Appendix A

Notation and Prerequisites

A.1 Notation

• $\mathbb{Z}$, $\mathbb{R}$, $\mathbb{C}$ stand for the sets of all integers, reals, and complex numbers, respectively.
• $\mathbb{C}^{m\times n}$, $\mathbb{R}^{m\times n}$ stand for the spaces of complex, respectively, real $m \times n$ matrices. We write $\mathbb{C}^n$ and $\mathbb{R}^n$ as shorthands for $\mathbb{C}^{n\times 1}$, $\mathbb{R}^{n\times 1}$, respectively.
For $A \in \mathbb{C}^{m\times n}$, A^T stands for the transpose, and A^H for the conjugate transpose of A:

$$(A^H)_{rs} = \overline{A_{sr}},$$

where $\overline{z}$ is the conjugate of $z \in \mathbb{C}$.
• Both $\mathbb{C}^{m\times n}$, $\mathbb{R}^{m\times n}$ are equipped with the inner product

$$\langle A, B\rangle = \mathrm{Tr}(AB^H) = \sum_{r,s} A_{rs}\overline{B_{rs}}.$$

The norm associated with this inner product is denoted by $\|\cdot\|_2$.
• For $p \in [1,\infty]$, we define the p-norms $\|\cdot\|_p$ on $\mathbb{C}^n$ and $\mathbb{R}^n$ by the relation

$$\|x\|_p = \left\{ \begin{array}{ll} \left(\sum_i |x_i|^p\right)^{1/p}, & 1 \le p < \infty \\ \lim_{p\to\infty} \|x\|_p = \max_i |x_i|, & p = \infty \end{array} \right., \; 1 \le p \le \infty.$$

Note that when $p, q \in [1,\infty]$ and $\frac{1}{p} + \frac{1}{q} = 1$, then the norms $\|\cdot\|_p$ and $\|\cdot\|_q$ are conjugates of each other:

$$\|x\|_p = \max_{y:\|y\|_q \le 1} |\langle x, y\rangle|.$$

In particular, $|\langle x, y\rangle| \le \|x\|_p\|y\|_q$ (Hölder inequality).
• We use the notation I_m, $0_{m\times n}$ for the unit $m \times m$, respectively, the zero $m \times n$ matrices.
• $\mathbf{H}^m$, $\mathbf{S}^m$ are real vector spaces of $m \times m$ Hermitian, respectively, real symmetric matrices. Both are Euclidean spaces w.r.t. the inner product $\langle\cdot,\cdot\rangle$.
• We use "MATLAB notation": when $A_1, ..., A_k$ are matrices with the same number of rows, $[A_1, ..., A_k]$ denotes the matrix with the same number of rows obtained by writing, from left to right, first the columns of A_1, then the columns of A_2, and so on. When $A_1, ..., A_k$ are matrices with the same number of columns, $[A_1; A_2; ...; A_k]$ stands for the matrix with the same number of columns obtained by writing, from top to bottom, first the rows of A_1, then the rows of A_2, and so on.

• For a Hermitian/real symmetric $m \times m$ matrix A, $\lambda(A)$ is the vector of eigenvalues $\lambda_r(A)$ of A taken with their multiplicities in the non-ascending order:

$$\lambda_1(A) \geq \lambda_2(A) \geq \ldots \geq \lambda_m(A).$$

• For an $m \times n$ matrix A, $\sigma(A) = (\sigma_1(A), \ldots, \sigma_n(A))^T$ is the vector of singular values of A:

$$\sigma_r(A) = \lambda_r^{1/2}(A^H A),$$

and

$$\|A\|_{2,2} = \|A\| = \sigma_1(A) = \max\{\|Ax\|_2 : x \in \mathbb{C}^n, \|x\|_2 \leq 1\}$$

(by evident reasons, when A is real, one can replace $\mathbb{C}^n$ in the right hand side with $\mathbb{R}^n$).

• For Hermitian/real symmetric matrices A, B, we write $A \succeq B$ ($A \succ B$) to express that $A - B$ is positive semidefinite (resp., positive definite).

A.2 Conic Programming

A.2.1 Euclidean Spaces, Cones, Duality

Euclidean spaces

A *Euclidean space* is a finite dimensional linear space over reals equipped with an *inner product* $\langle x, y\rangle_E$ — a bilinear and symmetric real-valued function of $x, y \in E$ such that $\langle x, x\rangle_E > 0$ whenever $x \neq 0$.

Example: The standard Euclidean space $\mathbb{R}^n$. This space is comprised of n-dimensional real column vectors with the standard coordinate-wise linear operations and the inner product $\langle x, y\rangle_{\mathbb{R}^n} = x^T y$. $\mathbb{R}^n$ is a universal example of an Euclidean space: for every Euclidean n-dimensional space $(E, \langle\cdot,\cdot\rangle_E)$ there exists a one-to-one linear mapping $x \mapsto Ax : \mathbb{R}^n \to E$ such that $x^T y \equiv \langle Ax, Ay\rangle_E$. All we need in order to build such a mapping, is to find an *orthonormal basis* $e_1, \ldots, e_n$, $n = \dim E$, in E, that is, a basis such that $\langle e_i, e_j\rangle_E = \delta_{ij} \equiv \begin{cases} 1, & i = j \\ 0, & i \neq j \end{cases}$; such a basis always exists. Given an orthonormal basis $\{e_i\}_{i=1}^n$, a one-to-one mapping $A : \mathbb{R}^n \to E$ preserving the inner product is given by $Ax = \sum_{i=1}^n x_i e_i$.

Example: The space $\mathbb{R}^{m\times n}$ of $m \times n$ real matrices with the Frobenius inner product. The elements of this space are $m \times n$ real matrices with the standard linear operations and the inner product $\langle A, B\rangle_F = \mathrm{Tr}(AB^T) = \sum_{i,j} A_{ij} B_{ij}$.

Example: The space $\mathbf{S}^n$ of $n \times n$ real symmetric matrices with the Frobenius inner product. This is the subspace of $\mathbb{R}^{n\times n}$ comprised of all symmetric $n \times n$ matrices; the inner product is inherited from the embedding space. Of course, for symmetric matrices, this product can be written down without transposition:

$$A, B \in \mathbf{S}^n \Rightarrow \langle A, B\rangle_F = \mathrm{Tr}(AB) = \sum_{i,j} A_{ij} B_{ij}.$$

Example: The space $\mathbf{H}^n$ of $n \times n$ Hermitian matrices with the Frobenius inner product. This is the real linear space comprised of $n \times n$ Hermitian matrices; the inner product is

$$\langle A, B\rangle = \mathrm{Tr}(AB^H) = \mathrm{Tr}(AB) = \sum_{i,j=1}^n A_{ij} \overline{B_{ij}}.$$

Linear forms on Euclidean spaces

Every homogeneous linear form $f(x)$ on a Euclidean space $(E, \langle\cdot,\cdot\rangle_E)$ can be represented in the form $f(x) = \langle e_f, x\rangle_E$ for certain vector $e_f \in E$ uniquely defined by $f(\cdot)$. The mapping $f \mapsto e_f$ is a one-to-one linear mapping of the space of linear forms on E onto E.

Conjugate mapping

Let $(E, \langle\cdot,\cdot\rangle_E)$ and $(F, \langle\cdot\cdot\rangle_F)$ be Euclidean spaces. For a linear mapping $A : E \to F$ and every $f \in F$, the function $\langle Ae, f\rangle_F$ is a linear function of $e \in E$ and as such it is representable as $\langle e, A^* f\rangle_E$ for certain uniquely defined vector $A^* f \in E$. It is immediately seen that the mapping $f \mapsto A^* f$ is a linear mapping of F into E; the characteristic identity specifying this mapping is

$$\langle Ae, f\rangle_F = \langle e, A^* f\rangle \ \forall (e \in E, f \in F).$$

The mapping A^* is called *conjugate* to A. It is immediately seen that the conjugation is a linear operation with the properties $(A^*)^* = A$, $(AB)^* = B^* A^*$. If $\{e_j\}_{j=1}^m$ and $\{f_i\}_{i=1}^n$ are orthonormal bases in E, F, then every linear mapping $A : E \to F$ can be associated with the matrix $[a_{ij}]$ (“matrix of the mapping in the pair of bases in question”) according to the identity

$$A \sum_{j=1}^{m} x_j e_j = \sum_i \left[\sum_j a_{ij} x_j\right] f_i$$

(in other words, a_{ij} is the i-th coordinate of the vector Ae_j in the basis $f_1, ..., f_n$). With this representation of linear mappings by matrices, the matrix representing A^* in the pair of bases $\{f_i\}$ in the argument and $\{e_j\}$ in the image spaces of A^* is the transpose of the matrix representing A in the pair of bases $\{e_j\}$, $\{f_i\}$.

Cones in Euclidean space

A nonempty subset $\mathbf{K}$ of a Euclidean space $(E, \langle\cdot,\cdot\rangle_E)$ is called a cone, if it is a convex set comprised of rays emanating from the origin, or, equivalently, whenever $t_1, t_2 \geq 0$ and $x_1, x_2 \in \mathbf{K}$, we have $t_1 x_1 + t_2 x_2 \in \mathbf{K}$.

A cone $\mathbf{K}$ is called *regular*, if it is closed, possesses a nonempty interior and is *pointed* — does not contain lines, or, which is the same, is such that $a \in \mathbf{K}$, $-a \in \mathbf{K}$ implies that $a = 0$.

Dual cone. If $\mathbf{K}$ is a cone in a Euclidean space $(E, \langle\cdot,\cdot\rangle_E)$, then the set

$$\mathbf{K}^* = \{e \in E : \langle e, h\rangle_E \geq 0 \, \forall h \in \mathbf{K}\}$$

also is a cone called the cone *dual* to $\mathbf{K}$. The dual cone always is closed. The cone dual to dual is the closure of the original cone: $(\mathbf{K}^*)^* = \mathrm{cl}\, \mathbf{K}$; in particular, $(\mathbf{K}^*)^* = \mathbf{K}$ for every closed cone $\mathbf{K}$. The cone $\mathbf{K}^*$ possesses a nonempty interior if and only if $\mathbf{K}$ is pointed, and $\mathbf{K}^*$ is pointed if and only if $\mathbf{K}$ possesses a nonempty interior; in particular, $\mathbf{K}$ is regular if and only if $\mathbf{K}^*$ is so.

Example: Nonnegative ray and nonnegative orthants. The simplest one-dimensional cone is the nonnegative ray $\mathbb{R}_+ = \{t \geq 0\}$ on the real line $\mathbb{R}^1$. The simplest cone in $\mathbb{R}^n$ is the *nonnegative orthant* $\mathbb{R}^n_+ = \{x \in \mathbb{R}^n : x_i \geq 0, 1 \leq i \leq n\}$. This cone is regular and self-dual: $(\mathbb{R}^n_+)^* = \mathbb{R}^n_+$.

Example: Lorentz cone $\mathbf{L}^n$. The cone $\mathbf{L}^n$ “lives” in $\mathbb{R}^n$ and is comprised of all vectors $x = [x_1; ...; x_n] \in \mathbb{R}^n$ such that $x_n \geq \sqrt{\sum_{j=1}^{n-1} x_j^2}$; same as $\mathbb{R}^n_+$, the Lorentz cone is regular and self-dual.

By definition, $\mathbf{L}^1 = \mathbb{R}_+$ is the nonnegative orthant; this is in full accordance with the "general" definition of a Lorentz cone combined with the standard convention "a sum over an empty set of indices is 0."

Example: Semidefinite cone $\mathbf{S}^n_+$. The cone $\mathbf{S}^n_+$ "lives" in the Euclidean space $\mathbf{S}^n$ of $n \times n$ symmetric matrices equipped with the Frobenius inner product. The cone is comprised of all $n \times n$ symmetric *positive semidefinite* matrices A, i.e., matrices $A \in \mathbf{S}^n$ such that $x^T Ax \geq 0$ for all $x \in \mathbb{R}^n$, or, equivalently, such that all eigenvalues of A are nonnegative. Same as $\mathbb{R}^n_+$ and $\mathbf{L}^n$, the cone $\mathbf{S}^n_+$ is regular and self-dual.

Example: Hermitian semidefinite cone $\mathbf{H}^n_+$. This cone "lives" in the space $\mathbf{H}^n$ of $n \times n$ Hermitian matrices and is comprised of all positive semidefinite Hermitian $n \times n$ matrices; it is regular and self-dual.

A.2.2 Conic Problems and Conic Duality

Conic problem

A *conic problem* is an optimization problem of the form

$$\mathrm{Opt}(P) = \min_x \left\{ \langle c, x \rangle_E : \begin{array}{l} A_i x - b_i \in \mathbf{K}_i,\ i = 1, ..., m, \\ Ax = b \end{array} \right\} \tag{P}$$

where

- $(E, \langle \cdot, \cdot \rangle_E)$ is a Euclidean space of *decision vectors* x and $c \in E$ is the *objective*;
- A_i, $1 \leq i \leq m$, are linear maps from E into Euclidean spaces $(F_i, \langle \cdot, \cdot \rangle_{F_i})$, $b_i \in F_i$ and $\mathbf{K}_i \subset F_i$ are regular cones;
- A is a linear mapping from E into a Euclidean space $(F, \langle \cdot, \cdot \rangle_F)$ and $b \in F$.

Examples: Linear, Conic Quadratic and Semidefinite Optimization. We will be especially interested in the three generic conic problems as follows:

- *Linear Optimization*, or *Linear Programming:* this is the family of all conic problems associated with nonnegative orthants $\mathbb{R}^m_+$, that is, the family of all usual LPs $\min_x \{c^T x : Ax - b \geq 0\}$;
- *Conic Quadratic Optimization*, or *Conic Quadratic Programming*, or *Second Order Cone Programming:* this is the family of all conic problems associated with the cones that are *finite direct products* of Lorentz cones, that is, the conic programs of the form

  $$\min_x \left\{ c^T x : [A_1; ...; A_m]x - [b_1; ...; b_m] \in \mathbf{L}^{k_1} \times ... \times \mathbf{L}^{k_m} \right\}$$

  where A_i are $k_i \times \dim x$ matrices and $b_i \in \mathbb{R}^{k_i}$. The "Mathematical Programming" form of such a program is

  $$\min_x \left\{ c^T x : \|\bar{A}_i x - \bar{b}_i\|_2 \leq \alpha_i^T x - \beta_i,\ 1 \leq i \leq m \right\},$$

  where $A_i = [\bar{A}_i; \alpha_i^T]$ and $b_i = [\bar{b}_i; \beta_i]$, so that α_i is the last row of A_i, and β_i is the last entry of b_i;

- *Semidefinite Optimization*, or *Semidefinite Programming:* this is the family of all conic problems associated with the cones that are *finite direct products* of Semidefinite cones, that is, the conic programs of the form

$$\min_x \left\{ c^T x : A_i^0 + \sum_{j=1}^{\dim x} x_j A_i^j \succeq 0,\ 1 \le i \le m \right\},$$

where A_i^j are symmetric matrices of appropriate sizes.

A.2.3 Conic Duality

Conic duality — derivation

The origin of conic duality is the desire to find a systematic way to bound from below the optimal value in a conic problem (P). This way is based on *linear aggregation* of the constraints of (P), namely, as follows. Let $y_i \in \mathbf{K}_i^*$ and $z \in F$. By the definition of the dual cone, for every x feasible for (P) we have

$$\langle A_i^* y_i, x\rangle_E - \langle y_i, b_i\rangle_{F_i} \equiv \langle y_i, A x_i - b_i\rangle_{F_i} \ge 0,\ 1 \le i \le m,$$

and of course

$$\langle A^* z, x\rangle_E - \langle z, b\rangle_F = \langle z, Ax - b\rangle_F = 0.$$

Summing up the resulting inequalities, we get

$$\langle A^* z + \sum_i A_i^* y_i, x\rangle_E \ge \langle z, b\rangle_F + \sum_i \langle y_i, b_i\rangle_{F_i}. \tag{C}$$

By its origin, this scalar linear inequality on x is a consequence of the constraints of (P), that is, it is valid for all feasible solutions x to (P). It may happen that the left hand side in this inequality is, identically in $x \in E$, equal to the objective $\langle c, x\rangle_E$; this happens if and only if

$$A^* z + \sum_i A_i^* y_i = c.$$

Whenever it is the case, the right hand side of (C) is a valid lower bound on the optimal value in (P). The dual problem is nothing but the problem

$$\mathrm{Opt}(D) = \max_{z,\{y_i\}} \left\{ \langle z, b\rangle_F + \sum_i \langle y_i, b_i\rangle_{F_i} : \begin{array}{l} y_i \in \mathbf{K}_i^*,\ 1 \le i \le m, \\ A^* z + \sum_i A_i^* y_i = c \end{array} \right\} \tag{D}$$

of maximizing this lower bound.

By the origin of the dual problem, we have

Weak Duality: *One has* $\mathrm{Opt}(D) \le \mathrm{Opt}(P)$.

We see that (D) is a conic problem. A nice and important fact is that *conic duality is symmetric.*

Symmetry of Duality: *The conic dual to* (D) *is (equivalent to)* (P).

Proof. In order to apply to (D) the outlined recipe for building the conic dual, we should rewrite (D) as a *minimization* problem

$$-\mathrm{Opt}(D) = \min_{z,\{y_i\}} \left\{ \langle z, -b\rangle_F + \sum_i \langle y_i, -b_i\rangle_{F_i} : \begin{array}{l} y_i \in \mathbf{K}_i^*, 1 \le i \le m \\ A^*z + \sum_i A_i^* y_i = c \end{array} \right\}; \qquad (D')$$

the corresponding space of decision vectors is the direct product $F \times F_1 \times ... \times F_m$ of Euclidean spaces equipped with the inner product

$$\langle [z; y_1, ..., y_m], [z'; y_1', ..., y_m'] \rangle = \langle z, z'\rangle_F + \sum_i \langle y_i, y_i'\rangle_{F_i}.$$

The above "duality recipe" as applied to (D') reads as follows: pick weights $\eta_i \in (\mathbf{K}_i^*)^* = \mathbf{K}_i$ and $\zeta \in E$, so that the scalar inequality

$$\underbrace{\langle \zeta, A^*z + \sum_i A_i^* y_i\rangle_E + \sum_i \langle \eta_i, y_i\rangle_{F_i}}_{=\langle A\zeta, z\rangle_F + \sum_i \langle A_i\zeta + \eta_i, y_i\rangle_{F_i}} \ge \langle \zeta, c\rangle_E \qquad (C')$$

in variables z, $\{y_i\}$ is a consequence of the constraints of (D'), and impose on the "aggregation weights" $\zeta, \{\eta_i \in \mathbf{K}_i\}$ an additional restriction that the left hand side in this inequality is, identically in $z, \{y_i\}$, equal to the objective of (D'), that is, the restriction that

$$A\zeta = -b,\ A_i\zeta + \eta_i = -b_i,\ 1 \le i \le m,$$

and maximize under this restriction the right hand side in (C'), thus arriving at the problem

$$\max_{\zeta,\{\eta_i\}} \left\{ \langle c, \zeta\rangle_E : \begin{array}{l} \mathbf{K}_i \ni \eta_i = A_i[-\zeta] - b_i, 1 \le i \le m \\ A[-\zeta] = b \end{array} \right\}.$$

Substituting $x = -\zeta$, the resulting problem, after eliminating η_i variables, is nothing but

$$\max_x \left\{ -\langle c, x\rangle_E : \begin{array}{l} A_i x - b_i \in \mathbf{K}_i,\ 1 \le i \le m \\ Ax = b \end{array} \right\},$$

which is equivalent to (P). □

Conic Duality Theorem

A conic program (P) is called *strictly feasible*, if it admits a feasible solution $\bar{x}$ such that $A_i\bar{x} = -b_i \in \mathrm{int}K_i$, $i = 1, ..., m$.

Conic Duality Theorem is the following statement resembling very much the standard Linear Programming Duality Theorem:

Theorem A.1 [Conic Duality Theorem] *Consider a primal-dual pair of conic problems* (P), (D). *Then*

(i) [Weak Duality] *One has* $\mathrm{Opt}(D) \le \mathrm{Opt}(P)$.

(ii) [Symmetry] *The duality is symmetric:* (D) *is a conic problem, and the problem dual to* (D) *is (equivalent to)* (P).

(iii) [Strong Duality] *If one of the problems* (P), (D) *is strictly feasible and bounded, then the other problem is solvable, and* $\mathrm{Opt}(P) = \mathrm{Opt}(D)$.

If both the problems are strictly feasible, then both are solvable with equal optimal values.

Proof. We have already verified Weak Duality and Symmetry. Let us prove the first claim in Strong Duality. By Symmetry, we can restrict ourselves to the case when the strictly feasible and bounded problem is (P).

Consider the following two sets in the Euclidean space $G = \mathbb{R} \times F \times F_1 \times ... \times F_m$:

$$\begin{array}{rcl} T &=& \{[t; z; y_1; ...; y_m] : \exists x : t = \langle c, x\rangle_E; y_i = A_i x - b_i, 1 \le i \le m; \\ && z = Ax - b\}, \\ S &=& \{[t; z; y_1; ...; y_m] : t < \mathrm{Opt}(P), y_1 \in \mathbf{K}_1, ..., y_m \in \mathbf{K}_m, z = 0\}. \end{array}$$

The sets T and S clearly are convex and nonempty; observe that they do not intersect. Indeed, assuming that $[t; z; y_1; ...; y_m] \in S \cap T$, we should have $t < \mathrm{Opt}(P)$, and $y_i \in \mathbf{K}_i$, $z = 0$ (since the point is in S), and at the same time for certain $x \in E$ we should have $t = \langle c, x\rangle_E$ and $A_i x - b_i = y_i \in \mathbf{K}_i$, $Ax - b = z = 0$, meaning that there exists a feasible solution to (P) with the value of the objective $< \mathrm{Opt}(P)$, which is impossible. Since the convex and nonempty sets S and T do not intersect, they can be separated by a linear form: there exists $[\tau; \zeta; \eta_1; ...; \eta_m] \in G = \mathbb{R} \times F \times F_1 \times ... \times F_m$ such that

$$\begin{array}{ll} (a) & \sup\limits_{[t;z;y_1;...;y_m]\in S} \langle [\tau; \zeta; \eta_1; ...; \eta_m], [t; z; y_1; ...; y_m]\rangle_G \\ & \le \inf\limits_{[t;z;y_1;...;y_m]\in T} \langle [\tau; \zeta; \eta_1; ...; \eta_m], [t; z; y_1; ...; y_m]\rangle_G, \\ (b) & \inf\limits_{[t;z;y_1;...;y_m]\in S} \langle [\tau; \zeta; \eta_1; ...; \eta_m], [t; z; y_1; ...; y_m]\rangle_G \\ & < \sup\limits_{[t;z;y_1;...;y_m]\in T} \langle [\tau; \zeta; \eta_1; ...; \eta_m], [t; z; y_1; ...; y_m]\rangle_G, \end{array}$$

or, which is the same,

$$\begin{array}{ll} (a) & \sup\limits_{t<\mathrm{Opt}(P), y_i\in\mathbf{K}_i} [\tau t + \sum_i \langle \eta_i, y_i\rangle_{F_i}] \\ & \le \inf\limits_{x\in E} [\tau\langle c, x\rangle_E + \langle \zeta, Ax - b\rangle_F + \sum_i \langle \eta_i, A_i x - b_i\rangle_{F_i}], \\ (b) & \inf\limits_{t<\mathrm{Opt}(P), y_i\in\mathbf{K}_i} [\tau t + \sum_i \langle \eta_i, y_i\rangle_{F_i}] \\ & < \sup\limits_{x\in E} [\tau\langle c, x\rangle + \langle \zeta, Ax - b\rangle_F + \sum_i \langle \eta_i, Ax - b_i\rangle_{F_i}]. \end{array} \tag{A.2.1}$$

Since the left hand side in (A.2.1.a) is finite, we have

$$\tau \ge 0,\ -\eta_i \in \mathbf{K}_i^*, 1 \le i \le m, \tag{A.2.2}$$

whence the left hand side in (A.2.1.a) is equal to $\tau\mathrm{Opt}(P)$. Since the right hand side in (A.2.1.a) is finite and $\tau \ge 0$, we have

$$A^*\zeta + \sum_i A_i^* \eta_i + \tau c = 0 \tag{A.2.3}$$

and the right hand side in (a) is $\langle -\zeta, b\rangle_F - \sum_i \langle \eta_i, b_i\rangle_{F_i}$, so that (A.2.1.$a$) reads

$$\tau\mathrm{Opt}(P) \le \langle -\zeta, b\rangle_F - \sum_i \langle \eta_i, b_i\rangle_{F_i}. \tag{A.2.4}$$

We claim that $\tau > 0$. Believing in our claim, let us extract from it Strong Duality. Indeed, setting $y_i = -\eta_i/\tau$, $z = -\zeta/\tau$, (A.2.2), (A.2.3) say that $z, \{y_i\}$ is a feasible solution for (D), and by (A.2.4) the value of the dual objective at this dual feasible solution is $\ge \mathrm{Opt}(P)$. By Weak

Duality, this value cannot be larger than Opt(P), and we conclude that our solution to the dual is in fact an optimal one, and that Opt(P) = Opt(D), as claimed.

It remains to prove that $\tau > 0$. Assume this is not the case; then $\tau = 0$ by (A.2.2). Now let $\bar{x}$ be a strictly feasible solution to (P). Taking inner product of both sides in (A.2.3) with $\bar{x}$, we have

$$\langle \zeta, A\bar{x}\rangle_F + \sum_i \langle \eta_i, A_i\bar{x}\rangle_{F_i} = 0,$$

while (A.2.4) reads

$$-\langle \zeta, b\rangle_F - \sum_i \langle \eta_i, b_i\rangle_{F_i} \geq 0.$$

Summing up the resulting inequalities and taking into account that $\bar{x}$ is feasible for (P), we get

$$\sum_i \langle \eta_i, A_i\bar{x} - b_i\rangle \geq 0.$$

Since $A_i\bar{x} - b_i \in \mathrm{int}\mathbf{K}_i$ and $\eta_i \in -\mathbf{K}_i^*$, the inner products in the left hand side of the latter inequality are nonpositive, and i-th of them is zero if and only if $\eta_i = 0$; thus, the inequality says that $\eta_i = 0$ for all i. Adding this observation to $\tau = 0$ and looking at (A.2.3), we see that $A^*\zeta = 0$, whence $\langle \zeta, Ax\rangle_F = 0$ for all x and, in particular, $\langle \zeta, b\rangle_F = 0$ due to $b = A\bar{x}$. The bottom line is that $\langle \zeta, Ax - b\rangle_F = 0$ for all x. Now let us look at (A.2.1.b). Since $\tau = 0$, $\eta_i = 0$ for all i and $\langle \zeta, Ax - b\rangle_F = 0$ for all x, both sides in this inequality are equal to 0, which is impossible. We arrive at a desired contradiction.

We have proved the first claim in Strong Duality. The second claim there is immediate: if both (P), (D) are strictly feasible, then both problems are bounded as well by Weak Duality, and thus are solvable with equal optimal values by the already proved part of Strong Duality. □

Optimality conditions in Conic Programming

Optimality conditions in Conic Programming are given by the following statement:

Theorem A.2 *Consider a primal-dual pair (P), (D) of conic problems, and let both problems be strictly feasible. A pair $(x, \xi \equiv [z; y_1; ...; y_m])$ of feasible solutions to (P) and (D) is comprised of optimal solutions to the respective problems if and only if*

(i) [Zero duality gap] *One has*

$$\begin{array}{rcl}\mathrm{DualityGap}(x;\xi) & := & \langle c, x\rangle_E - [\langle z, b\rangle_F + \sum_i \langle b_i, y_i\rangle_{F_i}] \\ & = & 0,\end{array}$$

same as if and only if

(ii) [Complementary slackness]

$$\forall i : \langle y_i, A_i x_i - b_i\rangle_{F_i} = 0.$$

Proof. By Conic Duality Theorem, we are in the situation when Opt(P) = Opt(D). Therefore

$$\mathrm{DualityGap}(x;\xi) = \underbrace{[\langle c, x\rangle_E - \mathrm{Opt}(P)]}_{a} + \underbrace{\left[\mathrm{Opt}(D) - \left[\langle z, b\rangle_F + \sum_i \langle b_i, y_i\rangle_{F_i}\right]\right]}_{b}$$

Since x and ξ are feasible for the respective problems, the duality gap is nonnegative and it can vanish if and only if $a = b = 0$, that is, if and only if x and ξ are optimal solutions to the respective problems, as claimed in (i). To prove (ii), note that since x is feasible, we have

$$Ax = b,\ A_i x - b_i \in \mathbf{K}_i,\ c = A^* z + \sum_i A_i^* y_i, y_i \in \mathbf{K}_i^*,$$

whence

$$\begin{array}{l} \mathrm{DualityGap}(x;\xi) = \langle c, x\rangle_E - [\langle z, b\rangle_F + \sum_i \langle b_i, y_i\rangle_{F_i}] \\ = \langle A^* z + \sum_i A_i^* y_i, x\rangle_E - [\langle z, b\rangle_F + \sum_i \langle b_i, y_i\rangle_{F_i}] \\ = \underbrace{\langle z, Ax - b\rangle_F}_{=0} + \sum_i \underbrace{\langle y_i, A_i x - b_i\rangle_{F_i}}_{\geq 0}, \end{array}$$

where the nonnegativity of the terms in the last $\sum_i$ follows from $y_i \in \mathbf{K}_i^*, A_i x_i - b_i \in \mathbf{K}_i$. We see that the duality gap, as evaluated at a pair of primal-dual feasible solutions, vanishes if and only if the complementary slackness holds true, and thus (ii) is readily given by (i). $\square$

A.2.4 Conic Representations of Sets and Functions

Conic representations of sets

When asked whether the optimization programs

$$\min_y \sum_{i=1}^m |a_i^T y - b_i| \tag{A.2.5}$$

and

$$\min_y \max_{1\leq i\leq m} |a_i^T y - b_i| \tag{A.2.6}$$

are Linear Optimization programs, the answer definitely will be ”yes”, in spite of the fact that an LO program is defined as

$$\min_x \left\{c^T x : Ax \geq b, Px = p\right\} \tag{A.2.7}$$

and neither (A.2.5), nor (A.2.6) are in this form. What the “yes” answer actually means, is that both (A.2.5) and (A.2.6) can be straightforwardly *reduced to*, or, which is the same, *represented by* LO programs, e.g., the LO program

$$\min_{y,u} \left\{\sum_{i=1}^m u_i : -u_i \leq a_i^T y - b_i \leq u_i,\ 1 \leq i \leq m\right\} \tag{A.2.8}$$

in the case of (A.2.5), and the LO program

$$\min_{y,t} \left\{t : -t \leq a_i^T y - b_i \leq t,\ 1 \leq i \leq m\right\} \tag{A.2.9}$$

in the case of (A.2.6).

An “in-depth” explanation of what actually takes place in these and similar examples is as follows.

1. The "initial form" of a typical Mathematical Programming problem is $\min_{v\in V} f(v)$, where $f(v) : V \to \mathbb{R}$ is the objective, and $V \subset \mathbb{R}^n$ is the feasible set of the problem. It is technically convenient to assume that the objective is "as simple as possible" — just linear: $f(v) = e^T v$; this assumption does not restrict generality, since we can always pass from the original problem, given in the form $\min_{v\in V} \phi(v)$, to the equivalent problem

$$\min_{y=[v;s]} \left\{ c^T y \equiv s : y \in Y = \{[v; s] : v \in V, s \geq \phi(v)\} \right\}.$$

Thus, from now on we assume w.l.o.g. that the original problem is

$$\min_y \left\{ d^T y : y \in Y \right\}. \tag{A.2.10}$$

2. All we need in order to reduce (A.2.10) to an LO program is what is called a *polyhedral representation* of Y, that is, a representation of the form

$$U = \{y \in \mathbb{R}^n : \exists u : Ay + Bu - b \in \mathbb{R}^N_+\}.$$

Indeed, given such a representation, we can reformulate (A.2.10) as the LO program

$$\min_{x=[y;u]} \left\{ c^T x := d^T y : \mathcal{A}(x) := Ay + Bu - b \geq 0 \right\}.$$

For example, passing from (A.2.5) to (A.2.8), we first rewrite the original problem as

$$\min_{t,y} \left\{ t : \sum_i |a_i^T y - b_i| \leq t \right\}$$

and then point out a polyhedral representation

$$\begin{array}{l} \{[y;t] : \sum_i |a_i^T y - b_i| \leq t\} \\ = \{[y;t] : \exists u : \underbrace{\left\{ \begin{array}{l} u_i - a_i^T y + b_i \geq 0, \\ u_i + a_i^T y - b_i \geq 0, \\ t - \sum_i u_i \geq 0 \end{array} \right.}_{A[y;t]+Bu-b\geq 0} \} \end{array}$$

of the feasible set of the latter problem, thus ending up with reformulating the problem of interest as an LO program in variables y, t, u. The course of actions for (A.2.6) is completely similar, up to the fact that after "linearizing the objective" we get the optimization problem

$$\min_{y,t} \left\{ t : -t \leq a_i^T y - b_i \leq t, \, 1 \leq i \leq m \right\}$$

where the feasible set is polyhedral "as it is" (i.e., with polyhedral representation not requiring u-variables).

The notion of polyhedral representation naturally extends to conic problems, specifically, as follows. Let $\mathcal{K}$ be a family of regular cones, every one "living" in its own Euclidean space. A set $Y \subset \mathbb{R}^n$ is called *$\mathcal{K}$-representable*, if it can be represented in the form

$$Y = \{y \in \mathbb{R}^n : \exists u \in \mathbb{R}^m : Ay + Bu - b \in \mathbf{K}\}, \tag{A.2.11}$$

where $\mathbf{K} \in \mathcal{K}$ and A, B, b are matrices and vectors of appropriate dimensions. A representation of Y of the form (A.2.11), (i.e., the corresponding collection $A, B, b, \mathbf{K}$), is called a *$\mathcal{K}$-representation* ($\mathcal{K}$-r. for short) of Y.

Geometrically, a $\mathcal{K}$-r. of Y is the representation of Y as the *projection* on the space of y variables of the set $Y_+ = \{[y;u] : Ax + Bu - b \in \mathbf{K}\}$, which, in turn, is given as the inverse image of a cone $\mathbf{K} \in \mathcal{K}$ under the affine mapping $[y;u] \mapsto Ay + Bu - b$.

The role of the notion of a conic representation stems from the fact that given a $\mathcal{K}$-r. of the feasible domain Y of (A.2.10), we can immediately rewrite this optimization program as a conic program involving a cone from the family $\mathcal{K}$, specifically, as the program

$$\min_{x=[y;u]} \left\{ c^T x := d^T y : \mathcal{A}(x) := Ay + Bu - b \in \mathbf{K} \right\}. \tag{A.2.12}$$

In particular,

- When $\mathcal{K} = \mathcal{LO}$ is the family of all nonnegative orthants (or, which is the same, the family of all finite direct products of nonnegative rays), a $\mathcal{K}$-representation of Y allows one to rewrite (A.2.10) as a Linear program;
- When $\mathcal{K} = \mathcal{CQO}$ is the family of all finite direct products of Lorentz cones, a $\mathcal{K}$-representation of Y allows one to rewrite (A.2.10) as a Conic Quadratic program;
- When $\mathcal{K} = \mathcal{SDO}$ is the family of all finite direct products of positive semidefinite cones, a $\mathcal{K}$-representation of Y allows one to rewrite (A.2.10) as a Semidefinite program.

Note that a $\mathcal{K}$-representable set is always convex.

Elementary calculus of $\mathcal{K}$-representations

It turns out that when the family of cones $\mathcal{K}$ is "rich enough," $\mathcal{K}$-representations admit a kind of simple "calculus" that allows to convert $\mathcal{K}$-r.'s of operands participating in a standard convexity-preserving operation, like taking intersection, into a $\mathcal{K}$-r. of the result of this operation. "Richness" here means that $\mathcal{K}$

- contains a nonnegative ray $\mathbb{R}_+$;
- is closed w.r.t. taking finite direct products: whenever $\mathbf{K}_i \in \mathcal{K}$, $1 \le i \le m < \infty$, one has $\mathbf{K}_1 \times ... \times \mathbf{K}_m \in \mathcal{K}$;
- is closed w.r.t. passing from a cone to its dual: whenever $\mathbf{K} \in \mathcal{K}$, one has $\mathbf{K}^* \in \mathcal{K}$.

In particular, every one of the three aforementioned families of cones $\mathcal{LO}$, $\mathcal{CQO}$, $\mathcal{SDO}$ is rich.

We present here the most basic and most frequently used "calculus rules" (for more rules and for instructive examples of $\mathcal{LO}$-, $\mathcal{CQO}$-, and $\mathcal{SDO}$-representable sets, see [9]). Let $\mathcal{K}$ be a rich family of cones. Then

1. [taking finite intersections] If the sets $Y_i \subset \mathbb{R}^n$ are $\mathcal{K}$-representable, $1 \le i \le m$, then so is their intersection $Y = \bigcap_{i=1}^{m} Y_i$.
   Indeed, if $Y_i = \{y \in \mathbb{R}^n : \exists u_i : A_i x + B_i u - b_i \in \mathbf{K}_i$ with $\mathbf{K}_i \in \mathcal{K}$, then

   $$\begin{array}{rl} Y = & \{y \in \mathbb{R}^n : \exists u = [u_1; ...; u_m] : \\ & [A_1; ...; A_m]y + \mathrm{Diag}\{B_1, ..., B_m\}[u_1; ...; u_m] - [b_1; ...; b_m] \\ & \in \mathbf{K} := \mathbf{K}_1 \times ... \times \mathbf{K}_m\}, \end{array}$$

   and $\mathbf{K} \in \mathcal{K}$, since $\mathcal{K}$ is closed w.r.t. taking finite direct products.

2. [taking finite direct products] If the sets $Y_i \subset \mathbb{R}^{n_i}$ are $\mathcal{K}$-representable, $1 \leq i \leq m$, then so is their direct product $Y = Y_1 \times ... \times Y_m$.
   Indeed, if $Y_i = \{y \in \mathbb{R}^n : \exists u_i : A_i x + B_i u - b_i \in \mathbf{K}_i$ with $\mathbf{K}_i \in \mathcal{K}$, then

$$\begin{array}{rl} Y = & \{y = [y_1; ...; y_m] \in \mathbb{R}^{n_1 + ... + n_m} : \exists u = [u_1; ...; u_m] : \\ & \quad \mathrm{Diag}\{A_1, ..., A_m\} y + \mathrm{Diag}\{B_1, ..., B_m\}[u_1; ...; u_m] - [b_1; ...; b_m] \\ & \quad \in \mathbf{K} := \mathbf{K}_1 \times ... \times \mathbf{K}_m\}, \end{array}$$

   and, as above, $\mathbf{K} \in \mathcal{K}$.

3. [taking inverse affine images] Let $Y \subset \mathbb{R}^n$ be $\mathcal{K}$-representable, let $z \mapsto Pz + p : \mathbb{R}^N \to \mathbb{R}^n$ be an affine mapping. Then the inverse affine image $Z = \{z : Pz + p \in Y\}$ of Y under this mapping is $\mathcal{K}$-representable.
   Indeed, if $Y = \{y \in \mathbb{R}^n : \exists u : Ay + Bu - b \in \mathbf{K}\}$ with $\mathbf{K} \in \mathcal{K}$, then

$$Z = \{z \in \mathbb{R}^N : \exists u : \underbrace{A[Pz + p] + Bu - b}_{\equiv \widetilde{A}z + Bu - \widetilde{b}} \in \mathbf{K}\}.$$

4. [taking affine images] If a set $Y \subset \mathbb{R}^n$ is $\mathcal{K}$-representable and $y \mapsto z = Py + p : \mathbb{R}^n \to \mathbb{R}^m$ is an affine mapping, then the image $Z = \{z = Py + p : y \in Y\}$ of Y under the mapping is $\mathcal{K}$-representable.
   Indeed, if $Y = \{y \in \mathbb{R}^n : \exists u : Au + Bu - b \in \mathbf{K}\}$, then

$$Z = \{z \in \mathbb{R}^m : \exists [y; u] : \underbrace{\left[\begin{array}{c} Py + p - z \\ -Py - p + z \\ Ay + Bu - b \end{array}\right]}_{\equiv \widetilde{A}z + \widetilde{B}[y;u] - \widetilde{b}} \in \mathbf{K}_+ := \mathbb{R}^m_+ \times \mathbb{R}^m_+ \times \mathbf{K}\},$$

   and the cone $\mathbf{K}_+$ belongs to $\mathcal{K}$ as the direct product of several nonnegative rays (every one of them belongs to $\mathcal{K}$) and the cone $\mathbf{K} \in \mathcal{K}$.

Note that the above "calculus rules" are "completely algorithmic" — a $\mathcal{K}$-r. of the result of an operation is readily given by $\mathcal{K}$-r.'s of the operands.

Conic representation of functions

By definition, the *epigraph* of a function $f(y) : \mathbb{R}^n \to \mathbb{R} \cup \{+\infty\}$ is the set

$$\mathrm{Epi}\{f\} = \{[y; t] \in \mathbb{R}^n \times \mathbb{R} : t \geq f(y)\}.$$

Note that a function is convex if and only if its epigraph is so.

Let $\mathcal{K}$ be a family of regular cones. A function f is called $\mathcal{K}$*-representable*, if its epigraph is so:

$$\mathrm{Epi}\{f\} := \{[y, t] : \exists u : Ay + ta + Bu - b \in \mathbf{K}\} \tag{A.2.13}$$

with $\mathbf{K} \in \mathcal{K}$. A $\mathcal{K}$-representation ($\mathcal{K}$-r. for short) of a function is, by definition, a $\mathcal{K}$-r. of its epigraph. Since $\mathcal{K}$-representable sets always are convex, so are $\mathcal{K}$-representable functions.

Examples of $\mathcal{K}$-r.'s of functions:

- the function $f(y) = |y| : \mathbb{R} \to \mathbb{R}$ is $\mathcal{LO}$-representable:

$$\{[y; t] : t \geq |y|\} = \{[y; t] : A[y; t] := [t - y; t + y] \in \mathbb{R}^2_+\};$$

- the function $f(y) = \|y\|_2 : \mathbb{R}^n \to \mathbb{R}$ is $\mathcal{CQO}$-representable:

$$\{[y;t] \in \mathbb{R}^{n+1} : t \geq \|y\|_2\} = \{[y;t] \in \mathbf{L}^{n+1}\};$$

- the function $f(y) = \lambda_{\max}(y) : \mathbf{S}^n \to \mathbb{R}$ (the maximal eigenvalue of a symmetric matrix y) is $\mathcal{SDO}$-representable:

$$\{[y;t] \in \mathbf{S}^n \times \mathbb{R} : t \geq \lambda_{\max}(y)\} = \{[y;t] : \mathcal{A}[y;t] := tI_n - y \in \mathbf{S}^n_+\}.$$

Observe that a $\mathcal{K}$-r. (A.2.13) of a function f induces $\mathcal{K}$-r.'s of its *level sets* $\{y : f(y) \leq c\}$:

$$\{y : f(y) \leq c\} = \{y : \exists u : Ay + Bu - [b - ca] \in \mathbf{K}\}.$$

This explains the importance of $\mathcal{K}$-representations of functions: usually, the feasible set Y of a convex problem (A.2.10) is given by a system of convex constraints:

$$Y = \{y : f_i(y) \leq 0,\ 1 \leq i \leq m\}.$$

If now all functions f_i are $\mathcal{K}$-representable, then, by the above observation and by the "calculus rule" related to intersections, Y is $\mathcal{K}$-representable as well, and a $\mathcal{K}$-r. of Y is readily given by $\mathcal{K}$-r.'s of f_i.

$\mathcal{K}$-representable functions admit simple calculus, which is similar to the one of $\mathcal{K}$-representable sets, and is equally algorithmic; for details and instructive examples, see [9].

A.3 Efficient Solvability of Convex Programming

The goal of this section is to explain the precise meaning of the informal (and in fact slightly exaggerated) claim,

> *An optimization problem with convex efficiently computable objective and constraints is efficiently solvable.*

that on many different occasions was reiterated in the main body of the book. Our exposition follows the one from [9, chapter 5].

A.3.1 Generic Convex Programs and Efficient Solution Algorithms

In what follows, it is convenient to represent optimization programs as

$$(p): \quad \mathrm{Opt}(p) = \min_x \left\{ p_0(x) : x \in X(p) \subset \mathbb{R}^{n(p)} \right\},$$

where $p_0(\cdot)$ and $X(p)$ are the objective, which we assume to be a real-valued function on $\mathbb{R}^{n(p)}$, and the feasible set of program (p), respectively, and $n(p)$ is the dimension of the decision vector.

A generic optimization problem

A *generic optimization program* $\mathcal{P}$ is a collection of optimization programs (p) ("instances of $\mathcal{P}$") such that every instance of $\mathcal{P}$ is identified by a finite-dimensional *data vector* $\mathrm{data}(p)$; the dimension of this vector is called the *size* $\mathrm{Size}(p)$ of the instance:

$$\mathrm{Size}(p) = \dim \mathrm{data}(p).$$

For example, *Linear Optimization* is a generic optimization problem $\mathcal{LO}$ with instances of the form

$$(p): \quad \min_x \left\{ c_p^T x : x \in X(p) := \{x : A_p x - b_p \geq 0\} \right\} \qquad [A_p : m(p) \times n(p)],$$

where $m(p), n(p), c_p, A_p, b_p$ can be arbitrary. The data of an instance can be identified with the vector

$$\mathrm{data}(p) = [m(p); n(p); c_p; b_p; A_p^1; ...; A_p^{n(p)}],$$

where A_p^i is i-th column in A_p.

Similarly, *Conic Quadratic Optimization* is a generic optimization problem $\mathcal{CQO}$ with instances

$$\begin{array}{c} (p): \quad \min_x \left\{ c_p^T x : x \in X(p) \right\}, \\ X(p) := \{x : \|A_{pi}x - b_{pi}\|_2 \leq e_{pi}^T x - d_{pi}, \ 1 \leq i \leq m(p)\} \end{array} \qquad [A_{pi} : k_i(p) \times n(p)].$$

The data of an instance can be defined as the vector obtained by listing, in a fixed order, the dimensions $m(p)$, $n(p)$, $\{k_i(p)\}_{i=1}^{m(p)}$ and the entries of the reals d_{pi}, vectors c_p, b_{pi}, e_{pi} and the matrices A_{pi}^ℓ.

Finally, *Semidefinite Optimization* is a generic optimization problem $\mathcal{SDO}$ with instances of the form

$$\begin{array}{c} (p): \ \min_x \left\{ c_p^T x : x \in X(p) := \{x : A_p^i(x) \succeq 0, 1 \leq i \leq m(p)\} \right\} \\ A_p^i(x) = A_{pi}^0 + x_1 A_{pi}^1 + ... + x_{n(p)} A_{pi}^{n(p)}, \end{array}$$

where A_{pi}^ℓ are symmetric matrices of size $k_i(p)$. The data of an instance can be defined in the same fashion as in the case of $\mathcal{CQO}$.

Approximate solutions

In order to quantify the quality of a candidate solution of an instance (p) of a generic problem $\mathcal{P}$, we assume that $\mathcal{P}$ is equipped with an *infeasibility measure* $\mathrm{Infeas}_{\mathcal{P}}(p, x)$ — a real-valued nonnegative function of an instance $(p) \in \mathcal{P}$ and a candidate solution $x \in \mathbb{R}^{n(p)}$ to the instance such that $x \in X(p)$ if and only if $\mathrm{Infeas}_{\mathcal{P}}(p, x) = 0$.

Given an infeasibility measure and a tolerance $\epsilon > 0$, we define an ϵ *solution* to an instance $(p) \in \mathcal{P}$ as a point $x_\epsilon \in \mathbb{R}^{n(p)}$ such that

$$p_0(x_\epsilon) - \mathrm{Opt}(p) \leq \epsilon \ \& \ \mathrm{Infeas}_{\mathcal{P}}(p, x_\epsilon) \leq \epsilon.$$

For example, a natural infeasibility measure for a generic optimization problem $\mathcal{P}$ with instances of the form

$$(p): \quad \min_x \{p_0(x) : x \in X(p) := \{x : p_i(x) \leq 0, \ 1 \leq i \leq m(p)\}\} \tag{A.3.1}$$

is

$$\mathrm{Infeas}_{\mathcal{P}}(p, x) = \max\left[0, p_1(x), p_2(x), ..., p_{m(p)}(x)\right]; \tag{A.3.2}$$

this recipe, in particular, can be applied to the generic problems $\mathcal{LO}$ and $\mathcal{CQO}$. A natural infeasibility measure for $\mathcal{SDO}$ is

$$\mathrm{Infeas}_{\mathcal{SDO}}(p, x) = \min\left\{ t \geq 0 : A_p^i(x) + tI_{k_i(p)} \succeq 0, \ 1 \leq i \leq m(p) \right\}.$$

Convex generic optimization problems

A generic problem $\mathcal{P}$ is called *convex*, if for every instance (p) of the problem, $p_0(x)$ and $\mathrm{Infeas}_{\mathcal{P}}(p,x)$ are convex functions of $x \in \mathbb{R}^{n(p)}$. Note that then $X(p) = \{x \in \mathbb{R}^{n(p)} : \mathrm{Infeas}_{\mathcal{P}}(p,x) \le 0\}$ is a convex set for every $(p) \in \mathcal{P}$.

For example, $\mathcal{LO}$, $\mathcal{CQO}$ and $\mathcal{SDO}$ with the just defined infeasibility measures are generic convex programs. The same is true for generic problems with instances (A.3.1) and infeasibility measure (A.3.2), provided that all instances are convex programs, i.e., $p_0(x), p_1(x), ..., p_{m(p)}(x)$ are restricted to be real-valued *convex* functions on $\mathbb{R}^{n(p)}$.

A solution algorithm

A solution algorithm $\mathcal{B}$ for a generic problem $\mathcal{P}$ is a code for the Real Arithmetic Computer — an idealized computer capable to store real numbers and to carry out the operations of Real Arithmetics (the four arithmetic operations, comparisons and computing elementary functions like $\sqrt{\cdot}$, $\exp\{\cdot\}$, $\sin(\cdot)$) with real arguments. Given on input the data vectors $\mathrm{data}(p)$ of an instance $(p) \in \mathcal{P}$ and a tolerance $\epsilon > 0$ and executing on this input the code $\mathcal{B}$, the computer should eventually stop and output

— either a vector $x_\epsilon \in \mathbb{R}^{n(p)}$ that must be an ϵ solution to (p),

— or a correct statement "(p) is infeasible"/"(p) is not below bounded."

The *complexity* of the generic problem $\mathcal{P}$ with respect to a solution algorithm $\mathcal{B}$ is quantified by the function $\mathrm{Compl}_{\mathcal{P}}(p,\epsilon)$; the value of this function at a pair $(p) \in \mathcal{P}$, $\epsilon > 0$ is exactly the number of elementary operations of the Real Arithmetic Computer in the course of executing the code $\mathcal{B}$ on the input $(\mathrm{data}(p), \epsilon)$.

Polynomial time solution algorithms

A solution algorithm for a generic problem $\mathcal{P}$ is called *polynomial time* ("efficient"), if the complexity of solving instances of $\mathcal{P}$ within (an arbitrary) accuracy $\epsilon > 0$ is bounded by a polynomial in the size of the instance and the *number of accuracy digits* $\mathrm{Digits}(p,\epsilon)$ in an ϵ solution:

$$\begin{gathered}\mathrm{Compl}_{\mathcal{P}}(p,\epsilon) \le \chi \left(\mathrm{Size}(p)\mathrm{Digits}(p,\epsilon)\right)^{\chi}, \\ \mathrm{Size}(p) = \dim \mathrm{data}(p),\ \mathrm{Digits}(p,\epsilon) = \ln\left(\frac{\mathrm{Size}(p)+\|\mathrm{data}(p)\|_1+\epsilon^2}{\epsilon}\right);\end{gathered}$$

from now on, χ stands for various "characteristic constants" (not necessarily identical to each other) of the generic problem in question, i.e., for positive quantities depending on $\mathcal{P}$ and independent of $(p) \in \mathcal{P}$ and $\epsilon > 0$. Note also that while the "strange" numerator in the fraction participating in the definition of Digits arises by technical reasons, the number of accuracy digits for small $\epsilon > 0$ becomes independent of this numerator and close to $\ln(1/\epsilon)$.

A generic problem $\mathcal{P}$ is called *polynomially solvable* ("computationally tractable"), if it admits a polynomial time solution algorithm.

A.3.2 Polynomial Solvability of Generic Convex Programming Problems

The main fact about generic convex problems that underlies the remarkable role played by these problems in Optimization is that *under minor non-restrictive technical assumptions, a generic convex problem*, in contrast to typical generic non-convex problems, *is computationally tractable*.

The just mentioned "minor non-restrictive technical assumptions" are those of *polynomial computability*, *polynomial growth*, and *polynomial boundedness of feasible sets*.

Polynomial computability

A generic convex optimization problem $\mathcal{P}$ is called *polynomially computable*, if it can be equipped with two codes, $\mathcal{O}$ and $\mathcal{C}$, for the Real Arithmetic Computer, such that:

• for every instance $(p) \in \mathcal{P}$ and any candidate solution $x \in \mathbb{R}^{n(p)}$ to the instance, executing $\mathcal{O}$ on the input $(\mathrm{data}(p), x)$ takes a polynomial in $\mathrm{Size}(p)$ number of elementary operations and produces a value and a subgradient of the objective $p_0(\cdot)$ at the point x;

• for every instance $(p) \in \mathcal{P}$, any candidate solution $x \in \mathbb{R}^{n(p)}$ to the instance and any $\epsilon > 0$, executing $\mathcal{C}$ on the input $(\mathrm{data}(p), x, \epsilon)$ takes a polynomial in $\mathrm{Size}(p)$ and $\mathrm{Digits}(p, \epsilon)$ number of elementary operations and results

— either in a correct claim that $\mathrm{Infeas}_{\mathcal{P}}(p, x) \leq \epsilon$,

— or in a correct claim that $\mathrm{Infeas}_{\mathcal{P}}(p, x) > \epsilon$ and in computing a linear form $e \in \mathbb{R}^{n(p)}$ that separates x and the set $\{y : \mathrm{Infeas}_{\mathcal{P}}(p, y) \leq \epsilon\}$, so that

$$\forall (y, \mathrm{Infeas}_{\mathcal{P}}(p, y) \leq \epsilon) : e^T y < e^T x.$$

Consider, for example, a generic convex program $\mathcal{P}$ with instances of the form (A.3.1) and the infeasibility measure (A.3.2) and assume that the functions $p_0(\cdot), p_1(\cdot), ..., p_{m(p)}(\cdot)$ are real-valued and convex for all instances of $\mathcal{P}$. Assume, moreover, that the objective and the constraints of instances are efficiently computable, meaning that there exists a code $\mathcal{CO}$ for the Real Arithmetic Computer, which being executed on an input of the form $(\mathrm{data}(p), x \in \mathbb{R}^{n(p)})$ computes in a polynomial in $\mathrm{Size}(p)$ number of elementary operations the values and subgradients of $p_0(\cdot)$, $p_1(\cdot)$,..., $p_{m(p)}(\cdot)$ at x. In this case, $\mathcal{P}$ is polynomially computable. Indeed, the code $\mathcal{O}$ allowing to compute in polynomial time the value and a subgradient of the objective at a given candidate solution is readily given by $\mathcal{CO}$. In order to build $\mathcal{C}$, let us execute $\mathcal{CO}$ on an input $(\mathrm{data}(p), x)$ and compare the quantities $p_i(x)$, $1 \leq i \leq m(p)$, with ϵ. If $p_i(x) \leq \epsilon$, $1 \leq i \leq m(p)$, we output the correct claim that $\mathrm{Infeas}_{\mathcal{P}}(p, x) \leq \epsilon$, otherwise we output a correct claim that $\mathrm{Infeas}_{\mathcal{P}}(p, x) > \epsilon$ and return, as e, a subgradient, taken at x, of a constraint $p_{i(x)}(\cdot)$, where $i(x) \in \{1, 2, ..., m(p)\}$ is such that $p_{i(x)}(x) > \epsilon$.

By the reasons outlined above, the generic problems $\mathcal{LO}$ and $\mathcal{CQO}$ of Linear and Conic Quadratic Optimization are polynomially computable. The same is true for Semidefinite Optimization, see [9, chapter 5].

Polynomial growth

We say that $\mathcal{P}$ is of *polynomial growth*, if for properly chosen $\chi > 0$ one has

$$\begin{array}{l}\forall ((p) \in \mathcal{P}, x \in \mathbb{R}^{n(p)}) : \\ \max\left[|p_0(x)|, \mathrm{Infeas}_{\mathcal{P}}(p, x)\right] \leq \chi \left(\mathrm{Size}(p) + \|\mathrm{data}(p)\|_1\right)^{\chi \mathrm{Size}^{\chi}(p)}.\end{array}$$

For example, the generic problems of Linear, Conic Quadratic and Semidefinite Optimization clearly are with polynomial growth.

Polynomial boundedness of feasible sets

We say that $\mathcal{P}$ is with polynomially bounded feasible sets, if for properly chosen $\chi > 0$ one has

$$\forall ((p) \in \mathcal{P}) : x \in X(p) \Rightarrow \|x\|_\infty \leq \chi \left(\mathrm{Size}(p) + \|\mathrm{data}(p)\|_1\right)^{\chi \mathrm{Size}^{\chi}(p)}.$$

While the generic convex problems $\mathcal{LO}$, $\mathcal{CQO}$, and $\mathcal{SDO}$ are polynomially computable and with polynomial growth, neither one of these problems (same as neither one of other natural generic convex problems) "as it is" possesses polynomially bounded feasible sets. We, however, can enforce the latter property by passing from a generic problem $\mathcal{P}$ to its "bounded version" $\mathcal{P}_{\rm b}$ as follows: the instances of $\mathcal{P}_{\rm b}$ are the instances (p) of $\mathcal{P}$ augmented by bounds on the variables; thus, an instance $(p_+) = (p, R)$ of $\mathcal{P}_{\rm b}$ is of the form

$$(p, R): \quad \min_x \left\{ p_0(x) : x \in X(p, R) = X(p) \cap \{x \in \mathbb{R}^{n(p)} : \|x\|_\infty \leq R\} \right\}$$

where (p) is an instance of $\mathcal{P}$ and $R > 0$. The data of (p, R) is the data of (p) augmented by R, and

$$\text{Infeas}_{\mathcal{P}_{\rm b}}((p, R), x) = \text{Infeas}_{\mathcal{P}}(p, x) + \max[\|x\|_\infty - R, 0].$$

Note that $\mathcal{P}_{\rm b}$ inherits from $\mathcal{P}$ the properties of polynomial computability and/or polynomial growth, if any, and always is with polynomially bounded feasible sets. Note also that R can be really large, like $R = 10^{100}$, which makes the "expressive abilities" of $\mathcal{P}_{\rm b}$, for all practical purposes, as strong as those of $\mathcal{P}$. Finally, we remark that the "bounded versions" of $\mathcal{LO}$, $\mathcal{CQO}$, and $\mathcal{SDO}$ are sub-problems of the original generic problems.

Main result

The main result on computational tractability of Convex Programming is the following:

Theorem A.3 *Let $\mathcal{P}$ be a polynomially computable generic convex program with a polynomial growth that possesses polynomially bounded feasible sets. Then $\mathcal{P}$ is polynomially solvable.*

As a matter of fact, "in real life" the only restrictive assumption in Theorem A.3 is the one of polynomial computability. This is the assumption that is usually violated when speaking about *semi-infinite* convex programs like the RCs of uncertain conic problems

$$\min_x \left\{ c_p^T x : x \in X(p) = \{x \in \mathbb{R}^{n(p)} : A_{p\zeta} x + a_{p\zeta} \in \mathbf{K} \, \forall (\zeta \in \mathcal{Z})\} \right\}.$$

associated with simple *non-polyhedral* cones $\mathbf{K}$. Indeed, when $\mathbf{K}$ is, say, a Lorentz cone, so that

$$X(p) = \{x : \|B_{p\zeta} x + b_{p\zeta}\|_2 \leq c_{p\zeta}^T x + d_{p\zeta} \, \forall (\zeta \in \mathcal{Z})\},$$

to compute the natural infeasibility measure

$$\min \left\{ t \geq 0 : \|B_{p\zeta} x + b_{p\zeta}\|_2 \leq c_{p\zeta}^T x + d_{p\zeta} + t \, \forall (\zeta \in \mathcal{Z}) \right\}$$

at a given candidate solution x means to *maximize* the function $f_x(\zeta) = \|B_{p\zeta} x + b_{p\zeta}\|_2 - c_{p\zeta}^T x - d_{p\zeta}$ over the uncertainty set $\mathcal{Z}$. When the uncertain data are affinely parameterized by ζ, this requires a *maximization of a nonlinear convex function* $f_x(\zeta)$ *over* $\zeta \in \mathcal{Z}$, and this problem can be (and generically is) computationally intractable, even when $\mathcal{Z}$ is a simple convex set. It becomes also clear why the outlined difficulty does not occur in uncertain LO with the data affinely parameterized by ζ: here $f_x(\zeta)$ is an affine function of ζ, and as such can be efficiently maximized over $\mathcal{Z}$, provided the latter set is convex and "not too complicated."

A.3.3 "What is Inside": Efficient Black-Box-Oriented Algorithms in Convex Optimization

Theorem A.3 is a direct consequence of a fact that is instructive in its own right and has to do with "black-box-oriented" Convex Optimization, specifically, with solving an optimization problem

$$\min_{x\in X} f(x), \tag{A.3.3}$$

where

- $X \subset \mathbb{R}^n$ is a solid (a convex compact set with a nonempty interior) known to belong to a given Euclidean ball $E_0 = \{x : \|x\|_2 \leq R\}$ and represented by a *Separation oracle* — a routine that, given on input a point $x \in \mathbb{R}^n$, reports whether $x \in X$, and if it is not the case, returns a vector $e \neq 0$ such that

$$e^T x \geq \max_{y\in X} e^T y;$$

- f is a convex real-valued function on $\mathbb{R}^n$ represented by a *First Order oracle* that, given on input a point $x \in \mathbb{R}^n$, returns the value and a subgradient of f at x.

In addition, we assume that we know in advance an $r > 0$ such that X contains a Euclidean ball of the radius r (the center of this ball can be unknown).

Theorem A.3 is a straightforward consequence of the following important fact:

Theorem A.4 [9, Theorem 5.2.1] *There exists a Real Arithmetic algorithm (the Ellipsoid method) that, as applied to* (A.3.3)*, the required accuracy being $\epsilon > 0$, finds a feasible ϵ solution x_ϵ to the problem (i.e., $x_\epsilon \in X$ and $f(x_\epsilon) - \min_X f \leq \epsilon$) after at most*

$$\begin{array}{c} N(\epsilon) = \mathrm{Ceil}\left(2n^2\left[\ln\left(\frac{R}{r}\right) + \ln\left(\frac{\epsilon + \mathrm{Var}_R(f)}{\epsilon}\right)\right]\right) + 1 \\ \mathrm{Var}_R(f) = \max_{\|x\|_2 \leq R} f(x) - \min_{\|x\|_2 \leq R} f(x) \end{array}$$

steps, with a step reducing to a single call to the Separation and to the First Order oracles accompanied by $O(1)n^2$ additional arithmetic operations to process the answers of the oracles. Here $O(1)$ is an absolute constant.

Recently, the Ellipsoid method was equipped with "on line" accuracy certificates, which yield a slightly strengthened version of the above theorem, namely, as follows:

Theorem A.5 [74] *Consider problem* (A.3.3) *and assume that*

• *$X \in \mathbb{R}^n$ is a solid contained in the centered at the origin Euclidean ball E_0 of a known in advance radius R and given by a Separation oracle that, given on input a point $x \in \mathbb{R}^n$, reports whether $x \in \mathrm{int}X$, and if it is not the case, returns a nonzero e such that $e^T x \geq \max_{y\in X} e^T y$;*

• *$f : \mathrm{int}X \to \mathbb{R}$ is a convex function represented by a First Order oracle that, given on input a point $x \in \mathrm{int}X$, reports the value $f(x)$ and a subgradient $f'(x)$ of f at x. In addition, assume that f is semibounded on X, meaning that $V_X(f) \equiv \sup_{x,y\in \mathrm{int}X} (y-x)^T f'(x) < \infty$.*

There exists an explicit Real Arithmetic algorithm that, given on input a desired accuracy $\epsilon > 0$, terminates with a strictly feasible ϵ-solution x_ϵ to the problem ($x_\epsilon \in \mathrm{int}X$, $f(x_\epsilon) - \inf_{x\in\mathrm{int}X} f(x) \leq \epsilon$) after at most

$$N(\epsilon) = O(1)\left(n^2\left[\ln\left(\frac{nR}{r}\right) + \ln\left(\frac{\epsilon + V_X(f)}{\epsilon}\right)\right]\right)$$

steps, with a step reducing to a single call to the Separation and to the First Order oracles accompanied by $O(1)n^2$ additional arithmetic operations to process the answers of the oracles. Here r is the supremum of the radii of Euclidean balls contained in X, and $O(1)$'s are absolute constants.

The progress, as compared to Theorem A.3, is that now we do not need a priori knowledge of $r > 0$ such that X contains a Euclidean ball of radius r, f is allowed to be undefined outside of $\mathrm{int}X$ and the role of $\mathrm{Var}_R(f)$ (the quantity that now can be $+\infty$) is played by $V_X(f) \leq \sup_{\mathrm{int}X} f - \inf_{\mathrm{int}X} f$.

A.4 Miscellaneous

A.4.1 Matrix Cube Theorems

Matrix Cube Theorem, Complex Case

The "Complex Matrix Cube" problem is as follows:

CMC: *Let m, $p_1, q_1,...,p_L, q_L$ be positive integers, and $A \in \mathbf{H}^m_+$, $L_j \in \mathbb{C}^{p_j\times m}$, $R_j \in \mathbb{C}^{q_j\times m}$ be given matrices, $L_j \neq 0$. Let also a partition $\{1,2,...,L\} = I^{\mathrm{r}}_{\mathrm{s}} \cup I^{\mathrm{c}}_{\mathrm{s}} \cup I^{\mathrm{c}}_{\mathrm{f}}$ of the index set $\{1,...,L\}$ into three non-overlapping sets be given, and let $p_j = q_j$ for $j \in I^{\mathrm{r}}_{\mathrm{s}} \cup I^{\mathrm{c}}_{\mathrm{s}}$. With these data, we associate a parametric family of "matrix boxes"*

$$\mathcal{U}[\rho] \quad = \quad \left\{ A + \rho \sum_{j=1}^{L} [L_j^H \Theta^j R_j + R_j^H [\Theta^j]^H L_j] : \begin{array}{c} \Theta^j \in \mathcal{Z}^j, \\ 1 \leq j \leq L \end{array} \right\} \subset \mathbf{H}^m, \tag{A.4.1}$$

where $\rho \geq 0$ is the parameter and

$$\mathcal{Z}^j = \left\{ \begin{array}{l} \{\Theta^j = \theta I_{p_j} : \theta \in \mathbb{R}, |\theta| \leq 1\}, \quad j \in I^{\mathrm{r}}_{\mathrm{s}} \\ \qquad [\text{“real scalar perturbation blocks”}] \\ \{\Theta^j = \theta I_{p_j} : \theta \in \mathbb{C}, |\theta| \leq 1\}, \quad j \in I^{\mathrm{c}}_{\mathrm{s}} \\ \quad [\text{“complex scalar perturbation blocks”}] \\ \{\Theta^j \in \mathbb{C}^{p_j\times q_j} : \|\Theta^j\|_{2,2} \leq 1\}, \quad j \in I^{\mathrm{c}}_{\mathrm{f}} \\ \qquad [\text{“full complex perturbation blocks”}]\ . \end{array} \right. \tag{A.4.2}$$

Given $\rho \geq 0$, check whether

$$\mathcal{U}[\rho] \subset \mathbf{H}^m_+. \tag{$\mathcal{A}(\rho)$}$$

Remark A.1 *We always assume that $p_j = q_j > 1$ for $j \in I^{\mathrm{c}}_{\mathrm{s}}$. Indeed, one-dimensional complex scalar perturbations can always be regarded as full complex perturbations.*

Our main result is as follows:

Theorem A.6 [The Complex Matrix Cube Theorem [3, section B.4]] *Consider, along with predicate $\mathcal{A}(\rho)$, the predicate*

$$\begin{array}{ll} \exists Y_j \in \mathbf{H}^m,\ j = 1,...,L \ \textit{such that:} & \\ (a) \quad Y_j \succeq L_j^H \Theta^j R_j + R_j^H [\Theta^j]^H L_j \quad \forall (\Theta^j \in \mathcal{Z}^j, 1 \leq j \leq L) & \\ (b) \quad A - \rho \sum\limits_{j=1}^{L} Y_j \succeq 0. & \end{array} \tag{$\mathcal{B}(\rho)$}$$

Then:

(i) *Predicate $\mathcal{B}(\rho)$ is stronger than $\mathcal{A}(\rho)$ — the validity of the former predicate implies the validity of the latter one.*

(ii) *$\mathcal{B}(\rho)$ is computationally tractable — the validity of the predicate is equivalent to the solvability of the system of LMIs*

$$\begin{array}{lrcl} (s.\mathbb{R}) & Y_j \pm \left[L_j^H R_j + R_j^H L_j\right] & \succeq & 0,\ j \in I_{\rm s}^{\rm r}, \\ (s.\mathbb{C}) & \left[\begin{array}{cc} Y_j - V_j & L_j^H R_j \\ R_j^H L_j & V_j \end{array}\right] & \succeq & 0,\ j \in I_{\rm s}^{\rm c}, \\ (f.\mathbb{C}) & \left[\begin{array}{cc} Y_j - \lambda_j L_j^H L_j & R_j^H \\ R_j & \lambda_j I_{p_j} \end{array}\right] & \succeq & 0,\ j \in I_{\rm f}^{\rm c} \\ (*) & A - \rho \sum\limits_{j=1}^{L} Y_j & \succeq & 0. \end{array} \tag{A.4.3}$$

in the matrix variables $Y_j \in \mathbf{H}^m$, $j = 1, ..., k$, $V_j \in \mathbf{H}^m$, $j \in I_{\rm s}^{\rm c}$, and the real variables λ_j, $j \in I_{\rm f}^{\rm c}$.

(iii) *"The gap" between $\mathcal{A}(\rho)$ and $\mathcal{B}(\rho)$ can be bounded solely in terms of the maximal size*

$$p^{\rm s} = \max\left\{p_j : j \in I_{\rm s}^{\rm r} \cup I_{\rm s}^{\rm c}\right\} \tag{A.4.4}$$

of the scalar perturbations (here the maximum over an empty set by definition is 0). Specifically, there exists a universal function $\vartheta_{\mathbb{C}}(\cdot)$ such that

$$\vartheta_{\mathbb{C}}(\nu) \leq 4\pi\sqrt{\nu},\ \nu \geq 1, \tag{A.4.5}$$

and

$$\text{if } \mathcal{B}(\rho) \text{ is not valid, then } \mathcal{A}(\vartheta_{\mathbb{C}}(p^{\rm s})\rho) \text{ is not valid.} \tag{A.4.6}$$

(iv) *Finally, in the case $L = 1$ of single perturbation block $\mathcal{A}(\rho)$ is equivalent to $\mathcal{B}(\rho)$.*

Remark A.2 *From the proof of Theorem A.6 it follows that $\vartheta_{\mathbb{C}}(0) = \frac{4}{\pi}$, $\vartheta_{\mathbb{C}}(1) = 2$. Thus,*

- *when there are no scalar perturbations: $I_{\rm s}^{\rm r} = I_{\rm s}^{\rm c} = \emptyset$, the factor ϑ in the implication*

$$\neg\mathcal{B}(\rho) \Rightarrow \neg\mathcal{A}(\vartheta\rho) \tag{A.4.7}$$

  can be set to $\frac{4}{\pi} = 1.27...$

- *when there are no complex scalar perturbations (cf. Remark A.1) and all real scalar perturbations are non-repeated ($I_{\rm s}^{\rm c} = \emptyset$, $p_j = 1$ for all $j \in I_{\rm s}^{\rm r}$), the factor ϑ in* (A.4.7) *can be set to* 2.

The following simple observation is crucial when applying Theorem A.6.

Remark A.3 *Assume that the data A, $R_1, ..., R_L$ of the Matrix Cube problem are affine in a vector of parameters y, while the data $L_1, ..., L_L$ are independent of y. Then* (A.4.3) *is a system of LMIs in the variables Y_j, V_j, λ_j* and *y.*

Matrix Cube Theorem, Real Case

The Real Matrix Cube problem is as follows:

> RMC: *Let* m, $p_1, q_1, ..., p_L, q_L$ *be positive integers, and* $A \in \mathbf{S}^m$, $L_j \in \mathbb{R}^{p_j \times m}$, $R_j \in \mathbb{R}^{q_j \times m}$ *be given matrices,* $L_j \neq 0$. *Let also a partition* $\{1, 2, ..., L\} = I_{\rm s}^{\rm r} \cup I_{\rm f}^{\rm r}$ *of the index set* $\{1, ..., L\}$ *into two non-overlapping sets be given. With these data, we associate a parametric family of "matrix boxes"*
>
> $$\mathcal{U}[\rho] = \left\{ A + \rho \sum_{j=1}^{L} [L_j^T \Theta^j R_j + R_j^T [\Theta^j]^T L_j] : \Theta^j \in \mathcal{Z}^j, 1 \leq j \leq L \right\} \subset \mathbf{S}^m, \tag{A.4.8}$$
>
> *where* $\rho \geq 0$ *is the parameter and*
>
> $$\mathcal{Z}^j = \left\{ \begin{array}{l} \{\theta I_{p_j} : \theta \in \mathbb{R}, |\theta| \leq 1\}, j \in I_{\rm s}^{\rm r} \\ \qquad [\text{“scalar perturbation blocks”}] \\ \{\Theta^j \in \mathbb{R}^{p_j \times q_j} : \|\Theta^j\|_{2,2} \leq 1\}, j \in I_{\rm f}^{\rm r} \\ \qquad [\text{“full perturbation blocks”}] \end{array} \right. . \tag{A.4.9}$$
>
> *Given* $\rho \geq 0$, *check whether*
>
> $$\mathcal{U}[\rho] \subset \mathbf{S}_+^m \qquad \mathcal{A}(\rho)$$

Remark A.4 *We always assume that* $p_j > 1$ *for* $j \in I_{\rm s}^{\rm r}$. *Indeed, non-repeated* ($p_j = 1$) *scalar perturbations always can be regarded as full perturbations.*

Consider, along with predicate $\mathcal{A}(\rho)$, the predicate

$$\begin{array}{ll} \exists Y_j \in \mathbf{S}^m, \; j = 1, ..., L : & \\ (a) \quad Y_j \succeq L_j^T \Theta^j R_j + R_j^T [\Theta^j]^T L_j \;\forall \left(\Theta^j \in \mathcal{Z}^j, 1 \leq j \leq L\right) & \\ (b) \quad A - \rho \sum_{j=1}^{L} Y_j \succeq 0. & \end{array} \qquad \mathcal{B}(\rho)$$

The Real case version of Theorem A.6 is as follows:

Theorem A.7 [The Real Matrix Cube Theorem [3, section B.4]] *One has:*

(i) *Predicate* $\mathcal{B}(\rho)$ *is stronger than* $\mathcal{A}(\rho)$ *— the validity of the former predicate implies the validity of the latter one.*

(ii) $\mathcal{B}(\rho)$ *is computationally tractable — the validity of the predicate is equivalent to the solvability of the system of LMIs*

$$\begin{array}{lrcl} (s) & Y_j \pm \left[L_j^T R_j + R_j^T L_j\right] & \succeq & 0, \; j \in I_{\rm s}^{\rm r}, \\ (f) & \left[\begin{array}{cc} Y_j - \lambda_j L_j^T L_j & R_j^T \\ R_j & \lambda_j I_{p_j} \end{array}\right] & \succeq & 0, \; j \in I_{\rm f}^{\rm r} \\ (*) & A - \rho \sum_{j=1}^{L} Y_j & \succeq & 0. \end{array} \tag{A.4.10}$$

in matrix variables $Y_j \in \mathbf{S}^m$, $j = 1, ..., L$, *and real variables* λ_j, $j \in I_{\rm f}^{\rm r}$.

(iii) *"The gap" between $\mathcal{A}(\rho)$ and $\mathcal{B}(\rho)$ can be bounded solely in terms of the maximal rank*

$$p^{\rm s} = \max_{j \in I_{\rm S}^{\rm r}} {\rm Rank}(L_j^T R_j + R_j^T L_j)$$

of the scalar perturbations. Specifically, there exists a universal function $\vartheta_{\mathbb{R}}(\cdot)$ satisfying the relations

$$\vartheta_{\mathbb{R}}(2) = \frac{\pi}{2};\, \vartheta_{\mathbb{R}}(4) = 2;\, \vartheta_{\mathbb{R}}(\mu) \leq \pi\sqrt{\mu}/2\, \forall \mu \geq 1$$

such that with $\mu = \max[2, p^{\rm s}]$ one has

$$\text{if } \mathcal{B}(\rho) \text{ is not valid, then } \mathcal{A}(\vartheta_{\mathbb{R}}(\mu)\rho) \text{ is not valid.} \tag{A.4.11}$$

(iv) *Finally, in the case $L = 1$ of single perturbation block $\mathcal{A}(\rho)$ is equivalent to $\mathcal{B}(\rho)$.*

A.4.2 Approximate $\mathcal{S}$-Lemma

Theorem A.8 [Approximate $\mathcal{S}$-Lemma [3, section B.3]] *Let $\rho > 0$, $A, B, B_1, ..., B_J$ be symmetric $m \times m$ matrices such that $B = bb^T$, $B_j \succeq 0$, $j = 1, ..., J \geq 1$, and $B + \sum_{j=1}^{J} B_j \succ 0$.*

Consider the optimization problem

$$\mathrm{Opt}(\rho) = \max_x \left\{ x^T A x : x^T B x \leq 1, x^T B_j x \leq \rho^2,\, j = 1, ..., J \right\} \tag{A.4.12}$$

along with its semidefinite relaxation

$$\begin{array}{rcl} \mathrm{SDP}(\rho) & = & \max\limits_X \big\{ \mathrm{Tr}(AX) : \mathrm{Tr}(BX) \leq 1, \mathrm{Tr}(B_j X) \leq \rho^2, \\ & & \quad j = 1, ..., J, X \succeq 0 \big\} \\ & = & \min\limits_{\lambda, \{\lambda_j\}} \big\{ \lambda + \rho^2 \sum\limits_{j=1}^{J} \lambda_j : \lambda \geq 0, \lambda_j \geq 0,\, j = 1, ..., J, \\ & & \quad \lambda B + \sum\limits_{j=1}^{J} \lambda_j B_j \succeq A \big\}. \end{array} \tag{A.4.13}$$

Then there exists $\bar{x}$ such that

$$\begin{array}{lrcl} (a) & \bar{x}^T B \bar{x} & \leq & 1 \\ (b) & \bar{x}^T B_j \bar{x} & \leq & \Omega^2(J)\rho^2, j = 1, ..., J \\ (c) & \bar{x}^T A \bar{x} & = & \mathrm{SDP}(\rho), \end{array} \tag{A.4.14}$$

where $\Omega(J)$ is a universal function of J such that $\Omega(1) = 1$ and

$$\Omega(J) \leq 9.19\sqrt{\ln(J)},\ J \geq 2. \tag{A.4.15}$$

In particular,

$$\mathrm{Opt}(\rho) \leq \mathrm{SDP}(\rho) \leq \mathrm{Opt}(\Omega(J)\rho). \tag{A.4.16}$$

A.4.3 Talagrand Inequality

Theorem A.9 [Talagrand Inequality] *Let $\eta_1, ..., \eta_m$ be independent random vectors taking values in unit balls of the respective finite-dimensional vector spaces $(E_1, \|\cdot\|_{(1)})$,...,$(E_m, \|\cdot\|_{(m)})$, and let $\eta = (\eta_1, ..., \eta_m) \in E = E_1 \times ... \times E_m$. Let us equip E with the norm $\|(z^1, ..., z^m)\| = \sqrt{\sum_{i=1}^m \|z^i\|_{(i)}^2}$, and let Q be a closed convex subset of E. Then*

$$\mathbf{E}\left\{\exp\{\frac{\mathrm{dist}^2_{\|\cdot\|}(\eta, Q)}{16}\}\right\} \le \frac{1}{\mathrm{Prob}\{\eta \in Q\}}.$$

For proof, see, e.g., [60].

A.4.4 A Concentration Result for Gaussian Random Vector

Theorem A.10 [3, Theorem B.5.1] *Let $\zeta \sim \mathcal{N}(0, I_m)$, and let Q be a closed convex set in $\mathbb{R}^m$ such that*

$$\mathrm{Prob}\{\zeta \in Q\} \ge \chi > \frac{1}{2}. \tag{A.4.17}$$

Then

(i) *Q contains the centered at the origin $\|\cdot\|_2$-ball of the radius*

$$r(\chi) = \mathrm{ErfInv}(1 - \chi) > 0. \tag{A.4.18}$$

(ii) *If Q contains the centered at the origin $\|\cdot\|_2$-ball of a radius $r \ge r(\chi)$, then*

$$\begin{array}{l}\forall \alpha \in [1, \infty) : \mathrm{Prob}\{\zeta \notin \alpha Q\} \le \mathrm{Erf}\left(\mathrm{ErfInv}(1 - \chi) + (\alpha - 1)r\right) \\ \quad \le \mathrm{Erf}\left(\alpha \mathrm{ErfInv}(1 - \chi)\right) \le \frac{1}{2}\exp\left\{-\frac{\alpha^2 \mathrm{ErfInv}^2(1-\chi)}{2}\right\}.\end{array} \tag{A.4.19}$$

In particular, for a closed and convex set Q, $\zeta \sim \mathcal{N}(0, \Sigma)$ and $\alpha \ge 1$ one has

$$\begin{array}{l}\mathrm{Prob}\left\{\zeta \notin Q\right\} \le \delta < \frac{1}{2} \Rightarrow \\ \mathrm{Prob}\left\{\zeta \notin \alpha Q\right\} \le \mathrm{Erf}(\alpha \mathrm{ErfInv}(\delta)) \le \frac{1}{2}\exp\{-\frac{\alpha^2 \mathrm{ErfInv}^2(\delta)}{2}\}.\end{array} \tag{A.4.20}$$