Mathematical Optimization and Polyhedral Approaches: SemiDefinite Programming and Conic Optmization Master EUR Maths & Computer Science

Lucas Létocart

Université Sorbonne Paris Nord
src: M. Anjos, A. Lambert, A. Wiegele

Semidefinite Optimization

Semidefinite optimization, or semidefinite programming (SDP), refers to the problem of optimizing a linear function over the intersection of the set of symmetric positive semidefinite matrices with an affine space.

The simplest example of semidefinite optimization is the familiar linear programming (LP) problem :

$$\begin{array}{lll|lll} \max & \langle \mathrm{c}, \mathrm{x} \rangle & & \min & \langle \mathrm{b}, \mathrm{y} \rangle & \\ \text{s.t.} & \langle \mathrm{a}_i, \mathrm{x} \rangle = b_i, \ i = 1, \ldots, m & & \text{s.t.} & \mathrm{z} = \sum_{i=1}^{m} y_i \mathrm{a}_i - \mathrm{c} & \\ & \mathrm{x} \geq 0 & & & \mathrm{z} \geq 0 & \end{array}$$

where $\langle \mathrm{a}, \mathrm{b} \rangle = \mathrm{a}^T \mathrm{b}$.

Semidefinite Optimization

The general SDP problem has the form :

$$\begin{array}{ll|ll} \max & \langle \mathrm{C}, \mathrm{X} \rangle & \min & \langle \mathrm{b}, \mathrm{y} \rangle \\ \text{s.t.} & \langle \mathrm{A}_i, \mathrm{X} \rangle = b_i, \; i = 1, \ldots, m & \text{s.t.} & \mathrm{Z} = \sum_{i=1}^{m} y_i \mathrm{A}_i - \mathrm{C} \\ & \mathrm{X} \succeq 0 & & \mathrm{Z} \succeq 0 \end{array}$$

where $\langle \mathrm{A}, \mathrm{B} \rangle = \mathrm{A} \bullet \mathrm{B} = \operatorname{trace} \mathrm{AB} = \sum_{i,j} A_{ij} B_{ij}$,
all matrices are square and symmetric ($\mathcal{S}^n$),
and $\mathrm{X} \succeq 0$ denotes that X is positive semidefinite.

Importance of SDP

SDP is an important class of optimization problems for several reasons :

1. Because SDP problems are solvable in polynomial time, any problem that can be expressed using SDP is also solvable in polynomial time.
2. SDP problems can be solved efficiently in practice. This can be done by using one of the software packages available, or alternatively by implementing a suitable algorithm.
3. SDP can be used to obtain tight approximations for hard problems in integer and global optimization.
4. SDP problems are useful for a wide range of practical applications in areas such as control theory, portfolio optimization, truss topology design, and principal component analysis.

Convex sets

Convex sets

- A set C is **convex** if and only if for all points x and y in C, the segment connecting x and y is included in C.

  Examples : cubes, balls or ellipsoids

- To use convex sets in algorithms, you need to be able to represent them in a machine :
  - implicit description by the intersection of half-spaces,
  - explicit description of the convex combination of its extremal points

- Here, we will be talking about conic programming and convex cones.
  Examples of convex cones for optimization : $\mathbb{R}^n_+$, the cone of positive semi-definite matrices or the cone of copositive matrices.

Context

- We're working in a n-dimensional Euclidean space E provided with a scalar product : here $\mathbb{R}^n$

- For column vectors $x = (x_1, \ldots, x_n)^T$ and $y = (y_1, \ldots, y_n)^T$ belonging to $\mathbb{R}^n$, the scalar product is

$$x^T y = \sum_{i=1}^{n} x_i y_i$$

.

Topology in metric spaces : definitions (1/2)

- We define the **ball of center $x \in \mathbb{R}^n$ and radius r** as

$$B(x,r) = \{y \in \mathbb{R}^n : (x,y) \leq r\}$$

- If $A \subset \mathbb{R}^n$, a point $x \in A$ is an **interior point** if there exists a radius $\epsilon > 0$ such that $B(x,\epsilon) \subseteq A$

- Let $\mathrm{int}\ A$ be the set of interior points of A, and we say that A is **open** if all points of A are interior.

- The set A is **closed** if its complement $\mathbb{R}^n \backslash A$ is open.

Topology in metric spaces : definitions (2/2)

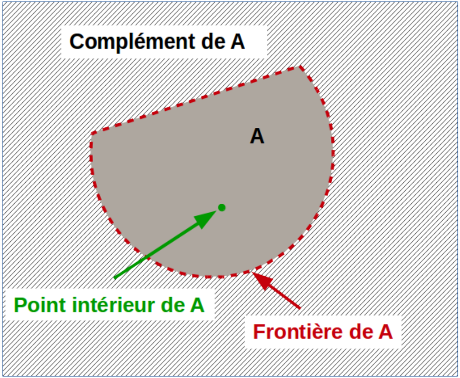
- The **closure** of A, noted $\bar{A}$, is the intersection of all close sets containing A, so it's the smallest close set containing A.

- A point x belongs to the **frontier** of A, noted ∂A, if for any $\epsilon > 0$, the ball $B(x, \epsilon)$ has points in A and in its complement $\mathbb{R}^n \backslash A$.

- The **frontier** ∂A is a close set and we have $\bar{A} = A \cup \partial A$ and $\partial A = \bar{A} \backslash \text{int } A$.

- The set A is **compact** if and only if it is closed and bounded. (i.e. it can be contained in a sufficiently large ball of finite radius).

Topology in metric spaces : Illustration



Convex sets : definitions (1/4)

- The **segment between points x and y** is the set of barycenters with positive or zero coefficients of x and y, it is defined by points z :

$$[x, y] = \{z = (1 - \alpha)x + \alpha y \ : \alpha \text{ element of } [0, 1]\}$$

- A set C is **convex** if and only if for all points x and y in C, the segment connecting x and y is included in C

- Any **intersection** of convex sets is a **convex set**.

Convex sets : definitions (2/4)

- Given $n \in \mathbb{N}^*$ and $x_1, \ldots, x_n \in C \subseteq \mathbb{R}^n$, a **convex combination** of x_i is a point p of the form :

$$p = \sum_{i=1}^{n} \alpha_i x_i \text{ where } \alpha \in \mathbb{R}_+^n \text{ and } \sum_{i=1}^{n} \alpha_i = 1$$

- A **convex subset is stable to convex combinations** :

$$\forall \tilde{n} \in \mathbb{N}, \forall x_1, \ldots, x_{\tilde{n}} \in C \subseteq \mathbb{R}^n, \forall \alpha \in \mathbb{R}_+^{\tilde{n}}, \text{ st } \sum_{i=1}^{\tilde{n}} \alpha_i = 1,$$

$$\text{we have } p = \sum_{i=1}^{\tilde{n}} \alpha_i x_i \in C$$

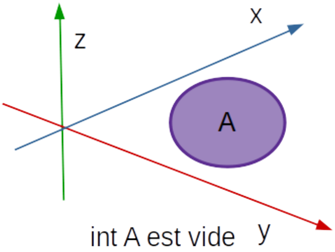
Convex sets : definitions (3/4)

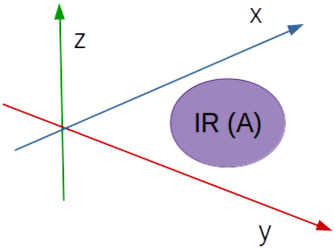
Let $C \subseteq \mathbb{R}^n$ be a convex set. A point $x \in C$ is in the **relative interior** of C if and only if

$$\forall y \in C, \exists z \in C, \alpha \in]0,1[: x = \alpha y + (1-\alpha) z$$

In other words, it's the interior of C relative to the affine subspace generated by C.

Example : Let $A = \{(x,y,z) \in \mathbb{R}^3 \mid z = 0, x^2 + y^2 \leq 1\}$.
$\forall, x \in A, \nexists \epsilon > 0 \text{ st } B(x,\epsilon) \subseteq A$, (as $z = 0$) $\Rightarrow \text{int } A = \emptyset$





Convex sets : definitions (4/4)

- The **convex envelope** of $A \in \mathbb{R}^n$ (denoted $\mathrm{conv}\ A$) is the smallest convex set containing A.
- This is the set of convex combinations of points $x_1, \ldots, x_{tilden}$, $\forall \tilde{n} \in \mathbb{N}$, i.e. their barycenters with positive coefficients :

$$\mathrm{conv}\ A = \left\{ p = \sum_{i=1}^{\tilde{n}} \alpha_i x_i \ : \tilde{n} \in \mathbb{N}, \forall x_1, \ldots, x_{\tilde{n}} \in C, \forall \alpha \in \mathbb{R}_+^{\tilde{n}}, \text{ st } \sum_{i=1}^{\tilde{n}} \alpha_i = 1 \right\}$$

- The convex envelope of a finite set of points is called a polytope (or polygon in 2 dimensions).

Implicit description : intersection of half-spaces (1/2)

- A **hyperplane H** corresponding to the normal vector $c \in \mathbb{R}^n \backslash \{0\}$ passing through $x \in \mathbb{R}^n$ is :

$$H = \{x \in \mathbb{R}^n \, : \, c^T x = \beta\}$$

it is an affine subspace of dimension $n - 1$.

- The hyperplane H divides $\mathbb{R}^n$ into 2 **close half spaces** :

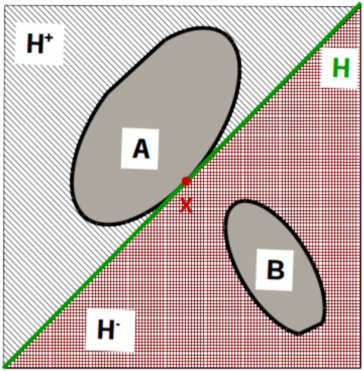
$$H^+ = \{y \in \mathbb{R}^n \, : \, c^T y \geq c^T x = \beta\} \text{ and } H^- = \{y \in \mathbb{R}^n \, : \, c^T y \leq c^T x = \beta\}$$

- H divides $A \subseteq \mathbb{R}^n$ and $B \subseteq \mathbb{R}^n$ if $A \subseteq H^+$ and $B \subseteq H^-$ (or vice-versa).
  i.e. if $\exists c \in \mathbb{R}^n \backslash \{0\}, \beta \in \mathbb{R} \, : \, \forall x \in A, y \in B \, : \, c^T x \leq \beta \leq c^T y$

- If the two inequalities are strict, the separation is called **strict**.

Implicit description : intersection of half-spaces (2/2)

- H is the **support of A in x**, if $x \in A$, $x \in H$ and A is contained in one of the 2 subspaces H^+ or H^-.
- Then we say that H^+ (or H^-) is the **support half-space**.



Example : Hyperplane H supports A at x and connects A and B

Orthogonal projection

Let C be a closed convex space. It is possible to project all points x of $\mathbb{R}^n$ onto C : we take the point of C that is closest to x.

- Let $x \in \mathbb{R}^n \backslash C$ (x is outside C), $exists! \pi_C(x) \in C$, which is closest to x. Furthermore, $\pi_C(x) \in \partial C$.
- The **projection** $\pi_C : \mathbb{R}^n \to C$ is characterized by the property

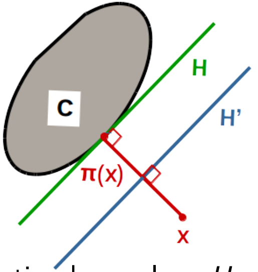
$$\forall y \in C \, : \, d(\pi_C(x), x) \leq d(y, x)$$

- For all $y \in \partial C$, it exists x exterior to C, such that $y = \pi_C(x)$

Support hyperplanes and separators

Let $C \subset \mathbb{R}^n$ be a convex space and closed, $x \in \mathbb{R}^n \setminus C$, $\pi_C(x) \in C$ the projection of x onto C, we have :

1. the hyperplane H which passes through x with normal $x - \pi_C(x)$ supports C at $pi_C(x)$ and thus separates $\{x\}$ and C.
2. the H' hyperplane passing through $(x + \pi_C(x))/2$ with normal $x - \pi_C(x)$ strictly separates $\{x\}$ and C.



So we can construct a supporting hyperplane H on each point $x \in \partial C$.
$\Rightarrow$ implicit description of the convex set.

Explicit description : set of extreme points (1/2)

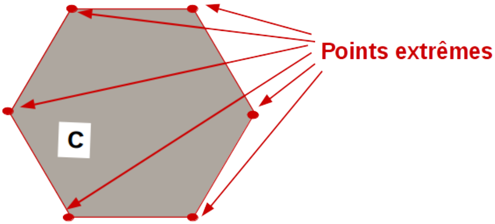
Definition : a point $x \in C$ is an **extreme point** if $C \setminus \{x\}$ is still convex (i.e. any segment s of C, x is not a point of the relative interior of s.)

i.e. : if x cannot be written in the form $x = (1-\alpha)y + \alpha z$, with $y, z \in C$ and $0 < \alpha < 1$.

Let $\mathbf{ext}\ C$ denote the set of all extreme points of C.

Property : Let $C \subset \mathbb{R}^n$ be a compact and convex set, then

$$C = \mathrm{conv}(\mathrm{ext}\ C)$$



The extreme points of a convex set.

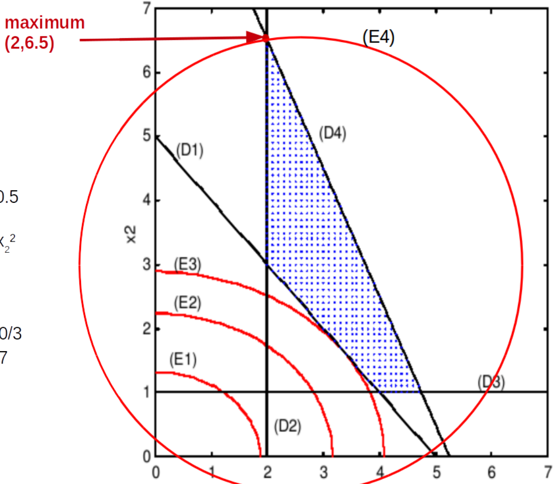
Explicit description : set of extreme points (2/2)

Application : Let $C \subset \mathbb{R}^n$ be compact and convex, and $f : \mathbb{R}^n \mapsto \mathbb{R}$ continuous and convex, then $\max_{x \in C} f(x)$ has a maximum which is an extreme point of C.

(D1) : $x_1 + x_2 \leq 5$
(D2) : $x_1 \geq 2$
(D3) : $x_2 \geq 1$
(D4) : $2x_1 + x_2 \leq 10.5$

$f(x_1,x_2) = 2x_1^2 + 4\,x_2^2$

(E1) : $f(x_1,x_2) = 7$
(E2) : $f(x_1,x_2) = 20$
(E3) : $f(x_1,x_2) = 100/3$
(E2) : $f(x_1,x_2) = 177$



Convex Cones

Convex Cones (1/4)

- A non-empty set $K \subset \mathbb{R}^n$ is a **convex cone** if and only if it is closed under positive linear combinations :

$$\forall \alpha, \beta \in \mathbb{R}_+, \forall x, y \in K \;:\; \alpha x + \beta y \in K$$

- A convex cone is **pointed** if it contains no line :

$$x \in K \cup -x \in K \Rightarrow x = 0$$

- The **dual cone** K^* of a convex cone K is defined as :

$$K^* = \{y \in \mathbb{R}^n \;:\; x^T y \geq 0, \;\forall x \in K\}$$

K^* is a closed and convex cone.

Convex Cones (2/4)

A **convex, closed, pointed cone with non-empty interior** $K \subset \mathbb{R}^n$ defines a partial order on $\mathbb{R}^n$, for $x, y \in \mathbb{R}^n$:

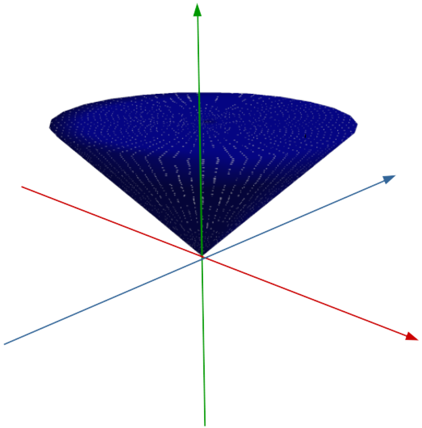
$$x \succeq y \iff x - y \in K$$

- This partial order satisfies the following properties :
  1. Reflexivity : $\forall x \in \mathbb{R}^n \ : \ x \succeq x$
  2. Antisymmetry : $\forall x, y \in \mathbb{R}^n \ : \ x \succeq y,\ y \succeq x \Longrightarrow x = y$
  3. Transitivity : $\forall x, y, z \in \mathbb{R}^n \ : \ x \succeq y,\ y \succeq z \Longrightarrow x \succeq z$
  4. Homogeneity : $\forall x, y \in \mathbb{R}^n, \forall \alpha \in \mathbb{R}_+ \ : \ x \succeq y \Longrightarrow \alpha x \succeq \alpha y$
  5. Additivity : $\forall x, y, x', y' \in \mathbb{R}^n \ : \ x \succeq y,\ x' \succeq y' \Longrightarrow x + x' \succeq y + y'$

- Strict inequality is defined as :

$$\forall x, y \in \mathbb{R}^n \ : \ x \succ y \iff x - y \in \mathrm{int}\ K$$

Convex Cones (3/4)

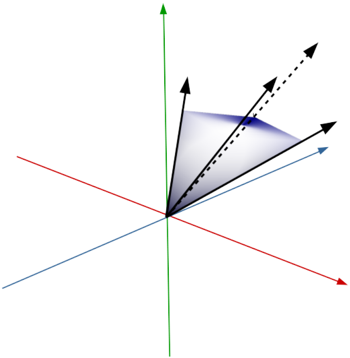


Example of a cone that is not the conic combination of a finite number of vectors.

Convex Cones (4/4)

The convex cone generated by a set of vectors $A \subset \mathbb{R}^n$ is the smallest convex cone containing A :

$$\text{cone } A = \{p = \sum_{i=1}^{n} \alpha_i x_i \ : \ n \in \mathbb{N}, \ x \in A, \ \alpha \in \mathbb{R}_+^n\}$$

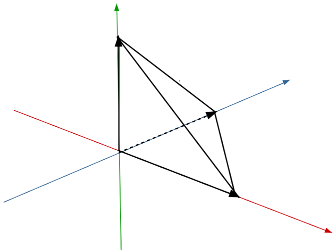


Example of a cone generated by the conic combination of 4 black vectors

Example : $\mathbb{R}^n_+$

$\mathbb{R}^n_+$ is a pointed convex cone, defined as :

$$\mathbb{R}^n_+ = \{x = (x_1, \ldots, x_n)^T \in \mathbb{R}^n \; : \; x_1, \ldots, x_n \geq 0\}$$



The cone $\mathbb{R}^3_+$: a polyhedron.

For the following LP :

$$(LP) \begin{cases} \max c^T x \\ \text{s.t.} \\ Ax \succeq b \end{cases}$$

The partial order $x \succeq y$ means $x_i \geq y_i$ for all $i \in \{1, \ldots, n\}$

Spectral Decomposition Theorem

Every matrix $X \in \mathcal{S}^n$ (the set of symmetric matrices of dimension n) can be decomposed as :

$$X = \sum_{i=1}^{n} \lambda_i u_i u_i^T$$

where $\lambda_i \in \mathbb{R}$ are the **eigenvalues** of X, and $u_i \in \mathbb{R}^n$ the corresponding **eigenvectors**, forming an orthonormal basis of $\mathbb{R}^n$.

In matrix form : $X = PDP^T$, where $D = diag(\lambda)$ is a diagonal matrix whose non-zero elements are the λ_i, and P is the orthogonal matrix whose columns are the u_i.

Principal and Symmetric Submatrices, Minors

- Definition : Let $X \in \mathcal{S}^n$, a **symmetric submatrix** of X is any submatrix obtained by deleting an equal number of rows and columns with the same indices.

  Example : $X = \begin{pmatrix} 3 & 5 & 6 \\ 5 & 1 & 9 \\ 6 & 9 & 2 \end{pmatrix}$, order 2 : $\begin{pmatrix} 3 & 5 \\ 5 & 1 \end{pmatrix}$, $\begin{pmatrix} 3 & 6 \\ 6 & 2 \end{pmatrix}$, $\begin{pmatrix} 1 & 9 \\ 9 & 2 \end{pmatrix}$; order 1 : (3) (1) (2)

- Definition : A **principal submatrix** of X is a symmetric submatrix obtained by deleting the last rows and columns of X.

  Example : $X = \begin{pmatrix} 3 & 5 & 6 \\ 5 & 1 & 9 \\ 6 & 9 & 2 \end{pmatrix}$, order 2 : $\begin{pmatrix} 3 & 5 \\ 5 & 1 \end{pmatrix}$, order 1 : (3)

- Definition : A **symmetric (or principal) minor** is the determinant of a symmetric (or principal) submatrix.

Positive Semi-Definite Matrices

Let $X \in \mathcal{S}^n$, the following statements are equivalent :

1. X is **positive semi-definite (PSD)**, denoted $X \succeq 0$, if
$$\forall x \in \mathbb{R}^n,\ x^T X x = \sum_{i=1}^{n} \sum_{j=1}^{n} X_{ij} x_i x_j = \langle X, xx^T \rangle \geq 0$$
2. The smallest eigenvalue of X is non-negative.
3. All symmetric minors of X are non-negative.
4. Cholesky decomposition : For $k \geq 1$, there exists a lower triangular matrix $L \in \mathbb{R}^{n \times k}$ such that $X = LL^T$

Positive Semi-Definite Matrices : Example

$$X = \begin{pmatrix} 3 & 1 & -1 \\ 1 & 3 & 1 \\ -1 & 1 & 1 \end{pmatrix}$$

2. The eigenvalues of X are $(0, 4, 3)$.
3. Symmetric minors :
   - Order 3 symmetric minor :
     $\begin{vmatrix} 3 & 1 & -1 \\ 1 & 3 & 1 \\ -1 & 1 & 1 \end{vmatrix} = 0 \geq 0$
   - Order 2 symmetric minors :
     $\begin{vmatrix} 3 & 1 \\ 1 & 3 \end{vmatrix} = 8 \geq 0$, $\begin{vmatrix} 3 & -1 \\ -1 & 1 \end{vmatrix} = 2 \geq 0$, $\begin{vmatrix} 3 & 1 \\ 1 & 1 \end{vmatrix} = 2 \geq 0$
   - Order 1 minors : $|3| = 3 \geq 0$, $|3| = 3 \geq 0$, $|1| = 1 \geq 0$

$\Longrightarrow$ The matrix X is positive semi-definite.

Positive Semi-Definite Matrices : Example

$$X = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 2 & 4 \\ 0 & 4 & 8 \end{pmatrix}$$

4. Non-unique Cholesky decompositions :
   - $\begin{pmatrix} 0 & 0 & 0 \\ 0 & 2 & 4 \\ 0 & 4 & 8 \end{pmatrix} = LL^T = \begin{pmatrix} 0 & 0 & 0 \\ 0 & \sqrt{2} & 0 \\ 0 & \frac{4}{\sqrt{2}} & 0 \end{pmatrix} \begin{pmatrix} 0 & 0 & 0 \\ 0 & \sqrt{2} & \frac{4}{\sqrt{2}} \\ 0 & 0 & 0 \end{pmatrix}$
   - $= L'L'^T = \begin{pmatrix} 0 & 0 & 0 \\ 1 & 1 & 0 \\ 2 & 2 & 0 \end{pmatrix} \begin{pmatrix} 0 & 1 & 2 \\ 0 & 1 & 2 \\ 0 & 0 & 0 \end{pmatrix}$

$\Longrightarrow$ The matrix X is positive semi-definite.

Positive Definite Matrices : Example

$$X = \begin{pmatrix} 4 & 1 & -1 \\ 1 & 3 & 1 \\ -1 & 1 & 1 \end{pmatrix}$$

4. Unique Cholesky decomposition :

$$\begin{pmatrix} 4 & 1 & -1 \\ 1 & 3 & 1 \\ -1 & 1 & 1 \end{pmatrix} = LL^T = \begin{pmatrix} 2 & 0 & 0 \\ \frac{1}{2} & \frac{\sqrt{11}}{2} & 0 \\ -\frac{1}{2} & \frac{5}{2\sqrt{11}} & \frac{\sqrt{2}}{\sqrt{11}} \end{pmatrix} \begin{pmatrix} 2 & \frac{1}{2} & -\frac{1}{2} \\ 0 & \frac{\sqrt{11}}{2} & \frac{5}{2\sqrt{11}} \\ 0 & 0 & \frac{\sqrt{2}}{\sqrt{11}} \end{pmatrix}$$

$\Longrightarrow$ The matrix X is positive definite.

Tensor Product of Two Vectors

- Definition : Let $u \in \mathbb{R}^n$, $v \in \mathbb{R}^n$, the tensor product of two vectors is defined as :

$$u \otimes v = uv^T = \begin{pmatrix} u_1 v_1 & \dots & u_1 v_n \\ \vdots & \ddots & \vdots \\ u_n v_1 & \dots & u_n v_n \end{pmatrix}$$

- Property : uv^T is not symmetric, and $u \otimes v$ is not commutative.

Example : Let $u = \begin{pmatrix} 2 \\ 1 \\ 3 \end{pmatrix}$ and $v = \begin{pmatrix} -1 \\ 0 \\ 2 \end{pmatrix}$, we have :

$u \otimes v = \begin{pmatrix} -2 & 0 & 4 \\ -1 & 0 & 2 \\ -3 & 0 & 6 \end{pmatrix}$ and $v \otimes u = \begin{pmatrix} -2 & -1 & -3 \\ 0 & 0 & 0 \\ 4 & 2 & 6 \end{pmatrix}$

Remarks

1. If X is SDP and $X_{ii} = 0$, then $X_{ij} = X_{ji} = 0$ for all $j = 1, \dots, n$
2. If X is a diagonal matrix, it is SDP if all its diagonal elements (its eigenvalues) are nonnegative
3. If, moreover, all diagonal elements are strictly positive, then it is PD.
4. For any $x \in \mathbb{R}^n$, $x \otimes x = xx^T = \begin{pmatrix} x_1^2 & \dots & x_1 x_n \\ \vdots & \ddots & \vdots \\ x_n x_1 & \dots & x_n^2 \end{pmatrix}$ is SDP.

Scalar Product and Trace

- The **trace of a matrix** $X \in \mathbb{R}^{n \times n}$ is $\mathrm{Tr}(X) = \sum_{i=1}^{n} X_{ii} = \sum_{i=1}^{n} \lambda_i$
- Properties
  - $\mathrm{Tr}(\lambda X) = \lambda \mathrm{Tr}(X)$
  - $\mathrm{Tr}(A + B) = \mathrm{Tr}(A) + \mathrm{Tr}(B)$
  - $\mathrm{Tr}(A) = \mathrm{Tr}(A^T)$
  - $\mathrm{Tr}(AB) = \mathrm{Tr}(BA)$
- The **scalar product** is defined as : $\langle A, B \rangle = \mathrm{Tr}(A^T B) = \sum_{i=1}^{n} \sum_{j=1}^{n} A_{ij} B_{ij}$

Block Matrices

- Definition : Let the matrices $X_i \in \mathcal{S}^{n_i}$ for all $i = 1, \ldots, r$. Then $X = X_1 \oplus \ldots \oplus X_r$ is the following **block matrix** :

$$X = X_1 \oplus \ldots \oplus X_r = \begin{pmatrix} X_1 & 0 & \ldots & 0 \\ 0 & X_2 & \ldots & 0 \\ \vdots & & \ddots & \vdots \\ 0 & 0 & \ldots & X_r \end{pmatrix}$$

- Property : $X \succeq 0$ if and only if $X_i \succeq 0$ for all $i = 1, \ldots, r$

The Cone of Semidefinite Matrices

- We denote by $\mathcal{S}_+^n \subset \mathcal{S}^n$ the convex cone of semidefinite positive (SDP) matrices :

$$\mathcal{S}_+^n = \{X \in \mathcal{S}^n : X \text{ is SDP, i.e. } \forall x \in \mathbb{R}^n \ : \ x^T X x \geq 0\}$$

- This is indeed a convex cone because :
  For all $\alpha, \beta > 0$ and $(X, Y) \in (\mathcal{S}_+^n)^2$, we have

$$x^T(\alpha X + \beta Y)x = \alpha x^T X x + \beta x^T Y x \geq 0,$$

  hence $\alpha X + \beta Y \in \mathcal{S}_+^n$.

- Let $X \in \mathcal{S}_+^n$, then $X = \sum_{i=1}^{n} \lambda_i u_i u_i^T$ with $\lambda_i \geq 0$.
  The SDP cone is generated by rank-1 matrices :

$$\mathcal{S}_+^n = \text{ cone } \{xx^T \ : \ x \in \mathbb{R}^n\}$$

- Inside the cone $\mathcal{S}_+^n$ lie the positive definite matrices.

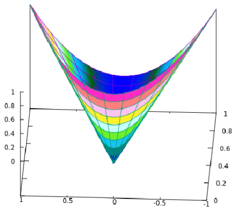
The Dual of the SDP Cone

- The dual of S_+^n is :

$$(S_+^n)^* = \left\{ B \in S^n \; : \; \langle A, B \rangle \geq 0, \, \forall A \in S_+^n \right\}$$

- Theorem : The cone $\mathcal{S}_+^n$ is self-dual : $(\mathcal{S}_+^n)^* = \mathcal{S}_+^n$,
i.e. $A \succeq 0 \Longleftrightarrow \forall B \in \mathcal{S}_+^n \; : \; \langle A, B \rangle \geq 0$ Exercise : Prove it.

Example : The Cone of Matrices $\mathcal{S}_+^2$



$$\begin{pmatrix} x & y \\ y & z \end{pmatrix} \succeq 0$$
$$\iff x \geq 0, z \geq 0, xz - y^2 \geq 0$$

The cone of matrices $X \in \mathcal{S}_+^2$

Conic Terminology

Both the non-negative orthant

$$\mathcal{N} = \{\mathrm{x} \in \mathbb{R}^n : x_i \geq 0 \text{ for all } i\}$$

and the set of symmetric positive semidefinite (psd) matrices

$$\mathcal{P} = \{\mathrm{X} \in \mathcal{S}^n : \lambda_i(\mathrm{X}) \geq 0 \text{ for all } i\}$$

are pointed closed convex cones.

A set $\mathcal{K}$ is

- a *cone* if $x \in \mathcal{K}$, then $\alpha x \in \mathcal{K}$ for all $\alpha \geq 0$
- *convex* if $\overline{x}, \overline{y} \in \mathcal{K}$, then $\alpha\overline{x} + (1-\alpha)\overline{y} \in \mathcal{K}$ for all $\alpha \in (0,1)$
- *closed* if it contains its boundary
- *pointed / proper* if $\mathcal{K} \cap -\mathcal{K} = \{0\}$

Interior and Boundary of $\mathcal{P}$

The interior of $\mathcal{P}$ is the set of positive definite (pd) matrices :

$$\{X \in \mathcal{S}^n : \lambda_i(X) > 0 \text{ for all } i\}$$

and the boundary of $\mathcal{P}$ are the singular psd matrices.

Algebraic Characterizations

There are many ways to express the psd (pd) condition on a matrix X.
A few of them are :

$$X \succeq 0 \iff \lambda_i(X) \geq 0 \text{ for all } i \qquad (\lambda_i(X) > 0)$$
$$\iff v^T X v \geq 0 \text{ for all } v \in \mathbb{R}^n \qquad (v^T X v > 0)$$

$$\iff \exists\ X^{\frac{1}{2}} \in \mathcal{S}^n \text{ s.t. } X^{\frac{1}{2}} X^{\frac{1}{2}} = X \qquad (\&\ X^{\frac{1}{2}} \text{ is invertible})$$

$$\iff \exists\ w_1, w_2, \ldots, w_n \text{ s.t. } X_{ij} = w_i^T w_j \qquad (\&\ w_1, \ldots, w_n \text{ lin. indep.})$$

$$\text{If } X_1 \succ 0 \text{ then } \begin{bmatrix} X_1 & X_2 \\ X_2^T & X_3 \end{bmatrix} \succeq 0 \qquad (\succ 0)$$
$$\iff X_3 - X_2^T X_1^{-1} X_2 \succeq 0 \qquad (X_3 - X_2^T X_1^{-1} X_2 \succ 0)$$

Sufficient (but not necessary) condition for psd (pd) :

$$X_{ii} \geq \sum_{j \neq i} |X_{ij}| \text{ for all } i \Rightarrow X \succeq 0 \qquad (X_{ii} > \sum_{j \neq i} |X_{ij}| \Rightarrow X \succ 0)$$

The Second-Order Cone (or Lorentz Cone)

An $(n+1)$-dimensional second-order cone (SOC) is the set of all vectors $(x_0, x_1, \ldots, x_n)$ that satisfy $x_0 \geq \sqrt{x_1^2 + \ldots + x_n^2}$,
or equivalently

$$SOC = \{\mathrm{x} \in \mathbb{R}^{n+1} : x_0^2 - x_1^2 - \ldots - x_n^2 \geq 0, x_0 \geq 0\}.$$

The SOC is also a pointed closed convex cone, and second-order cone programming (SOCP) consists of optimizing a linear function subject to linear equality constraints and one or more SOC constraints.

Relationships Between $\mathcal{N}$, $\mathcal{P}$, and the SOC

The SOC constraint

$$x_0^2 - x_1^2 - \ldots - x_n^2 \geq 0, x_0 \geq 0$$

is equivalent to the positive semidefinite constraint

$$\begin{pmatrix} x_0 & & & & x_1 \\ & x_0 & & & x_2 \\ & & x_0 & & x_3 \\ & & & \ddots & \vdots \\ x_1 & x_2 & x_3 & \cdots & x_0 \end{pmatrix} \succeq 0$$

Furthermore, the k-dimensional non-negative orthant is the direct (Cartesian) product of k 1-dimensional SOC cones.

Hence, SOCP is a special case of SDP ;
and LP is a special case of SOCP.

A Fundamental Structure : The Elliptope

The $\binom{n}{2}$-dimensional elliptope (or spectrahedron) is the feasible set of the SDP problem

$$\begin{array}{ll} \min & \langle \mathrm{C}, \mathrm{X} \rangle \\ \text{s.t.} & \mathrm{diag}(\mathrm{X}) = \mathrm{e} \\ & \mathrm{X} \succeq 0. \end{array}$$

where diag(X) denotes a vector with the diagonal elements of X, and e is the vector of all ones.

In other words, the elliptope of dimension $\binom{n}{2}$ is the set of all symmetric $n \times n$ matrices that are psd and have ones on the diagonal.

This special set comes up in many applications of SDP.

Small Elliptopes

If $X \in \mathcal{S}^2$, we obtain the elliptope in $\mathbb{R}^1$:

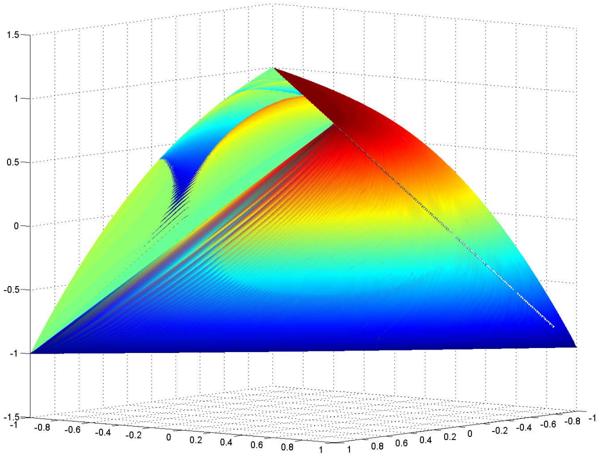
$$\begin{pmatrix} 1 & x \\ x & 1 \end{pmatrix} \succeq 0 \Rightarrow x \in [-1, 1].$$

If $X \in \mathcal{S}^3$, we obtain the elliptope in $\mathbb{R}^3$:

$$\left\{ \begin{pmatrix} x \\ y \\ z \end{pmatrix} \in \mathbb{R}^3 : \begin{pmatrix} 1 & x & y \\ x & 1 & z \\ y & z & 1 \end{pmatrix} \succeq 0 \right\}.$$

We can visualize this set in $\mathbb{R}^3$.

The Elliptope in 3 Dimensions



Geometry of the Elliptope

The vertices of the elliptope correspond to the psd matrices with all entries equal to ± 1.

For $n = 2$, we have two vertices :

$$\begin{pmatrix} 1 & 1 \\ 1 & 1 \end{pmatrix} \quad \text{and} \quad \begin{pmatrix} 1 & -1 \\ -1 & 1 \end{pmatrix}.$$

For $n = 3$, there are four vertices :

$$\begin{pmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{pmatrix}, \begin{pmatrix} 1 & 1 & -1 \\ 1 & 1 & -1 \\ -1 & -1 & 1 \end{pmatrix}, \begin{pmatrix} 1 & -1 & 1 \\ -1 & 1 & -1 \\ 1 & -1 & 1 \end{pmatrix}, \begin{pmatrix} 1 & -1 & -1 \\ -1 & 1 & 1 \\ -1 & 1 & 1 \end{pmatrix}.$$

Geometry of the Elliptope

Unlike a polyhedron, the elliptope has extreme points that are *not* vertices.

This occurs first when $n = 3$: the matrix

$$\begin{pmatrix} 1 & \frac{1}{2} & \frac{1}{2} \\ \frac{1}{2} & 1 & \frac{1}{2} \\ \frac{1}{2} & \frac{1}{2} & 1 \end{pmatrix}$$

is not a vertex, but it is an extreme point of the elliptope since it cannot be expressed as a convex combination of the four vertices.

The Dual Cone

For any convex cone $\mathcal{K}$, the dual cone $\mathcal{K}^*$ is defined as

$$\mathcal{K}^* := \{y \in \mathbb{R}^n : \langle x, y \rangle \geq 0 \quad \forall x \in \mathcal{K}\}.$$

We have that :

- For any cone $\mathcal{K}$, $\mathcal{K}^*$ is a closed convex cone ;
- The non-negative orthant is self-dual (obvious) ;
- The SOC is self-dual, by the Cauchy-Schwarz inequality ;
- The psd cone is self-dual, by Fejer's Theorem :

$$X \succeq 0 \Leftrightarrow X \bullet Z \geq 0 \text{ for all } Z \succeq 0$$

Other Important Examples of Cones of Matrices

- The completely positive cone :

$$\mathcal{C} := \{\mathrm{X} \in \mathcal{S}^n : \mathrm{X} = \sum_{i=1}^{k} \mathrm{v}_i \mathrm{v}_i^T, \mathrm{v}_i \geq 0\}$$

- The copositive cone :

$$\mathcal{C}^* := \{\mathrm{X} \in \mathcal{S}^n : \mathrm{v}^T \mathrm{X} \mathrm{v} \geq 0 \text{ for all } \mathrm{v} \geq 0\}$$

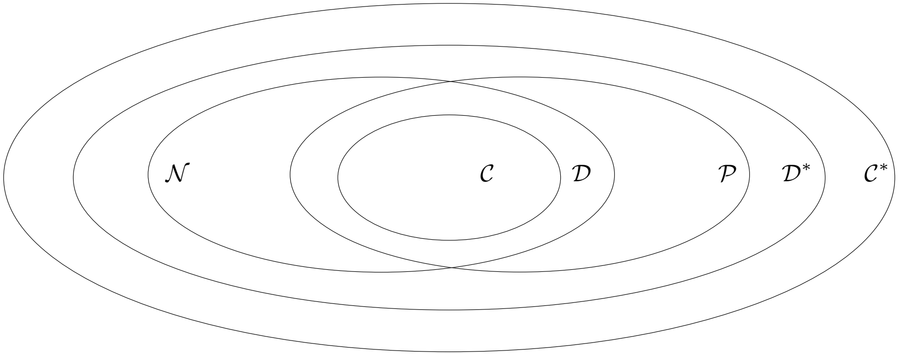
- The doubly non-negative cone :

$$\mathcal{D} := \mathcal{P} \cap \mathcal{N}$$

- and its dual

$$\mathcal{D}^* = \mathcal{P} \oplus \mathcal{N}$$

Relationships Between the Cones for $n \geq 5$



Conic Optimization Duality

Consider the general conic optimization problem

$$\begin{array}{ll} \min & \langle c, x \rangle \\ \text{s.t.} & \langle a_i, x \rangle = b_i, \;\; i = 1, \ldots, m \quad \leftarrow y_i \\ & x \in \mathcal{K} \end{array}$$

The Lagrangian dual is :

$$\max_y \left\{ \min_{x \in \mathcal{K}} \langle c, x \rangle + \sum_{i=1}^{m} y_i \left(b_i - \langle a_i, x \rangle \right) \right\} = \max_y \left\{ \sum_{i=1}^{m} b_i y_i + \min_{x \in \mathcal{K}} \langle c - \sum_{i=1}^{m} y_i a_i, x \rangle \right\}$$

The inner minimization is unbounded below unless

$$\langle c - \sum_{i=1}^{m} y_i a_i, x \rangle \geq 0 \text{ for all } x \in \mathcal{K}$$

in which case the minimum is zero.

Conic Optimization Duality (ctd)

Since the outer problem is a maximization problem, it is therefore equivalent to

$$\begin{aligned} \max \quad & \sum_{i=1}^{m} b_i y_i \\ \text{s.t.} \quad & \mathrm{c} - \sum_{i=1}^{m} y_i \mathrm{a}_i \in \mathcal{K}^* \end{aligned}$$

which is equivalent to

$$\begin{aligned} \max \quad & \sum_{i=1}^{m} b_i y_i \\ \text{s.t.} \quad & \sum_{i=1}^{m} y_i \mathrm{a}_i + \mathrm{z} = \mathrm{c} \\ & \mathrm{z} \in \mathcal{K}^* \end{aligned}$$

This is the dual cone optimization problem.

Why Focus on Semidefinite Optimization ?

Convexity is of paramount importance in optimization.

Convex optimization problems have many of the advantageous properties of LP, including :

- an elegant and powerful duality theory, and
- polynomial-time solvability using interior-point methods (IPMs) – but with a major caveat.

Semidefinite Programming

- **Semidefinite Programming (SDP)** is an extension of Linear Programming (LP).
- SDP was first studied in the 1960s.
- An SDP problem is a program with a linear objective function in terms of the variable matrix X and constraints that are also linear in terms of this matrix.

Linear Matrix Inequalities (LMI)

Definition : Let $A_i \in \mathcal{S}^n$ and $x \in \mathbb{R}^n$, a **Linear Matrix Inequality (LMI)** is of the form : $g(x) = \sum_{i=1}^{m} x_i A_i \succeq A_0$

Example : Let x_1 and x_2 be two real numbers in $\mathbb{R}$, and consider the following matrix inequality :

$$x_1 \begin{pmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} + x_2 \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix} \succeq \begin{pmatrix} -1 & 0 & -1 \\ 0 & -2 & 0 \\ -1 & 0 & 0 \end{pmatrix}$$

is equivalent to : $\begin{pmatrix} x_2 + 1 & x_1 & 1 \\ x_1 & 2 & 0 \\ 1 & 0 & x_2 \end{pmatrix} \succeq 0$

or, equivalently, to an infinite set of inequalities : $\forall v \in \mathbb{R}^3$,

$$v^T \begin{pmatrix} x_2 + 1 & x_1 & 1 \\ x_1 & 2 & 0 \\ 1 & 0 & x_2 \end{pmatrix} v \geq 0$$

Linear Matrix Inequalities (LMI)

$$g(x) = \sum_{i=1}^{m} x_i A_i - A_0 \succeq 0$$

$g(x)$ defines a convex set, which is the set $\mathcal{C}$:

$$\mathcal{C} = \{x \in \mathbb{R}^n \ : \ g(x) \succeq 0\}$$

$\mathcal{C}$ lies at the intersection of an affine space and the cone of positive semidefinite matrices.

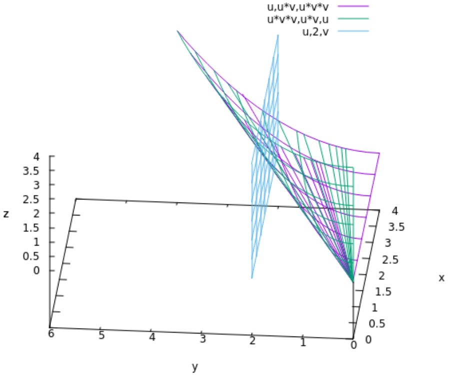
$\mathcal{C}$ is generally not a polytope for $n \geq 2$.

Linear Matrix Inequalities (LMI) : Example

In $\mathcal{S}_+^2$, we look for matrices $\begin{pmatrix} x & y \\ y & z \end{pmatrix} \succeq 0$ such that $y = 2$, i.e. :

$$\begin{pmatrix} x & 2 \\ 2 & z \end{pmatrix} \succeq 0 \iff x \geq 0, z \geq 0, xz \geq 4$$

These matrices lie at the intersection of the cone $\mathcal{S}_+^2$ and the hyperplane $y = 2$.



The Primal Semidefinite Problem

Let $Q, A_1, \ldots, A_m \in \mathcal{S}^n$, and $b \in \mathbb{R}^m$, the primal (SDP) is written :

$$(SDP) \begin{cases} \max f(X) = \langle Q, X \rangle \\ \text{s.t.} \\ \quad g_r(X) = \langle A_r, X \rangle = b_r \quad \forall r = 1, \ldots, m \\ \quad X \succeq 0 \end{cases}$$

The constraints $g_r(x)$ define the structure of the matrix X.

Reminder : $\forall A, B \in \mathcal{S}^n$, $\langle A, B \rangle = \sum_{i=1}^{n} \sum_{j=1}^{n} a_{ij} b_{ij}$

Example : A Primal SDP Problem (1/2)

$$
(Ex_1)\begin{cases}
\min \langle \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}, X \rangle \\
\text{s.t.} \\
\quad \langle \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}, X \rangle = 2 \\
\quad X \succeq 0
\end{cases}
\qquad
(Ex_1)\begin{cases}
\min X_{11} + X_{22} \\
\text{s.t.} \\
\quad X_{12} + X_{21} = 2 \\
\quad X \succeq 0
\end{cases}
$$

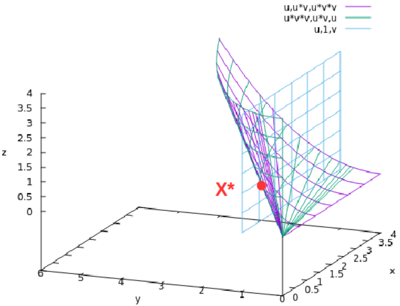
Which is equivalent to the problem :

$$
(Ex_1)\begin{cases}
\min X_{11} + X_{22} \\
\text{s.t.} \\
\quad \begin{pmatrix} X_{11} & 1 \\ 1 & X_{22} \end{pmatrix} \succeq 0
\end{cases}
$$

Example : A Primal SDP Problem (2/2)

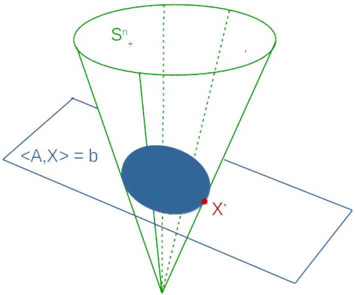
$$
(Ex_1)\begin{cases} \min X_{11} + X_{22} \\ \text{s.t.} \\ \qquad \begin{pmatrix} X_{11} & 1 \\ 1 & X_{22} \end{pmatrix} \succeq 0 \end{cases}
$$

The optimal solution is the matrix $X^* = \begin{pmatrix} 1 & 1 \\ 1 & 1 \end{pmatrix}$ with value 2.



The Primal SDP Problem : Geometry

The optimal matrix X^* lies at the intersection of the hyperplane $\langle A, X \rangle = b$ and the cone $\mathcal{S}^n_+$ of SDP matrices.



Example of a feasible region for an SDP

Primal Problem Not Necessarily Bounded

- The SDP problem may be unbounded.
- Example : Let $\alpha \in \mathbb{R}$ be a scalar, and consider the family of problems (Ex^α)

$$(Ex^\alpha)\begin{cases} \max X_{11} \\ \text{s.t.} \\ \qquad \begin{pmatrix} X_{11} & \alpha \\ \alpha & 0 \end{pmatrix} \succeq 0 \end{cases}$$

  - $\det(X) = -\alpha^2$ and so the feasible region of (Ex^α) is empty if $\alpha \neq 0$.
  - When $\alpha = 0$, (Ex^α) admits a feasible solution, but no strictly feasible solution.

  The optimal value is $+\infty$; (Ex^α) is unbounded.

- In the case of a minimization of the objective function, the optimal value is 0 with the solution $X = \begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix}$.

Optimum Not Necessarily Attained

- The optimal value of an SDP problem may not be attained.
- Example

$$(SDP)\begin{cases} \min X_{11} \\ \text{s.t.} \\ \quad \begin{pmatrix} X_{11} & 1 \\ 1 & X_{22} \end{pmatrix} \succeq 0 \end{cases}$$

The optimal value is 0, achieved when $X_{11} = \frac{1}{X_{22}}$ with $X_{22} \to +\infty$, and thus it is never attained.

SDP Programming and Linear Programming

Consider a SDP problem where :

- $Q, A_1, \ldots, A_m \in \mathcal{S}^n$ are diagonal matrices, whose diagonal elements are the vectors q and a_r $(r = 1, \ldots, m)$, and $b \in \mathbb{R}^m$,
- the variable matrix X is diagonal, and its diagonal elements are the entries of the vector x

The SDP problem can be written as

$$(SDP_d)\begin{cases} \max\langle q, x\rangle \\ \text{s.t.} \\ \quad \langle a_r, x\rangle = b_r \\ \quad x \succeq 0 \end{cases} \qquad \Longleftrightarrow \qquad (LP)\begin{cases} \max q^T x \\ \text{s.t.} \\ \quad a_r^T x = b_r \\ \quad x \geq 0 \end{cases}$$

Therefore, SDP programming includes linear programming.

SDP Duality

Consider a primal-dual SDP pair in the following form :

$\min \quad \langle \mathrm{C}, \mathrm{X} \rangle$	$\max \quad \langle \mathrm{b}, \mathrm{y} \rangle$
$\text{s.t.} \quad \langle \mathrm{A}_i, \mathrm{X} \rangle = b_i, \; i = 1, \ldots, m$	$\text{s.t.} \quad \sum_{i=1}^{m} y_i \mathrm{A}_i + \mathrm{Z} = \mathrm{C}$
$\mathrm{X} \succeq 0$	$\mathrm{Z} \succeq 0$

Like in LP, we have a weak duality theorem.

Theorem

If $\tilde{\mathrm{X}}$ is primal feasible and $(\tilde{\mathrm{y}}, \tilde{\mathrm{Z}})$ is dual feasible then $\langle \mathrm{C}, \tilde{\mathrm{X}} \rangle \geq \langle \mathrm{b}, \tilde{\mathrm{y}} \rangle$.

The proof is just like for LP :

$$\langle \mathrm{C}, \tilde{\mathrm{X}} \rangle - \langle \mathrm{b}, \tilde{\mathrm{y}} \rangle = \langle \tilde{\mathrm{Z}}, \tilde{\mathrm{X}} \rangle + \sum_{i=1}^{m} \tilde{y}_i \langle \mathrm{A}_i, \tilde{\mathrm{X}} \rangle - \sum_{i=1}^{m} \tilde{y}_i \langle \mathrm{A}_i, \tilde{\mathrm{X}} \rangle = \langle \tilde{\mathrm{Z}}, \tilde{\mathrm{X}} \rangle \geq 0.$$

The difference between the primal and dual objective values for feasible solutions $\tilde{\mathrm{X}}$ and $(\tilde{\mathrm{y}}, \tilde{\mathrm{Z}})$ is called the duality gap.

Beyond weak duality, however, the picture differs. For example, for LP,

- if the primal is feasible and bounded, or
- if the dual is feasible and bounded,

then both primal and dual have optimal solutions, and the duality gap is zero at optimality.

For SDP, the situation is more complicated, as the following two examples demonstrate.

Zero Duality Gap Without Attainment

Consider the SDP problem

$$\begin{array}{ll} \inf & X_{11} \\ \text{s.t.} & \begin{pmatrix} X_{11} & 1 \\ 1 & X_{22} \end{pmatrix} \succeq 0. \end{array}$$

It is feasible and bounded yet the optimal value zero cannot be attained because $\begin{pmatrix} 0 & 1 \\ 1 & X_{22} \end{pmatrix}$ is not psd for any value of X_{22}.

The dual problem is

$$\begin{array}{ll} \max & y_1 \\ \text{s.t.} & \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix} - y_1 \begin{pmatrix} 0 & \frac{1}{2} \\ \frac{1}{2} & 0 \end{pmatrix} \succeq 0 \end{array}$$

or equivalently

$$\begin{array}{ll} \max & y_1 \\ \text{s.t.} & \begin{pmatrix} 1 & -\frac{y_1}{2} \\ -\frac{y_1}{2} & 0 \end{pmatrix} \succeq 0 \end{array}$$

with $y_1^* = 0$ optimal. (In fact, $y_1 = 0$ is the only feasible solution.)

Positive Duality Gap

The SDP problem

$$
\begin{array}{ll}
\min & X_{11} \\
\text{s.t.} & X_{11} + 2X_{23} = 1 \\
& X_{22} = 0 \\
& \begin{pmatrix} X_{11} & X_{12} & X_{13} \\ X_{12} & X_{22} & X_{23} \\ X_{13} & X_{23} & X_{33} \end{pmatrix} \succeq 0
\end{array}
$$

has optimal value 1.
The dual SDP problem is

$$
\begin{array}{ll}
\max & y_1 \\
\text{s.t.} & \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} - y_1 \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & \frac{1}{2} \\ 0 & \frac{1}{2} & 0 \end{pmatrix} - y_2 \begin{pmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{pmatrix} = \begin{pmatrix} 1 - y_1 & 0 & 0 \\ 0 & -y_2 & -\frac{y_1}{2} \\ 0 & \frac{y_1}{2} & 0 \end{pmatrix}
\end{array}
$$

and the psd constraint implies $y_1 = 0$ for every feasible solution, hence the optimal value is 0. (Take e.g. $y_1^* = 0, y_2^* = 0$.)

Constraint Qualification

To avoid this kind of difficulty and obtain a strong duality result for SDP, we must require that the primal-dual SDP pair satisfy a *constraint qualification (CQ)*.

This is a standard concept in non-linear optimization. Arguably the most commonly used CQ is Slater's CQ.

Definition

Slater's CQ holds if both primal and dual have feasible positive definite matrices.

We then have the following result.

Theorem

Under Slater's CQ, both primal and dual have optimal solutions, and the duality gap is zero at optimality.

Verifying Slater's CQ

Example

$$\begin{array}{ll|ll} \min & \langle C, X\rangle & \max & \langle e, y\rangle \\ s.t. & \langle e_i e_i^T, X\rangle = 1,\ i = 1,\dots,m & s.t. & \sum_{i=1}^{m} y_i e_i e_i^T + Z = C \\ & X \succeq 0 & & Z \succeq 0 \end{array}$$

Optimality Conditions

From the weak duality of SDP, we have that the duality gap equals

$$\langle \mathrm{C}, \tilde{\mathrm{X}} \rangle - \mathrm{b}^T \tilde{\mathrm{y}} = \langle \tilde{\mathrm{Z}}, \tilde{\mathrm{X}} \rangle \geq 0.$$

Since both X and Z are psd, $\langle \mathrm{X}, \mathrm{Z} \rangle = 0$ implies $\mathrm{XZ} = \mathrm{ZX} = 0$, thus we obtain the sufficient optimality conditions :

$$\begin{array}{ll} \langle \mathrm{A}_i, \mathrm{X} \rangle = b_i, i = 1, \ldots, m, & \text{(primal feasibility)} \\ \mathrm{X} \succeq 0 & \\ \mathrm{Z} + \sum_{i=1}^{m} y_i \mathrm{A}_i = \mathrm{C} & \text{(dual feasibility)} \\ \mathrm{Z} \succeq 0 & \\ \mathrm{XZ} = 0 & \text{(complementarity)} \end{array}$$

If Slater's CQ holds, they are also necessary for optimality.
These optimality conditions can be used as the starting point for defining IPMs to solve SDP problems.

The Dual SDP Problem

Let $\beta \in \mathbb{R}^m$ be the dual variables of (SDP), the dual $(DSDP)$ is :

$$(SDP)\begin{cases} \max f(X) = \langle Q, X\rangle \\ \text{s.t.} \\ \quad g_r(X) = \langle A_r, X\rangle = b_r \leftarrow \beta_r \\ \quad X \succeq 0 \end{cases} \qquad (DSDP)\begin{cases} \min h(\beta) = \langle b, \beta\rangle \\ \text{s.t.} \\ \quad \sum_{r=1}^{m} \beta_r A_r - S = Q \\ \quad \beta \in \mathbb{R}^m \\ \quad S \succeq 0 \end{cases}$$

Remarks :

- $\sum_{r=1}^{m} \beta_r A_r - Q$ must belong to $\mathcal{S}_+^n$
- The matrix S plays the role of a slack variable here.

Example : The Dual SDP Problem

$$(Ex_1)\begin{cases} \min X_{11} + X_{22} \\ \text{s.t.} \\ \quad \begin{pmatrix} X_{11} & 1 \\ 1 & X_{22} \end{pmatrix} \succeq 0 \end{cases}$$

$$(Ex_1)\begin{cases} \max -\langle \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}, X \rangle \\ \text{s.t.} \\ \quad \langle \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}, X \rangle = 2 \leftarrow \beta \\ \quad X \succeq 0 \end{cases}$$

$$(DEx_1)\begin{cases} \min 2\beta \\ \text{s.t.} \\ \quad \begin{pmatrix} 1 & \beta \\ \beta & 1 \end{pmatrix} \succeq 0 \\ \quad \beta \in \mathbb{R}^2 \end{cases}$$

Dual of an SDP, Relation with the Lagrangian

$$(SDP) : \max_X \{\langle Q, X\rangle : \langle A_r, X\rangle = b_r, r = 1, \ldots, m\,,\; X \succeq 0\}$$

For each constraint, we introduce β_r ; its Lagrangian is :

$$\mathcal{L}(X, \beta) = \langle Q, X\rangle - \sum_{r=1}^{m} \beta_r(\langle A_r, X\rangle - b_r) = \langle b, \beta\rangle - \langle \sum_{r=1}^{m} \beta_r A_r - Q, X\rangle$$

Its dual function is : $L(\beta) = \max_{X \succeq 0} \langle b, \beta\rangle - \langle \sum_{r=1}^{m} \beta_r A_r - Q, X\rangle$

And the Lagrangian dual is :

$(DL) = \min_{\beta\in\mathbb{R}^m} \max_{X\succeq 0} \langle b, \beta\rangle - \langle \sum_{r=1}^{m} \beta_r A_r - Q, X\rangle$

Now,

$$\max_{X\succeq 0} \mathcal{L}(X, \beta) = \begin{cases} \langle b, \beta\rangle & \text{if } (\sum_{r=1}^{m} \beta_r A_r - Q) \succeq 0 \\ \infty & \text{otherwise} \end{cases}$$

Therefore : $(DL) = \min_{\beta\in\mathbb{R}^m} \{\langle b, \beta\rangle : (\sum_{r=1}^{m} \beta_r A_r - Q) \succeq 0\}$, which is exactly $(DSDP)$

Feasibility Conditions

Primal Feasibility

We say that $X \in \mathbb{S}_+^n$ is primal feasible if

$$\langle A_r, X \rangle = b_r, \quad \forall r = 1, \ldots, m$$

i.e., if it satisfies all the linear equality constraints.

Dual Feasibility

We say that $(\beta, S) \in \mathbb{R}^m \times \mathbb{S}_+^n$ is dual feasible if

$$\sum_{r=1}^{m} \beta_r A_r - S = Q$$

i.e., the slack variable S must be positive semidefinite.

Duality Gap

Definition

Let X be primal feasible and (β, S) be dual feasible. Then the difference between their objective values is :

$$\langle b, \beta \rangle - \langle Q, X \rangle$$

Duality Gap

Definition

Let X be primal feasible and (β, S) be dual feasible. Then the difference between their objective values is :

$$\langle b, \beta \rangle - \langle Q, X \rangle$$

Theorem : Weak Duality

If X is primal feasible and (β, S) is dual feasible, then

$$\langle Q, X \rangle \leq \langle b, \beta \rangle$$

i.e., the dual objective is always an upper bound on the primal objective.

Slater's Condition

Slater's Condition for the Primal

There exists $X \succ 0$ (positive definite) such that $\langle A_r, X\rangle = b_r$, for all r.
This ensures that the feasible region of the primal has a non-empty interior.

Slater's Condition for the Dual

There exists (β, S) such that $S \succ 0$ and $\sum_{r=1}^{m} \beta_r A_r - S = Q$.
This ensures the dual feasible region has a non-empty interior.

Slater's Condition

Slater's Condition for the Primal

There exists $X \succ 0$ (positive definite) such that $\langle A_r, X \rangle = b_r$, for all r.
This ensures that the feasible region of the primal has a non-empty interior.

Slater's Condition for the Dual

There exists (β, S) such that $S \succ 0$ and $\sum_{r=1}^{m} \beta_r A_r - S = Q$.
This ensures the dual feasible region has a non-empty interior.

Theorem : Strong Duality

If Slater's condition holds for either the primal or the dual, then strong duality holds, i.e., there is no duality gap and both problems attain their optimum.

Optimality Conditions

KKT Conditions (Karush-Kuhn-Tucker)

Let $X \in \mathbb{S}^n$, $S \in \mathbb{S}^n_+$, and $\beta \in \mathbb{R}^m$. Then (X, β, S) is optimal for (SDP) and (DSDP) if and only if :

- **Primal feasibility :** $\langle A_r, X \rangle = b_r$, for all $r = 1, \ldots, m$
- **Dual feasibility :** $\sum_{r=1}^m \beta_r A_r - S = Q$
- **Complementary slackness :** $\langle S, X \rangle = 0$

Solving SDP Problems

Numerous algorithms have been proposed (and, in most cases, implemented) for solving SDP problems :

- Ellipsoid method
- Interior-point methods (IPMs)
- Spectral bundle method
- Low-rank method
- Augmented Lagrangian methods
- Semi-infinite LP methods
- Boundary point method
- and more...

Solving SDP problems

Numerous algorithms have been proposed (and, in most cases, implemented) for solving SDP problems :

- Ellipsoid method
- Interior-point methods (IPMs)
- Spectral bundle method
- Low-rank method
- Augmented Lagrangian methods
- Semi-infinite LP methods
- Boundary point method
- and more...

The first two are the only ones with provable polynomial-time convergence.
IPMs are the only ones that are polynomial-time *and* efficient in practice.

Which algorithms to solve (*SDP*)

- **Interior-point methods** (implementations : CSDP, SeDuMi, SDPA). In general, they are not able to solve instances of dimension greater than 1000 or with more than 10,000 constraints.
- **Spectral bundle methods** based on eigenvalue optimization (implementations : Spectral Bundle).
  The idea is to exploit the fact that $X \succeq 0 \iff \lambda_{\min}(X) \geq 0$.
- **Bundle methods** (e.g., callable via the Conic Bundle library). Can solve significantly larger instances, but must be properly tuned, as convergence can be very slow.
- Other classes of methods such as **augmented Lagrangian algorithms**.

Interior-point algorithms

Self-Concordant Barrier Functions

Use of an IPM requires a self-concordant barrier function for the cone underlying the feasible set.

Although such a function (the Universal Barrier Function) exists for a large variety of convex cones, it is very hard to compute in general.

However, efficient self-concordant barriers exist for symmetric cones.

Symmetric Cones

Symmetric cones arise from direct products of the following five types of cones :

- SOC
- symmetric psd matrices over the reals (psd cone)
- Hermitian psd matrices over the complex numbers (can be expressed as a psd cone of 2 times the size) ;
- Hermitian psd matrices over the quaternions (can be expressed as a psd cone of 4 times the size) ;
- One exceptional 27-dimensional cone (3×3 Hermitian psd matrices over the octonions).

Thus, SDP is (basically) the most general class of symmetric cones, and these are the cones over which we know how to optimize in polynomial time.

Interior-point methods - Principle

- Methods developed in the 1990s, with several variants available.
- **Primal-dual method** : We assume that both the primal (*SDP*) and the dual (*DSDP*) satisfy Slater's conditions (zero duality gap).
- **Principle :** follow a central path inside the feasible region, obtained by replacing the optimality conditions with approximate optimality conditions.

Interior-point methods - Principle

- Idea : get rid of the difficult constraint $X \succeq 0$
- Method : penalize the objective function with a barrier function that tends to 0 inside the cone $\mathcal{S}_+^n$ and to infinity outside.

  $\Longrightarrow$ If we start with a solution $X \succ 0$, we can remove the constraint $X \succeq 0$, as it will be enforced by the barrier function (we remain in the interior of the cone $\mathcal{S}_+^n$).

Interior-point methods - Auxiliary problem

Let the barrier function be : $B(X) = \begin{cases} \log\left(\det(X)\right) \text{ if } X \succ 0 \\ -\infty \text{ otherwise} \end{cases}$

Remark : For $X \succ 0$, we have $\det(X) = \prod_{i=1}^{n} \lambda_i > 0$.

For $\mu \geq 0$, we define the following auxiliary problem :

$$(SDP_\mu) \begin{cases} \max f_\mu(X) = \langle Q, X \rangle + \mu \log\left(\det(X)\right) \\ \text{s.t.} \\ \quad g_r(X) = \langle A_r, X \rangle - b_r = 0 \qquad \forall r = 1, \dots, m \\ \quad X \succ 0 \end{cases}$$

Lemma : The function $X \mapsto \log\left(\det(X)\right)$ is strictly concave in the interior of $\mathcal{S}_+^n$, i.e., when $X \succ 0$.
$\Longrightarrow$ For fixed $\mu \geq 0$, the optimal solution X_μ^* of (SDP_μ) is unique.

Solving (SDP) via the auxiliary problem

As $\mu \to 0$, we want $\langle Q, X_\mu^* \rangle$ to converge to the optimal value of (SDP).

We will :

1. Write the necessary conditions for the existence of X_μ^*, in the form of a system that X_μ^* must satisfy. These conditions will also imply the convergence of $\langle Q, X_\mu^* \rangle$ to the optimal value of (SDP).

2. Observe that the necessary conditions are also sufficient : the system derived in (i) has a unique solution $X = X_\mu^*$.

$\forall g_r(X) = 0$, we associate a Lagrange multiplier $\beta_r \in \mathbb{R}$.

We then consider the Lagrangian :

$$\mathcal{L}_\mu(X, \beta) = \langle Q, X \rangle + \mu \log\left(\det(X)\right) - \sum_{r=1}^{m} \langle \beta_r, g_r(X) \rangle$$

Lagrange Theorem : X_μ^* is a (local) maximum if :

1. (i) $g_r(X_\mu^*) = 0$ for all r
2. (ii) $\exists! \beta^* \in \mathbb{R}^m$ such that $\nabla \mathcal{L}_\mu(X_\mu^*, \beta^*) = \nabla f_\mu(X_\mu^*) + \sum_{r=1}^{m} \beta_r^* \nabla g_r(X_\mu^*) = 0$

This applies if :

1. f_μ and g_r are defined on an open subset of $\mathbb{R}$
2. f_μ and g_r are differentiable
3. the $\nabla g_r(X)$ are linearly independent (which holds since g_r are linear functions).

Step (i) : Necessary conditions for existence of X_μ^*

$$(SDP_\mu)\begin{cases} \max f_\mu(X) = \langle Q, X\rangle + \mu \log(\det(X)) \\ \text{s.t.} \\ \quad g_r(X) = \langle A_r, X\rangle - b_r = 0 & \forall r = 1,\ldots,m \\ \quad X \succ 0 \end{cases}$$

Property : $\nabla \log(\det(X)) = (X^T)^{-1}$

Let's compute $\nabla \mathcal{L}_\mu(X,\beta) = \nabla f_\mu(X) + \sum_{r=1}^{m} \beta_r \nabla g_r(X)$

- We have $\nabla \log(\det(X)) = (X^T)^{-1}$ and $\nabla\langle Q, X\rangle = \nabla \mathrm{Tr}(Q^T X) = Q^T$
- Thus, $\nabla f_\mu(X) = Q^T + \mu(X^T)^{-1}$
- And $\nabla g_r(X) = \nabla(\langle A_r, X\rangle - b_r) = \nabla(\mathrm{Tr}(A_r^T X) - b_r) = A_r^T$

Therefore, we get $\nabla \mathcal{L}_\mu(X,\beta) = Q^T + \mu(X^T)^{-1} - \sum_{r=1}^{m} \beta_r A_r^T = 0$

Optimality Conditions for (SDP_μ)

By introducing the slack variable $S = \sum_{r=1}^{m} \beta_r A_r^T - Q^T = \mu (X^T)^{-1}$, we obtain the following optimality conditions :

$$(OPT_\mu) \begin{cases} g_r(X) = \langle A_r, X \rangle = b_r & \forall r,\ X \succ 0 \text{ (primal feasibility)} \\ \sum_{r=1}^{m} \beta_r A_r - S = Q & \beta \in \mathbb{R}^m, S \succ 0 \text{ (dual feasibility)} \\ SX = \mu I_n & \text{complementarity of } X \text{ and } S \end{cases}$$

Compared to (SDP), we replaced the last condition $SX = 0$ with $SX = \mu I_n$, where $\mu > 0$ and $\mu \to 0$.

A Primal-Dual Interpretation

- Since X_μ^* satisfies (OPT_μ) if it exists, we can deduce that $f_\mu(X_\mu^*) = \langle Q, X_\mu^* \rangle + \mu \log(\det(X_\mu^*))$ converges to the value of the original problem (SDP) : $f(X) = \langle Q, X \rangle$ as $\mu \to 0$.

- Moreover, we have a primal-dual pair (X, β, S) whose duality gap depends on μ :
  If $(\hat{X}, \hat{\beta}, \hat{S})$ satisfy (OPT_μ), then $\hat{X}$ is a strictly feasible solution of (SDP), $(\hat{\beta}, \hat{S})$ is a strictly feasible solution of $(DSDP)$, and the duality gap is :

$$\langle b, \hat{\beta} \rangle - \langle Q, \hat{X} \rangle = n\mu$$

Exercise : proof

A Primal-Dual Interpretation

- Therefore, if we can compute X_μ^* for very small μ, we obtain an "almost" optimal solution to (SDP).

- Moreover, due to weak duality, $\langle Q, X \rangle \leq \langle b, \beta \rangle$, so the value $\langle Q, X_\mu^* \rangle$ will be at most $n\mu$ away from the optimal value of (SDP).

Step (ii) : KKT Conditions for Optimality of X_μ^*

Suppose that (SDP) has a strictly feasible solution $\hat{X} \succ 0$ and that its dual $(DSDP)$ also has a strictly feasible solution $\hat{\beta}$ such that the corresponding slack matrix $\hat{S} = \sum_{r=1}^{m} \beta_r A_r - Q$ is positive definite (i.e., $S \succ 0$).

Furthermore, if the A_r are linearly independent, then for all $\mu \geq 0$, (SDP_μ) has a unique solution $X^* = X_\mu^*$, $\beta^* = \beta_\mu^*$, and $S^* = S_\mu^*$, and X_μ^* is the unique maximizer of f_μ under the constraints $g_r(X) = 0$ and $X \succ 0$.

Interior Point Method : Algorithm

Let $(X, \beta, S) = (X^0, \beta^0, S^0)$ be a strictly feasible point, and let $\epsilon > 0$.

1. Compute the current value of μ : $\mu = \frac{SX}{n}$
2. Compute new search directions for the new μ, i.e., compute $(\Delta X, \Delta\beta, \Delta S)$ by solving (OPT_μ) :

$$(OPT_\mu)\begin{cases} \langle A_r, X + \Delta X\rangle = b_r & \forall r,\ X \succ 0 \\ \sum_{r=1}^{m}(\beta_r + \Delta\beta_r)A_r - (S + \Delta S) = Q & \beta \in \mathbb{R}^m, S \succ 0 \\ (S + \Delta S)(X + \Delta X) = \mu I_n & \end{cases}$$

3. Compute step sizes : calculate step lengths α_p and α_d such that :

$$X + \alpha_p \Delta X \succ 0 \qquad S + \alpha_d \Delta S \succ 0$$

4. Update : $X = X + \alpha_p \Delta X$, $\beta = \beta + \alpha_d \Delta\beta$ and $S = S + \alpha_d \Delta S$
5. If $SX < \epsilon$, stop.

Step 2 : Approximation of Search Directions

Solve the system :

$$(OPT_\mu)\begin{cases} \langle A_r, X+\Delta X\rangle = b_r & \forall r,\ X \succeq 0 \\ \sum_{r=1}^{m}(\beta_r+\Delta\beta_r)A_r - (S+\Delta S) = Q & \beta \in \mathbb{R}^m \\ (S+\Delta S)(X+\Delta X) = \mu I_n & \end{cases}$$

The first two equations are linear.
Approximate the third one with a linear function (from Helmberg et al. 1996) :

$$\begin{aligned} \mu I_n &= (S+\Delta S)(X+\Delta X) \\ &= SX + S\Delta X + \Delta S X + \Delta S \Delta X \approx SX + S\Delta X + X\Delta S \end{aligned}$$

Remark : even if X and S are symmetric, the product XS is not necessarily symmetric. When that happens, we consider the matrix $\hat{X}\hat{S} = \frac{SX+(SX)^T}{2}$.

IPM variants

Within the framework of IPMs, many variants have been proposed, analyzed, and implemented :

- Path-following
- Infeasible
- Potential reduction
- Dual scaling
- Primal-dual completion-based
- and more...

Modern LP software contains both simplex and interior-point solvers, often several variants for each.

Spectral bundle method

assume m and n are large
$\longrightarrow$ avoid Cholesky factorization, matrix multiplication,...

idea : get rid of $Z \succeq 0$ by using eigenvalue arguments.

Spectral bundle method

$\mathcal{A}$ has constant trace property if I is in the range of $\mathcal{A}^\top$, i.e.,
$\exists \mu$ such that $\mathcal{A}^\top(\mu) = I$.

Spectral bundle method

$\mathcal{A}$ has constant trace property if I is in the range of $\mathcal{A}^\top$, i.e.,
$\exists \mu$ such that $\mathcal{A}^\top(\mu) = I$.

The constant trace property implies :

$$\mathcal{A}(X) = b, \mathcal{A}^\top(\mu) = I$$

Spectral bundle method

$\mathcal{A}$ has constant trace property if I is in the range of $\mathcal{A}^\top$, i.e., $\exists\mu$ such that $\mathcal{A}^\top(\mu) = I$.

The constant trace property implies :

$$\begin{aligned}
&\mathcal{A}(X) = b, \mathcal{A}^\top(\mu) = I \\
&\implies \operatorname{trace}(X) = \langle I, X\rangle = \langle \mathcal{A}^\top(\mu), X\rangle = \langle \mu, \mathcal{A}(X)\rangle = \mu^\top b =: a
\end{aligned}$$

Spectral bundle method

$\mathcal{A}$ has constant trace property if I is in the range of $\mathcal{A}^\top$, i.e., $\exists\mu$ such that $\mathcal{A}^\top(\mu) = I$.

The constant trace property implies :

$$\mathcal{A}(X) = b, \mathcal{A}^\top(\mu) = I$$
$$\implies \operatorname{trace}(X) = \langle I, X\rangle = \langle \mathcal{A}^\top(\mu), X\rangle = \langle \mu, \mathcal{A}(X)\rangle = \mu^\top b =: a$$

constant trace property holds for many SDP derived from combinatorial optimization problems.

Spectral bundle method

Reformulate dual as follows :

$$\min\{b^\top y : \mathcal{A}^\top(y) - C = Z \succeq 0\}$$

Spectral bundle method

Reformulate dual as follows :

$$\min\{b^\top y : \mathcal{A}^\top(y) - C = Z \succeq 0\}$$

Adding (redundant) primal constraint $tr(X) = a$ (thus introducing new dual variable, say λ) and dual becomes :

$$\min\{b^\top y + a\lambda : \mathcal{A}^\top(y) - C + \lambda I = Z \succeq 0\}$$

Spectral bundle method

Reformulate dual as follows :

$$\min\{b^\top y \colon \mathcal{A}^\top(y) - C = Z \succeq 0\}$$

Adding (redundant) primal constraint $tr(X) = a$ (thus introducing new dual variable, say λ) and dual becomes :

$$\min\{b^\top y + a\lambda \colon \mathcal{A}^\top(y) - C + \lambda I = Z \succeq 0\}$$

X^*,Z^* optimal

$$\Longrightarrow X^* Z^* = 0, \text{ hence } Z^* \text{ is singular and } \lambda_{min}(Z^*) = 0.$$

Spectral bundle method

Reformulate dual as follows :

$$\min\{b^\top y \colon \mathcal{A}^\top(y) - C = Z \succeq 0\}$$

Adding (redundant) primal constraint $tr(X) = a$ (thus introducing new dual variable, say λ) and dual becomes :

$$\min\{b^\top y + a\lambda \colon \mathcal{A}^\top(y) - C + \lambda I = Z \succeq 0\}$$

X^*,Z^* optimal

$\Longrightarrow X^* Z^* = 0$, hence Z^* is singular and $\lambda_{min}(Z^*) = 0$.

$\longrightarrow$ used to compute dual variable λ explicitely

Spectral bundle method

$$\lambda_{min}(Z^*) = 0 \iff \lambda_{\max}(-Z^*) = 0$$

Spectral bundle method

$$\lambda_{min}(Z^*) = 0 \iff \lambda_{\max}(-Z^*) = 0$$
$$\iff \lambda_{\max}(C - \mathcal{A}^\top(y^*) - \lambda^* I) = 0$$

Spectral bundle method

$$
\begin{aligned}
\lambda_{min}(Z^*) = 0 \quad &\iff \quad \lambda_{\max}(-Z^*) = 0 \\
&\iff \quad \lambda_{\max}(C - \mathcal{A}^\top(y^*) - \lambda^* I) = 0 \\
&\iff \quad \lambda_{\max}(C - \mathcal{A}^\top(y^*)) - \lambda^* = 0
\end{aligned}
$$

Spectral bundle method

$$\begin{aligned}\lambda_{min}(Z^*) = 0 \quad &\iff \quad \lambda_{\max}(-Z^*) = 0 \\ &\iff \quad \lambda_{\max}(C - \mathcal{A}^\top(y^*) - \lambda^* I) = 0 \\ &\iff \quad \lambda_{\max}(C - \mathcal{A}^\top(y^*)) - \lambda^* = 0 \\ &\iff \quad \lambda^* = \lambda_{max}(C - \mathcal{A}^\top(y^*))\end{aligned}$$

Spectral bundle method

$$\begin{aligned}\lambda_{min}(Z^*) = 0 &\iff \lambda_{\max}(-Z^*) = 0\\ &\iff \lambda_{\max}(C - \mathcal{A}^\top(y^*) - \lambda^* I) = 0\\ &\iff \lambda_{\max}(C - \mathcal{A}^\top(y^*)) - \lambda^* = 0\\ &\iff \lambda^* = \lambda_{max}(C - \mathcal{A}^\top(y^*))\end{aligned}$$

rewrite the dual :

$$\min\{b^\top y + \lambda_{\max}(C - \mathcal{A}^\top(y)) \colon y \in \mathbb{R}^m\}$$

$\longrightarrow$ non-smooth unconstrained convex problem in y.

Spectral bundle method

$$
\begin{aligned}
\lambda_{min}(Z^*) = 0 &\iff \lambda_{\max}(-Z^*) = 0 \\
&\iff \lambda_{\max}(C - \mathcal{A}^\top(y^*) - \lambda^* I) = 0 \\
&\iff \lambda_{\max}(C - \mathcal{A}^\top(y^*)) - \lambda^* = 0 \\
&\iff \lambda^* = \lambda_{max}(C - \mathcal{A}^\top(y^*))
\end{aligned}
$$

rewrite the dual :

$$\min\{b^\top y + \lambda_{\max}(C - \mathcal{A}^\top(y)) \colon y \in \mathbb{R}^m\}$$

$\longrightarrow$ non-smooth unconstrained convex problem in y.

note : evaluating $f(y) = b^\top y + \lambda_{\max}(C - \mathcal{A}^\top(y))$ amounts in computing largest eigenvalue of $C - \mathcal{A}^\top(y)$

Spectral bundle method

$$
\begin{aligned}
\lambda_{min}(Z^*) = 0 \quad &\iff \quad \lambda_{\max}(-Z^*) = 0 \\
&\iff \quad \lambda_{\max}(C - \mathcal{A}^\top(y^*) - \lambda^* I) = 0 \\
&\iff \quad \lambda_{\max}(C - \mathcal{A}^\top(y^*)) - \lambda^* = 0 \\
&\iff \quad \lambda^* = \lambda_{max}(C - \mathcal{A}^\top(y^*))
\end{aligned}
$$

rewrite the dual :

$$
\min\{b^\top y + \lambda_{\max}(C - \mathcal{A}^\top(y)) \colon y \in \mathbb{R}^m\}
$$

$\longrightarrow$ non-smooth unconstrained convex problem in y.

note : evaluating $f(y) = b^\top y + \lambda_{\max}(C - \mathcal{A}^\top(y))$ amounts in computing largest eigenvalue of $C - \mathcal{A}^\top(y)$

$\longrightarrow$ can be done by iterative methods even for very large (sparse) matrices.

Bundle method

We would like to compute

$$z^* = \max\{\langle C, X\rangle : \mathcal{A}(X) = a,\ \mathcal{B}(X) = b,\ X \succeq 0\}$$

Optimizing over $\mathcal{A}(X) = a, X \succeq 0$ without $\mathcal{B}(X) = b$ is "easy", but inclusion of $\mathcal{B}(X) = b$ makes SDP difficult.

Bundle method

We would like to compute

$$z^* = \max\{\langle C, X\rangle : \mathcal{A}(X) = a,\ \mathcal{B}(X) = b,\ X \succeq 0\}$$

Optimizing over $\mathcal{A}(X) = a, X \succeq 0$ without $\mathcal{B}(X) = b$ is “easy”, but inclusion of $\mathcal{B}(X) = b$ makes SDP difficult.
partial Lagrangian dual (y dual to $\mathcal{B}(X) = b$) :

$$\mathcal{L}(X; y) = \langle C, X\rangle + y^\top (b - \mathcal{B}(X))$$

dual functional :

$$f(y) = \max_{\mathcal{A}(X)=a, X\succeq 0} \mathcal{L}(X; y)$$

Bundle method

We would like to compute

$$z^* = \max\{\langle C, X\rangle : \mathcal{A}(X) = a,\ \mathcal{B}(X) = b,\ X \succeq 0\}$$

Optimizing over $\mathcal{A}(X) = a, X \succeq 0$ without $\mathcal{B}(X) = b$ is "easy", but inclusion of $\mathcal{B}(X) = b$ makes SDP difficult.
partial Lagrangian dual (y dual to $\mathcal{B}(X) = b$) :

$$\mathcal{L}(X; y) = \langle C, X\rangle + y^\top (b - \mathcal{B}(X))$$

dual functional :

$$f(y) = \max_{\mathcal{A}(X)=a, X\succeq 0} \mathcal{L}(X; y)$$

and thus

$$z^* = \min_{y\in\mathbb{R}^m} f(y)$$

Bundle method

$$z^* = \min_{y \in \mathbb{R}^m} f(y)$$

with

$$f(y) \quad = \quad \max_{\mathcal{A}(X)=a, X \succeq 0} \langle C, X \rangle + y^\top (b - \mathcal{B}(X))$$

Bundle method

$$z^* = \min_{y \in \mathbb{R}^m} f(y)$$

with

$$\begin{aligned} f(y) &= \max_{\mathcal{A}(X)=a, X \succeq 0} \langle C, X \rangle + y^\top (b - \mathcal{B}(X)) \\ &= b^\top y + \max_{\mathcal{A}(X)=a, X \succeq 0} \langle C - \mathcal{B}^\top(y), X \rangle \end{aligned}$$

Bundle method

$$z^* = \min_{y \in \mathbb{R}^m} f(y)$$

with

$$\begin{aligned} f(y) &= \max_{\mathcal{A}(X)=a, X \succeq 0} \langle C, X \rangle + y^\top (b - \mathcal{B}(X)) \\ &= b^\top y + \max_{\mathcal{A}(X)=a, X \succeq 0} \langle C - \mathcal{B}^\top(y), X \rangle \end{aligned}$$

evaluating $f(y)$ amounts in solving an SDP.

Bundle method

$$z^* = \min_{y \in \mathbb{R}^m} f(y)$$

with

$$\begin{aligned} f(y) &= \max_{\mathcal{A}(X)=a, X \succeq 0} \langle C, X \rangle + y^\top (b - \mathcal{B}(X)) \\ &= b^\top y + \max_{\mathcal{A}(X)=a, X \succeq 0} \langle C - \mathcal{B}^\top(y), X \rangle \end{aligned}$$

evaluating $f(y)$ amounts in solving an SDP.

basic assumption : we can evaluate $f(y)$ easily, yielding also a maximizer X^* and $g^* = b - \mathcal{B}(X^*)$.

Augmented Lagrange algorithm

$$\min f(x) \quad \text{such that} \quad x \in \mathcal{X},\ h(x) = 0$$

Augmented Lagrange algorithm

$$\min f(x) \quad \text{such that} \quad x \in \mathcal{X},\ h(x) = 0$$

$f : \mathbb{R}^n \mapsto \mathbb{R}$, $h : \mathbb{R}^n \mapsto \mathbb{R}^m$ sufficiently smooth functions, $\mathcal{X} \subseteq \mathbb{R}^n$ nonempty closed convex set of simple structure

Augmented Lagrange algorithm

$$\min f(x) \quad \text{such that} \quad x \in \mathcal{X},\ h(x) = 0$$

$f : \mathbb{R}^n \mapsto \mathbb{R}$, $h : \mathbb{R}^n \mapsto \mathbb{R}^m$ sufficiently smooth functions, $\mathcal{X} \subseteq \mathbb{R}^n$ nonempty closed convex set of simple structure

$$\mathcal{L}_\sigma(x, y) := f(x) + y^\top h(x) + \frac{\sigma}{2}\|h(x)\|^2$$

Augmented Lagrange algorithm

$$\min f(x) \quad \text{such that} \quad x \in \mathcal{X},\ h(x) = 0$$

$f : \mathbb{R}^n \mapsto \mathbb{R}$, $h : \mathbb{R}^n \mapsto \mathbb{R}^m$ sufficiently smooth functions, $\mathcal{X} \subseteq \mathbb{R}^n$ nonempty closed convex set of simple structure

$$\mathcal{L}_\sigma(x, y) := f(x) + y^\top h(x) + \frac{\sigma}{2}\|h(x)\|^2$$

repeat until convergence
(a) Keep y fixed : solve $\min_x \mathcal{L}_\sigma(x, y)$ to get x
(b) update y : $y \leftarrow y + \sigma h(x)$
(c) update σ

Original version : Powell, Hestenes, 1969

Augmented Lagrange algorithm

$$\text{(DSDP)} \qquad \min\ b^\top y \quad \text{s.t. } \mathcal{A}^\top(y) - C = Z,\ Z \succeq 0$$

Augmented Lagrange algorithm

$$\text{(DSDP)} \qquad \min \; b^\top y \quad \text{s.t. } \mathcal{A}^\top(y) - C = Z, \; Z \succeq 0$$

$$\mathcal{L}_\sigma(y, Z; X) = \; b^\top y + \langle X, Z + C - \mathcal{A}^\top(y) \rangle + \frac{\sigma}{2} \| Z + C - \mathcal{A}^\top(y) \|^2$$

Projection methods

inner minimization

$$\min_{y,Z\succeq 0} \mathcal{L}_{\sigma_k}(y, Z; X)$$

Projection methods

inner minimization

$$\min_{y, Z \succeq 0} \mathcal{L}_{\sigma_k}(y, Z; X)$$

$$\text{Define } \mathcal{W}(y) := \mathcal{A}^\top(y) - C - \frac{1}{\sigma} X$$

Projection methods

inner minimization

$$\min_{y, Z \succeq 0} \mathcal{L}_{\sigma_k}(y, Z; X)$$

$$\text{Define } \mathcal{W}(y) := \mathcal{A}^\top(y) - C - \frac{1}{\sigma} X$$

$$\begin{aligned} \mathcal{L}_\sigma(y, Z; X) &= b^\top y + \langle X, Z + C - \mathcal{A}^\top(y) \rangle + \tfrac{\sigma}{2} \| Z + C - \mathcal{A}^\top(y) \|^2 \\ &= b^\top y + \tfrac{1}{\sigma} \| Z - \mathcal{W}(y) \|^2 - \tfrac{1}{2\sigma} \| X \|^2 \end{aligned}$$

Projection methods

inner minimization

$$\min_{y, Z \succeq 0} \mathcal{L}_{\sigma_k}(y, Z; X)$$

$$\text{Define } \mathcal{W}(y) := \mathcal{A}^\top(y) - C - \frac{1}{\sigma} X$$

$$\begin{aligned} \mathcal{L}_\sigma(y, Z; X) &= b^\top y + \langle X, Z + C - \mathcal{A}^\top(y) \rangle + \tfrac{\sigma}{2} \| Z + C - \mathcal{A}^\top(y) \|^2 \\ &= b^\top y + \tfrac{1}{\sigma} \| Z - \mathcal{W}(y) \|^2 - \tfrac{1}{2\sigma} \| X \|^2 \end{aligned}$$

$$\min_{y, Z \succeq 0} b^\top y + \frac{1}{\sigma} \| Z - \mathcal{W}(y) \|^2$$

Projection methods

inner minimization : optimality conditions

$$\mathcal{L}_\sigma(y, Z; V) = b^\top y + \frac{1}{\sigma}\|Z - \mathcal{W}(y)\|^2 - \langle V, Z\rangle$$

Projection methods

inner minimization : optimality conditions

$$\mathcal{L}_\sigma(y, Z; V) = b^\top y + \frac{1}{\sigma}\|Z - \mathcal{W}(y)\|^2 - \langle V, Z\rangle$$

$$\nabla_y \mathcal{L} = 0 \Rightarrow \sigma\mathcal{A}(\mathcal{A}^\top(y)) = \sigma\mathcal{A}(Z + C) + (\mathcal{A}(X) - b)$$

Projection methods

inner minimization : optimality conditions

$$\mathcal{L}_\sigma(y, Z; V) = b^\top y + \frac{1}{\sigma}\|Z - \mathcal{W}(y)\|^2 - \langle V, Z\rangle$$

$$\nabla_y \mathcal{L} = 0 \Rightarrow \sigma\mathcal{A}(\mathcal{A}^\top(y)) = \sigma\mathcal{A}(Z + C) + (\mathcal{A}(X) - b)$$

$$\nabla_Z \mathcal{L} = 0 \Rightarrow V = \sigma(Z - \mathcal{W}(y))$$

Projection methods

inner minimization : optimality conditions

$$\mathcal{L}_\sigma(y, Z; V) = b^\top y + \frac{1}{\sigma}\|Z - \mathcal{W}(y)\|^2 - \langle V, Z\rangle$$

$$\nabla_y \mathcal{L} = 0 \Rightarrow \sigma\mathcal{A}(\mathcal{A}^\top(y)) = \sigma\mathcal{A}(Z + C) + (\mathcal{A}(X) - b)$$

$$\nabla_Z \mathcal{L} = 0 \Rightarrow V = \sigma(Z - \mathcal{W}(y))$$

$$V \succeq 0,\;\; Z \succeq 0, VZ = 0.$$

Conclusions on algorithms

- solution method to choose depends on sizes and on structure of problems

Conclusions on algorithms

- solution method to choose depends on sizes and on structure of problems
- interior point methods : many implementations available, limit $n \approx$ thousands, $m \approx$ tens of thousands

Conclusions on algorithms

- solution method to choose depends on sizes and on structure of problems
- interior point methods : many implementations available, limit $n \approx$ thousands, $m \approx$ tens of thousands
- spectral bundle method : general tool for matrices of large dimension

Conclusions on algorithms

- solution method to choose depends on sizes and on structure of problems
- interior point methods : many implementations available, limit $n \approx$ thousands, $m \approx$ tens of thousands
- spectral bundle method : general tool for matrices of large dimension
- bundle method : if partial Lagrangian dual is "nice"

Conclusions on algorithms

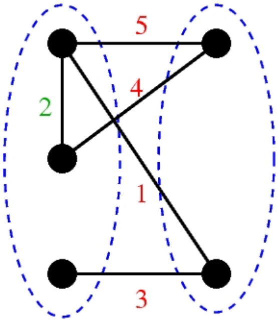
- solution method to choose depends on sizes and on structure of problems
- interior point methods : many implementations available, limit $n \approx$ thousands, $m \approx$ tens of thousands
- spectral bundle method : general tool for matrices of large dimension
- bundle method : if partial Lagrangian dual is "nice"
- more methods : augmented Lagrangian methods, projection methods, low-rank methods,. . .

Conclusions on algorithms

- solution method to choose depends on sizes and on structure of problems
- interior point methods : many implementations available, limit $n \approx$ thousands, $m \approx$ tens of thousands
- spectral bundle method : general tool for matrices of large dimension
- bundle method : if partial Lagrangian dual is “nice”
- more methods : augmented Lagrangian methods, projection methods, low-rank methods,...
- SDP is standard tool in optimization and sufficiently easy to use(?)

The Max-Cut Problem

Given a graph $G = (V, E)$ and weights w_{ij} for all edges $(i, j) \in E$, find an edge-cut of maximum weight, i.e. find a set $S \subseteq V$ s.t. the sum of the weights of the edges with one end in S and the other in $V \setminus S$ is maximum. We assume wlog that $w_{ii} = 0$ for all $i \in V$, and that G is complete (assign $w_{ij} = 0$ if edge $ij \notin E$).



Standard Integer LP Formulation

$$
\begin{aligned}
\max \quad & \sum_{i=1}^{n} \sum_{j=i+1}^{n} w_{ij} y_{ij} \\
\text{s.t.} \quad & y_{ij} + y_{ik} + y_{jk} \leq 2, 1 \leq i < j < k \leq n \\
& y_{ij} - y_{ik} - y_{jk} \leq 0, 1 \leq i < j \leq n, k \neq i, j \\
& y_{ij} \in \{0, 1\}, 1 \leq i < j \leq n
\end{aligned}
$$

where

$$
y_{ij} = \begin{cases} 1 & \text{if edge } ij \text{ is cut} \\ 0 & \text{otherwise,} \end{cases}
$$

$y_{ij} = y_{ji}$, and w_{ij} denotes the weight of edge ij.

This formulation is the basis for a highly successful branch-and-cut algorithm for solving spin glass problems in physics (Liers, Jünger, Reinelt and Rinaldi (2005)).
The solver can be accessed online at the Spin Glass Server :

`http://www.informatik.uni-koeln.de/spinglass/`

Quadratic Formulation of Max-Cut

0-1 formulation :
$z_i = 1$ if i is in V_1 and $z_i = 0$ otherwise.

$$(MC_{0,1})\begin{cases} \max \displaystyle\sum_{i=1}^{n}\sum_{j=i+1}^{n} w_{ij}\Big[z_i(1-z_j)+z_j(1-z_i)\Big] \\ \quad z \in \{0,1\}^n \end{cases}$$

We count w_{ij} if z_i is in V_1 and z_j is not $(z_i(1-z_j))$, or the reverse $(z_j(1-z_i))$.

MaxCut - $\{-1, 1\}$ Formulation

We now consider variables $x_i = -1$ for all vertices in V_1 and 1 for those in V_2.

Exercise : Perform the change of variable. What formulation do we obtain ?

MaxCut - $\{-1, 1\}$ Formulation

We now consider variables $x_i = -1$ for all vertices in V_1 and 1 for those in V_2.

Exercise : Perform the change of variable. What formulation do we obtain ?

We obtain the following formulation :

$$(MC_{-1,1}) \begin{cases} \max & \sum_{i=1}^{n} \sum_{j=i+1}^{n} \frac{1}{2} w_{ij}(1 - x_i x_j) \\ & x \in \{-1, 1\}^n \end{cases}$$

MaxCut - Formulation Using the Laplacian Matrix

Laplacian matrix of an undirected graph :

Unweighted case :

$$L_{ij}\begin{cases} deg(i) & \text{if } i=j \\ -1 & \text{if } i\neq j \text{ and } [e_i,e_j]\in E \\ 0 & \text{otherwise} \end{cases}$$

Weighted case :

$$L_{ij}\begin{cases} \sum_{k=1}^{n} w_{ik} & \text{if } i=j \\ -w_{ij} & \text{if } i\neq j \text{ and } [e_i,e_j]\in E \\ 0 & \text{otherwise} \end{cases}$$

If we denote M as the weighted adjacency matrix of the graph and D its weighted degree matrix, then $L = D - M$

Exercise : Show that the MaxCut problem can be formulated using the Laplacian matrix.

$$(MC_{-1,1})\begin{cases} \max & \sum_{i=1}^{n}\sum_{j=i+1}^{n}\frac{1}{2}w_{ij}(1-x_ix_j) \\ & x\in\{-1,1\}^n \end{cases} \iff (MC_L)\begin{cases} \max & \sum_{i=1}^{n}\sum_{j=1}^{n}\frac{1}{4}L_{ij}x_ix_j \\ & x\in\{-1,1\}^n \end{cases}$$

Semidefinite Relaxation : Example of the $\{-1, 1\}$ case

$$\left\{ \min_{(x,y)\in\{-1,1\}} -2x^2 + xy \right.$$

```
optimal value: -3
optimal solution: (-1,1)
```

Semidefinite Relaxation : Example of the $\{-1, 1\}$ case

$$\left\{ \min_{(x,y)\in\{-1,1\}} -2x^2 + xy \right. \iff \left\{ \begin{array}{ll} \min & -2Z_{11} + 0.5(Z_{12} + Z_{21}) \\ \text{s.t.} & Z = \begin{pmatrix} Z_{11} & Z_{12} \\ Z_{21} & Z_{22} \end{pmatrix} = \begin{pmatrix} x^2 & xy \\ yx & y^2 \end{pmatrix} \\ & (x, y) \in \{-1, 1\}^2 \\ & Z \in \mathcal{S}^n \end{array} \right.$$

- Introduce Z, a variable matrix modeling the products xy^T.
  $\rightarrow$ This linearizes the objective function.

Semidefinite Relaxation : Example of the $\{-1, 1\}$ case

$$\left\{ \min_{(x,y)\in\{-1,1\}} -2x^2 + xy \right. \quad \Longleftrightarrow \quad \left\{ \begin{array}{ll} \min & -2Z_{11} + 0.5(Z_{12} + Z_{21}) \\ \text{s.t.} & \text{rank}(Z) = 1 \\ & (x, y) \in \{-1, 1\}^2 \\ & Z \in \mathcal{S}_+^n \end{array} \right.$$

- Introduce Z, a variable matrix modeling the products xy^T.
  $\rightarrow$ This linearizes the objective function.
- Rewrite the constraint $Z = \begin{pmatrix} x \\ y \end{pmatrix} (x \ y)$ as $\text{rank}(Z) = 1$ and $Z \succeq 0$

Semidefinite Relaxation : Example of the $\{-1,1\}$ case

$$\left\{ \min_{(x,y)\in\{-1,1\}} -2x^2 + xy \right. \quad \Longleftrightarrow \quad \left\{ \begin{array}{ll} \min & -2Z_{11} + 0.5(Z_{12} + Z_{21}) \\ \text{s.t.} & \mathrm{rank}(Z) = 1 \\ & Z_{ii} = 1 \\ & Z \in \mathcal{S}_+^n \end{array} \right.$$

- Introduce Z, a variable matrix modeling the products xy^T.
  $\rightarrow$ This linearizes the objective function.
- Rewrite the constraint $Z = \begin{pmatrix} x \\ y \end{pmatrix} (x \ y)$ as $\mathrm{rank}(Z) = 1$ and $Z \succeq 0$
- Rewrite the $\{-1,1\}$ constraints as $x^2 = 1$ and $y^2 = 1$.

Semidefinite Relaxation : Example of the $\{-1, 1\}$ case

$$\left\{ \min_{(x,y)\in\{-1,1\}} -2x^2 + xy \right. \quad \xrightarrow{relax} \quad \begin{cases} \min & -2Z_{11} + 0.5(Z_{12} + Z_{21}) \\ \text{s.t.} & Z_{ii} = 1 \\ & Z \in \mathcal{S}_+^n \end{cases}$$

- Introduce Z, a variable matrix modeling the products xy^T.
  $\rightarrow$ This linearizes the objective function.
- Rewrite the constraint $Z = \begin{pmatrix} x \\ y \end{pmatrix} (x\ y)$ as $\mathrm{rank}(Z) = 1$ and $Z \succeq 0$
- Rewrite the $\{-1, 1\}$ constraints as $x^2 = 1$ and $y^2 = 1$.
- To obtain a relaxation, drop the constraint $\mathrm{rank}(Z) = 1$

SDP Formulation for MaxCut in $\{-1, 1\}$

The MaxCut problem can be formulated as an SDP program :

$$(MC_L)\begin{cases} \max & \displaystyle\sum_{i=1}^{n}\sum_{j=1}^{n}\frac{1}{4}L_{ij}x_ix_j \\ & x \in \{-1,1\}^n \end{cases} \iff (F - SDP_{MC})\begin{cases} \max & \dfrac{1}{4}\langle L, X\rangle \\ & X_{ii} = 1 \\ & \mathrm{rank}(X) = 1 \\ & X \succeq 0 \end{cases}$$

Proof : Exercise

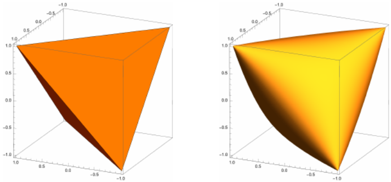
SDP Relaxation for MaxCut in $\{-1, 1\}$

By relaxing the rank constraint, we obtain the relaxation :

$$(MC_L)\begin{cases} \max & \frac{1}{4}\sum_{i=1}^{n}\sum_{j=i+1}^{n} L_{ij}x_i x_j \\ & x \in \{-1,1\}^n \end{cases} \xrightarrow{relax} (R - SDP_{MC})\begin{cases} \max & \frac{1}{4}\langle L, X\rangle \\ & X_{ii} = 1 \\ & X \succeq 0 \end{cases}$$

Proof : If x is feasible for (MC_L), then $X = xx^T$ is feasible for $(R - SDP_{MC})$, hence $v(R\text{-}SDP_{MC}) \geq v(MC_L)$.

Illustration : Graph with 3 nodes :



Source : "Topics in Convex Optimization (Michaelmas 2018)"

The Basic Semidefinite Relaxation of Max-Cut

Consider the change of variable $X = vv^T, v \in \{\pm 1\}^n$.
Then $X_{ij} = v_i v_j$ and max-cut is equivalent to

$$\begin{array}{ll} \max & \langle Q, X\rangle \\ \text{s.t.} & \mathrm{diag}(X) = e \\ & \mathrm{rank}(X) = 1 \\ & X \succeq 0, \end{array}$$

where $Q = \frac{1}{4}(\mathrm{Diag}(We) - W)$.

Removing the rank constraint, we obtain the basic SDP relaxation of max-cut.

Question : How good is this SDP relaxation ?

Goemans and Williamson (1995) : 0.878-approximation algorithm

Theorem (Goemans and Williamson (1995))

If $w_{ij} \geq 0$ for all edges ij, then

$$\frac{\textit{max-cut opt value}}{\textit{SDP relax opt value}} \geq \alpha$$

where $\alpha := \min_{0 \leq \xi \leq \pi} \frac{2}{\pi} \frac{\xi}{1 - \cos \xi} \approx 0.87856.$

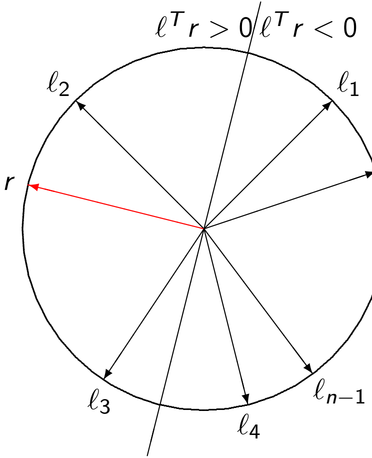
This performance ratio is best-possible if the Unique Games Conjecture is true.

Goemans and Williamson (1995) : 0.878-approximation algorithm (ctd)

In fact, Goemans and Williamson proved a stronger result :
they described a randomized algorithm that

- from an optimal solution X^* of the SDP relaxation
- *generates a cut* with expected weight $\geq \alpha$ (SDP relax opt value).

Using the fact that $X^* \succeq 0 \Rightarrow \exists\ \ell_1, \ell_2, \ldots, \ell_n$ s.t. $X_{ij} = \ell_i^T \ell_j$



Since the optimal max-cut value is at least the expected value of this cut, the theorem follows.

Constant Relative Accuracy Estimate

With no restriction on the edge weights, Nesterov (1998) proved constant relative accuracy estimates for the basic SDP bound.

Define

$$\mu^* = \max \; v^T Q v \quad \text{s.t.} \; v \in \{-1,1\}^n \qquad \mu_* = \min \; v^T Q v \quad \text{s.t.} \; v \in \{-1,1\}^n$$

$$\psi^* = \max \; \langle Q, X \rangle \quad \text{s.t.} \; \mathrm{diag}(X) = e, \; X \succeq 0 \qquad \psi_* = \min \; \langle Q, X \rangle \quad \text{s.t.} \; \mathrm{diag}(X) = e, X \succeq 0$$

and

$$s(\beta) := \beta\,\psi^* + (1-\beta)\,\psi_*, \;\; \beta \in [0,1].$$

Theorem (Nesterov (1998))

Without any assumption on the matrix Q,

$$\frac{|s(\frac{2}{\pi}) - \mu^*|}{\mu^* - \mu_*} \le \frac{\pi}{2} - 1 < \frac{4}{7}.$$

Branch and Bound Based on the Relaxation (R-SDP_{MC})

Separation mechanism :
Select two vertices i and j and consider the following two subproblems :

1. $S_1 := \{x \in \{-1,1\}^n \; : \; x_i - x_j = 0\}$ (i.e. i and j are in the same partition)
2. $S_2 := \{x \in \{-1,1\}^n \; : \; x_i + x_j = 0\}$ (i.e. i and j are in different partitions)

A branch and bound is built using the bound obtained by solving (R-SDP_{MC}).

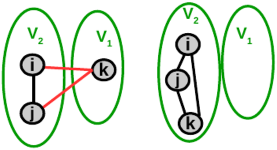
- Efficient solution of (R-SDP_{MC}) using interior-point methods
- However : the root bound is not tight enough

$\Longrightarrow$ Not efficient enough for graphs with more than 50 nodes

Strengthening ($R\text{-}SDP_{MC}$) : Triangle Inequalities

Let $s_{ij} = 1$ if edge $[e_i, e_j]$ is included in the maximum cut, 0 otherwise.
Consider the triangle inequalities, $\forall\, 1 \leq i < j < k \leq n$:

$$\mathcal{T}_{0,1} \begin{cases} s_{ij} + s_{ik} + s_{jk} \leq 2 & (i) \\ s_{ij} - s_{ik} - s_{jk} \leq 0 & (ii) \\ -s_{ij} + s_{ik} - s_{jk} \leq 0 & (iii) \\ -s_{ij} - s_{ik} + s_{jk} \leq 0 & (iv) \end{cases}$$

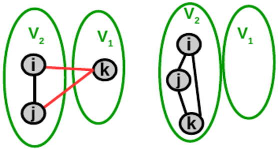


- (i) If $[e_i, e_j]$ and $[e_i, e_k]$ are in the cut, then $[e_j, e_k]$ cannot be (and vice versa).
- (ii) If $[e_i, e_j]$ is in the cut, then at least one of $[e_i, e_k]$ or $[e_j, e_k]$ must be in it.
- (iii) If $[e_i, e_k]$ is in the cut, then at least one of $[e_i, e_j]$ or $[e_j, e_k]$ must be in it.
- (iv) If $[e_j, e_k]$ is in the cut, then at least one of $[e_i, e_k]$ or $[e_i, e_j]$ must be in it.

Strengthening (R-SDP_{MC}) : Triangle Inequalities

Let $s_{ij} = 1$ if edge $[e_i, e_j]$ is included in the maximum cut, 0 otherwise.
We consider triangle inequalities, $\forall\, 1 \leq i < j < k \leq n$:

$$\mathcal{T}_{0,1} \begin{cases} s_{ij} + s_{ik} + s_{jk} \leq 2 \\ s_{ij} - s_{ik} - s_{jk} \leq 0 \\ -s_{ij} + s_{ik} - s_{jk} \leq 0 \\ -s_{ij} - s_{ik} + s_{jk} \leq 0 \end{cases}$$

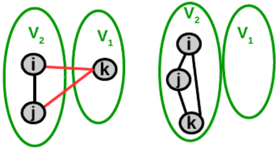


- In our model, the variables are in $\{-1, 1\}$, where $x_i = 1$ if node i is in partition V_1 and $x_i = -1$ if it is in partition V_2.
- If $s_{ij} = 1$, this corresponds to $x_i = 1$ and $x_j = -1$ (or vice versa), hence $x_i x_j = X_{ij} = -1$.
- We can apply the change of variables : $X_{ij} = 1 - 2s_{ij}$.
  Indeed, if $s_{ij} = 1 \Rightarrow X_{ij} = -1$ and if $s_{ij} = 0 \Rightarrow X_{ij} = 1$

Strengthening $(R\text{-}SDP_{MC})$: Triangle Inequalities

We now consider triangle inequalities, $\forall\, 1 \leq i < j < k \leq n$:

$$\mathcal{T}_{0,1}\begin{cases} s_{ij} + s_{ik} + s_{jk} \leq 2 \\ s_{ij} - s_{ik} - s_{jk} \leq 0 \\ -s_{ij} + s_{ik} - s_{jk} \leq 0 \\ -s_{ij} - s_{ik} + s_{jk} \leq 0 \end{cases}$$



where $s_{ij} = 1$ if edge $[e_i, e_j]$ is in the maximum cut, 0 otherwise.
We apply the variable change : $X_{ij} = 1 - 2s_{ij}$, yielding the following inequalities :

$$\mathcal{T}_{-1,1}\begin{cases} X_{ij} + X_{ik} + X_{jk} \geq -1 & (i) \\ X_{ij} - X_{ik} - X_{jk} \geq -1 & (ii) \\ -X_{ij} + X_{ik} - X_{jk} \geq -1 & (iii) \\ -X_{ij} - X_{ik} + X_{jk} \geq -1 & (iv) \end{cases} \quad (SDP - T_{MC})\begin{cases} \max & \frac{1}{4}\langle L, X\rangle \\ & X_{ii} = 1 \\ & X \in \mathcal{T}_{-1,1} \\ & X \succeq 0 \end{cases}$$

We have $v(MC_L) \leq v(SDP\text{-}T_{MC}) \leq v(R\text{-}SDP_{MC})$

Branch and Bound Based on the Relaxation (SDP-T_{MC})

- Challenge : solving (R-SDP_{MC}) at every node since the number of triangle inequalities is huge : $4\binom{n}{3}$
  $\Longrightarrow$ Requires a separation algorithm to identify active triangle inequalities

- Same branching rules used as in the previous B&B algorithm

This algorithm is implemented in `BiqMac` (Rinaldi, Rendl, Wiegele)
Currently one of the most efficient methods for solving MaxCut.

A Very Brief Introduction to Polynomial Optimization...

... with a focus on binary problems.

Polynomial Optimization

Polynomial optimization problems (POPs) consist of optimizing a multivariate polynomial subject to multivariate polynomial constraints :

$$\begin{array}{rll} z = & \sup & f(x) \\ & \text{s.t.} & g_i(x) \geq 0 \qquad i = 1, \ldots, m. \end{array}$$

Numerous classes of problems can be modelled as POPs, including :

- Linear Problems
- Quadratic Problems (Convex / Non-convex)
- Mixed-Binary Problems

$$x_i \in \{0, 1\} \quad \Leftrightarrow \quad x_i^2 - x_i = 0$$

Thus, solving POPs is in general NP-hard.

General POP Perspective

Given a general POP problem :

$$\text{(POP)} \quad z = \sup \; f(x) \quad \text{s.t.} \quad g_i(x) \geq 0 \qquad i = 1, \dots, m.$$

If λ is the optimal value of POP, then POP is equivalent to

$$\begin{array}{ll} \inf & \lambda \\ \text{s.t.} & \lambda - f(x) \geq 0 \quad \forall x \in S := \{x : g_i(x) \geq 0, \; i = 1, \dots, m\} \end{array}$$

which we rewrite as

$$\begin{array}{ll} \inf & \lambda \\ \text{s.t.} & \lambda - f(x) \in \mathcal{P}_d(S) \end{array}$$

where

$$\mathcal{P}_d(S) = \{p(x) \in \mathrm{R}_d[x] : p(s) \geq 0 \text{ for all } s \in S\}$$

is the cone of polynomials of degree $\leq d$ that are non-negative over S.

Main question :

How to relax the condition $\lambda - f(x) \in \mathcal{P}_d(S)$?

Conic Relaxations of POP

We relax $\lambda - f(x) \in \mathcal{P}_d(S)$ to

$$\lambda - f(x) \in \mathcal{K} \text{ for a suitable cone } \mathcal{K} \subseteq \mathcal{P}_d(S).$$

Then the conic optimization problem

$$\begin{array}{ll} \inf & \lambda \\ \text{s.t.} & \lambda - f(x) \in \mathcal{K} \end{array}$$

provides an upper bound for the original problem.

- The choice of $\mathcal{K}$ is key to obtaining good bounds on the problem

- Optimizing over $\mathcal{K}$ should (must ?) be tractable

Example : Application to Max-cut

The formulation

$$\max \mathrm{x}^T Q \mathrm{x}$$
$$\text{s.t. } x_i^2 = 1, i = 1, \ldots, n$$

can be recast as

$$\min \lambda$$
$$\text{s.t. } \lambda - \mathrm{x}^T Q \mathrm{x} \in \mathcal{P}_2 \left(\{ \mathrm{x} : x_i^2 = 1, i = 1, \ldots, n \} \right)$$

The $r = 2$ SOS relaxation is precisely the dual SDP problem of the basic SDP relaxation.

SOS Approach - Lasserre (2001), Parrilo (2000)

For each $r > 0$, define the approximation $\mathcal{K}_r \subseteq \mathcal{P}_d(S)$ as

$$\mathcal{K}_r := \left(\mathrm{SOS}_r + \sum_{i=1}^{m} g_i(x) \mathrm{SOS}_{r-\deg(g_i)} \right) \cap \mathrm{R}_d[x]$$

where SOS_d denotes the cone of real polynomials of degree at most d that are SOSs of polynomials, and $\mathrm{R}_d[x]$ denotes the set of polynomials in the variables x of degree at most d.

The corresponding relaxation can be written as

$$\begin{array}{llll} (\mathrm{L}_r) & z_r = \inf\limits_{\lambda,\sigma_i} & \lambda & \\ & \text{s.t.} & \lambda - f(x) = \sigma_0(x) + \sum_{i=1}^m \sigma_i(x) g_i(x) \\ & & \sigma_0(x) \text{ is SOS of degree} \leq r \\ & & \sigma_i(x) \text{ is SOS of degree} \leq r - \deg(g_i(x)), \; i = 1, \ldots, m. \end{array}$$

Solving the SOS Relaxation

For each r, the relaxation (L_r) can be cast as an SDP problem, since $\sigma(x)$ is a SOS of degree $2k$ if and only if

$$\sigma(x) = \begin{pmatrix} 1 \\ \vdots \\ x_i \\ \vdots \\ x_i x_j \\ \vdots \\ \prod_{|k|} x \end{pmatrix}^T M \begin{pmatrix} 1 \\ \vdots \\ x_i \\ \vdots \\ x_i x_j \\ \vdots \\ \prod_{|k|} x \end{pmatrix} \quad \text{with} \quad M \succeq 0.$$

Note that $\mathrm{SOS}_d = \mathrm{SOS}_{d-1}$ for every odd degree d.

Lasserre's Hierarchy for a Small Example

To solve

$$
\begin{array}{ll}
\sup_{x,y} & -(x-1)^2-(y-1)^2 \\
\text{s.t.} & x^2-4xy-1 \geq 0 \\
& yx-3 \geq 0 \\
& y^2-4 \geq 0 \\
& 12^2-(x-2)^2-4(y-1)^2 \geq 0
\end{array}
$$

r	2	4	6
# vars	14	73	245
# constraints	6	15	28
Bound	9.40	36.06	51.73

There is no need to run relaxations for $r > 6$, because an optimal solution (and optimality certificate) can be extracted from the solution to the SDP problem L_6.

Assessing the SOS Relaxation

Good news : Under mild conditions, $z_r \to z$.

Bad news : For a problem with n variables and m inequality constraints, the size of the relaxation is :

- One psd matrix of dimension $\binom{n+r}{r}$;
- m psd matrices, each of dimension $\binom{n+r-\deg(g_i)}{r-\deg(g_i)}$
- $\binom{n+r}{r}$ linear constraints.

To overcome the size blow-up one may exploit the structure (sparsity, symmetry, convexity) to get smaller SDP programs.
Many authors have contributed here : Gatermann, Helton, Kim, Kojima, Lasserre, Netzer, Nie, Parrilo, Pasechnik, Riener, Schweighofer, Sotirov, Theobald, etc.

One idea : Improve the bound without growing r

Recall

$$\text{(POP)} \quad z = \sup \; f(x) \quad \text{s.t.} \quad x \in S := \{x : g_i(x) \geq 0, \; i = 1, \ldots, m\}$$

$$(\mathrm{L}_r(G)) \quad z_r(G) = \inf_{\lambda, \sigma_i} \; \lambda$$
$$\text{s.t.} \quad \lambda - f(x) = \sigma_0(x) + \textstyle\sum_{i=1}^m \sigma_i(x) g_i(x)$$
$$\sigma_0(x) \text{ is SOS of degree} \leq r$$
$$\sigma_i(x) \text{ is SOS of degree} \leq r - \deg(g_i(x)),$$
$$i = 1, \ldots, m.$$

Observe that

- (L_r) is defined in terms of the functions used to describe S
- Call this set $G = \{g_i(x) : i = 1, \ldots, m\}$

Ghaddar, Vera, Anjos (2011) : Improve the description of S by growing G in such a way that the bound obtained from L_r improves for fixed r.

Another idea : Pure SOC Relaxations

For binary quadratic POPs, it is possible to obtain purely second-order cone (SOC) relaxations.

Let us separate the various types of constraints as follows :

$$
\begin{array}{lll}
\max & x^T Q x + p^T x & \\
\text{s.t.} & a_j^T x = b_j & \forall j \in \{1, \ldots, t\} \\
& c_j^T x \leq d_j & \forall j \in \{1, \ldots, u\} \\
& x^T F_j x + e_j^T x = k_j & \forall j \in \{1, \ldots, v\} \\
& x^T G_j x + h_j^T x \leq l_j & \forall j \in \{1, \ldots, w\} \\
& x_i \in \{-1, 1\} & \forall i \in \{1, \cdots, n\}
\end{array}
$$

Useful Lemma

$x \in \{-1,1\}^n \Rightarrow \|x\|^2 = n$.

This leads to the following specialized lemma that we will use :

Lemma

If $f(x)$ is a polynomial of degree one and $\mathcal{B}' := \{x : \|x\|^2 = n\}$, then

$$f(x) \in \mathcal{P}_1(\mathcal{B}') \text{ if and only if } f(x) = f^T \begin{pmatrix} \sqrt{n} \\ x \end{pmatrix}$$

with $f \in SOC^{n+1}$.

First Relaxation

Using the previous lemma, we obtain **(BQPP$_{ss}$)** :

$$\min \lambda$$

$$\text{s.t. } \lambda - (x^T Q x + p^T x) = \begin{pmatrix} 1 & x^T \end{pmatrix} M \begin{pmatrix} 1 \\ x \end{pmatrix}$$

$$+ \sum_i (1 + x_i)\alpha_i^T \begin{pmatrix} \sqrt{n} \\ x \end{pmatrix} + \sum_i (1 - x_i)\beta_i^T \begin{pmatrix} \sqrt{n} \\ x \end{pmatrix} + \sum_i \gamma_i (1 - x_i^2)$$

$$+ \sum_j \delta_j(x)(b_j - a_j^T x) + \sum_j (d_j - c_j^T x)\eta_j^T \begin{pmatrix} \sqrt{n} \\ x \end{pmatrix}$$

$$+ \sum_j \theta_j (k_j - x^T F_j x - e_j^T x) + \sum_j \xi_j (l_j - x^T G_j x - h_j^T x),$$

$$M \in \mathcal{S}_+^{n+1}, \quad \alpha_i, \beta_i, \eta_j \in \mathcal{L}^{n+1}, \quad \delta_j \in \mathbb{R}_1[x], \quad \gamma_i, \theta_j \in \mathbb{R}, \quad \xi_j \in \mathbb{R}_+$$

Second (Improved) Relaxation

Adding products of linear constraints strengthens further : $\mathbf{(BQPP_{ss^+})}$

$$\min \lambda \quad \text{s.t. } \lambda - (x^T Q x + p^T x) = \begin{pmatrix} 1 & x^T \end{pmatrix} M \begin{pmatrix} 1 \\ x \end{pmatrix}$$

$$+ \sum_i (1 + x_i)\alpha_i^T \begin{pmatrix} \sqrt{n} \\ x \end{pmatrix} + \sum_i (1 - x_i)\beta_i^T \begin{pmatrix} \sqrt{n} \\ x \end{pmatrix} + \sum_i \gamma_i (1 - x_i^2)$$

$$+ \sum_j \delta_j(x)(b_j - a_j^T x) + \sum_j (d_j - c_j^T x)\eta_j^T \begin{pmatrix} \sqrt{n} \\ x \end{pmatrix}$$

$$+ \sum_j \theta_j (k_j - x^T F_j x - e_j^T x) + \sum_j \xi_j (l_j - x^T G_j x - h_j^T x)$$

$$+ \sum_{i,k} \sigma_{ik}(d_k - c_k^T x)(1 + x_i) + \sum_{i,k} \mu_{ik}(d_k - c_k^T x)(1 - x_i)$$

$$+ \sum_{k \le l} \nu_{kl}(d_k - c_k^T x)(d_l - c_l^T x) + \sum_{i \le j} \tau_{ij}(1 - x_i)(1 - x_j)$$

$$+ \sum_{i \le j} \omega_{ij}(1 + x_i)(1 + x_j) + \sum_{i,j} \phi_{ij}(1 - x_i)(1 + x_j)$$

$$M \in \mathcal{S}_+^{n+1}, \quad \alpha_i, \beta_i, \eta_j \in \mathcal{L}^{n+1}, \quad \gamma_i, \theta_j \in \mathbb{R}, \quad \xi_j, \sigma_{ik}, \mu_{ik}, \nu_{kl}, \tau_{ij}, \omega_{ij}, \phi_{ij} \in \mathbb{R}_+$$

Pure SOC Relaxation

We can relax ($BQPP_{ss}$) by removing the SOS term.
In the absence of the SOS term, the valid inequalities

$$-1 \leq x_i x_j \leq 1$$

are no longer satisfied, and may strengthen the relaxation **($BQPP_{soc}$)** :

$$\min \lambda \qquad \text{s.t. } \lambda - (x^T Q x + p^T x) =$$

$$\sum_i (1 + x_i)\alpha_i^T \begin{pmatrix} \sqrt{n} \\ x \end{pmatrix} + \sum_i (1 - x_i)\beta_i^T \begin{pmatrix} \sqrt{n} \\ x \end{pmatrix} + \sum_i \gamma_i (1 - x_i^2)$$

$$+ \sum_j \delta_j(x)(b_j - a_j^T x) + \sum_j (d_j - c_j^T x)\eta_j^T \begin{pmatrix} \sqrt{n} \\ x \end{pmatrix}$$

$$+ \sum_j \theta_j (k_j - x^T F_j x - e_j^T x) + \sum_j \xi_j (l_j - x^T G_j x - h_j^T x)$$

$$+ \sum_{i<j} \nu_{ij}^+ (1 + x_i x_j) + \sum_{i<j} \nu_{ij}^- (1 + x_i x_j)$$

$$\alpha_i, \beta_i, \eta_j \in \mathcal{L}^{n+1}, \quad \delta_j \in \mathbb{R}_1[x], \quad \gamma_i, \theta_j \in \mathbb{R}, \quad \xi_j, \nu_{ij}^+, \nu_{ij}^- \in \mathbb{R}_+$$

Lasserre Relaxation for BQPPs

Lasserre introduced SDP relaxations for binary polynomial programs of the form :

$$\Gamma_r := \Big(\Psi_{r+2} + \sum_i (1 - x_i^2)\Psi_r + \sum_i (x_i^2 - 1)\Psi_r + \sum_i (b_i - a_i^T x)\Psi_r + \sum_i (a_i^T x - b_i)\Psi_r + \sum_i (d_i - c_i^T x)\Psi_r + \sum_i (k_i - x^T F_i x - e_i^T x)\Psi_r + \sum_i (x^T F_i x + e_i^T x - k_i)\Psi_r + \sum_i (l_i - x^T G_i x - h_i^T x)\Psi_r \Big) \cap \mathbb{R}_2[x],$$

for even $r \geq 0$. Taking $r = 0$, we obtain a relaxation for BQPP :

$$(\mathbf{BQPP}_{\mathbf{Las}}) \ \min \lambda \text{s.t.} \lambda - q(x) \in \Gamma_0$$

Theorem

$$\lambda^*_{BQPP_{Las}} \geq \lambda^*_{BQPP_{SS}} \geq \lambda^*_{BQPP_{SS^+}} \geq z^*_{BQPP}$$

In conclusion...

- Semidefinite, conic and polynomial optimization is a very active and exciting research area.
- There are still numerous open questions, both theoretical and algorithmic.
- Furthermore, this area is ripe for real-world applications.