Convex Optimization in Machine Learning and Computational Finance Lecture 7: Conic Optimization and Applications

Dr. Michel Baes, Pr. Patrick Cheridito

RiskLab / ETH Zürich

Quick recall of last week's lecture

- Combining subgradients:
  Subgradient of a maximum
  Subgradient of a sum
- KKT Conditions.
  Link with duality.
  Geometrical view of KKT Conditions.
- Application: projection on a hyperplane.
  Another interpretation of dual variables: reaction forces.

Classes of problems and complexity

$$\min\Big\{f(x) : g(x) \succeq_K b, x \in X\Big\}.$$

$f : \mathbb{R}^n \to \mathbb{R}\cup\{+\infty\}$, $g : \mathbb{R}^n \to (\mathbb{R}\cup\{+\infty\})^m$, $b \in \mathbb{R}^m$, $X \subseteq \mathbb{R}^n$,
$K \subseteq \mathbb{R}^m$ is a closed convex cone with no straight line.

A *class* of problems is a subset of these problems
Convex problems: f convex, g K-concave, X convex.
Integer programming: f, g linear, $K = \mathbb{R}^k_+ \times \{0\}$, $X \subseteq \mathbb{Z}^n$.
Semidefinite problems: f, g linear, $K = \mathbb{S}^N_+$, $X = \mathbb{R}^n$.
(and so on...)

A class is usually parametrized by n, m, (k, N if needed) the required accuracy, and other relevant quantities (e.g. various condition numbers, radius of X,...)

Classes of problems and complexity

$$\min \left\{ f(x) : g(x) \succeq_K b, x \in X \right\}.$$

$f : \mathbb{R}^n \to \mathbb{R} \cup \{+\infty\}$, $g : \mathbb{R}^n \to (\mathbb{R} \cup \{+\infty\})^m$, $b \in \mathbb{R}^m$, $X \subseteq \mathbb{R}^n$,
$K \subseteq \mathbb{R}^m$ is a closed convex cone with no straight line.

Let $\mathcal{M}$ be an optimization method for a class $\mathcal{C}$ of problems.

The *worst-case complexity* of $\mathcal{M}$ on $\mathcal{C}$
is the number of *elementary operations*
(simple scalar function evaluations)
that $\mathcal{M}$ takes to solve the most difficult problem of $\mathcal{C}$.

It is usually a function of n, m, the required accuracy,
and the other parameters defining $\mathcal{C}$.

The Big O notation:
dropping unnecessary details

A sequence $(x_n)_{n\geq 0}$ is in $\mathcal{O}(f(n))$ (or $x_n = \mathcal{O}(f(n))$) iff $\exists M > 0$, $n' \in \mathbb{N}$ such that $|x_n| \leq Mf(n)$ for $n \geq n'$.

- We don't say what M is.
- Example: $x_n = 3n^3 + n^2 \leq 4n^3 = \mathcal{O}(n^3)$
- $\mathcal{O}(f(n)) = x_n$ has no meaning.
- $\mathcal{O}(f(n))\mathcal{O}(g(n)) = \mathcal{O}(f(n)g(n))$.

Illustration: how does $n(\sqrt[n]{n} - 1)$ grow as $n \to \infty$?
$\sqrt[n]{n} = \exp(\ln n^{1/n}) = \exp(\ln n/n) = 1 + \ln n/n + \mathcal{O}(\ln^2 n/n^2)$,
thus $n(\sqrt[n]{n} - 1) = n(\ln n/n + \mathcal{O}(\ln^2 n/n^2)) = \ln n + \mathcal{O}(\ln^2 n/n)$,
and grows as $\ln n$.

Problems you can solve

Least-square problems

$$\min_{x \in \mathbb{R}^n} ||Ax - b||_2^2,$$

where $A \in \mathbb{R}^{m \times n}$ is full rank, $b \in \mathbb{R}^m$.

- Analytical solution (use optimality conditions):
  $x^* = A^{\dagger} b = (A^T A)^{-1} A^T b$.
- Solved e.g. by the `backslash` operation ("\") in Matlab.
- Complexity: $\mathcal{O}(mn^2)$, or even less if A is structured (with many zeros, ...).

Problems you can solve

Linear optimization problems

$$\min\{c^T x : a_i^T x \leq b_i,\ 1 \leq i \leq m\},$$

where $a_i \in \mathbb{R}^n$ for $1 \leq i \leq m$, $b \in \mathbb{R}^m$, $c \in \mathbb{R}^n$.

- No analytical solution.
- Solved reliably by mature software.
- Complexity: theoretically $\mathcal{O}(m^{3/2}n^2)$,
  in practice, a few dozen of least-square problems $\mathcal{O}(mn^2)$.
  or even less if A is structured (with many zeros, ...).

Problems you can solve
Convex optimization problems

$$\min\{f(x) : g_i(x) \le b_i,\ 1 \le i \le m\},$$

where $f, g_1, \ldots, g_m : \mathbb{R}^n \to \mathbb{R} \cup \{+\infty\}$ are convex, $b \in \mathbb{R}^m$.

- No analytical solution. (In fact, NP-Hard in general).
- **Most** of them are well-solved by efficient software.
- Complexity: polynomial in m, n, and in the complexity of evaluating f, g_i and their (sub)gradient/Hessian.
- Convexity is often difficult to recognize.

How do we recognize those convex problems
that can be efficiently solved?
How to use the software that can solve them?

Every convex optimization problem is a conic one

$$\begin{array}{ll} \min & \langle c, x \rangle \\ \text{s.t.} & Ax = b \\ & x \in K \end{array} \geq \begin{array}{ll} \max & \langle y, b \rangle \\ \text{s.t.} & A^*y + s = c \\ & s \in K^* \end{array}$$

Here, $A \in \mathbb{R}^{m \times n}$ with full row rank, $b \in \mathbb{R}^m$, $c \in \mathbb{R}^n$,
$K \subseteq \mathbb{R}^n$ is a closed convex cone with nonempty interior,
$\langle \cdot, \cdot \rangle$ is the scalar product on $\mathbb{R}^m$ or on $\mathbb{R}^n$,
A^* satisfies $\langle Au, v \rangle = \langle A^*v, u \rangle$ for $u \in \mathbb{R}^n, v \in \mathbb{R}^m$,
and K^* is the dual of K wrt $\langle \cdot, \cdot \rangle$.

Motivations:

1. Concentrate all the nonlinearities in just one object (K).

2. Benefitting from a powerful duality theory.

Some conic problems you can actually solve

$$\begin{array}{ll} \min & \langle c, x\rangle \\ \text{s.t.} & Ax = b \\ & x \in K \end{array} \quad \geq \quad \begin{array}{ll} \max & \langle y, b\rangle \\ \text{s.t.} & A^* y + s = c \\ & s \in K^* \end{array}$$

Important case: $K = K^*$ for a well chosen scalar product.

e.g. $K = \mathbb{R}^n_+$ with $\langle u, v\rangle = u^T v$ (*linear programming*)

$K = \mathbb{L}^k_+$, the light-cone of the k-dimensional Euclidean unit ball, with $\langle u, v\rangle = 2u^T v$ (*second-order cone programming*)

$K = \mathbb{S}^N_+$ with $\langle \cdot, \cdot\rangle_F$ (*semidefinite programming*)

K is any product of these.

These problems can be solved efficiently with:
SeDuMi (written in C, runs on Matlab, well maintained)
Other solvers: SDPT3 (Matlab), CSDP (written in C), MOSEK (commercial software).

Second-order cone problems

$$f^* = \min\{f^T x : ||A_i x + b_i||_2 \leq c_i^T x + d_i, 1 \leq i \leq \ell\},$$

where $A_i \in \mathbb{R}^{k_i \times N}$, $b_i \in \mathbb{R}^{k_i}$, $c_i \in \mathbb{R}^N$, $d_i \in \mathbb{R}$, $f \in \mathbb{R}^N$.

- It corresponds exactly to conic optimization with $K = \mathbb{L}_+^{k_1} \times \cdots \times \mathbb{L}_+^{k_\ell} \times \mathbb{R}^N$ (see blackboard).
- More general than linear programming $(A_i = 0, b_i = 0)$.
- More general than quadratic programming ($c_i = 0$ for all i except one).
- If one inequality "$\leq$" is reversed, the problem might **not be convex** anymore.

An important application: robust linear programming

$$f^* = \min\{c^T x : a^T x \le b\},$$

Sometimes, (a, b, c) are not known exactly.
Suppose we only know that $a \in \mathcal{E} = \{Py + \hat{a} : ||y||_2 \le 1\}$.
Ascertaining that the constraint is **always** satisfied:

$$\begin{aligned}
f^*_{\text{robust}} &= \min\{c^T x : a^T x \le b \; \forall a \in \mathcal{E}\} \\
&= \min\{c^T x : \max_{a \in \mathcal{E}} a^T x \le b\} \\
&= \min\{c^T x : \max_{||y||_2 \le 1} (Py)^T x + \hat{a}^T x \le b\} \\
&= \min\{c^T x : ||P^T x||_2 + \hat{a}^T x \le b\}
\end{aligned}$$

Generalizable to many constraints;
we can also have ellipsoid uncertainty on b, c
[Ben-Tal, Nemirovski].

The minimal surface problem

Given a function $f : C \subseteq \mathbb{R}^2 \to \mathbb{R}$, the *surface of its graph* is given by:

$$S(f) = \int_C \sqrt{1 + ||f'(x)||^2} dx$$

Given some values for f e.g. on the boundary of C, find the function f with smallest surface.

- Discretize f on a regular mesh: $\{x_{ij}\}_{i,j} = \{(hi, hj)\}_{i,j}$
- Approximate $f'(x_{ij}) \simeq \widehat{f}'(x_{ij}) := \frac{1}{h} \begin{pmatrix} f(x_{i+1,j}) - f(x_{ij}) \\ f(x_{i,j+1}) - f(x_{ij}) \end{pmatrix}$
- The objective is approximated by $\sum_{ij} \left\| \begin{pmatrix} \widehat{f}'(x_{ij}) \\ 1 \end{pmatrix} \right\|_2$.
- Constraints are linear in the variables f_{ij}. $\Rightarrow$ SOCP

What can be expressed with SOC constraints?

$$f^* = \min\{f^T x : ||A_i x + b_i||_2 \leq c_i^T x + d_i, 1 \leq i \leq m\},$$

A *conic quadratic representable set (CQr set)* is a set of the form:

$$S = \{y : \exists z \left\| A_i \begin{pmatrix} y \\ z \end{pmatrix} + b_i \right\|_2 \leq c_i^T \begin{pmatrix} y \\ z \end{pmatrix} + d_i,\ 1 \leq i \leq m\}.$$

As $\min\{f^T y : y \in S\}$ is a second-order cone problem, we can solve it using an SOC solver.

A function F is *CQr* if epi$F$ is CQr; $\min\{F(y) : y \in S\}$ is an SOCP, and thus also solvable.

What can be expressed with SOC constraints?

- **Affine functions**
- **Euclidean norm**
- **Ellipsoids**
- **Squared Euclidean norm**: $x \mapsto ||x||_2^2$

$$\{(t,x) : ||x||_2^2 \le t\} = \left\{(t,x) : \left\| \begin{pmatrix} x \\ (t-1)/2 \end{pmatrix} \right\|_2 \le \frac{t+1}{2} \right\}.$$

- **Hyperbola branch**: $S := \{(s,t) : st \ge 1, t > 0\}$

$$S = \left\{(s,t) : \left\| \begin{pmatrix} 1 \\ (t-s)/2 \end{pmatrix} \right\|_2 \le \frac{t+s}{2} \right\}.$$

- Intersections, Minkowski sums, light cones, and cartesian products of CQr sets are CQr.
- Sums, affine substitution of the argument of CQr fcts.

Semidefinite Programming

Since people realized that such optimization problems can be solved efficiently *(Nesterov, Nemirovski, 1992)*, they discovered hundreds of new applications.

$$f^* = \min\{\langle C, X\rangle_F : \mathcal{A}(X) = b, X \in \mathbb{S}^N_+\},$$

where $C \in \mathbb{S}^N$, $b \in \mathbb{R}^m$, $\mathcal{A} : \mathbb{S}^N \to \mathbb{R}^m$ is linear

(**e.g.** $\mathcal{A}(X) = \text{diag}(X)$, $\mathcal{A}(X) = X_{ij}, \mathcal{A}(X) = \text{Tr}(X), \ldots$)

What can we solve with an SDP solver?

$$f^* = \min\{\langle C, X\rangle_F : \mathcal{A}(X) = b, X \in \mathbb{S}^N_+\}$$

Dual: $d^* = \max\{\langle b, y\rangle : C - \mathcal{A}^* y \in \mathbb{S}^N_+\}$

A *semidefinite representable set (SDr set)*
is a set of the form (with $\hat{\mathcal{A}}$ linear $\mathbb{R}^{m+s} \to \mathbb{S}^N$ and $\hat{B} \in \mathbb{S}^N$):

$$S = \{y \in \mathbb{R}^m : \exists z \in \mathbb{R}^s \text{ for which } \hat{\mathcal{A}}\begin{pmatrix} y \\ z \end{pmatrix} - \hat{B} \in \mathbb{S}^N_+\},$$

A function F is *SDr* if epi$F$ is SDr.

What can be expressed with SDP constraints?

- Intersection, Minkowski sum, light cone, and cartesian product of SDr sets is SDr.
- Sum, affine substitution of the argument keep SDr
- **The maximal eigenvalue** $\lambda_{\max}(\cdot)$ is SDr (and therefore a convex function): $(t, X) \in \text{epi}\lambda_{\max}$ iff $t \geq \lambda_{\max}(X)$ iff $tI - X \in \mathbb{S}^N_+$.

Proof: use the spectral decomposition theorem for X: $X = \sum_{i=1}^N \lambda_i(X) P_i$. Then $tI - X = \sum_{i=1}^N (t - \lambda_i(X)) P_i$ is a spectral decomposition of$tI - X$.
This matrix is in $\mathbb{S}^N_+$ iff $t - \lambda_i(X) \geq 0$ for all i iff $t \geq \lambda_{\max}(X)$. ∎

- Similarly, $-\lambda_{\min}$ is SDr.

An ubiquitous tool in SDP: Schur's Lemma

Notation simplifications:
$X \in \mathbb{S}^N_+$ iff $X \succeq 0$, $X \in \mathbb{S}^N_{++}$ iff $X \succ 0$.

Lemma 1 (Schur) *Let*

$$X := \begin{pmatrix} A & B^T \\ B & C \end{pmatrix} \in \mathbb{S}^N,$$

and assume $C \in \mathbb{S}^k_{++}$.
Then $X \succeq 0$ *iff* $A \succeq B^T C^{-1} B$ *and* $X \succ 0$ *iff* $A \succ B^T C^{-1} B$.

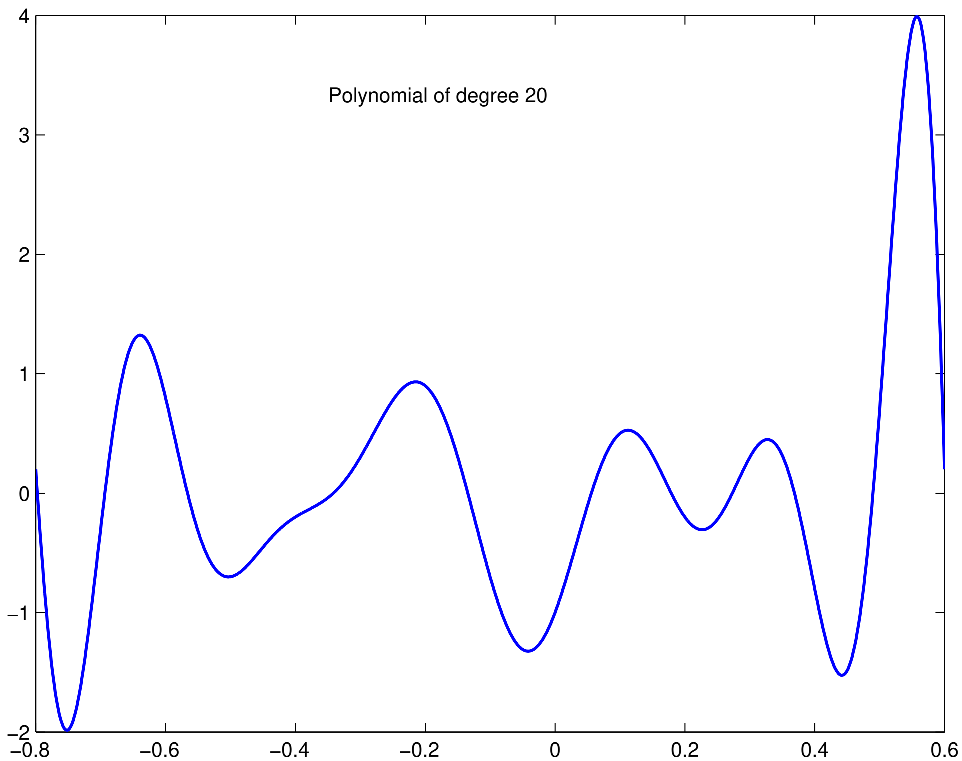
Proof: Do a Gauss elimination of the off-diagonal block.
i.e. compute $U^T X U$, where $U := \begin{pmatrix} I & 0 \\ -C^{-1}B & I \end{pmatrix}$. ∎

Just a few consequences of Schur's Lemma

- Every CQr set/function is an SDr set/function.
  It suffices to show that $S := \{(t,x) : ||x||_2 \le t\}$ is SDr.
  By Schur, $(t,x) \in S$ iff $\begin{pmatrix} t & x^T \\ x & tI \end{pmatrix} \succeq 0$.
- $S := \{(X,Y) : X \succeq Y^{-1}\}$ is SDr:
  $(X,Y) \in S$ iff $\begin{pmatrix} X & I \\ I & Y \end{pmatrix} \succeq 0$.
- $S := \{(X,Y) : X \succeq Y^{2}\}$ is SDr:
  $(X,Y) \in S$ iff $\begin{pmatrix} X & Y \\ Y & I \end{pmatrix} \succeq 0$.
- In fact, $\mathrm{Tr}(X^k)$ is SDr for all $k \in \mathbb{Z}$.

Many others consequences are known [Ben-Tal, Nemirovski]

I. Minimizing a univariate polynomial on an interval



As counterintuitive as it seems,
finding the minimal value of a univariate polynomial
is a convex problem

(and many other problems
involving univariate polynomials).

The first question: Showing that a polynomial is nonnegative

Is $p(t) = t^4 + 4t^3 + 8t^2 + 4t + 5$ nonnegative?

Observation:
a sum of squared polynomials is nonnegative.

> Can every nonnegative polynomial of degree $2d$ with n variables be written as a sum of squares?

Theorem 1 (Hilbert) *True for and only for:*

- *univariate polynomials ($n = 1$, $d \geq 0$);*
- *polynomials of degree 2 ($d = 1$, $n \geq 1$);*
- *polynomial of degree 4 with 2 variables ($d = n = 2$).*

For univariate polynomials, this is a semidefinite problem

Theorem 2 (Shor, 1983) *A univariate polynomial $p(t)$ of degree $2d$ is nonnegative iff $\exists Q \in \mathbb{S}_+^{d+1}$ such that*

$$p(t) = \pi(t)^T Q \pi(t),$$

with $\pi(t) := (1, t, \cdots, t^d)^T$.

Proof: $\Rightarrow$: Suppose $p(t) \geq 0 \; \forall t \in \mathbb{R}$. By Hilbert:

$$p(t) = \sum_{i=1}^{N} (a_i^T \pi(t))^2 = \sum_{i=1}^{N} \pi(t)^T a_i a_i^T \pi(t) = \pi(t)^T Q \pi(t),$$

with $Q := \sum_{i=1}^{N} a_i a_i^T$. Of course, $Q \in \mathbb{S}_+^{d+1}$.

$\Leftarrow$: If Q is semidefinite, then $p(t) = \pi(t)^T Q \pi(t) \geq 0$ for all $t \in \mathbb{R}$. ∎

Indeed, you have linear and semidefinite constraints

$$\begin{array}{ll} \min_h & \langle c, h\rangle \\ \text{s.t.} & Ah = b \\ & p(t) \geq 0 \text{ for all } t \in \mathbb{R} \end{array}$$

$$\Updownarrow$$

$$\begin{array}{ll} \min_{h,Q} & \langle c, h\rangle \\ \text{s.t.} & Ah = b \\ & q_{00} = h_0 \\ & q_{01} + q_{10} = h_1 \\ & q_{02} + q_{11} + q_{20} = h_2 \\ & \cdots \\ & q_{dd} = h_{2d} \\ & Q \in \mathbb{S}_+^{d+1} \end{array}$$

Here $p(t) = \sum_{i=0}^{2d} h_i t^i$.

$Ah = b$ can represent constraints such as:

- $h_i = \alpha$,
- $p(\bar{t}) = \beta$,
- $p'(\bar{t}) = \gamma$,
- $\int_l^u p(t)dt = \delta$, ...

For minimizing $p(t)$

(wlog) with $p(0) = 0$:

$$\begin{array}{lll} \min & h_0 & \\ \text{s.t.} & p(t) + h_0 \geq 0 & \forall t \end{array}$$

Polynomials positive on an interval

Let $p(t) := \sum_{i=0}^{d} h_i t^i$.

Observation: $p(t) \geq 0$ for $t \in \mathbb{R}_+$ iff $p(t^2) \geq 0$ for $t \in \mathbb{R}$.

Another observation:
$p(t) \geq 0$ for $t \geq a$ iff $p(t^2 + a) \geq 0$ for $t \in \mathbb{R}$.

Again another observation:
$p(t) \geq 0$ for $0 \leq t \leq 1$ iff $p(1/(t^2 + 1)) \geq 0$ for $t \in \mathbb{R}$. Also,

$$g(t) := (t^2 + 1)^d \cdot p\left(\frac{1}{t^2 + 1}\right) = \sum_{i=0}^{d} h_i (t^2 + 1)^{d-i}$$

is a polynomial, and $p(1/(t^2 + 1)) \geq 0$ iff $g(t) \geq 0$.

Exercise: how to ensure that $p(t) \geq 0$ for $a \leq t \leq b$.

II. Splitting a graph in two

Scheduling tasks between two processors

Tasks

#	length	weight
1	10'	3
2	5'	6
3	10'	4
4	15'	1

p_j: length of task j in min
C_j: completion time of task j
The total cost of task j is $w_j C_j$,
where w_j is the *weight* of task j

If #1 and #2 are performed on the same processor,

- (#1, #2) costs $10 \times 3 + 15 \times 6 = 120$, $[C_1 = 10, C_2 = 15]$
- (#2, #1) costs $5 \times 6 + 15 \times 3 = 75$. $[C_2 = 5, C_1 = 15]$

The extra cost of having #1 and #2 on the same processor is $\min\{120, 75\} - 10 \times 3 - 5 \times 6 = 15$.

Scheduling tasks between two processors

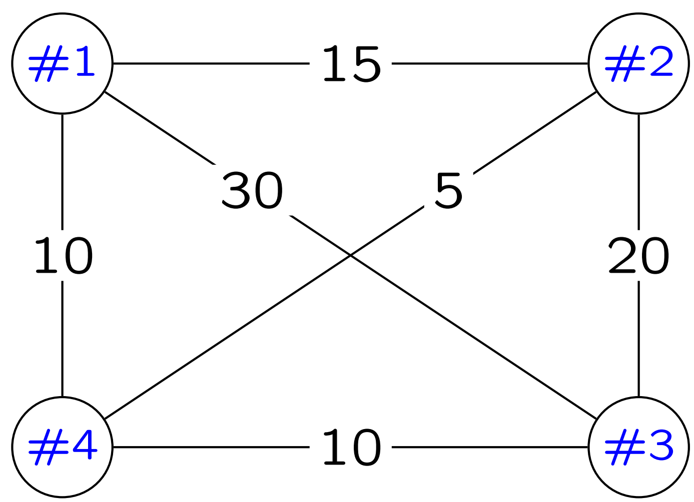
Tasks

#	length	weight
1	10'	3
2	5'	6
3	10'	4
4	15'	1

p_j: length of task j in min
C_j: completion time of task j
The total cost of task j is $w_j C_j$, where w_j is the *weight* of task j

Let $G = (V, E)$ be a complete undirected graph, with $V :=$ {**tasks**}, weight on arc $\{i, j\}$ is $\min\{w_i p_j, w_j p_i\}$.

Find the maximal cut



Scheduling tasks between two processors

Tasks

#	length	weight
1	10'	3
2	5'	6
3	10'	4
4	15'	1

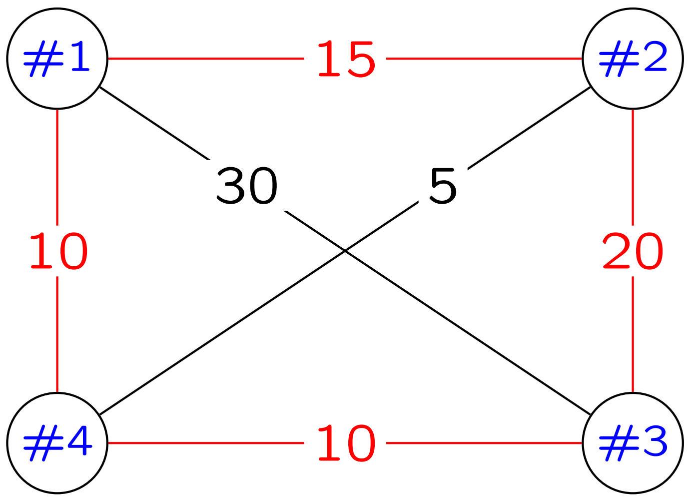
p_j: length of task j in min

C_j: completion time of task j

The total cost of task j is $w_j C_j$, where w_j is the *weight* of task j

Let $G = (V, E)$ be a complete undirected graph, with $V :=$ {**tasks**}, weight on arc $\{i, j\}$ is $\min\{w_i p_j, w_j p_i\}$.

Find the maximal cut



55

Scheduling tasks between two processors

Tasks

#	length	weight
1	10'	3
2	5'	6
3	10'	4
4	15'	1

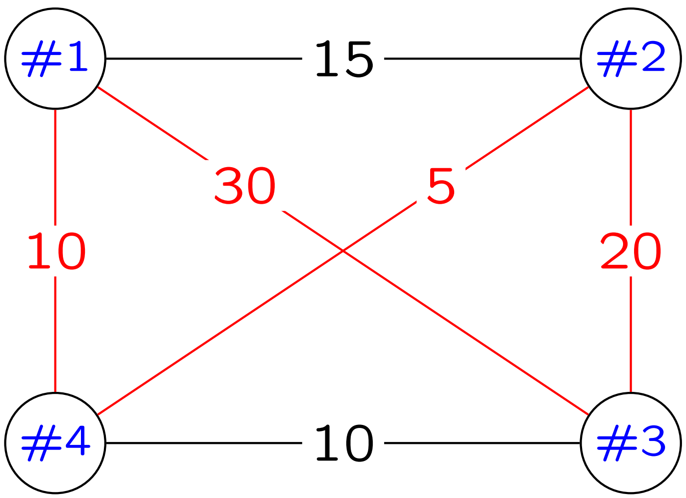
p_j: length of task j in min

C_j: completion time of task j

The total cost of task j is $w_j C_j$, where w_j is the *weight* of task j

Let $G = (V, E)$ be a complete undirected graph, with $V :=$ {**tasks**}, weight on arc $\{i, j\}$ is $\min\{w_i p_j, w_j p_i\}$.

Find the maximal cut



65

The MAXCUT problem

Let $G = (V, E)$ be an undirected graph, $n := |V|$,
$w_{\{ij\}} \geq 0$: weight of arc $\{ij\} \in E$.

$$\forall S \subseteq V, \quad \text{cut}(S) := \sum \{w_{\{ij\}} : i \in S, j \in V \setminus S\}$$

Problem: find $S \subseteq V$ maximizing cut(S).

Modeling:

$$f^* := \max \left\{ \sum_{i<j} w_{ij} \frac{1 - x_i x_j}{2} : x_i^2 = 1, 1 \leq i \leq n \right\} :$$

two nodes have different values -1 and $+1$
iff they are in different classes (S and $V \setminus S$)

▶ **Highly non-convex problem**, (and even NP-Hard)

MAXCUT is well-approximated by SDP

$$f^* := \max \left\{ \sum_{i<j} w_{ij} \frac{1 - x_i x_j}{2} : x_i^2 = 1, 1 \leq i \leq n \right\}.$$

Idea: replace xx^T by a matrix X (i.e. $X_{ij} = x_i x_j$).

$$f^* = \max \left\{ \sum_{i<j} w_{ij} \frac{1 - X_{ij}}{2} : X_{ii} = 1, 1 \leq i \leq n, \; X \in \mathbb{S}^n_+, \mathrm{rank}(X) = 1 \right\}.$$

The rank constraint is not convex. We just suppress it.

$$f^*_{\mathrm{SDP}} := \max \left\{ \sum_{i<j} w_{ij} \frac{1 - X_{ij}}{2} : X_{ii} = 1, 1 \leq i \leq n, \; X \in \mathbb{S}^n_+ \right\}.$$

Theorem: *(Goemans, Williamson)* $f^*_{\mathrm{SDP}} \geq f^* \geq 0.878 f^*_{\mathrm{SDP}}$.

What do you do with the optimal X^*?

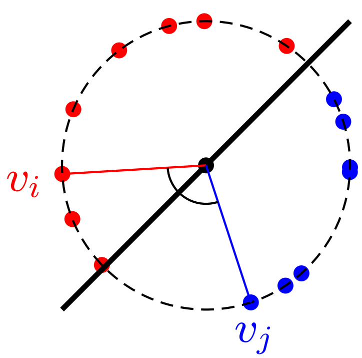
$$X^* := \arg\max \left\{ \sum_{i<j} w_{ij} \frac{1 - X_{ij}}{2} : X_{ii} = 1, 1 \le i \le n,\ X \in \mathbb{S}^n_+ \right\}.$$

If $\text{rank}(X^*) = 1$, then $X^* = x^* x^{*T}$, and x^* is the MAXCUT solution.

Otherwise, let $k := \text{rank}(X^*)$ and $X^* = VV^T$, $V = \mathbb{R}^{n \times k}$.

Let $V^T = [v_1, \ldots, v_n]$, with $v_i \in \mathbb{R}^k$, so that $X_{ij} = v_i^T v_j$ (and $||v_i||_2 = 1$).

Pick a random $u \in \mathbb{R}^k$ such that $||u||_2 = 1$, and set $\hat{x}_i := \text{sign}\langle u, v_i \rangle$.



What is $\mathbb{E}\left[\sum_{i<j} w_{ij}(1 - \hat{x}_i \hat{x}_j)/2\right]$ $(\le f^*)$?

$E_{ij} = \mathbb{E}\left[(1 - \hat{x}_i \hat{x}_j)/2\right]$ is the probability that i and j are in different classes:

$$E_{ij} = P[\text{sign}\langle u, v_i \rangle \ne \text{sign}\langle u, v_j \rangle]$$
$$= \text{acos}(v_j^T v_i)/\pi = \text{acos}(X_{ij})/\pi.$$

Now, $\text{acos}(X_{ij})/\pi \ge \alpha(1 - X_{ij})/2$

for $\alpha = 0.87856... = \min_{-1 \le t \le 1} \frac{2\text{acos}(t)}{\pi(1-t)}$.

III. An application in control
Proving the stability of some constrained systems

There are many applications of SDP in Control

$$\begin{aligned} dx(t)/dt &= Ax(t) + Bu(t) \\ y(t) &= Cx(t) \\ x(0) &= x_0. \end{aligned}$$

$x : \mathbb{R}^n \to \mathbb{R}$ is the *state* of the system, accessible only indirectly.

$u : \mathbb{R}^m \to \mathbb{R}$ is the *input* of the system, which we control.

$y : \mathbb{R}^m \to \mathbb{R}$ is the *output* of the system, which we measure.

Many results in linear control use SDP in a crucial way:

- Lyapunov stability $B = 0$, $C = 0$.
- Extension of Lyapunov: the S-Lemma.
- Kalman-Yakubovitch-Popov Lemma and positive real systems.

The idea of Lyapunov: creating a potential function

Assume first
$B = 0$ and $C = 0$:

$$\begin{aligned} dx(t)/dt &= Ax(t) \\ x(0) &= x_0. \end{aligned}$$

Will $x(t) \to 0$ as $t \to \infty$?

▶ **Applications (a.o.)**: RLC electrical circuits, mechanical systems with masses, dampeners and springs

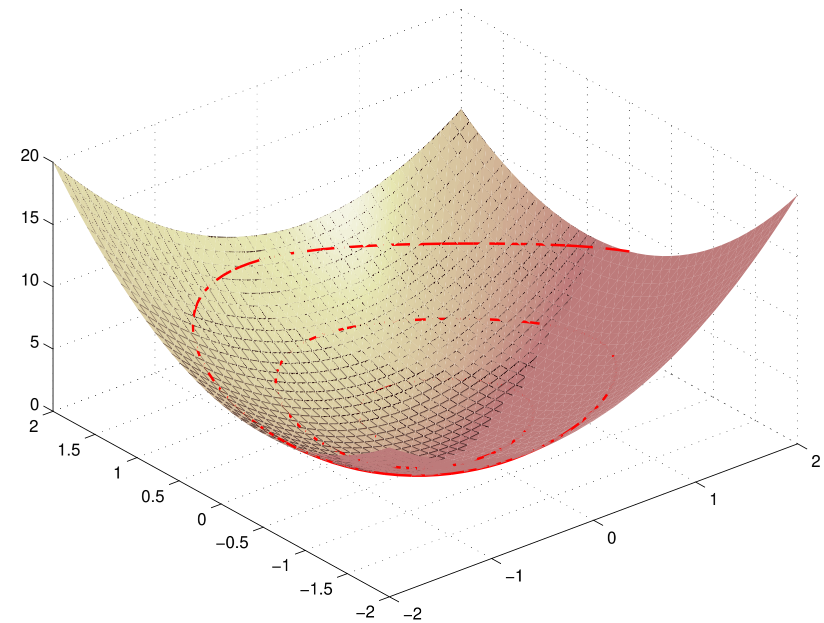
The idea of Lyapunov: creating a potential

$$dx(t)/dt = Ax(t), \;\; x(0) = x_0.$$

Find a *Lyapunov function* $V : \mathbb{R}^n \to \mathbb{R}_+$ such that:

1. $V(x) = 0$ implies $x = 0$

2. $dV(x(t))/dt < 0$ for any trajectory $x(t) \neq 0$



The idea of Lyapunov: creating a potential

$$dx(t)/dt = Ax(t), \;\; x(0) = x_0.$$

Find a *Lyapunov function* $V : \mathbb{R}^n \to \mathbb{R}_+$ such that:
1. $V(x) = 0$ implies $x = 0$
2. $dV(x(t))/dt < 0$ for any trajectory $x(t) \neq 0$.

Consider quadratic functions: $V(x) = \langle x, Px \rangle$, with $P \in \mathbb{S}^n$.
We need $P \in \mathbb{S}^n_{++}$. Also, with $\langle u, v \rangle := u^T v$:

$$dV(x(t))/dt = \langle dx/dt, Px \rangle + \langle x, P dx/dt \rangle$$

$$= \langle Ax, Px \rangle + \langle x, PAx \rangle = \langle x, (A^T P + PA)x \rangle < 0,$$

i.e. the system is stable when there exists a matrix
$P \in \mathbb{S}^n_{++}$ such that $-(A^T P + PA) \in \mathbb{S}^n_{++}$.

- We have a **Convex feasibility problem**:
  find a point in a given convex set

What happens if we can measure ($C \neq 0$) and control ($B \neq 0$) the system

$$
\begin{aligned}
dx(t)/dt &= Ax(t) + Bu(t) \\
y(t) &= Cx(t) \\
x(0) &= x_0.
\end{aligned}
$$

Questions:

Will $x(t)$ eventually vanish for a well-chosen $u(t)$?

Will $x(t)$ eventually vanish if $u(t)$ can be anything in a set?

We consider here a quadratic domain:

$$q(y, u) = q(Cx, u) \geq 0,$$

where q is a polynomial of degree 2.

What happens if we can control the system

Again, we try to find a quadratic Lyapunov function
$L(x) := \langle x, Px \rangle$ with $P \succ 0$
such that $dL(x(t))/dt < 0$ for feasible $x(t)$:

$$f_P(x, w) := dL(x(t))/dt = \langle dx/dt, Px \rangle + \langle x, P dx/dt \rangle$$

$$= \langle x, P(Ax + Bw) \rangle + \langle (Ax + Bw), Px \rangle < 0$$

for $q(Cx, w) \geq 0$.

Is $\{(x, w) : f_P(x, w) \geq 0, q(Cx, w) \geq 0\}$
empty for a matrix $P \succ 0$?

Interestingly, the S-Lemma gives you the matrix P as well.

A convex cone in $\mathbb{R}^2$

Let $A, B \in \mathbb{S}^n$. The set $Q := \{(x^T Ax, x^T Bx) : x \in \mathbb{R}^n\}$
is a convex cone [Dines]

Proof: Q is trivially a cone. We'll show its convexity.
We write $f(z) = z^T Az$, $g(z) := z^T Bz$. Let $\lambda \in [0,1]$, $u = (f(x), g(x))$, $v = (f(y), g(y)) \in Q$, and $w := \lambda u + (1-\lambda)v$. It suffices to find $z \in \text{span}\{x, y\}$ for which $w = (f(z), g(z))$. Wlog, $f(y)g(x) > g(y)f(x)$ (if equality, we can take $z = Cy$ for some C). In polar coordinates, $z = \rho z(\theta) = \rho(x \cos(\theta) + y \sin(\theta))$. We need to solve in ρ, θ:

$$\rho^2 f(z(\theta)) = \lambda f(x) + (1-\lambda) f(y), \quad \rho^2 g(z(\theta)) = \lambda g(x) + (1-\lambda) g(y).$$

Eliminating ρ^2 (from which we get $\rho(\theta)$) and isolating λ, we get:

$$\lambda = \frac{f(z(\theta))g(y) - f(y)g(z(\theta))}{(f(x) - f(y))g(z(\theta)) - (g(x) - g(y))f(z(\theta))} = \frac{N(\theta)}{D(\theta)} = \lambda(\theta)$$

D is quadratic in $\sin(\theta), \cos(\theta)$ and $D(\pi/2) = D(0) > 0$,

so $D(\theta) = a + b \sin(2\theta)$. Assume $b \geq 0$, so $D(\theta) > 0$ on $I = [0, \pi/2]$.

Then $\theta \mapsto \lambda(\theta)$ is continuous on I. As $\lambda(0) = 0$ and $\lambda(\pi/2) = 1$,

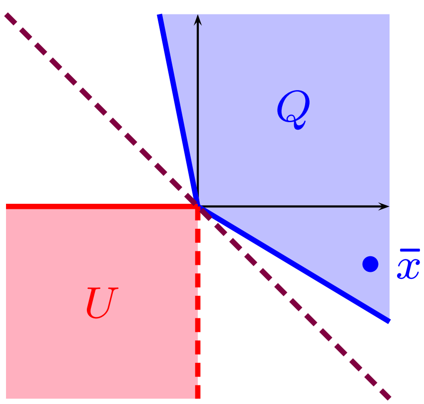
there exists a $\theta^* \in I$ with $\lambda(\theta^*) = \lambda$. Then $z := \rho(\theta^*)z(\theta^*)$ works. ∎

The S-Lemma by separation

Theorem 3 *Let $f, g : \mathbb{R}^n \to \mathbb{R}$ be two quadratic functions such that there exists $\bar{x} \in \mathbb{R}^n$ for which $g(\bar{x}) < 0$.*

$$S := \{x \in \mathbb{R}^n : f(x) < 0, g(x) \leq 0\} = \emptyset$$

$$\textbf{iff} \quad \exists\, y \geq 0 : f(x) + yg(x) \geq 0 \;\; \forall x \in \mathbb{R}^n.$$



Proof: $\Leftarrow$ is evident. $\Rightarrow$ (for f, g homogeneous: $f(x) = x^T Ax$, $g(x) = x^T Bx$).

Let $U := \{(a, b) : a < 0, b \leq 0\}$ and Q as above.

Since $S = \emptyset$, $Q \cap U = \emptyset$. As Q is convex (and $0 \in Q$), there is $(\mu, \nu) \neq 0$ for which $\mu a + \nu b \geq 0 \geq \mu a' + \nu b'$ when $(a, b) \in Q$, $(a', b') \in U$. Thus $\nu \geq 0$ ($a' \uparrow 0$) and $\mu \geq 0$ ($b' = 0$). Thanks to $\bar{x}$, $\mu \neq 0$. Wlog, $\mu = 1$. ∎

The S-Lemma: some observations

Remarks:

- Consider $f^* = \min\{f(x) : g(x) \leq 0\}$
  with $f(x) = x^T Ax$, $g(x) = x^T Bx$, and $\lambda_{\min}(B) < 0$.
  Then $f^* \geq 0$ iff $\max\{y : A + yB \in \mathbb{S}^N_+\} \geq 0$.
- A simple homogenization trick proves the full statement
  (Exercise: how to extend the above pair
  of optimization problems to non-homogenous f, g?)
- *Slater condition* ($\exists\, \bar{x} : g(\bar{x}) < 0$) is essential:
  consider $f(x) = -x^2$, $g(x) = 1$.
- There is no result so amenable for SDP
  when there is more than one quadratic constraint.
- Deep result, with a lot of applications in many fields.

[Polik, Terlaky, "A survey of the S-Lemma", SIAM Review 49 (2007).]

For next time

Other applications of convex optimization
in Machine Learning and Mathematical finance
(Tentative list:
Approximation,
Lasso optimization,
Coalition and Shapley's Core Theorem,
Maximum likelihood problems)