ADVANCES IN CONVEX OPTIMIZATION: CONIC PROGRAMMING

Arkadi Nemirovski

Georgia Institute of Technology
and
Technion – Israel Institute of Technology

- Convex Programming – “solvable case” in Optimization
- Revealing structure of convex programs: Conic Programming
- Exploiting structure of convex programs: Interior Point polynomial time algorithms
- Conic Quadratic and Semidefinite Programming: expressive abilities and applications

Revealing Structure: Conic Form of a Convex Problem

- When passing from a Linear Programming program

$$\min_{x\in\mathbb{R}^n}\left\{c^Tx : b - Ax \leq 0\right\}$$

to convex ones, the traditional way is to replace linear objective c^Tx and linear left hand sides of the constraints with convex functions.

- A much more productive way is to "make nonlinear" the coordinate-wise vector inequality $u \leq v \Leftrightarrow v - u \in \mathbb{R}^n_+$ in $b - Ax \leq 0$ by replacing it with a more general vector inequality

$$u \leq_{\mathbf{K}} v \;\Leftrightarrow\; v - u \in \mathbf{K}$$
$$[\mathbf{K} \subset \mathbb{R}^n\text{: convex pointed closed cone, } \mathrm{int}\,\mathbf{K} \neq \emptyset]$$

thus arriving at convex programs in the conic form:

$$\min_{x\in\mathbb{R}^n}\left\{c^Tx : b - Ax \leq_{\mathbf{K}} 0\right\}$$

- (c, A, b): data • $\mathbf{K}$: structure

• Conic problem: $\min_{x\in\mathbf{R}^n}\left\{c^Tx : Ax - b \geq_{\mathbf{K}} 0\right\}$

• Every convex problem can be reformulated equivalently as a conic one. However: a general convex cone has no more structure than a general convex function. So what is the point?

Fact: "Nearly all" interesting for applications convex problems are covered by *just 3* generic conic problems:

• Linear Programming: K is a direct product $\mathbb{R}^n_+$ of rays $\mathbb{R}_+$:

$$\min_x\left\{c^Tx : Ax - b \geq 0\right\} \qquad \text{(LP)}$$

• Conic Quadratic Programming: K is a direct product of Lorentz cones $\mathbf{L}^n = \left\{x \in \mathbb{R}^n : x_n \geq \left(\sum_{i=1}^{n-1} x_i^2\right)^{1/2}\right\}$:

$$\min_x\left\{c^Tx : \|A_ix - b_i\|_2 \leq c_i^Tx - d_i,\ i = 1, \ldots, m\right\} \qquad \text{(CQP)}$$

• SemiDefinite Programming: K is a direct product of semidefinite cones $\mathbf{S}^n_+ = \{X = X^T \in \mathbf{R}^{n\times n} : X \succeq 0 [\Leftrightarrow x^TXx \geq 0 \forall x]\}$:

$$\min_x\left\{c^Tx : A_0 + \sum_{i=1}^n x_iA_i \succeq 0\right\} \qquad \text{(SDP)}$$

Note: LP $\subset$ CQP $\subset$ SDP

- Good news about Conic Programming, especially LP/CQP/SDP:
- Fully symmetric and "algorithmic" duality allowing for instructive processing of conic programs "on paper" and heavily utilized by solution algorithms
- Existence of theoretically *and practically* powerful algorithms — Polynomial Time Interior Point Methods
- Extremely powerful "expressive abilities" of CQP/SDP
  $\Rightarrow$ huge spectrum of applications

Conic Duality

• Duality in MP is about building *lower bounds* on the optimal value in an optimization program, i.e., about certifying *negative* statements "*there is no feasible solution with the value of the objective < ...*"

• For conic problems, Fenchel-Lagrange duality becomes fully symmetric and "algorithmic":

$$(P): \quad \mathrm{Opt}(P) = \min_x \{c^T x : \overbrace{Ax - b}^{\xi} \geq_{\mathbf{K}} 0\} \quad \Leftrightarrow \quad \min_\xi \{e^T \xi : \xi \in [\mathcal{L} - b] \bigcap \mathbf{K}\}$$

$$[e : A^T e = c, \mathcal{L} = \mathrm{Im} A]$$

$$\Downarrow \text{[F.–L. Duality]}$$

$$(D): \; \mathrm{Opt}(D) = \max_\lambda \{b^T \lambda : A^T \lambda = c, \lambda \geq_{\mathbf{K}_*} 0\} \Leftrightarrow \max_\lambda \{b^T \lambda : \lambda \in [\mathcal{L}^\perp + e] \bigcap \mathbf{K}_*\}$$

$$[\mathbf{K}_* = \{\lambda : \lambda^T \xi \geq 0 \; \forall \xi \in \mathbf{K}\}]$$

$$\mathrm{Opt}(P) = \min_x \left\{ c^T x : Ax - b \geq_{\mathbf{K}} 0 \right\} \qquad (P)$$

$$\mathrm{Opt}(D) = \max_\lambda \left\{ b^T \lambda : A^T \lambda = c, \lambda \geq_{\mathbf{K}_*} 0 \right\} \quad (D)$$

Conic Duality Theorem:

- [Symmetry] **Conic duality is fully symmetric: the dual problem is conic, and its dual is (equivalent to) the primal problem**
- [Weak Duality] $\mathrm{Opt}(D) \leq \mathrm{Opt}(P)$
- [Strong Duality] **Let one of the problems (P), (D) be strictly feasible and bounded. Then the other problem is solvable, and**

$$\mathrm{Opt(D)} = \mathrm{Opt(P)}.$$

In particular, if both (P), (D) are strictly feasible, then both are solvable with equal optimal values, and a primal-dual feasible pair (x, λ) is primal-dual optimal iff

$$c^T x - b^T \lambda = 0 \qquad \Leftrightarrow \qquad [Ax - b]^T \lambda = 0.$$

• Conic Duality

$$\mathrm{Opt}(P) = \min_x \{c^T x : Ax - b \geq_{\mathbf{K}} 0\} \quad (P)$$

$$\mathrm{Opt}(D) = \max_\lambda \{b^T\lambda : A^T\lambda = c, \lambda \geq_{\mathbf{K}_*} 0\} \quad (D)$$

is a special case of **Lagrange Duality: If convex problem**

$$\mathrm{Opt}(\mathrm{Pr}) = \min_x \{f(x) : g_i(x) \leq 0, 1 \leq i \leq m\}$$

is strictly feasible and bounded, then its Lagrange dual

$$\mathrm{Opt}(\mathrm{Dl}) = \max_{\lambda \geq 0} L(\lambda), \quad L(\lambda) \equiv \inf_x \{f(x) + \sum_i \lambda_i g_i(x)\}$$

is solvable, and $\mathrm{Opt}(\mathrm{Pr}) = \mathrm{Opt}(\mathrm{Dl})$.

In contrast to the general Lagrange Duality, Conic Duality is

- **fully symmetric** – (D) "remembers" (P).
- **completely algorithmic** – passing from (P) to (D) is a purely mechanical process.

• Algorithmic nature of Convex Duality makes it a powerful tool for instructive analytical – "on paper" – processing conic programs.

Example: Truss Topology Design. A *truss* is a mechanical construction, like electric mast, railroad bridge, or Eiffel Tower, comprised of thin elastic *bars* linked to each other at *nodes*.

In a TTD problem, one is given

- a 2D/3D *nodal set*,
- a set of *tentative bars* – allowed pair connections of nodes,
- a set of *loading scenarios* – collections of forces acting at the nodes,

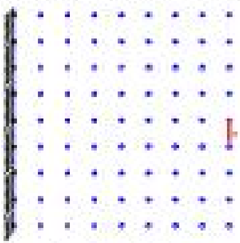
and looks for a construction of a given weight which is the most stiffest w.r.t. the scenario loads.

• Stiffness of a truss w.r.t. a load is quantified by *compliance* – the potential energy capacitated by the truss as a result of its deformation under the load (the less is compliance, the better).

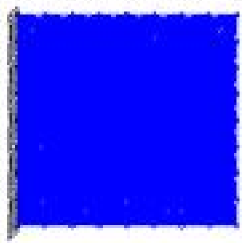
$$\min_{\tau, t_i}\left\{\tau : \left[\begin{array}{c|c} 2\tau & f_\ell^T \\ \hline f_\ell & \sum_{i=1}^{N} t_i b_i b_i^T \end{array}\right] \succeq 0, 1 \le \ell \le K, t \ge 0, \sum_i t_i \le W \right\}$$

- t_i: bar volumes
- $f_\ell \in \mathbb{R}^M$: loads (M: total # of nodal degrees of freedom)
- τ: upper bound on the worst-case, w.r.t. loads f_ℓ, $1 \le \ell \le K$, compliance

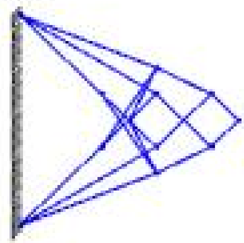
- In TTD, one starts with a "dense" nodal grid and allows for all air connections of nodes by bars. At the optimum, most of the bars get zero volume, thus revealing the optimal topology:



9×9 nodal set ($M = 144$) and loading scenario



81 tentative nodes and 2,039 tentative bars



optimal console 12 nodes, 32 bars

• In order to capture topology design, one should work with dense grids (M of order of few thousands)
$\Rightarrow$ The design dimension $N = O(M^2)$ of the TTD is in the range of millions...

• **Cure: Semidefinite Duality.** In the dual of TTD, most of the variables can be eliminated analytically, which results in the problem of dimension $\approx MK << N = O(M^2)$:

$$\min_{\alpha,v,\gamma}\left\{-2\sum_\ell f_\ell^T v_\ell + W\gamma : \begin{array}{l} \left[\begin{array}{c|ccc} \gamma & b_i^T v_1 & \ldots & b_i^T v_K \\ \hline b_i^T v_1 & \alpha_1 & & \\ \vdots & & \ddots & \\ b_i^T v_K & & & \alpha_K \end{array}\right] \succeq 0 \, \forall i \\ 2\sum_\ell \alpha_\ell = 1 \end{array}\right\}$$

• Taking dual to the (processed!) dual of TTD, we end up with instructive (and unexpected) equivalent *bar-stress* reformulation of the TTD problem:

$$\min_{\tau,t_i}\left\{\tau:\ \left[\begin{array}{c|c}2\tau & f_\ell^T\\ \hline f_\ell & \sum_{i=1}^N t_i b_i b_i^T\end{array}\right]\succeq 0\ \forall \ell,\ t\geq 0, \sum_i t_i \leq W\right\}$$

$$\Rightarrow \min_{\alpha,v,\gamma}\left\{-2\sum_\ell f_\ell^T v_\ell + W\gamma:\ \begin{array}{c}\left[\begin{array}{c|ccc}\gamma & b_i^T v_1 & \dots & b_i^T v_K\\ \hline b_i^T v_1 & \alpha_1 & & \\ \vdots & & \ddots & \\ b_i^T v_K & & & \alpha_K\end{array}\right]\succeq 0\ \forall i\\ 2\sum_\ell \alpha_\ell = 1\end{array}\right\}$$

$$\Rightarrow \min_{\tau,t,q}\left\{\tau:\ \sum_i \frac{q_{i\ell}^2}{2t_i}\leq\tau,\ \sum_i q_{i\ell} b_i = f_\ell\ \forall \ell,\ t\geq 0, \sum_i t_i \leq W\right\}$$

- Convex Programming – "solvable case" in Optimization
- Revealing structure of convex programs: Conic Programming
- Exploiting structure of convex programs: Interior Point polynomial time algorithms
- Conic Quadratic and Semidefinite Programming: expressive abilities and applications

- Convex Programming – “solvable case” in Optimization
- Revealing structure of convex programs: Conic Programming
- Exploiting structure of convex programs: Interior Point polynomial time algorithms
- Conic Quadratic and Semidefinite Programming: expressive abilities and applications

1979: polynomial time solvability of LP (Khachiyan) via the Ellipsoid Method

1984: the first IPM for LP (Karmarkar): theoretical efficiency *+ practical performance competitive with the one of the Simplex Method*

1986: first polynomial *path-following* IPMs for LP (Renegar, Gonzaga): *improved complexity bounds + transparent construction with potential for nonlinear extensions*

"Interior Point Revolution" (mid-1980's – late 1990's):

- developing new IPMs
- nonlinear extensions: general theory of IPMs in Convex Programming
- advanced theory (Nesterov & Todd, 1997-98) of IPMs for conic problems on *homogeneous self-dual cones* (LP/CQP/SDP)

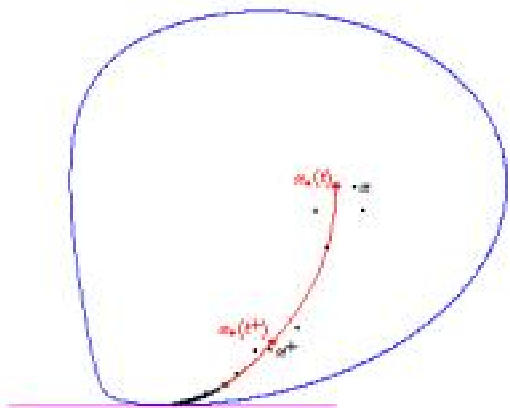
Polynomial Time IPMs: Path-Following Scheme

- **Path-Following Scheme (Fiacco & McCormic, 1968)** for solving convex program

$$\min_x \{c^T x : x \in G\}$$

- Equip G with a *barrier* – a C^2 function $F : \operatorname{int} G \to \mathbb{R}$ with $F''(\cdot) \succ 0$ and closed level sets $\{x \in \operatorname{int} G : F(x) \le a\}$;
- Trace the *path* $x_*(t) = \operatorname*{argmin}_{x \in \operatorname{int} G} F_t(x), \quad F_t(x) = tc^T x + F(x)$ as the *penalty parameter* $t \to \infty$:

Given (x, t) with x close to $x_*(t)$,
- replace t with $t^+ > t$;
- minimize $F_{t^+}(\cdot)$ by Newton method, x being the starting point, until a point x^+ close to $x_*(t^+)$ is built;
- replace $(x, t) \leftarrow (x^+, t^+)$ and loop



• It was discovered in late 1980's that the path-following scheme becomes polynomial when specific *self-concordant* barriers are used:

Let $G \subset \mathbb{R}^n$ be a convex domain. A C^3 convex function $F : \operatorname{int} G \to \mathbb{R}$ is called a *ϑ-self-concordant barrier* for G, if F is a barrier for G and $\forall (x \in \operatorname{int} G, h \in \mathbb{R}^n)$:

A. [self-concordance] $\left|D^3F(x)[h,h,h]\right| \leq 2\left(D^2F(x)[h,h]\right)^{3/2}$

B. [s.-c.b. quantification] $|DF(x)[h]| \leq \vartheta^{1/2}\left(D^2F(x)[h,h]\right)^{1/2}$

Interpretation: $D^2F(x)$ defines a local Euclidean metrics

$$\|h\|_x = \left(D^2F(x)[h,h]\right)^{1/2}.$$

A, B mean that $D^2F(\cdot)$ and $F(\cdot)$ are Lipschitz continuous w.r.t. this local metrics with constants 2 and $\vartheta^{1/2}$, respectively.

<u>Theorem.</u> Let $G \subset \mathbb{R}^n$ be a closed convex domain not containing lines, $c \in \mathbb{R}^n$ be such that the level sets $\{x \in G : c^T x \le a\}$ are bounded, and F be a ϑ-s.-c.b. for G. Then

(i) The path $x_*(t) = \underset{\mathrm{int}\,G}{\operatorname{argmin}}[tc^T x + F(x)]$, $t > 0$, is well-defined

(ii) Let us say that (x, t) is close to the path, if $t > 0$ and

$$\lambda(x,t) \equiv \left([\nabla F_t(x)]^T [\nabla^2 F_t(x)]^{-1} \nabla F_t(x)\right)^{1/2} \le 0.1 \qquad [F_t(x) = tc^T x + F(x)]$$

Given (x_0, t_0) close to the path, consider the recurrence

$$\begin{bmatrix} t_{i-1} \\ x_{i-1} \end{bmatrix} \mapsto \begin{bmatrix} t_i &= \exp\{0.1/\sqrt{\vartheta}\} t_{i-1} \\ x_i &= x_{i-1} - \frac{1}{1+\lambda(x_{i-1},t_i)} [\nabla^2 F_{t_i}(x_{i-1})]^{-1} \nabla F_{t_i}(x_{i-1}) \end{bmatrix}$$

Then all (x_i, t_i) are well-defined and close to the path, and

$$\forall i : c^T x_i - \min_G c^T x \le \frac{2\vartheta}{t_i} = \frac{2\vartheta}{t_0} \exp\{-0.1 i/\sqrt{\vartheta}\}.$$

Thus, every $O(1)\sqrt{\vartheta}$ steps add an accuracy digit.

• <u>Conclusion:</u> When we are smart enough to equip the feasible domain G of a convex problem $\min\limits_{x\in G} c^T x$ with an efficiently computable ϑ-s.-c.b. F with not too large ϑ, we get a polynomial time IPM for solving the problem.

<u>Note:</u> Every convex domain $G \subset \mathbb{R}^n$ admits $O(n)$-s.-c.b.. E.g., when G is a pointed cone, we can set

$$F(x) = O(1) \log \int\limits_{G_*} \exp\{-x^T \xi\} d\xi$$

- "Good" – efficiently computable – s.-c.b.'s are known for a wide variety of "basic" convex domains
- All standard convexity-preserving operations can be equipped with simple rules to combine good s.-c.b.'s for the operands into a good s.-c.b. for the result.

$\Rightarrow$ Essentially, the entire Convex Programming is within the grasp of polynomial time IPMs.

• The Interior Point constructions become maximally flexible as applied to conic problems on cones with many symmetries, most notably on *homogeneous self-dual cones*, which covers LP/SDP/CQP. The related theory is intrinsically linked to the theory of Euclidean Jordan Algebras.

In LP/CQP/SDP, one uses the self-concordant barriers as follows:

$\mathbf{K}$	$F_{\mathbf{K}}$	ϑ
$\mathbb{R}_+$	$-\ln(x)$	1
$\mathbf{L}^n$	$-\ln(x_n^2 - \sum_{i=1}^{n-1} x_i^2)$	2
$\mathbf{S}_+^n$	$-\ln\det X$	n
$\mathbf{K}_1 \times ... \times \mathbf{K}_m$	$F_{\mathbf{K}_1}(x^1) + ... + F_{\mathbf{K}_m}(x^m)$	$\sum_i \vartheta(F_{\mathbf{K}_i})$

and solves simultaneously the problem of interest and its dual ("*primal-dual* IPMs").

- Primal-dual LP/CQP/SDP IPMs underly the best known so far polynomial time complexity bounds for these generic problems and, in addition, allow for
  - on-line adjustable "long step" path-tracing policies
    $\Rightarrow$ in practice, much faster convergence than for the "off-line" worst-case-oriented penalty updating rule, with no risk to violate the $O(\sqrt{\vartheta})$ complexity bound
  - elegant way ("self-dual embedding") to initialize path-tracing
  - building infeasibility/unboundedness certificates,...
- Practical performance of primal-dual IPM's for LP/CQP/SDP is usually much better than the one predicted by the worst-case-oriented theoretical complexity analysis.

Challenge: *On extremely large-scale CQP/SDP problems ($10^4 - 10^6$ design variables), IPMs become too time-consuming. What to do?*

- Convex Programming – "solvable case" in Optimization
- Revealing structure of convex programs: Conic Programming
- Exploiting structure of convex programs: Interior Point polynomial time algorithms
- Conic Quadratic and Semidefinite Programming: expressive abilities and applications

- The initial form of a convex program usually is

$$\min_x \left\{ c^T x : x \in X = \bigcap_{i=1}^{m} X_i \right\} \qquad (*)$$

$[X_i = \{x : g_i(x) \leq 0\}$ **with convex** $g_i]$

⇒ **How to recognize that (∗) can be reformulated as a CQP/SDP program ?**

- <u>**Definition:**</u> **Let $\mathcal{F}$ be a family of cones. A set $X \subset \mathbb{R}^n$ is called** *$\mathcal{F}$-representable*, **if there exists a** *$\mathcal{F}$-representation* **of X:**

$$X = \{x : \exists u : Ax + Bu + d \geq_{\mathbf{K}} 0\}, \ \mathbf{K} \in \mathcal{F}$$

A function $f : \mathbb{R}^n \to \mathbb{R} \cup \{+\infty\}$ is called $\mathcal{F}$-representable, if so is its epigraph $\text{Epi}\{f\} = \{(t, x) : t \geq f(x)\}$.

• Mathematical Programming is about solving optimization problems of the form

$$\mathrm{Opt} = \min_{x\in\mathbb{R}^n}\{f(x) : g_i(x) \le 0, 1 \le i \le m\}$$

with "good enough" (usually C^1) objective $f(\cdot)$ and constraints $g_i(\cdot)$.

• MP is primarily operational: while the descriptive issues (existence/ uniqueness/characterization of a solution) are of definite importance, the major goal is to approximate an optimal solution numerically

⇒ The primary role in MP Theory is played by investigating complexity of generic MP problems and developing efficient solution algorithms.

- **Facts:**
  - $\mathcal{F}$-representations of functions g_i can be straightforwardly converted into $\mathcal{F}$-representations of the sets $X_i = \{x : g_i(x) \le 0\}$
  - $\mathcal{F}$-representations of sets X_i can be straightforwardly converted into an $\mathcal{F}$-representation of the set $X = \bigcap_{i=1}^{m} X_i$
  - Given a $\mathcal{F}$-representation $X = \{x : \exists u : Ax + Bu + d \in \mathbf{K}\}$ of X, a program

$$\min_X c^T x$$

can be reformulated equivalently as the $\mathcal{F}$-conic program

$$\min_{x,u} \left\{ c^T x : Ax + Bu + d \ge_{\mathbf{K}} 0 \right\}.$$

⇒ The question

"What can be expressed via CQP/SDP?"

can be posed as

"What are CQP/SDP-representable sets/functions?"

• Fact: Let $\mathcal{F}$ be a family of cones closed w.r.t. taking (finite) direct products and passing from a cone to its dual. There exists a simple "calculus" which shows that the family of $\mathcal{F}$-representable sets/functions is closed w.r.t. all basic convexity-preserving operations.

The calculus is "fully algorithmic" – an $\mathcal{F}$-representation of the result is readily given by $\mathcal{F}$-representations of the operands.

The convexity-preserving operations in question include:

- For sets: taking finite intersections, arithmetic sums, direct products, images/inverse images under affine mappings, conic hulls, convex hulls of finite unions, polars,...
- For functions: taking combinations with nonnegative coefficients, affine substitutions of arguments, partial minimization, superpositions with monotone outer functions, Legendre transforms,...

• To recognize $\mathcal{F}$-representability of a convex problem, one applies the outlined calculus to "raw materials" – basic $\mathcal{F}$-representable sets and functions. *"Expressive abilities" of the generic $\mathcal{F}$-conic problem depend on how rich is the collection of the associated basic sets/functions.*

$$
\left.
\begin{array}{l}
\text{(CQP)}\left\{
\begin{array}{|l|}
\hline
\min c^T x \text{ subject to} \\
\hline
(a): \quad Ax = b,\ x \geq 0 \\
\hline
(b): \quad
\begin{array}{l}
\left(\sum\limits_{i=1}^{8} |x_i|^5\right)^{1/5} \leq x_2^{1/7} x_3^{2/7} x_4^{3/7} + 2 x_1^{1/5} x_5^{2/5} x_6^{1/5} \\
5x_2 \geq \dfrac{1}{x_1^{1/2} x_3^2} + \dfrac{2}{x_2^{1/3} x_3^3 x_4^{5/8}} \\
\begin{bmatrix} x_2 & x_1 & & \\ x_1 & x_4 & x_3 & \\ & x_3 & x_6 & x_3 \\ & & x_3 & x_8 \end{bmatrix} \preceq 5I
\end{array} \\
\hline
\end{array}
\right. \\
\begin{array}{|l|}
(c): \quad
\begin{array}{l}
\begin{bmatrix} x_1 & x_2 - x_1 & x_3 - x_2 & x_4 - x_3 \\ x_2 - x_1 & x_2 & x_3 - x_2 & x_4 - x_3 \\ x_3 - x_2 & x_3 - x_2 & x_3 & x_4 - x_3 \\ x_4 - x_3 & x_4 - x_3 & x_4 - x_3 & x_4 \end{bmatrix} \succeq 0 \\
x_1 + x_2 \sin(\phi) + x_3 \sin(2\phi) + x_4 \sin(4\phi) \geq 0,\ 0 \leq \phi \leq \frac{\pi}{2}
\end{array} \\
\hline
\end{array}
\end{array}
\right\} \text{(SDP)}
$$

- The problem can be converted, *in a systematic way*, into SDP

$$\min_{z \in \mathbb{R}^{93}} \left\{ d^T z : A_0 + \sum\nolimits_i z_i A_i \succeq 0 \right\},$$

- Removing constraints (c), the problem becomes a CQP

$$\min_{w \in \mathbb{R}^{48}} \left\{ e^T w : \|P_i w + p_i\|_2 \leq q_i^T w + r_i, 1 \leq i \leq m \right\}$$

and can be approximated, *in a polynomial time fashion*, by LP.

Expressive Abilities of CQP

$$\min_x \left\{ c^T x : \|A_i x - b_i\|_2 \le c_i^T x - d_i,\ i = 1, ..., m \right\} \qquad \text{(CQP)}$$

- **Sample of CQP-representable functions/sets:**
- $\|\cdot\|_p$, $p \in \mathbf{Q}$
  $\Rightarrow$ *Approximation in* $\|\cdot\|_p$
- convex quadratic forms
  $\Rightarrow$ *Convex Quadratic Programming*
- power monomials $-x_1^{p_1} x_2^{p_2} ... x_n^{p_n}$, $x \ge 0$ ($p_i \in \mathbf{Q}_+$, $\sum_i p_i \le 1$),
  power monomials $x_1^{-p_1} x_2^{-p_2} ... x_n^{-p_n}$, $x > 0$ ($p_i \in \mathbf{Q}_+$)
  $\Rightarrow$ *Geometric Programming in power form*
- $f(x, t) = x^T (B^T \mathrm{Diag}\{t\} B)^{-1} x$, $t \in \mathbb{R}^n_{++}$
  $\Rightarrow$ *Truss Topology/Electric Circuit Design*

• "Whether CQP does exist?"

<u>Theorem.</u> The Lorentz cones admit fast polyhedral approximation. Specifically, for every $\epsilon \in (0, 0.1)$ and every n, one can point out

- a polyhedral cone $\mathrm{P} \subset \mathbb{R}^{\rfloor 2n\ln(1/\epsilon)\lfloor}$ given by an explicit system of $\rfloor 5n\ln(1/\epsilon)\lfloor$ homogeneous linear inequalities, and
- an explicit linear mapping $\mathcal{M} : \mathbb{R}^{\rfloor 2n\ln(1/\epsilon)\lfloor} \to \mathbb{R}^n$

such that $\mathcal{M}(\mathrm{P})$ is in-between L^n and the "$(1+\epsilon)$-extension" of L^n:

$$\underbrace{\mathrm{L}^n}_{\left\{ \begin{array}{c} z=(v,t)\in\mathbb{R}^n: \\ \|v\|_2 \le t \end{array} \right\}} \subset \mathcal{M}(\mathrm{P}) \subset \underbrace{\mathrm{L}^n_\epsilon}_{\left\{ \begin{array}{c} z=(v,t)\in\mathbb{R}^n: \\ \|v\|_2 \le (1+\epsilon)t \end{array} \right\}}$$

$\Rightarrow$ CQP can be reduced, *in a polynomial time fashion*, to LP.

Expressive Abilities of SDP

$$\min_x\{c^T x : \sum_i x_i A_i \succeq B\} \qquad \text{(SDP)}$$

- **Sample of SDP-representable functions/sets:**
- All CQP-representable functions/sets
- Symmetric functions of eigenvalues of symmetric matrices/singular values of rectangular matrices

  Theorem. Let $f(x) : \mathbb{R}^n \to \mathbb{R} \cup \{+\infty\}$ be symmetric and SDP-representable. Then $F(X) = f(\lambda_1(X), ..., \lambda_n(X)) : \mathbf{S}^n \to \mathbb{R} \cup \{+\infty\}$ is SDP-representable as well.

- The cones of (coefficients of) univariate algebraic/trigonometric polynomials of a given degree nonnegative on a given segment

  Theorem. For a segment $\Delta \subset \mathbb{R}$, the sets

  $$P_d^n(\Delta) = \{(A_0, ..., A_d) \in (\mathbf{S}^n)^{d+1} : A_0 + tA_1 + ... + t^d A_d \succeq 0 \forall t \in \Delta\}$$

  are SDP-representable with explicit SDP representations.

  $\Rightarrow$ Minimization of a univariate algebraic/trigonometric polynomial over a segment is an SDP program.

- **Challenge:** *Complete description of SDP-representable sets.*

Is it true that

— a *convex* semialgebraic set is SDP-representable?

— the epigraph of a *convex* algebraic polynomial is SDP-representable?

(true in the univariate case)

- Due to its tremendous expressive abilities, SDP has a wide variety of applications, including those in
  - Relaxations of difficult combinatorial problems
  - Ellipsoidal approximations of convex sets
  - Statistics
  - Robust Control
  - Structural Design
  - Communications
  - Signal Processing,...

Permanent challenge: Extending the scope of applications – building SDP models for various problems of Engineering and Management

• **Example: Semidefinite Relaxations of Difficult problems**

A (nonconvex) quadratically constrained quadratic problem

$$\mathrm{Opt} = \max_x \left\{ x^T A_0 x + 2b_0^T x + c_0 : x^T A_i x + 2b_i^T x + c_i \le 0,\ 1 \le i \le m \right\} \qquad (*)$$

can be NP-hard. E.g., quadratic constraints can model Boolean restrictions on variables: $x_j^2 = x_j \Leftrightarrow x_j \in \{0,1\}$.

• Passing to the matrix variable $X = \begin{bmatrix} 1 & x^T \\ x & xx^T \end{bmatrix}$, (*) becomes

$$\max_X \left\{ \mathrm{Tr}(\mathcal{A}_0 X) : \begin{array}{l} \mathrm{Tr}(\mathcal{A}_i X) \le 0,\ 1 \le i \le m, \\ X \succeq 0,\ X_{11} = 1,\ \mathrm{Rank}(X) = 1 \end{array} \right\} \qquad \left[\mathcal{A}_i = \begin{bmatrix} c_i & b_i^T \\ b_i & A_i \end{bmatrix} \right]$$

Eliminating the "troublemaking" rank constraint, we arrive at the *SDP relaxation of* (*)

$$[\mathrm{Opt} \le] \quad \mathrm{SDP} = \max_X \left\{ \mathrm{Tr}(\mathcal{A}_0 X) : \mathrm{Tr}(\mathcal{A}_i X) \le 0,\ 1 \le i \le m, X \succeq 0, X_{11} = 1 \right\}$$

• **Interpretation:** In the relaxation, we maximize the *expected value* of the original objective over *random* solutions satisfying *at average* the original constraints.

• In good cases, SDP relaxations yield *provably tight* bounds.

<u>Example:</u> It is NP-hard to compute

$$\begin{array}{l}\mathrm{Opt} = \max\limits_x \{x^T A x : \|x\|_\infty \le 1\} \equiv \max\limits_x \{x^T A x : x_i^2 \le 1, 1 \le i \le n\} \\ \qquad \le \mathrm{SDP} = \max\limits_X \{\mathrm{Tr}(AX) : X_{ii} \le 1, 1 \le i \le n, X \succeq 0\}\end{array}$$

even when 4%-accuracy is sought. However:

- A **is diagonal-dominated with nonpositive off-diagonal entries**

$\Rightarrow \mathrm{Opt} \le \mathrm{SDP} \le 1.1382\,\mathrm{Opt}$ [Goemans & Williamson, '95]

- $A \succeq 0 \Rightarrow \mathrm{Opt} \le \mathrm{SDP} \le \frac{\pi}{2}\mathrm{Opt}$ [Nesterov, '98]

$\Rightarrow$ **Tight approximations of matrix norms:** When $p > 2 > r \ge 1$, SDP yields a computable upper bound on the (computationally intractable!) matrix norm $\|A\|_{pr} = \max\{\|Ax\|_r : \|x\|_p \le 1\}$ tight within factor $\theta(p,r) \le \frac{3\pi}{6\sqrt{3}-2\pi} = 2.2936...$ (cf. the Grothendieck inequality ('53) dealing with $p = \infty, r = 1$; here the constant can be improved to $\frac{\pi}{2\ln(1+\sqrt{2})} \approx 1.7822...$)

- $\forall A$: $\mathrm{Opt} \le \mathrm{SDP} \le O(1)\ln(n+1)\mathrm{Opt}$ (valid with the unit box in $\mathbb{R}^n$ replaced by intersection of n centered at the origin ellipsoids in $\mathbb{R}^m$).

$$\mathrm{Opt} = \min_{x \in \mathbf{R}^n} \{f(x) : g_i(x) \le 0, 1 \le i \le m\}$$

- In late 1970's it was understood that

— Convex Programming (f and g_i are convex) is "computationally tractable": under mild computability and boundedness assumptions, generic Convex Programming problems admit efficient solution algorithms.

— In contrast to this, typical generic nonconvex problems seem to be intractable: no efficient algorithms for these problems are known, and, unless P=NP, no such algorithms exist.

- **A generic convex problem:** a family $\mathcal{P}$ of instances

$$\mathrm{Opt}(p) = \min_{x\in\mathbb{R}^{n(p)}} \left\{ f_p(x) : g_{i,p}(x) \le 0, 1 \le i \le m(p) \right\} \qquad (p)$$

such that

- within $\mathcal{P}$, an instance p can be identified by its **data vector** $\mathrm{Data}(p) \in \mathbb{R}^{N(p)}$
- all instances $p \in \mathcal{P}$ are convex.

Example: Linear Programming. The objective and the constraints in (p) are affine functions of x, and

$$\mathrm{Data}(p) = \left(m(p), n(p), \text{coefficients of } f_p, g_{1,p}, \ldots, g_{m(p),p} \right).$$

$$\mathcal{P} = \left\{ \mathrm{Opt}(p) = \min_{x\in\mathbb{R}^{n(p)}} \{f_p(x) : g_{i,p}(x) \le 0, 1 \le i \le m(p)\} \right\}$$

• **A solution algorithm for a generic problem $\mathcal{P}$:** a code $\mathcal{B}$ for an idealized Real Arithmetic computer which, given on input

- the data $\mathrm{Data}(p)$ of an instance $p \in \mathcal{P}$,
- a required accuracy $\epsilon > 0$,

produces in finitely many operations of precise Real Arithmetics

- either an ϵ-solution x_ϵ to p: $f_p(x_\epsilon) \le \mathrm{Opt} + \epsilon$ & $g_{i,p}(x_\epsilon) \le \epsilon \, \forall i$,
- or a correct claim that p is infeasible/below unbounded.

• **A solution algorithm is** *efficient* (≡*polynomial time*)**, if the # of operations is bounded by**

$$\mathrm{Poly}\Big(\underbrace{\dim \mathrm{Data}(p)}_{\mathrm{Size}(p)}, \underbrace{\log\left(\frac{\mathrm{Size}(p) + \|\mathrm{Data}(p)\|_\infty}{\epsilon}\right)}_{\mathrm{Digits}(p,\epsilon)}\Big).$$

- Theorem. Let $\mathcal{P}$ be a generic convex problem with instances

$$\mathrm{Opt}(p) = \min_{x\in\mathbb{R}^{n(p)}} \{f_p(x) : g_{i,p}(x) \le 0, i \le m(p), \|x\|_\infty \le 1\} \qquad (p)$$

normalized by the requirement

$$\forall (x \in \mathbb{R}^{n(p)}, \|x\|_\infty \le 1) : |f_p(x)| \le 1, |g_{i,p}(x)| \le 1, 1 \le i \le m(p).$$

There exists an explicit algorithm (Ellipsoid Method) which finds an ϵ-solution to (p), $0 < \epsilon < 1$, or detects correctly that (p) is infeasible, by computing $\left(\frac{\epsilon}{5n(p)}\right)$-accurate approximations to the values and the subgradients of $f_p, g_{i,p}$ along $3n^2(p)\ln(2n(p)/\epsilon)$ successively generated search points, with additional $O(1)n(p)(n(p)+m(p))$ operations per search point.

- Corollary. Under

  Computability Assumption: *Given the data* Data(p) *of an instance* $p \in \mathcal{P}$, *a tolerance* $\delta \in (0,1)$, *and* $x \in \mathbb{R}^{n(p)}$, $\|x\|_\infty \le 1$, *the values and subgradients of* $f_p, g_{i,p}$ *at* x *can be computed within accuracy* δ *in*

$$\mathrm{Poly}(\mathrm{Size}(p), \mathrm{Digits}(p,\delta))$$

  operations

$\mathcal{P}$ admits a polynomial time solution algorithm.

• A convex problem always has a lot of structure (otherwise, how could we know that the problem is convex?)

• "Universal" polynomial time algorithms, like the Ellipsoid method, are black box oriented: they utilize detailed a priori knowledge of the structure and the data of a convex problem for the only purpose to compute the objective and the constrains at a point.

⇒ Poor (although polynomial time) performance: the arithmetic cost of accuracy digit is *at least* $O(n^4)$, which makes impossible to solve in realistic time problems with just few hundreds of variables...

- *How to reveal and to utilize the structure?*

An answer is given by conic reformulations of Convex Programming problems.

- Convex Programming – "solvable case" in Optimization
- Revealing structure of convex programs: Conic Programming
- Exploiting structure of convex programs: Interior Point polynomial time algorithms
- Conic Quadratic and Semidefinite Programming: expressive abilities and applications