

School of Mathematics



Interior Point Methods: Second-Order Cone Programming and Semidefinite Programming

Jacek Gondzio

Email: J.Gondzio@ed.ac.uk

URL: <http://www.maths.ed.ac.uk/~gondzio>

Outline

- **Self-concordant Barriers**
- **Second-Order Cone Programming**
 - example cones
 - example SOCP problems
 - logarithmic barrier function
 - IPM for SOCP
- **Semidefinite Programming**
 - background (linear matrix inequalities)
 - duality
 - example SDP problems
 - logarithmic barrier function
 - IPM for SDP
- **Final Comments**

Self-concordant Barrier

Def: Let $C \in \mathcal{R}^n$ be an open nonempty convex set.

Let $f : C \mapsto \mathcal{R}$ be a 3 times continuously diff'able convex function.

A function f is called **self-concordant** if there exists a constant $p > 0$ such that

$$|\nabla^3 f(x)[h, h, h]| \leq 2p^{-1/2}(\nabla^2 f(x)[h, h])^{3/2},$$

$\forall x \in C, \forall h : x+h \in C$. (We then say that f is p -self-concordant).

Note that a self-concordant function is always well approximated by the quadratic model because the error of such an approximation can be bounded by the $3/2$ power of $\nabla^2 f(x)[h, h]$.

Self-concordant Barrier

Lemma The barrier function $-\log x$ is self-concordant on \mathcal{R}_+ .

Proof:

Consider $f(x) = -\log x$ and compute

$$f'(x) = -x^{-1}, \quad f''(x) = x^{-2} \quad \text{and} \quad f'''(x) = -2x^{-3}$$

and check that the self-concordance condition is satisfied for $p = 1$.

Lemma

The barrier function $1/x^\alpha$, with $\alpha \in (0, \infty)$ is not self-concordant on \mathcal{R}_+ .

Lemma

The barrier function $e^{1/x}$ is not self-concordant on \mathcal{R}_+ .

Use self-concordant barriers in optimization

Second-Order Cone Programming (SOCP)

SOCP: Second-Order Cone Programming

- Generalization of QP.
- Deals with conic constraints.
- Solved with IPMs.
- Numerous applications:
quadratically constrained quadratic programs,
problems involving sums and maxima/minima of norms,
SOC-representable functions and sets,
matrix-fractional problems,
problems with hiperbolic constraints,
robust LP/QP,
robust least-squares.

SOCP: Second-Order Cone Programming

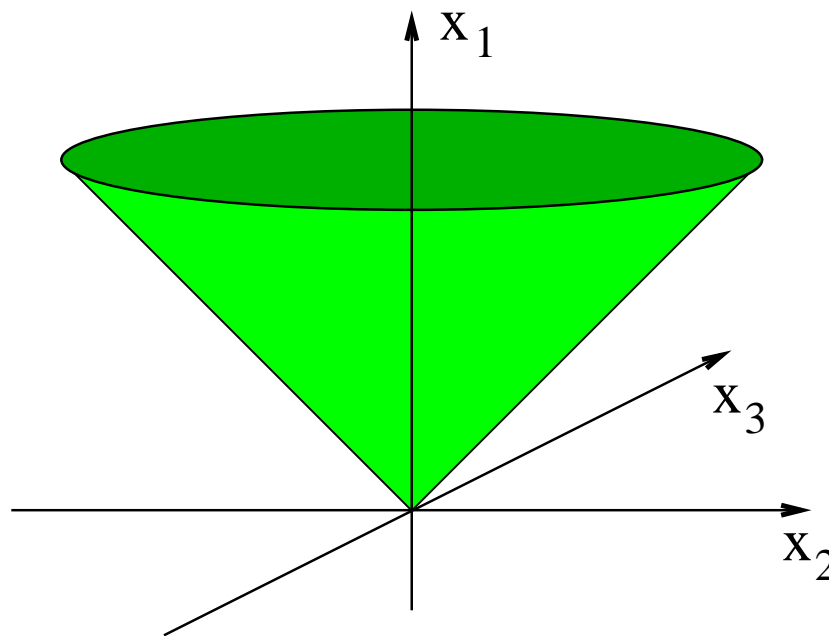
This lecture is based on three papers:

- **M. Lobo, L. Vandenberghe, S. Boyd and H. Lebret**, Applications of Second-Order Cone Programming, *Linear Algebra and its Appls* 284 (1998) pp. 193-228.
- **L. Vandenberghe and S. Boyd**, Semidefinite Programming, *SIAM Review* 38 (1996) pp. 49-95.
- **E.D. Andersen, C. Roos and T. Terlaky**, On Implementing a Primal-Dual IPM for Conic Optimization, *Mathematical Programming* 95 (2003) pp. 249-273.

Cones: Background

Def. A set $K \in \mathcal{R}^n$ is called a cone if for any $x \in K$ and for any $\lambda \geq 0$, $\lambda x \in K$.

Convex Cone:



Example:

$$K = \left\{ x \in \mathcal{R}^n : x_1^2 \geq \sum_{j=2}^n x_j^2, x_1 \geq 0 \right\}.$$

Example: Three Cones

R_+ :

$$R_+ = \{x \in \mathcal{R} : x \geq 0\}.$$

Quadratic Cone:

$$K_q = \{x \in \mathcal{R}^n : x_1^2 \geq \sum_{j=2}^n x_j^2, x_1 \geq 0\}.$$

Rotated Quadratic Cone:

$$K_r = \{x \in \mathcal{R}^n : 2x_1x_2 \geq \sum_{j=3}^n x_j^2, x_1, x_2 \geq 0\}.$$

Matrix Representation of Cones

Each of the three most common cones has a matrix representation using orthogonal matrices T and/or Q .

(Orthogonal matrix: $Q^T Q = I$).

Quadratic Cone K_q . Define

$$Q = \begin{bmatrix} 1 & & & & \\ & -1 & & & \\ & & -1 & & \\ & & & \ddots & \\ & & & & -1 \end{bmatrix}$$

and write:

$$K_q = \{x \in \mathcal{R}^n : x^T Q x \geq 0, x_1 \geq 0\}.$$

Example: $x_1^2 \geq x_2^2 + x_3^2 + \cdots + x_n^2$.

Matrix Representation of Cones (cont'd)

Rotated Quadratic Cone K_r . Define

$$Q = \begin{bmatrix} 0 & 1 & & & \\ 1 & 0 & & & \\ & & -1 & & \\ & & & \ddots & \\ & & & & -1 \end{bmatrix}$$

and write:

$$K_r = \{x \in \mathcal{R}^n : x^T Q x \geq 0, x_1, x_2 \geq 0\}.$$

Example: $2x_1x_2 \geq x_3^2 + x_4^2 + \cdots + x_n^2$.

Matrix Representation of Cones (cont'd)

Consider a linear transformation $T : \mathcal{R}^2 \mapsto \mathcal{R}^2$:

$$T_2 = \begin{bmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} \end{bmatrix}.$$

It corresponds to a rotation by $\pi/4$. Indeed, write:

$$\begin{bmatrix} z \\ y \end{bmatrix} = T_2 \begin{bmatrix} u \\ v \end{bmatrix}$$

that is

$$z = \frac{u + v}{\sqrt{2}}, \quad y = \frac{u - v}{\sqrt{2}}$$

to get

$$2yz = u^2 - v^2.$$

Matrix Representation of Cones (cont'd)

Now, define

$$T = \begin{bmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} & & & \\ \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} & & & \\ & & 1 & & \\ & & & \dots & \\ & & & & 1 \end{bmatrix}$$

and observe that the rotated quadratic cone satisfies

$$Tx \in K_r \quad \text{iff} \quad x \in K_q.$$

Example: Conic constraint

Consider a constraint:

$$\frac{1}{2}\|x\|^2 + a^T x \leq b.$$

Observe that $g(x) = \frac{1}{2}x^T x + a^T x - b$ is convex hence the constraint defines a convex set.

The constraint may be reformulated as an intersection of an affine (linear) constraint and a quadratic one:

$$\begin{aligned} a^T x + z &= b \\ y &= 1 \\ \|x\|^2 &\leq 2yz, \quad y, z \geq 0. \end{aligned}$$

Example: Conic constraint (cont'd)

Now, substitute:

$$z = \frac{u + v}{\sqrt{2}}, \quad y = \frac{u - v}{\sqrt{2}}$$

to get

$$\begin{aligned} a^T x + \frac{u + v}{\sqrt{2}} &= b \\ u - v &= \sqrt{2} \\ \|x\|^2 + v^2 &\leq u^2. \end{aligned}$$

Dual Cone

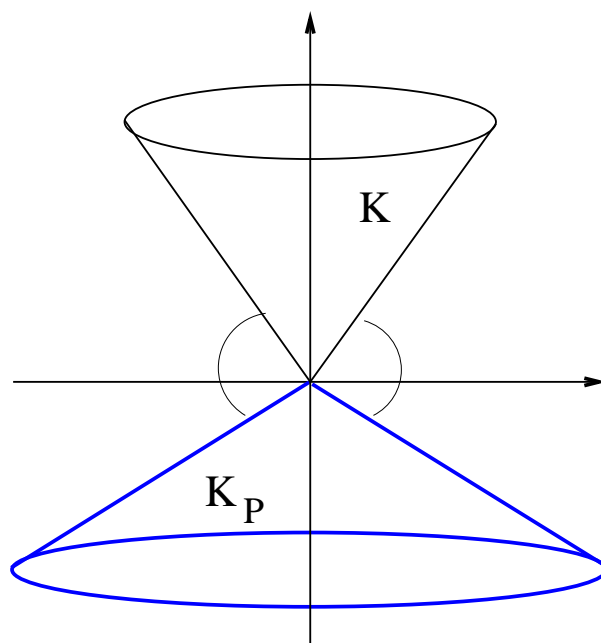
Let $K \in \mathcal{R}^n$ be a cone.

Def. The set:
$$K_* := \{s \in \mathcal{R}^n : s^T x \geq 0, \forall x \in K\}$$

is called the **dual** cone.

Def. The set:
$$K_P := \{s \in \mathcal{R}^n : s^T x \leq 0, \forall x \in K\}$$

is called the **polar** cone (Fig below).



Conic Optimization

Consider an optimization problem:

$$\begin{array}{ll} \min & c^T x \\ \text{s.t.} & Ax = b, \\ & x \in K, \end{array}$$

where K is a convex closed cone.

We assume that

$$K = K^1 \times K^2 \times \cdots \times K^k,$$

that is, cone K is a product of several individual cones each of which is one of the three cones defined earlier.

Primal and Dual SOCPs

Consider a **primal** SOCP

$$\begin{aligned} \min \quad & c^T x \\ \text{s.t.} \quad & Ax = b, \\ & x \in K, \end{aligned}$$

where K is a convex closed cone.

The associated **dual** SOCP

$$\begin{aligned} \max \quad & b^T y \\ \text{s.t.} \quad & A^T y + s = c, \\ & s \in K_*. \end{aligned}$$

Weak Duality:

If (x, y, s) is a primal-dual feasible solution, then

$$c^T x - b^T y = x^T s \geq 0.$$

IPM for Conic Optimization

Conic Optimization problems can be solved in polynomial time with IPMs.

Consider a quadratic cone

$$K_q = \{(x, t) : x \in \mathcal{R}^{n-1}, t \in \mathcal{R}, t^2 \geq \|x\|^2, t \geq 0\},$$

and define the (convex) **logarithmic barrier function** for this cone $f : \mathcal{R}^n \mapsto \mathcal{R}$

$$f(x, t) = \begin{cases} -\ln(t^2 - \|x\|^2) & \text{if } \|x\| < t \\ +\infty & \text{otherwise.} \end{cases}$$

Logarithmic Barrier Fctn for Quadratic Cone

Its derivatives are given by:

$$\nabla f(x, t) = \frac{2}{t^2 - x^T x} \begin{bmatrix} x \\ -t \end{bmatrix},$$

and

$$\nabla^2 f(x, t) = \frac{2}{(t^2 - x^T x)^2} \begin{bmatrix} (t^2 - x^T x)I + 2xx^T & -2tx \\ -2tx^T & t^2 + x^T x \end{bmatrix}.$$

Theorem:

$f(x, t)$ is a self-concordant barrier on K_q .

Exercise: Prove it in case $n = 2$.

Examples of SOCP

LP, QP use the cone \mathcal{R}_+ (positive orthant).

SDP uses the cone $\mathcal{SR}_+^{n \times n}$ (symmetric positive definite matrices).

SOCP uses two quadratic cones K_q and K_r .

Quadratically Constrained Quadratic Programming (QCQP) is a particular example of SOCP.

Typical trick to replace a quadratic constraint as a conic one!!!

Consider a constraint:

$$\frac{1}{2}\|x\|^2 + a^T x \leq b.$$

Rewrite it as:

$$\|x\|^2 + v^2 \leq u^2.$$

QCQP and SOCP

Let $P_i \in \mathcal{R}^{n \times n}$ be a symmetric positive definite matrix and $q_i \in \mathcal{R}^n$. Define a quadratic function $f_i(x) = x^T P_i x + 2q_i^T x + r_i$ and an associated ellipsoid $\mathcal{E}_i = \{x \mid f_i(x) \leq 0\}$.

The set of constraints $f_i(x) \leq 0$, $i = 1, 2, \dots, m$ defines an intersection of (convex) ellipsoids and of course defines a convex set.

The optimization problem

$$\begin{array}{ll} \min & f_0(x) \\ \text{s.t.} & f_i(x) \leq 0, \quad i = 1, 2, \dots, m, \end{array}$$

is an example of quadratically constrained quadratic program (QCQP).

QCQP can be reformulated as SOCP.

QCQP can be also reformulated as SDP.

SOCP Example: Linear Regression

The **least squares solution** of a linear system of equations $Ax = b$ is the solution of the following optimization problem

$$\min_x \|Ax - b\|$$

and it can be recast as:

$$\begin{array}{ll} \min & t \\ \text{s.t.} & \|Ax - b\| \leq t. \end{array}$$

Ellipsoids: Background

Sphere with $(0, 0)$ centre:

$$x_1^2 + x_2^2 \leq 1$$

Ellipsoid, centre at $(0, 0)$, radii a, b :

$$\frac{x_1^2}{a^2} + \frac{x_2^2}{b^2} \leq 1$$

Ellipsoid, centre at (p, q) , radii a, b :

$$\frac{(x_1 - p)^2}{a^2} + \frac{(x_2 - q)^2}{b^2} \leq 1$$

General ellipsoid:

$$(x - x_0)^T H (x - x_0) \leq 1,$$

where H is a positive definite matrix. Let $H = LL^T$. Then we can rewrite the ellipsoid as

$$\|L^T(x - x_0)\| \leq 1.$$

Ellipsoids are everywhere



Obélix



Gérard Depardieu as Obélix

SOCP Example: Robust LP

Consider an LP:

$$\begin{array}{ll} \min & c^T x \\ \text{s.t.} & a_i^T x \leq b_i, \quad i = 1, 2, \dots, m, \end{array}$$

and assume that the values of a_i are uncertain.

Suppose that $a_i \in \mathcal{E}_i$, $i = 1, 2, \dots, m$, where \mathcal{E}_i are given ellipsoids

$$\mathcal{E}_i = \{\bar{a}_i + P_i u : \|u\| \leq 1\},$$

where P_i is a symmetric positive definite matrix.

SOCP Example: Robust LP (cont'd)

Observe that

$$a_i^T x \leq b_i \quad \forall a_i \in \mathcal{E}_i \quad \text{iff} \quad \bar{a}_i^T x + \|P_i x\| \leq b_i,$$

because for any $x \in \mathcal{R}^n$

$$\begin{aligned} \max\{a^T x : a \in \mathcal{E}\} &= \bar{a}^T x + \max\{u^T P x : \|u\| \leq 1\} \\ &= \bar{a}^T x + \|P x\|. \end{aligned}$$

Hence **robust LP** formulated as SOCP is:

$$\begin{aligned} \min \quad & c^T x \\ \text{s.t.} \quad & \bar{a}_i^T x + \|P_i x\| \leq b_i, \quad i = 1, 2, \dots, m. \end{aligned}$$

SOCP Example: Robust QP

Consider a QP with “uncertain” objective:

$$\min_x \max_{P \in \mathcal{E}} x^T P x + 2q^T x + r$$

subject to linear constraints. “Uncertain” symmetric positive definite matrix P belongs to the ellipsoid:

$$P \in \mathcal{E} = \left\{ P_0 + \sum_{i=1}^m P_i u_i : \|u\| \leq 1 \right\},$$

where P_i are symmetric positive semidefinite matrices. The definition of ellipsoid \mathcal{E} implies that

$$\max_{P \in \mathcal{E}} x^T P x = x^T P_0 x + \max_{\|u\| \leq 1} \sum_{i=1}^m (x^T P_i x) u_i.$$

SOCP Example: Robust QP (cont'd)

From Cauchy-Schwartz inequality:

$$\sum_{i=1}^m (x^T P_i x) u_i \leq \left(\sum_{i=1}^m (x^T P_i x)^2 \right)^{1/2} \|u\|$$

hence

$$\max_{\|u\| \leq 1} \sum_{i=1}^m (x^T P_i x) u_i \leq \left(\sum_{i=1}^m (x^T P_i x)^2 \right)^{1/2}.$$

We get a reformulation of robust QP:

$$\min_x x^T P_0 x + \left(\sum_{i=1}^m (x^T P_i x)^2 \right)^{1/2} + 2q^T x + r.$$

SOCP Example: Robust QP (cont'd)

This problem can be written as:

$$\begin{aligned} \min \quad & t + v + 2q^T x + r \\ \text{s.t.} \quad & \|z\| \leq t, \quad x^T P_0 x \leq v, \quad x^T P_i x \leq z_i, \quad i = 1, \dots, m. \end{aligned}$$

SOCP reformulation:

$$\begin{aligned} \min \quad & t + v + 2q^T x + r \\ \text{s.t.} \quad & \|z\| \leq t, \\ & \|(2P_i^{1/2} x, z_i - 1)\| \leq z_i + 1, \quad z_i \geq 0, \quad i = 1..m, \\ & \|(2P_0^{1/2} x, v - 1)\| \leq v + 1, \quad v \geq 0. \end{aligned}$$

Semidefinite Programming (SDP)

SDP: Semidefinite Programming

- Generalization of LP.
- Deals with symmetric positive semidefinite matrices (Linear Matrix Inequalities, LMI).
- Solved with IPMs.
- Numerous applications:
eigenvalue optimization problems,
quasi-convex programs,
convex quadratically constrained optimization,
robust mathematical programming,
matrix norm minimization,
combinatorial optimization (provides good relaxations),
control theory,
statistics.

SDP: Semidefinite Programming

This lecture is based on two survey papers:

- **L. Vandenberghe and S. Boyd**,
Semidefinite Programming,
SIAM Review 38 (1996) pp. 49-95.
- **M.J. Todd**,
Semidefinite Optimization,
Acta Numerica 10 (2001) pp. 515-560.

SDP: Background

Def. A matrix $H \in \mathcal{R}^{n \times n}$ is positive semidefinite if $x^T H x \geq 0$ for any $x \neq 0$. We write $H \succeq 0$.

Def. A matrix $H \in \mathcal{R}^{n \times n}$ is positive definite if $x^T H x > 0$ for any $x \neq 0$. We write $H \succ 0$.

We denote with $\mathcal{SR}^{n \times n}$ ($\mathcal{SR}_+^{n \times n}$) the set of symmetric and symmetric positive semidefinite matrices.

Let $U, V \in \mathcal{SR}^{n \times n}$. We define the inner product between U and V as $U \bullet V = \text{trace}(U^T V)$, where $\text{trace}(H) = \sum_{i=1}^n h_{ii}$.

The associated norm is the Frobenius norm, written $\|U\|_F = (U \bullet U)^{1/2}$ (or just $\|U\|$).

Linear Matrix Inequalities

Def. *Linear Matrix Inequalities*

Let $U, V \in \mathcal{SR}^{n \times n}$.

We write $U \succeq V$ iff $U - V \succeq 0$.

We write $U \succ V$ iff $U - V \succ 0$.

We write $U \preceq V$ iff $U - V \preceq 0$.

We write $U \prec V$ iff $U - V \prec 0$.

Properties

1. If $P \in \mathcal{R}^{m \times n}$ and $Q \in \mathcal{R}^{n \times m}$, then $\text{trace}(PQ) = \text{trace}(QP)$.
2. If $U, V \in \mathcal{SR}^{n \times n}$, and $Q \in \mathcal{R}^{n \times n}$ is orthogonal (i.e. $Q^T Q = I$), then $U \bullet V = (Q^T U Q) \bullet (Q^T V Q)$.
More generally, if P is nonsingular, then $U \bullet V = (P U P^T) \bullet (P^{-T} V P^{-1})$.
3. Every $U \in \mathcal{SR}^{n \times n}$ can be written as $U = Q \Lambda Q^T$, where Q is orthogonal and Λ is diagonal. Then $U Q = Q \Lambda$.
In other words the columns of Q are the eigenvectors, and the diagonal entries of Λ the corresponding eigenvalues of U .
4. If $U \in \mathcal{SR}^{n \times n}$ and $U = Q \Lambda Q^T$, then $\text{trace}(U) = \text{trace}(\Lambda) = \sum_i \lambda_i$.

Properties (cont'd)

5. For $U \in \mathcal{SR}^{n \times n}$, the following are equivalent:

- (i) $U \succeq 0$ ($U \succ 0$)
- (ii) $x^T U x \geq 0, \forall x \in \mathcal{R}^n$ ($x^T U x > 0, \forall 0 \neq x \in \mathcal{R}^n$).
- (iii) If $U = Q \Lambda Q^T$, then $\Lambda \succeq 0$ ($\Lambda \succ 0$).
- (iv) $U = P^T P$ for some matrix P ($U = P^T P$ for some square nonsingular matrix P).

6. Every $U \in \mathcal{SR}^{n \times n}$ has a square root $U^{1/2} \in \mathcal{SR}^{n \times n}$.

Proof: From Property 5 (ii) we get $U = Q \Lambda Q^T$.

Take $U^{1/2} = Q \Lambda^{1/2} Q^T$, where $\Lambda^{1/2}$ is the diagonal matrix whose diagonal contains the (nonnegative) square roots of the eigenvalues of U , and verify that $U^{1/2} U^{1/2} = U$.

Properties (cont'd)

7. Suppose

$$U = \begin{bmatrix} A & B^T \\ B & C \end{bmatrix},$$

where A and C are symmetric and $A \succ 0$.

Then $U \succeq 0$ ($U \succ 0$) iff $C - BA^{-1}B^T \succeq 0$ ($\succ 0$).

The matrix $C - BA^{-1}B^T$ is called the *Schur complement* of A in U .

Proof: follows easily from the factorization:

$$\begin{bmatrix} A & B^T \\ B & C \end{bmatrix} = \begin{bmatrix} I & 0 \\ BA^{-1} & I \end{bmatrix} \begin{bmatrix} A & 0 \\ 0 & C - BA^{-1}B^T \end{bmatrix} \begin{bmatrix} I & A^{-1}B^T \\ 0 & I \end{bmatrix}.$$

8. If $U \in \mathcal{SR}^{n \times n}$ and $x \in \mathcal{R}^n$, then $x^T U x = U \bullet x x^T$.

Primal-Dual Pair of SDPs

Primal

$$\begin{aligned} \min \quad & C \bullet X \\ \text{s.t.} \quad & A_i \bullet X = b_i, \quad i = 1..m \\ & X \succeq 0; \end{aligned}$$

Dual

$$\begin{aligned} \max \quad & b^T y \\ \text{s.t.} \quad & \sum_{i=1}^m y_i A_i + S = C, \\ & S \succeq 0, \end{aligned}$$

where $A_i \in \mathcal{SR}^{n \times n}$, $b \in \mathcal{R}^m$, $C \in \mathcal{SR}^{n \times n}$ are given;
and $X, S \in \mathcal{SR}^{n \times n}$, $y \in \mathcal{R}^m$ are the variables.

Theorem: Weak Duality in SDP

If X is feasible in the primal and (y, S) in the dual, then

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

Proof:

$$\begin{aligned} C \bullet X - b^T y &= \left(\sum_{i=1}^m y_i A_i + S \right) \bullet X - b^T y \\ &= \sum_{i=1}^m (A_i \bullet X) y_i + S \bullet X - b^T y \\ &= S \bullet X = X \bullet S. \end{aligned}$$

Further, since X is positive semidefinite, it has a square root $X^{1/2}$ (Property 6), and so

$$X \bullet S = \text{trace}(XS) = \text{trace}(X^{1/2} X^{1/2} S) = \text{trace}(X^{1/2} S X^{1/2}) \geq 0.$$

We use Property 1 and the fact that S and $X^{1/2}$ are positive semidefinite, hence $X^{1/2} S X^{1/2}$ is positive semidefinite and its trace is nonnegative.

SDP Example 1: Minimize the Max. Eigenvalue

We wish to choose $x \in \mathcal{R}^k$ to minimize the maximum eigenvalue of $A(x) = A_0 + x_1 A_1 + \dots + x_k A_k$, where $A_i \in \mathcal{R}^{n \times n}$ and $A_i = A_i^T$. Observe that

$$\lambda_{max}(A(x)) \leq t$$

if and only if

$$\lambda_{max}(A(x) - tI) \leq 0 \quad \iff \quad \lambda_{min}(tI - A(x)) \geq 0.$$

This holds iff

$$tI - A(x) \succeq 0.$$

So we get the SDP in the *dual* form:

$$\begin{aligned} \max \quad & -t \\ \text{s.t.} \quad & tI - A(x) \succeq 0, \end{aligned}$$

where the variable is $y := (t, x)$.

SDP Example 2: Logarithmic Chebyshev Approx.

Suppose we wish to solve $Ax \approx b$ approximately,

where $A = [a_1 \dots a_n]^T \in \mathcal{R}^{n \times k}$ and $b \in \mathcal{R}^n$.

In Chebyshev approximation we minimize the ℓ_∞ -norm of the residual, i.e., we solve

$$\min \max_i |a_i^T x - b_i|.$$

This can be cast as an LP, with x and an auxiliary variable t :

$$\begin{aligned} \min \quad & t \\ \text{s.t.} \quad & -t \leq a_i^T x - b_i \leq t, \quad i = 1..n. \end{aligned}$$

In some applications b_i has a dimension of a power of intensity, and it is typically expressed on a logarithmic scale. In such cases the more natural optimization problem is

$$\min \max_i |\log(a_i^T x) - \log(b_i)|$$

(assuming $a_i^T x > 0$ and $b_i > 0$).

Logarithmic Chebyshev Approximation (cont'd)

The logarithmic Chebyshev approximation problem can be cast as a semidefinite program. To see this, note that

$$|\log(a_i^T x) - \log(b_i)| = \log \max(a_i^T x / b_i, b_i / a_i^T x).$$

Hence the problem can be rewritten as the following (nonlinear) program

$$\begin{aligned} \min \quad & t \\ \text{s.t.} \quad & 1/t \leq a_i^T x / b_i \leq t, \quad i = 1..n. \end{aligned}$$

or,

$$\begin{aligned} \min \quad & t \\ \text{s.t.} \quad & \begin{bmatrix} t - a_i^T x / b_i & 0 & 0 \\ 0 & a_i^T x / b_i & 1 \\ 0 & 1 & t \end{bmatrix} \succeq 0, \quad i = 1..n \end{aligned}$$

which is a semidefinite program.

Logarithmic Barrier Function

Define the **logarithmic barrier function** for the cone $\mathcal{SR}_+^{n \times n}$ of positive definite matrices.

$$f : \mathcal{SR}_+^{n \times n} \mapsto \mathcal{R}$$

$$f(X) = \begin{cases} -\ln \det X & \text{if } X \succ 0 \\ +\infty & \text{otherwise.} \end{cases}$$

Let us evaluate its derivatives.

Let $X \succ 0, H \in \mathcal{SR}^{n \times n}$. Then

$$\begin{aligned} f(X + \alpha H) &= -\ln \det[X(I + \alpha X^{-1}H)] \\ &= -\ln \det X - \ln(1 + \alpha \text{trace}(X^{-1}H) + \mathcal{O}(\alpha^2)) \\ &= f(X) - \alpha X^{-1} \bullet H + \mathcal{O}(\alpha^2), \end{aligned}$$

so that $f'(X) = -X^{-1}$ and $Df(X)[H] = -X^{-1} \bullet H$.

Logarithmic Barrier Function (cont'd)

Similarly

$$\begin{aligned} f'(X + \alpha H) &= -[X(I + \alpha X^{-1}H)]^{-1} \\ &= -[I - \alpha X^{-1}H + \mathcal{O}(\alpha^2)]X^{-1} \\ &= f'(X) + \alpha X^{-1}HX^{-1} + \mathcal{O}(\alpha^2), \end{aligned}$$

so that $f''(X)[H] = X^{-1}HX^{-1}$

and $D^2f(X)[H, G] = X^{-1}HX^{-1} \bullet G$.

Finally,

$$f'''(X)[H, G] = -X^{-1}HX^{-1}GX^{-1} - X^{-1}GX^{-1}HX^{-1}.$$

Logarithmic Barrier Function (cont'd)

Theorem: $f(X) = -\ln \det X$ is a convex barrier for $\mathcal{SR}_+^{n \times n}$.

Proof: Define $\phi(\alpha) = f(X + \alpha H)$. We know that f is convex if, for every $X \in \mathcal{SR}_+^{n \times n}$ and every $H \in \mathcal{SR}^{n \times n}$, $\phi(\alpha)$ is convex in α .

Consider a set of α such that $X + \alpha H \succ 0$. On this set

$$\phi''(\alpha) = D^2 f(\bar{X})[H, H] = \bar{X}^{-1} H \bar{X}^{-1} \bullet H,$$

where $\bar{X} = X + \alpha H$.

Since $\bar{X} \succ 0$, so is $V = \bar{X}^{-1/2}$ (Property 6), and

$$\begin{aligned} \phi''(\alpha) &= V^2 H V^2 \bullet H = \text{trace}(V^2 H V^2 H) \\ &= \text{trace}((V H V)(V H V)) = \|V H V\|_F^2 \geq 0. \end{aligned}$$

So ϕ is convex.

When $X \succ 0$ approaches a singular matrix, its determinant approaches zero and $f(X) \rightarrow \infty$.

Simplified Notation

Define $\mathcal{A} : \mathcal{SR}^{n \times n} \mapsto \mathcal{R}^m$

$$\mathcal{A}X = (A_i \bullet X)_{i=1}^m \in \mathcal{R}^m.$$

Note that, for any $X \in \mathcal{SR}^{n \times n}$ and $y \in \mathcal{R}^m$,

$$(\mathcal{A}X)^T y = \sum_{i=1}^m (A_i \bullet X) y_i = \left(\sum_{i=1}^m y_i A_i \right) \bullet X,$$

so the adjoint of \mathcal{A} is given by

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

\mathcal{A}^* is a mapping from \mathcal{R}^m to $\mathcal{SR}^{n \times n}$.

Simplified Notation (cont'd)

With this notation the **primal** SDP becomes

$$\begin{aligned} \min \quad & C \bullet X \\ \text{s.t.} \quad & \mathcal{A}X = b, \\ & X \succeq 0, \end{aligned}$$

where $X \in \mathcal{SR}^{n \times n}$ is the variable.

The associated **dual** SDP writes

$$\begin{aligned} \max \quad & b^T y \\ \text{s.t.} \quad & \mathcal{A}^* y + S = C \\ & S \succeq 0, \end{aligned}$$

where $y \in \mathcal{R}^m$ and $S \in \mathcal{SR}^{n \times n}$ are the variables.

Solving SDPs with IPMs

Replace the **primal SDP**

$$\begin{aligned} \min \quad & C \bullet X \\ \text{s.t.} \quad & \mathcal{A}X = b, \\ & X \succeq 0, \end{aligned}$$

with the **primal barrier SDP**

$$\begin{aligned} \min \quad & C \bullet X + \mu f(X) \\ \text{s.t.} \quad & \mathcal{A}X = b, \end{aligned}$$

(with a barrier parameter $\mu \geq 0$).

Formulate the Lagrangian

$$L(X, y, S) = C \bullet X + \mu f(X) - y^T (\mathcal{A}X - b),$$

with $y \in \mathcal{R}^m$, and write the first order conditions (FOC) for a stationary point of L :

$$C + \mu f'(X) - \mathcal{A}^* y = 0.$$

Solving SDPs with IPMs (cont'd)

Use $f(X) = -\ln \det(X)$ and $f'(X) = -X^{-1}$.

Therefore the FOC become:

$$C - \mu X^{-1} - \mathcal{A}^* y = 0.$$

Denote $S = \mu X^{-1}$, i.e., $XS = \mu I$.

For a positive definite matrix X its inverse is also positive definite.

The FOC now become:

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

with $X \succ 0$ and $S \succ 0$.

Newton direction

We derive the Newton direction for the system:

$$\begin{aligned} \mathcal{A}X &= b, \\ \mathcal{A}^*y + S &= C, \\ -\mu X^{-1} + S &= 0. \end{aligned}$$

Recall that the variables in FOC are (X, y, S) , where $X, S \in \mathcal{SR}_+^{n \times n}$ and $y \in \mathcal{R}^m$.

Hence we look for a direction $(\Delta X, \Delta y, \Delta S)$, where $\Delta X, \Delta S \in \mathcal{SR}_+^{n \times n}$ and $\Delta y \in \mathcal{R}^m$.

Newton direction (cont'd)

The differentiation in the above system is a **nontrivial** operation. The direction is the solution of the system:

$$\begin{bmatrix} \mathcal{A} & 0 & 0 \\ 0 & \mathcal{A}^* & \mathcal{I} \\ \mu(X^{-1} \odot X^{-1}) & 0 & \mathcal{I} \end{bmatrix} \cdot \begin{bmatrix} \Delta X \\ \Delta y \\ \Delta S \end{bmatrix} = \begin{bmatrix} \xi_b \\ \xi_C \\ \xi_\mu \end{bmatrix}.$$

We introduce a useful notation $P \odot Q$ for $n \times n$ matrices P and Q . This is an operator from $\mathcal{SR}^{n \times n}$ to $\mathcal{SR}^{n \times n}$ defined by

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

Logarithmic Barrier Function

for the cone $\mathcal{SR}_+^{n \times n}$ of positive definite matrices, $f : \mathcal{SR}_+^{n \times n} \mapsto \mathcal{R}$

$$f(X) = \begin{cases} -\ln \det X & \text{if } X \succ 0 \\ +\infty & \text{otherwise.} \end{cases}$$

LP: Replace $x \geq 0$ with $-\mu \sum_{j=1}^n \ln x_j$.

SDP: Replace $X \succeq 0$ with $-\mu \sum_{j=1}^n \ln \lambda_j = -\mu \ln(\prod_{j=1}^n \lambda_j)$.

Nesterov and Nemirovskii,

Interior Point Polynomial Algorithms in Convex Programming: Theory and Applications, SIAM, Philadelphia, 1994.

Lemma The barrier function $f(X)$ is self-concordant on $\mathcal{SR}_+^{n \times n}$.

Interior Point Methods:

- Logarithmic barrier functions for SDP and SOCP
Self-concordant barriers
→ polynomial complexity (predictable behaviour)
- Unified view of optimization
→ from LP via QP to NLP, SDP, SOCP
- Efficiency
 - good for SOCP
 - problematic for SDP because solving the problem of size n involves linear algebra operations in dimension n^2
→ and this requires n^6 flops!

Use IPMs in your research!