we will apply first a particular componentwise transformation, to obtain a matrix Y, and then apply hyperplane rounding to Y. The reason is that this will considerably simplify the computation of the expected value of the objective function.

To do this, we use a "block" version of Lemma 2.39.

Lemma 2.40. *Let $f, g : \mathbb{R} \to \mathbb{R}$ be functions such that both $f+g$ and $f-g$ have nonnegative Taylor coefficients. Let*

$$X = \begin{bmatrix} X_{11} & X_{12} \\ X_{12}^T & X_{22} \end{bmatrix}, \quad Y = \begin{bmatrix} f(X_{11}) & g(X_{12}) \\ g(X_{12}^T) & f(X_{22}) \end{bmatrix}. \tag{2.30}$$

Then $X \succeq 0$ implies $Y \succeq 0$.

The result now follows from a clever choice of f and g. Let

$$f(t) = \sinh(c_K \pi t/2), \qquad g(t) = \sin(c_K \pi t/2),$$

where the constant $c_K = \frac{2}{\pi}\sinh^{-1}(1) = \frac{2}{\pi}\log(1+\sqrt{2}) \approx 0.5611$ is chosen so $f(1) = 1$. Since

$$\sinh(t) = \sum_{k=0}^{\infty} \frac{t^{2k+1}}{(2k+1)!}, \qquad \sin(t) = \sum_{k=0}^{\infty} (-1)^k \frac{t^{2k+1}}{(2k+1)!},$$

both $f+g$ and $f-g$ have nonnegative Taylor expansions.

Let X be the optimal solution of the SDP relaxation, and define Y as in (2.30). Notice that the matrix Y satisfies $Y \succeq 0$ and $Y_{ii} = 1$. We can therefore apply hyperplane rounding to it to obtain a vector y. Computing the expected value of this solution, we have

$$\mathbf{E}[y^T A y] = \frac{2}{\pi} \operatorname{Tr} A \arcsin[Y] = \frac{2}{\pi} \cdot \operatorname{Tr} S(c_K \pi X_{12}/2) = c_K \cdot \operatorname{Tr} S X_{12},$$

and therefore this gives us a randomized algorithm with expected value c_K times the value of the SDP relaxation. Notice that no inequalities are used in the analysis, so the expected cost of the solution y for this rounding scheme is *exactly* equal to c_K times the optimal value of the SDP:

$$c_K \cdot \operatorname{Tr} S X_{12} = \mathbf{E}[y^T A y] \le f_\star \le \operatorname{Tr} S X_{12}.$$

This analysis gives an upper bound for the Grothendieck constant of $\frac{\operatorname{Tr} S X_{12}}{f_\star} \le K_G \le 1/c_K \approx 1.7822$. It has been recently shown that this rounding method (and thus, the value $1/c_K$) is not the best possible one [8], but the exact approximation ratio is not currently known.

Exercise 2.41. Show that the optimal values of the primal and dual semidefinite programs in (2.27) are equal, i.e., there is no duality gap.

Exercise 2.42. The *entrywise product* $A \circ B$ of two matrices is given by $(A \circ B)_{ij} = A_{ij} B_{ij}$. This product is also known as the *Hadamard* or *Schur* product. The Schur product theorem says that if two matrices A, B are positive semidefinite, so is their product $A \circ B$.

1. Prove the Schur product theorem. (Hint: What happens if one of the matrices is rank one?)
2. Prove Lemmas 2.39 and 2.40.

2.2.3 Stable Sets and the Theta Function

Given an undirected graph $G = (V, E)$, a *stable set* (or *independent set*) is a subset of the set of vertices V with the property that the induced subgraph has no edges. In other words, none of the selected vertices are adjacent to each other.

The *stability number* of a graph, usually denoted by $\alpha(G)$, is the cardinality of the largest stable set. Computing the stability number of a graph is NP-hard. There are many interesting applications of the stable set problem. In particular, it can be used to provide upper bounds on the *Shannon capacity of a graph* [28], a problem that appears in coding theory (when computing the zero-error capacity of a noisy channel [43]). In fact, this was one of the first appearances of semidefinite programming.

In many problems, it is of interest to compute upper bounds on $\alpha(G)$. The Lovász theta function of the graph G is denoted by $\vartheta(G)$ and is defined as the solution of the primal-dual SDP pair:

$$\boxed{\begin{array}{rll} \text{maximize} & \operatorname{Tr} JX & \\ \text{subject to} & \operatorname{Tr} X = 1 & \\ & X_{ij} = 0, & (i,j) \in E, \\ & X \succeq 0, & \end{array} \qquad \begin{array}{rll} \text{minimize} & t & \\ \text{subject to} & Y \preceq tI & \\ & Y_{ii} = 1, & i \in V, \\ & Y_{ij} = 1, & (i,j) \notin E, \end{array}} \tag{2.31}$$

where J is the matrix with all entries equal to one.

The theta function is an upper bound on the stability number, i.e.,

$$\alpha(G) \leq \vartheta(G).$$

The inequality is easy to prove. Consider the indicator vector $\chi(S)$ of any stable set S, and define the matrix $X := \frac{1}{|S|}\chi(S)\chi(S)^T$. It is easy to see that this X is a feasible solution of the primal SDP in (2.31), and it achieves an objective value equal to $|S|$. As a consequence, the inequality above directly follows.

For a class of graphs known as *perfect graphs*,[1] the upper bound given by the theta function is exact; i.e., it is equal to the stability number. Many classes of graphs, such as bipartite, chordal, and comparability graphs, are perfect. Thus, for these graphs one can compute in polynomial time the size of the largest stable set (and a maximum stable set) by solving the SDPs (2.31). Interestingly, at this time no polynomial-time combinatorial methods (not based on semidefinite programming) are known to compute this quantity for all perfect graphs. Further material

[1] A graph is *perfect* if, for every induced subgraph, the chromatic number is equal to size of the largest clique.

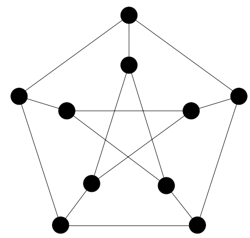

Figure 2.8. *Petersen graph.*

on the theta function of a graph and its applications in combinatorial optimization can be found in Lovász's original paper [28], or the references [19, 21].

Exercise 2.43. Consider the graph in Figure 2.8, known as the Petersen graph. Compute the semidefinite programming upper bound on the size of its largest stable subset (i.e., the Lovász theta function). Is this bound tight? Can you find a stable set that achieves this value?

Exercise 2.44. The *chromatic number* $\chi(G)$ of a graph G is the minimum number of colors needed to color all vertices, in such a way that adjacent vertices receive distinct colors. Show that the inequality

$$\vartheta(G) \leq \chi(\bar{G})$$

holds, where $\bar{G}$ is the complement of the graph G.

Hint: Given a coloring of $\bar{G}$, construct a feasible solution of the dual SDP in (2.31).

2.2.4 Bounded analytic interpolation

In many applications, one tries to find a function in a given function class, that takes specific values at prescribed points. These kinds of questions are known as *interpolation* problems. A classical and important class of interpolation problems involves *bounded analytic functions*. The mathematical background for these problems is reviewed and developed further in Chapter 9. Good general references include [3] for the theoretical aspects, and [24, 47] for specific applications of interpolation in systems and control theory.

We discuss here two specific problems related to this area. The first is the computation of the $\mathcal{H}_\infty$-norm of an analytic function, and the second is the classical Nevanlinna–Pick interpolation problem. Additional connections between analytic interpolation and convex optimization can be found in [6].

Norms of rational analytic functions. Let $\mathbb{D}$ be the complex open unit disk $\mathbb{D} = \{z \in \mathbb{C} : |z| < 1\}$. Consider a scalar rational function of a complex variable z given by

$$f(z) = c^T(z^{-1}I - A)^{-1}b + d, \tag{2.32}$$

where $A \in \mathbb{R}^{n\times n}$, $b, c \in \mathbb{R}^{n\times 1}$, and $d \in \mathbb{R}$. We assume that all the eigenvalues of A are in $\mathbb{D}$: $|\lambda_i(A)| < 1$ (i.e., A is *Schur stable*). It follows that $z^{-1}I - A$ is nonsingular on $|z| \leq 1$, and thus $f(z)$ is analytic[2] on the domain $\mathbb{D}$.

The question of interest is to compute the $\mathcal{H}_\infty$-norm of the function $f(z)$, i.e., its maximum absolute value on the unit disk:

$$\|f\|_\infty = \sup_{z\in\mathbb{D}} |f(z)|. \tag{2.33}$$

It can be shown, by using the maximum principle in complex analysis, that it is enough to compute the supremum of $f(z)$ on the boundary of the domain, i.e., the unit circle $|z| = 1$. A fairly complete characterization of this question is available. It is known in the literature under several names, such as the *Kalman–Yakubovich–Popov* lemma [38], or the *bounded real lemma*, or (as a special case of) the *structured singular value* theory [34], among others. The statement, presented below, characterizes this norm in terms of the solution of a semidefinite programming problem.

Theorem 2.45. *Consider a function $f(z)$ as in (2.32), with $|\lambda_i(A)| < 1$. Then, $\|f\|_\infty < \gamma$ if and only if the semidefinite program*

$$\begin{bmatrix} A & b \\ c^T & d \end{bmatrix}^T \begin{bmatrix} P & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} A & b \\ c^T & d \end{bmatrix} \prec \begin{bmatrix} P & 0 \\ 0 & \gamma^2 \end{bmatrix}, \qquad P \succ 0, \tag{2.34}$$

is feasible, where the decision variable is the matrix $P \in \mathcal{S}^n$.

A full proof can be found, for instance, in [3, 47]. We present here only the "easy" direction, i.e., showing that if (2.34) holds, then we have $\|f(z)\|_\infty < \gamma$. For this, let $v = (z^{-1}I - A)^{-1}b$, and multiply the first inequality in (2.34) left and right by $[v^* \; 1]$ and its conjugate transpose, respectively. From the identity

$$\begin{bmatrix} A & b \\ c^T & d \end{bmatrix} \begin{bmatrix} v \\ 1 \end{bmatrix} = \begin{bmatrix} z^{-1}v \\ f(z) \end{bmatrix},$$

we have that

$$(|z^{-1}|^2 - 1)(v^*Pv) + (|f(z)|^2 - \gamma^2) < 0,$$

and thus the conclusion directly follows. The converse direction takes a bit more work; see Chapter 9. There are extensions of this result to the matrix case, i.e., where $f(z)$ is matrix-valued.

Exercise 2.46. Use the given formulation to compute the $\mathcal{H}_\infty$-norm of the analytic function $f(z) = \frac{z-2}{z^3+z^2-z+3}$. How can you compute, from the semidefinite formulation, a value of z at which the maximum is achieved?

[2] We remark that the notation used here is slightly different from the usual notation in systems and control theory, where z is used instead of z^{-1} in (2.32). The reason is that for interpolation, it is more natural to use functions that are analytic on $\mathbb{D}$ (poles outside the unit circle) than functions that are analytic outside $\mathbb{D}$. To avoid distracting technical issues of controllability and/or observability, we use strict inequalities throughout.

Exercise 2.47. Formulate a similar statement for the matrix case. Do the same formulas work?

Nevanlinna–Pick interpolation. Consider now the following problem. We want to find an analytic function on $\mathbb{D}$ satisfying the interpolation constraints:

$$f(a_k) = c_k \qquad \text{for} \quad k = 1, \ldots, m, \tag{2.35}$$

where $a_k \in \mathbb{D}$. When does there exist an analytic function, satisfying the interpolation conditions, whose absolute value is bounded by 1 on the unit disk?

Clearly, a necessary condition is that the interpolated values c_k must satisfy $|c_k| \leq 1$ for all k. However, due to the analyticity constraint, this is not sufficient. Consider, for instance, the case $m = 2$ and the constraints $f(0) = 0$ and $f(1/2) = c$. In this case, a necessary condition is $|c| \leq 1/2$, which is stronger than the "obvious" condition $|c| \leq 1$. To see this, notice that, due to the first interpolation constraint, $f(z)$ must have the form $f(z) = zg(z)$, where $g(z) = f(z)/z$ is also analytic on $\mathbb{D}$ and bounded by one (by the maximum modulus theorem, since $|f(z)| = |g(z)|$ on the unit circle). Thus, we must have $1 \geq |g(1/2)| = 2|c|$, and thus $|c| \leq 1/2$.

Necessary and sufficient conditions for the interpolation problem to be feasible are given by the Nevanlinna–Pick theorem; see Chapter 9. The formulation below is convenient from the optimization viewpoint.

Theorem 2.48. *There exists a function $f(z)$ analytic on $\mathbb{D}$, satisfying the norm bound $\|f(z)\|_\infty \leq \gamma$ and the interpolation constraints* (2.35) *if and only if*

$$\begin{bmatrix} \gamma Z & C^* \\ C & \gamma Z^{-1} \end{bmatrix} \succeq 0, \tag{2.36}$$

where $Z_{jk} = \frac{1}{1 - a_j^* a_k}$ and $C = \operatorname{Diag}(c_1, \ldots, c_m)$.

Using Schur complements, it can be easily seen that this formulation is equivalent to the more usual characterization where the $m \times m$ Pick matrix P given by

$$P_{jk} = \frac{\gamma^2 - c_j^* c_k}{1 - a_j^* a_k}$$

is required to be positive semidefinite (e.g., Section 9.8). The advantage of condition (2.36) is that it is *linear* in the interpolation values c_k. This allows its use in a variety of system identification problems; see, for instance, [11, 35]. The Nevanlinna–Pick interpolation problem has many important applications in systems and control theory; see, for instance, [14] and [47] and the references therein.

2.2.5 Euclidean Distance Matrices

Assume we are given a list of pairwise distances between a finite number of points. Under what conditions can the points be embedded in some finite-dimensional space and those distances be realized as the *Euclidean* metric between the embedded points? This problem appears in a large number of applications, including distance geometry, computational chemistry, sensor network localization, and machine learning.

Concretely, assume we have a list of distances d_{ij} for $1 \leq i < j \leq n$. We would like to find points $x_i \in \mathbb{R}^k$ (for some value of k) such that $\|x_i - x_j\| = d_{ij}$ for all i, j. What are necessary and sufficient conditions for such an embedding to exist? In 1935, Schoenberg [41] gave an exact characterization in terms of the semidefiniteness of the matrix of squared distances.

Theorem 2.49. *The distances d_{ij} can be embedded in a Euclidean space if and only if the $n \times n$ matrix*

$$D := \begin{bmatrix} 0 & d_{12}^2 & d_{13}^2 & \dots & d_{1n}^2 \\ d_{12}^2 & 0 & d_{23}^2 & \dots & d_{2n}^2 \\ d_{13}^2 & d_{23}^2 & 0 & \dots & d_{3n}^2 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ d_{1n}^2 & d_{2n}^2 & d_{3n}^2 & \dots & 0 \end{bmatrix}$$

is negative semidefinite on the subspace orthogonal to the vector $e := (1, 1, \dots, 1)$.

Proof. We show only the necessity of the condition. Assume an embedding exists, i.e., there are points $x_i \in \mathbb{R}^k$ such that $d_{ij} = \|x_i - x_j\|$. Consider now the Gram matrix G of inner products

$$G := \begin{bmatrix} \langle x_1, x_1 \rangle & \langle x_1, x_2 \rangle & \dots & \langle x_1, x_n \rangle \\ \langle x_2, x_1 \rangle & \langle x_2, x_2 \rangle & \dots & \langle x_2, x_n \rangle \\ \vdots & \vdots & \ddots & \vdots \\ \langle x_n, x_1 \rangle & \langle x_n, x_2 \rangle & \dots & \langle x_n, x_n \rangle \end{bmatrix} = [x_1, \dots, x_n]^T [x_1, \dots, x_n],$$

which is positive semidefinite by construction. Since $D_{ij} = \|x_i - x_j\|^2 = \langle x_i, x_i \rangle + \langle x_j, x_j \rangle - 2\langle x_i, x_j \rangle$, we have

$$D = \operatorname{Diag}(G) \cdot e^T + e \cdot \operatorname{Diag}(G)^T - 2G,$$

from which the result directly follows. $\square$

Notice that the dimension of the embedding is given by the rank k of the Gram matrix G.

For more on this and related embedding problems, good starting points are Schoenberg's original paper [41] as well as the book [15].

Exercise 2.50. Consider the Euclidean distance matrix characterization in Theorem 2.49. Show that it implies the triangle inequality $d_{ik} \leq d_{ij} + d_{jk}$ for all triples (x_i, x_j, x_k) of points. Is the converse true?

2.2.6 Rank Minimization and Nuclear Norm

An interesting class of optimization problems appearing in many application domains is *rank minimization* problems. These have the form

$$\begin{array}{ll} \text{minimize} & \operatorname{rank} X \\ \text{subject to} & X \in \mathcal{C}, \end{array} \tag{2.37}$$

where the matrix $X \in \mathbb{R}^{m\times n}$ is the decision variable, and $\mathcal{C}$ is a given convex constraint set. Notice that the cost function is integer-valued, and thus (unless the problem is trivial) these optimization problems are not convex.

Rank minimization questions arise in many different areas, since notions such as order, complexity, and dimensionality can often be expressed by means of the rank of an appropriate matrix. For example, a low-rank matrix could correspond to a low-degree statistical model for a random process (e.g., factor analysis), a low-order realization of a linear dynamical system, or a low-dimensional embedding of data in Euclidean space (as in Section 2.2.5). If the set of models that satisfy the desired constraints is convex, then choosing the simplest one in a given family can be formulated as a rank minimization problem of the form (2.37).

In general, rank minimization problems can be quite difficult to solve, both in theory and practice. However, several researchers have proposed heuristic techniques to obtain good approximate solutions. A particularly interesting method is the *nuclear norm* heuristic, originally proposed in [17, 16]. In this method, instead of directly solving the problem (2.37), one solves instead

$$\begin{array}{ll} \text{minimize} & \|X\|_* \\ \text{subject to} & X \in \mathcal{C}, \end{array} \tag{2.38}$$

where $\|\cdot\|_*$ is the *nuclear norm* defined earlier in (2.11). In other words, the "difficult" objective function (rank) is replaced by a "nicer" cost function (nuclear norm) which is convex, and thus the resulting problem is convex.

Under certain conditions on the set $\mathcal{C}$, it has been shown that the solution of the problem (2.38) coincides with the lowest-rank solution, i.e., the "true" solution of (2.37). For example, a typical formulation (see, e.g., [39] for a specific statement) would establish that if the set $\mathcal{C}$ is a subspace of dimension $O(n \log n)$, uniformly chosen according to a natural rotation-invariant probability measure, then the nuclear norm heuristic succeeds with high probability.

Atomic norms. An interesting generalization of these methods is obtained by considering more general *atomic norms* [10]. Consider a set $\mathcal{A}$ of *atoms* v_i in some vector space V (the set $\mathcal{A}$ can be finite or infinite). Given an element $a \in V$, we are interested in the "smallest" decomposition of a in terms of the elements v_i, i.e., the one that satisfies

$$\begin{array}{ll} \text{minimize} & \sum_i |\alpha_i| \\ \text{subject to} & a = \sum_i \alpha_i v_i. \end{array} \tag{2.39}$$

We can then define the *atomic norm* $\|a\|_{\mathcal{A}}$ as the optimal value of this optimization problem. If the set of atoms is finite, this is a linear programming problem. In most situations of interest, however, the set $\mathcal{A}$ is either infinite or exponentially large, in which case an LP formulation is impractical. In certain cases, however, we can still compute this norm efficiently. For instance, in the case where the set of atoms $\mathcal{A}$ corresponds to the rank one matrices uv^T, where $\|u\| = \|v\| = 1$, then this norm corresponds exactly to the matrix nuclear norm defined earlier.

For many problems, however, we would like to consider more general sets of atoms. A particularly interesting case is when the atoms are the rank one matrices with ± 1 entries. In other words, the atoms are given by $\mathcal{A} = \{vw^T \in \mathbb{R}^{m\times n} : v \in \mathbb{R}^m, v_i^2 = 1, w \in \mathbb{R}^n, w_i^2 = 1\}$. In this case, the norm (2.39) is in general NP-hard to compute. However, a nice computable approximation is available, known as the γ_2 or max-norm. This norm is defined as $\|A\|_{\gamma_2} := \max_{\|u\|=1,\|v\|=1} \|A \circ uv^T\|_*$, where $\circ$ is the entrywise product, and can be computed as the optimal value of the primal-dual pair of semidefinite programs:

$$
\begin{array}{llcll}
\text{maximize} & \operatorname{Tr} A^T Y & \qquad & \text{minimize} & t \\
\text{subject to} & \begin{bmatrix} \operatorname{Diag}(p) & Y \\ Y^T & \operatorname{Diag}(q) \end{bmatrix} \succeq 0, & & \text{subject to} & \begin{bmatrix} V & A \\ A^T & W \end{bmatrix} \succeq 0, \\
& \displaystyle\sum_{i=1}^m p_i + \sum_{i=1}^n q_i = 2, & & & V_{ii} = t, \\
& & & & W_{ii} = t.
\end{array}
\tag{2.40}
$$

It can be easily seen that (2.40) gives a lower bound on the optimal value of (2.39), i.e., $\|A\|_{\gamma_2} \le \|A\|_{\mathcal{A}}$. Indeed, if $A = \sum_i \alpha_i v_i w_i^T$, where the v_i and w_i are ± 1 vectors, then choosing $V = \sum_i |\alpha_i| v_i v_i^T$, $W = \sum_i |\alpha_i| w_i w_i^T$, and $t = \sum_i |\alpha_i|$ gives a feasible solution for the right-hand side of (2.40). As discussed in Exercise 2.53, the γ_2-norm actually yields a constant approximation ratio to the atomic norm for this specific set of atoms. The γ_2-norm is of great importance in a number of applications, including communication complexity; see, e.g., [27].

Exercise 2.51. Check that the expression (2.39) correctly defines a matrix norm by verifying homogeneity and the triangle inequality. What properties are needed on the atom set $\mathcal{A}$ to ensure that the norm is well defined and nonzero at every nontrivial point?

Exercise 2.52. Let the set of atoms $\mathcal{A}$ be the rank one matrices of the form vw^T, where $\|v\| = \|w\| = 1$. Show that the corresponding atomic norm is the "standard" nuclear norm (sum of singular values).

Exercise 2.53. Using the results in Section 2.2.2, show that in the case where the atoms are the rank one matrices with ± 1 entries, the following inequality holds:

$$\|A\|_{\gamma_2} \le \|A\|_{\mathcal{A}} \le K_G \|A\|_{\gamma_2},$$

where K_G is the Grothendieck constant.

Exercise 2.54. Based on the previous exercise, explain the geometric relationship between the unit ball of the γ_2-norm in $\mathbb{R}^{m\times n}$ and the elliptope $\mathcal{E}_{m+n}$ defined earlier in Section 2.1.3.

2.3 Algorithms and Software

2.3.1 Algorithms

In this section we describe a few algorithmic and complexity aspects of the numerical solution of semidefinite optimization problems. For a complete treatment, we refer the reader to articles and monographs such as [13, 32, 44, 45].

Semidefinite programs are convex optimization problems and, as such, can be solved using general convex optimization techniques. Under "natural" assumptions (e.g., to rule out doubly exponentially small solutions), semidefinite optimization is solvable in polynomial time, in the sense that ϵ-suboptimal, weakly feasible solutions can be computed in time polynomial in $\log \frac{1}{\epsilon}$. This follows, for instance, from general results about the ellipsoid method [21].

Despite these nice theoretical results, the ellipsoid method is often too slow in practice. Since SDP is a generalization of linear programming, it is natural that some of the most effective practical methods for SDP have been inspired by state-of-the-art techniques from LP. This has led to the development of interior-point methods [1, 32] for SDP. The basic idea of interior-point methods is to consider the optimality conditions of Lemma 2.12 and to perturb the complementarity slackness condition to $(C - \sum_i A_i y_i)X = \mu I$. As μ varies, these equations implicitly define a curve (X_μ, y_μ) called the *central path*, and to solve the original problem we need to compute (X_μ, y_μ) as $\mu \to 0$. These equations are relatively easy to solve for large μ, and by carefully decreasing the value of μ, it is possible to use Newton's method to efficiently track solutions as μ decreases to zero. There are several different versions of these methods (depending on the exact form of the equations to which Newton's method is applied), although they all share fairly similar features. In particular, primal-dual interior-point methods of this kind are among the most efficient known methods for small- and medium-scale SDP problems.

Besides interior-point methods, there are several alternative techniques for solving SDPs that are sometimes preferable to "pure" primal-dual methods due to speed or memory efficiency issues. Examples of these are techniques based on low-rank factorizations [9], spectral bundle methods [23], or augmented Lagrangian methods for large-scale problems [46], among others.

2.3.2 Software

There are a number of useful software packages for polyhedral computations, linear and semidefinite programming, and algebraic visualization. We present below a partial annotated selection. A few good up-to-date web resources for general information about semidefinite programming include Christoph Helmberg's SDP page www-user.tu-chemnitz.de/$\sim$helmberg/semidef.html and the SDPA website sdpa.sourceforge.net.

Polyhedral computations. The first class of software packages we discuss is polyhedral manipulation codes and libraries. Almost all of them allow us to convert an inequality representation of a polyhedron (usually called an H-representation) into vertices/extreme rays (V-representation), and vice versa, as well as much more complicated operations between polyhedra.

- `cdd`, by Komei Fukuda.
  www.ifor.math.ethz.ch/~fukuda/cdd_home.
- `lrs`, by David Avis.
  cgm.cs.mcgill.ca/~avis/C/lrs.html.
- `polymake`, by Ewgenij Gawrilow and Michael Joswig (main authors).
  polymake.org.
- `PORTA`, by Thomas Christof and Andreas Löbel.
  typo.zib.de/opt-long_projects/Software/Porta.

Linear programming. For formulating and solving linear programs, many codes are available, ranging from academic implementations suitable for relatively small problems to industrial-scale solvers. The following is a necessarily partial list:

- `GLPK` – GNU Linear Programming Kit
  www.gnu.org/s/glpk. This is an open-source package for solving large-scale linear programming problems, using either simplex or interior-point methods. GLPK can also solve integer programming problems and can be used as a callable C library.
- `CLP` – LP solver, part of the COIN-OR (COmputational INfrastructure for Operations Research) suite of open source software. www.coin-or.org
- `CPLEX` – Perhaps the best-known commercial solver, now being developed and marketed by IBM.

Semidefinite programming. Although SDP is much more recent than linear programming, fortunately many good software packages are already available. Among the most well-known are the following:

- CSDP, originally by Brian Borchers, now a COIN-OR project:
  projects.coin-or.org/Csdp
- SDPA, by the research group of Masakazu Kojima, sdpa.sourceforge.net. Several versions of the SDPA solver are available, including parallel and variable-precision floating-point arithmetic, in MATLAB and C++ versions.
- SDPT3, by Kim-Chuan Toh, Reha Tütüncü, and Michael Todd.
  www.math.nus.edu.sg/~mattohkc/sdpt3.html. SDPT3 is a MATLAB package for linear, quadratic, and semidefinite programming. It can also handle determinant maximization problems, as well as problems with complex data.

- SeDuMi, originally by Jos Sturm, currently being maintained by the optimization group at Lehigh University (sedumi.ie.lehigh.edu), is a widely used MATLAB package for linear, quadratic, second order conic, and semidefinite optimization, and any combination of these.

An easy and convenient way to "try out" many of these packages, without installing them in a local machine, is through the NEOS Optimization server (neos-server.org), currently hosted by the University of Wisconsin-Madison.

Parsers. In practice, specifying a semidefinite programming problem by explicitly defining matrices A_i, C, and b in (SDP-P) can be cumbersome and error-prone. A much more convenient and reliable way is to use a "natural" description of the variables and inequalities and to automatically translate these into standard form using a parser or modeling language. Two well-known and convenient modeling environments for semidefinite programming are the following:

- CVX, by Michael Grant and Stephen Boyd.
  cvxr.com/cvx. CVX is a MATLAB-based "disciplined convex programming" software. It is particularly well suited to conic optimization, including semidefinite and geometric programming.

- YALMIP, by Johan Löfberg.
  yalmip.org. YALMIP is a MATLAB-based parser and solver for the modeling and solution of convex and nonconvex optimization problems.

Bibliography

[1] F. Alizadeh. Interior point methods in semidefinite programming with applications to combinatorial optimization. *SIAM J. Optim.*, 5(1):13–51, 1995.

[2] N. Alon and A. Naor. Approximating the cut-norm via Grothendieck's inequality. In *Proceedings of the Thirty-Sixth Annual ACM Symposium on Theory of Computing*, ACM, New York, 2004, pp. 72–80.

[3] J.A. Ball, I. Gohberg, and L. Rodman. *Interpolation of Rational Matrix Functions*. Birkhäuser, Basel, 1990.

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

[5] D. Bertsimas and J. N. Tsitsiklis. *Introduction to Linear Optimization*. Athena Scientific, Cambridge, MA, 1997.

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

[7] S. Boyd and L. Vandenberghe. *Convex Optimization*. Cambridge University Press, Cambridge, UK, 2004.

[8] M. Braverman, K. Makarychev, Y. Makarychev, and A. Naor. The Grothendieck constant is strictly smaller than Krivine's bound. In the *IEEE 52nd Annual Symposium on Foundations of Computer Science (FOCS)*, IEEE, Washington, DC, 2011, pp. 453–462.

[9] S. Burer and R. D.C. Monteiro. A nonlinear programming algorithm for solving semidefinite programs via low-rank factorization. *Mathematical Programming*, 95(2):329–357, 2003.

[10] V. Chandrasekaran, B. Recht, P. A. Parrilo, and A.S. Willsky. The convex geometry of linear inverse problems. *Foundations of Computational Mathematics*, 12:805–849, 2012.

[11] J. Chen, C.N. Nett, and M.K.H. Fan. Worst case system identification in $\mathcal{H}_\infty$: Validation of a priori information, essentially optimal algorithms, and error bounds. *IEEE Transactions on Automatic Control*, 40(7):1260–1265, 1995.

[12] V. Chvátal. *Linear Programming*. W.H. Freeman, New York, 1983.

[13] E. de Klerk. *Aspects of Semidefinite Programming: Interior Point Algorithms and Selected Applications*, Applied Optimization 65. Kluwer Academic Publishers, Dordrecht, The Netherlands, 2002.

[14] P. Delsarte, Y. Genin, and Y. Kamp. On the role of the Nevanlinna–Pick problem in circuit and system theory. *International Journal of Circuit Theory and Applications*, 9(2):177–187, 1981.

[15] M. M. Deza and M. Laurent. *Geometry of Cuts and Metrics*, Algorithms and Combinatorics 15. Springer-Verlag, Berlin, 1997.

[16] M. Fazel. *Matrix Rank Minimization with Applications*. Ph.D. thesis, Stanford University, Stanford, CA, 2002.

[17] M. Fazel, H. Hindi, and S.P. Boyd. A rank minimization heuristic with application to minimum order system approximation. In *Proceedings of the American Control Conference*, volume 6, IEEE, Washington, DC, 2001, pp. 4734–4739.

[18] M. R. Garey and D. S. Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness*. W. H. Freeman, New York, 1979.

[19] M. X. Goemans. Semidefinite programming in combinatorial optimization. *Math. Programming*, 79(1–3):143–161, 1997.

[20] M. X. Goemans and D. P. Williamson. Improved approximation algorithms for maximum cut and satisfiability problems using semidefinite programming. *Journal of the ACM*, 42(6):1115–1145, 1995.

[21] M. Grötschel, L. Lovász, and A. Schrijver. *Geometric Algorithms and Combinatorial Optimization*, 2nd ed., Algorithms and Combinatorics 2. Springer-Verlag, Berlin, 1993.

[22] O. Güler. Hyperbolic polynomials and interior point methods for convex programming. *Math. Oper. Res.*, 22(2):350–377, 1997.

[23] C. Helmberg and F. Rendl. A spectral bundle method for semidefinite programming. *SIAM Journal on Optimization*, 10(3):673–696, 2000.

[24] J.W. Helton. *Operator Theory, Analytic Functions, Matrices, and Electrical Engineering*. CBMS Regional Conference Series in Mathematics 68. AMS, Providence, RI, 1987.

[25] J.L. Krivine. Constantes de Grothendieck et fonctions de type positif sur les spheres. *Adv. Math*, 31:16–30, 1979.

[26] M. Laurent and S. Poljak. On a positive semidefinite relaxation of the cut polytope. *Linear Algebra and Its Applications*, 223:439–461, 1995.

[27] T. Lee and A. Shraibman. Lower bounds in communication complexity. *Foundations and Trends in Theoretical Computer Science*, 3(4), 2009.

[28] L. Lovász. On the Shannon capacity of a graph. *IEEE Transactions on Information Theory*, 25(1):1–7, 1979.

[29] J. Matoušek and B. Gärtner. *Understanding and Using Linear Programming.* Springer-Verlag, New York, 2007.

[30] A. Megretski. Relaxations of quadratic programs in operator theory and system analysis. In *Systems, Approximation, Singular Integral Operators, and Related Topics (Bordeaux,* 2000*)*, Oper. Theory Adv. Appl. 129. Birkhäuser, Basel, 2001, pp. 365–392.

[31] Y. Nesterov. Semidefinite relaxation and nonconvex quadratic optimization. *Optimization Methods and Software*, 9:141–160, 1998.

[32] Y. E. Nesterov and A. Nemirovski. *Interior Point Polynomial Methods in Convex Programming*, Studies in Applied Mathematics 13. SIAM, Philadelphia, 1994.

[33] J. Nie, P. A. Parrilo, and B. Sturmfels. Semidefinite representation of the k-ellipse. *IMA Volumes in Mathematics and Its Applications*, 146:117–132, 2008.

[34] A. Packard and J. C. Doyle. The complex structured singular value. *Automatica J. IFAC*, 29(1):71–109, 1993.

[35] P. A. Parrilo, M. Sznaier, R.S. Sánchez Peña, and T. Inanc. Mixed time/frequency-domain based robust identification. *Automatica J. IFAC*, 34(11):1375–1389, 1998.

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

[37] M. V. Ramana, L. Tunçel, and H. Wolkowicz. Strong duality for semidefinite programming. *SIAM J. Optim.*, 7(3):641–662, 1997.

[38] A. Rantzer. On the Kalman-Yakubovich-Popov lemma. *Systems & Control Letters*, 28:7–10, 1996.

[39] B. Recht, M. Fazel, and P. A. Parrilo. Guaranteed minimum-rank solutions of linear matrix equations via nuclear norm minimization. *SIAM Review*, 52(3):471–501, 2010.

[40] J. Renegar. Hyperbolic programs, and their derivative relaxations. *Found. Comput. Math.*, 6(1):59–79, 2006.

[41] I. J. Schoenberg. Remarks to Maurice Fréchet's article "Sur la définition axiomatique d'une classe d'espace distanciés vectoriellement applicable sur l'espace de Hilbert." *Ann. of Math.* (2), 36(3):724–732, 1935.

[42] A. Schrijver. *Theory of Linear and Integer Programming*. Wiley, New York, 1986.

[43] C. Shannon. The zero error capacity of a noisy channel. *IRE Transactions on Information Theory*, 2(3):8–19, 1956.

[44] M. Todd. Semidefinite optimization. *Acta Numerica*, 10:515–560, 2001.

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

[46] X.Y. Zhao, D. Sun, and K.C. Toh. A Newton-CG augmented Lagrangian method for semidefinite programming. *SIAM Journal on Optimization*, 20:1737–1765, 2010.

[47] K. Zhou, K. Glover, and J. C. Doyle. *Robust and Optimal Control*. Prentice Hall, Englewood Cliffs, NJ, 1995.

[48] G. M. Ziegler. *Lectures on Polytopes*, Graduate Texts in Mathematics 152. Springer-Verlag, New York, 1995.