

Course:

Advanced Nonlinear Programming

a.k.a.

Lectures on Modern Convex Optimization

ISyE 7683 A Spring 2021

Remote Synchronous Course

- **Instructor:** Dr. Arkadi Nemirovski nemirovs@isye.gatech.edu,
Groseclose 446, In-person office hours: N/A
*You always can contact me by e-mail nemirovs@isye.gatech.edu and by
phone 404-429-1528 10:00am-9:00pm, except for time of our classes*
- **Teaching Assistant:** TBA
- **Classes:** Monday Wednesday 11:00-12:15
- **Lecture Notes, Transparencies, (Non-obligatory) Exercises:** course
site on Canvas and
<https://www2.isye.gatech.edu/~nemirovs/LMCOLN2021WithSol.pdf>
<https://www2.isye.gatech.edu/~nemirovs/LMCOTR2021.pdf>
<https://www2.isye.gatech.edu/~nemirovs/LMCOEXERCISES.pdf>
- **Grading Policy:**
Take Home Final Exam: 100%

Comments on Remote Lecturing

♠ *Given the circumstances, it could be worse:* my teaching *always* was based on self-contained transparencies and therefore does not suffer much by passing from in-person to remote mode.

♠ Our classes will be hold via **BlueJeans**.

Please ask questions!!!

(I always ask for questions and never got enough of them in in-person class...)

• *There is no such thing as “stupid question;” I’ll do my best to avoid stupid answers...*

♠ *Errare humanum est — To err is human.* When speaking, I can pronounce something opposite to what I wanted to pronounce.

Rule: *Text dominates speech.* If I pronounce “white,” and on transparency it is written “black,” you should trust in transparency.

♠ Pre-recorded lectures are available, and records of BlueJeans classes will become available as classes go, via Canvas.

A man searches for a lost wallet at the place where the wallet was lost.

A wise man searches at a place with enough light...

♣ Where should we search for a wallet? Where is “enough light” – what Optimization can do well?

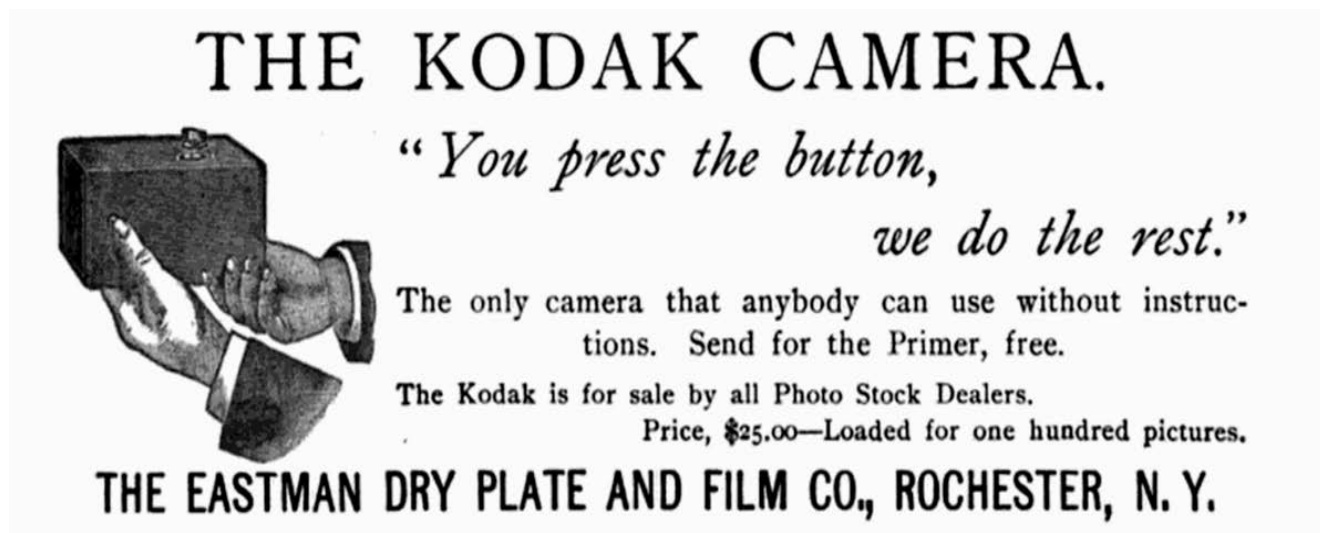
The most straightforward answer is: we can solve well *convex optimization problems*.

The very existence of what is called Mathematical Programming stemmed from discovery of Linear Programming (George Dantzig, late 1940's) – a modeling methodology accompanied by extremely powerful in practice (although “theoretically bad”) computational tool – Simplex Method. Linear Programming, which is a special case of Convex Programming, still underlies the majority of real life applications of Optimization, especially large-scale ones.

♠ When photography was invented in XIX Century, processing pictures was very sophisticated and required skills and training.

- Kodak Company changed the situation completely by offering (1888) centralized processing of films. Their slogan was

You press the button, we do the rest



♠ In the realm of Mathematical Programming, Convex Optimization is the area most close to this slogan, with

“pressing the button” = *creating convex optimization model of the problem at hand and feeding this model with necessary data*

♣ Around mid-1970's, it was shown that

- Linear and, more generally, Convex Programming problems are *efficiently solvable* – under mild computability and boundedness assumptions, generic Convex Programming problems admit numerical methods capable to approximate *globally optimal* solutions to *whatever high* accuracy in *reasonable* time — are *polynomial time* algorithms.

As applied to an instance of a generic problem, like Linear Programming

$$\mathcal{LP} = \left\{ \overbrace{\min_x \{c^T x : Ax \geq b\}}^{\text{instance}} : \begin{array}{l} A \in \mathbf{R}^{m \times n}, b \in \mathbf{R}^m, \\ c \in \mathbf{R}^n, m, n \in \mathbf{Z} \end{array} \right\},$$

a polynomial time algorithm solves it to a *whatever high required accuracy* ϵ in a number of arithmetic operations *polynomial* in the *size* of the instance (the number of data entries specifying the instance, $O(1)mn$ in the case of \mathcal{LP}) and the number $\ln(1/\epsilon)$ of required accuracy digits.

\Rightarrow *Theoretical (and to some extent – also practical) possibility to solve convex programs of reasonable size to high accuracy in reasonable time*

- No polynomial time algorithms for general-type nonconvex problems are known, and there are strong reasons to believe that no such methods exist.

⇒ Solving general nonconvex problems of not too small sizes is usually a highly unpredictable process: with luck, we can improve somehow the solution we start with, but we never know how nonoptimal we still are, and never have a reasonable a priori bound on how long it will take to achieve desired accuracy.

Polynomial Time Solvability of Convex Programming

♣ From purely academical viewpoint, polynomial time solvability of Convex Programming is a straightforward consequence of the following statement:

Theorem [circa 1976] *Consider a convex problem*

$$\text{Opt} = \min_{x \in \mathbb{R}^n} \left\{ f(x) : \begin{array}{l} g_i(x) \leq 0, \ 1 \leq i \leq m \\ |x_j| \leq 1, \ 1 \leq j \leq n \end{array} \right\}$$

normalized by the restriction

$$|f(x)| \leq 1, |g_j(x)| \leq 1 \ \forall x \in B = \{|x_j| \leq 1 \ \forall j\}.$$

For every $\epsilon \in (0, 1)$, one can find an ϵ -solution

$$x_\epsilon \in B : f(x_\epsilon) - \text{Opt} \leq \epsilon, g_i(x_\epsilon) \leq \epsilon \ \forall i$$

or to conclude correctly that the problem is infeasible at the cost of at most

$$3n^2 \ln \left(\frac{2n}{\epsilon} \right)$$

computations of the objective and the constraints, along with their (sub)gradients, at subsequently generated points of $\text{int } B$, with $O(1)n(n + m)$ additional arithmetic operations per every such computation.

♣ The outlined Theorem is sufficient to establish theoretical efficient solvability of generic Convex Programming problems. In particular, it underlies the famous result (Leo Khachiyan, 1979) on polynomial time solvability of \mathcal{LP} – the first ever mathematical result which made the C2 page of *New York Times* (Nov 27, 1979).

♣ From practical perspective, however, polynomial type algorithms suggested by Theorem are too slow: the arithmetic cost of an accuracy digit is at least

$$O(n^2n(m+n)) \geq O(n^4),$$

which, even with modern computers, allows to solve in reasonable time problems with hardly more than 100 – 200 design variables.

♣ Poor from practical viewpoint (although polynomial time) performance of the algorithms in question stems from their *black box oriented* nature – these algorithms do not adjust themselves to the structure of the problem and use a priori knowledge of this structure solely to mimic *First Order oracle* reporting the values and (sub)gradients of the objective and the constraints at query points.

Note: A convex program *always* has a lot of structure – otherwise how could we know that the problem is convex?

A good algorithm should utilize a priori knowledge of problem's structure in order to accelerate the solution process.

Example: The LP Simplex Method is fully adjusted to the particular structure of an LP problem and *works directly on problem's data rather than on values and gradients of objective and constraints at search points.*

Although not a polynomial time one, this algorithm in reality is capable to solve LP's with tens and hundreds of thousands of variables and constraints – a task which is by far out of reach of the theoretically efficient “universal” black box oriented algorithms underlying the Theorem.

"A good optimization algorithm should utilize problems's structure..."

Difficulty: *What is structure?*

There is no formal definition; we recognize "what is structure" *after* we see it, on a case-by-case basis...

- For example: What does the usual Mathematical Programming form

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

of a convex problem say about problem's structure?

— It says that there is objective called f , m constraints called g_i , and that the objective and constraints are convex. Not much...

♣ Since mid-1970's, Convex Programming is the most rapidly developing area in Optimization, with intensive and successful research primarily focusing on

- discovery and investigation of novel well-structured generic Convex Programming problems (‘Conic Programming’, especially *Conic Quadratic* and *Semidefinite*)
- developing theoretically efficient and powerful in practice algorithms for solving well-structured convex programs, including large-scale non-linear ones
- building Convex Programming models for a wide spectrum of problems arising in Engineering, Signal Processing, Machine Learning, Statistics, Management, Medicine, etc.
- extending modelling methodologies in order to capture factors like data uncertainty typical for real world situations
- adjusting algorithms to distributed organization of data and computations (‘cloud computing’)
- software implementation of novel optimization techniques at academic and industry levels

“Structure-Revealing” Representation of Convex Problem: Conic Programming

♣ When passing from a Linear Programming program

$$\min_x \{c^T x : Ax - b \geq 0\} \quad (*)$$

to a nonlinear convex one, the traditional wisdom is to replace linear inequality constraints

$$a_i^T x - b_i \geq 0$$

with nonlinear ones:

$$g_i(x) \geq 0 \quad [g_i \text{ are concave}]$$

♠ There exists, however, another way to introduce nonlinearity, namely, to replace the coordinate-wise vector inequality

$$y \geq z \Leftrightarrow y - z \in \mathbf{R}_+^m = \{u \in \mathbf{R}^m : u_i \geq 0 \forall i\} \quad [y, z \in \mathbf{R}^m]$$

with another vector inequality

$$y \geq_{\mathbf{K}} z \Leftrightarrow y - z \in \mathbf{K} \quad [y, z \in \mathbf{R}^m]$$

where \mathbf{K} is a *regular cone* (i.e., closed, pointed and convex cone with a nonempty interior) in \mathbf{R}^m .

♣ LP problem:

$$\min_x \{c^T x : Ax - b \geq 0\} \Leftrightarrow \min_x \{c^T x : Ax - b \in \mathbf{R}_+^m\}$$

♣ General Conic problem:

$$\min_x \{c^T x : Ax - b \succeq_{\mathbf{K}} 0\} \Leftrightarrow \min_x \{c^T x : Ax - b \in \mathbf{K}\}$$

- (c, A, b) – *data* of conic problem
- \mathbf{K} – structure of conic problem

♠ Note: Every convex problem admits equivalent conic reformulation

♠ Note: With conic formulation, convexity is “built in”; with the standard MP formulation convexity should be kept in mind as an additional property.

♣ **Question:** A general convex cone has no more structure than a general convex function. Why conic reformulation is “structure-revealing”?

♣ **Answer:** As a matter of fact, just 3 types of cones allow to represent an extremely wide spectrum (“essentially all”) of convex problems!

$$\min_x \left\{ c^T x : Ax - b \succeq_{\mathbf{K}} 0 \right\} \Leftrightarrow \min_x \left\{ c^T x : Ax - b \in \mathbf{K} \right\}$$

♠ Three Magic Families of cones:

- \mathcal{LP} : Nonnegative orthants \mathbf{R}_+^m – direct products of m nonnegative rays $\mathbf{R}_+ = \{s \in \mathbf{R} : s \geq 0\}$ giving rise to Linear Programming programs

$$\min_s \left\{ c^T x : a_\ell^T x - b_\ell \geq 0, 1 \leq \ell \leq q \right\}.$$

- \mathcal{QCP} : Direct products of Lorentz cones

$$\mathbf{L}_+^p = \{u \in \mathbf{R}^p : u_p \geq \|[u_1; \dots; u_{p-1}]\|_2\}$$

giving rise to Conic Quadratic programs

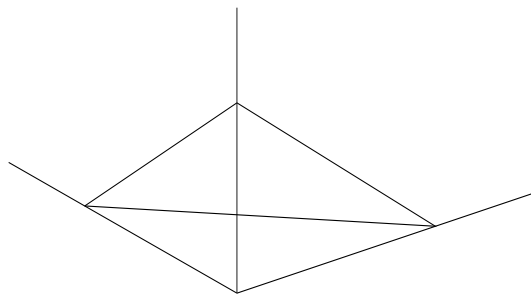
$$\min_x \left\{ c^T x : \|A_\ell x - b_\ell\|_2 \leq c_\ell^T x - d_\ell, 1 \leq \ell \leq q \right\}.$$

- \mathcal{SDP} : Direct products of Semidefinite cones $\mathbf{S}_+^p = \{M \in \mathbf{S}^p : M \succeq 0\}$ giving rise to Semidefinite programs

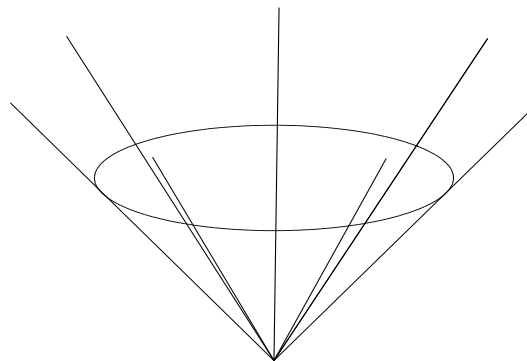
$$\min_x \left\{ c^T x : \mathcal{A}_\ell(x) \succeq 0, 1 \leq \ell \leq q \right\}.$$

where \mathbf{S}^p is the space of $p \times p$ real symmetric matrices, $M \succeq 0$ means that M is symmetric positive semidefinite, and $\mathcal{A}_\ell(x)$ are affine in x symmetric matrices.

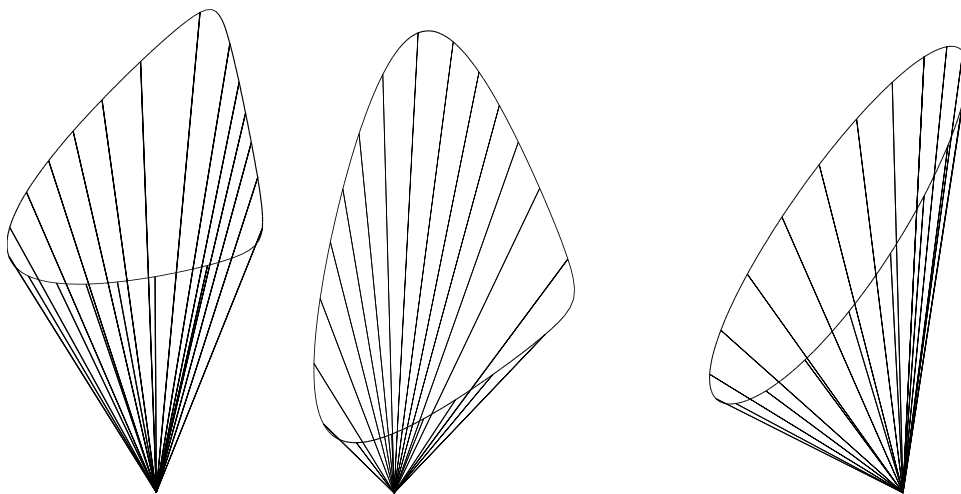
Note: Constraint stating that a symmetric matrix affinely depending on decision variables is $\succeq 0$ is called **LMI** – Linear Matrix Inequality.



The nonnegative orthant \mathbf{R}^3



The Lorentz cone \mathbf{L}^3



3 random 3D cross-sections of the semidefinite cone \mathbf{S}_+^3

Facts:

- ♠ Three “magic” families of conic problems – \mathcal{LP} , \mathcal{CQP} , \mathcal{SDP} – possess extremely strong “expressive abilities” and for all practical purposes cover the entire Convex Programming
- ♠ At the same time, the cones underlying the magic families are well understood and possess deep intrinsic mathematical similarity allowing for unified design of theoretically and practically efficient Interior Point polynomial time methods for $\mathcal{LP}/\mathcal{CQP}/\mathcal{SDP}$.
- ♠ To enjoy the power of “computational toolbox” of $\mathcal{LP}/\mathcal{CQP}/\mathcal{SDP}$, one should reformulate the problem of interest as a conic problem from a “magic” family, and *this is where a priori knowledge of problem’s structure is used.*

Illustration: The “maiden” form, as given by Statistics, of optimization problem responsible for extracting certain specific information from medical records reads

$$\min_{\alpha > 0, \phi} \left\{ \max_{x: Rx \leq r, y: Ry \leq r} \frac{1}{2} \left[\alpha \ln \left(\sum_i e^{\phi_i/\alpha} [\mathcal{A}x]_i \right) + \alpha \ln \left(\sum_i e^{-\phi_i/\alpha} [\mathcal{A}y]_i \right) + g^T[y - x] \right] + C\alpha \right\}$$

Due to specific structure of \mathcal{A} and constraints $Rx \leq r$, the problem is convex and thus efficiently solvable; however, implicit nature of the objective (presence of $\max_{x: Rx \leq r, y: Ry \leq r}$) prevents processing the problem by existing high-performance commercial solvers.

• Applying techniques to be presented in our course, the problem can be rewritten equivalently as

$$\min_{\alpha > 0, \phi, \lambda^\pm, u_\pm, \mu_\pm, \xi^\pm, \eta^\pm} \left\{ \frac{1}{2}[u_+ + u_-] + C\alpha : \begin{cases} \xi^+ \geq \lambda^+, \eta^+ \geq 0, \xi^- \geq \lambda^-, \eta^- \geq 0; \\ R^T \eta^+ - \mathcal{A}^T \xi^+ = -g, r^T \eta^+ \leq \alpha - \mu_+ + u_+, \\ R^T \eta^- - \mathcal{A}^T \xi^- = g, r^T \eta^- \leq \alpha - \mu_- + u_-; \\ \phi_i - \mu_+ + \alpha \ln(\alpha/\lambda_i^+) \leq 0, -\phi_i - \mu_- + \alpha \ln(\alpha/\lambda_i^-) \leq 0, \forall i \end{cases} \right\}.$$

The reformulated problem possesses explicitly given objective and constraints and can be processed by existing commercial solvers.

Fact: Modern Interior Point Polynomial Time methods for $\mathcal{LP}/\mathcal{CQP}/\mathcal{SDP}$ are the best known so far techniques for finding high accuracy solutions to convex programs – after the program is reformulated as $\mathcal{LP}/\mathcal{CQP}/\mathcal{SDP}$, such a solution usually is found in a moderate (few tens) number of iterations, an iteration reducing to assembling and solving a system of linear equations.

However: For extremely large-scale problems, the linear systems arising in Interior Point methods become too large to be solved in reasonable time

⇒ *In the large-scale case, utilizing "computationally cheap" optimization techniques becomes a must.*

As far as constrained/nonsmooth large-scale convex problems are concerned, the scope of these “computationally cheap” techniques – *First Order algorithms* – is restricted to search for medium-accuracy solutions.

In our course, the emphasis will be on

♣ Theory of Conic Programming, primarily *Conic Programming Duality*
Duality is indispensable tool in

- (a) processing conic problems “on paper,” allowing in numerous cases to gain deep theoretical understanding of the situation
- (b) design of solution algorithms aimed at “getting number.”

♣ “How to press the button” — how to pose the problem at hand as a conic problem from Magic Family. Specifically, we will investigate *expressive abilities* and *typical applications*, primarily in Engineering, of Linear, Conic Quadratic, and Semidefinite Programming

♣ Demonstrating polynomial time solvability of Convex Programming

♣ “How we do the rest” — what are typical convex programming algorithms, specifically,

— polynomial time Interior Point methods for $\mathcal{LP}/\mathcal{CQP}/\mathcal{SDP}$

— “computationally cheap” First Order algorithms for deterministic and stochastic problems with convex structure

Main Notational Conventions

♣ $O(1)$'s. Below $O(1)$'s denote properly selected *positive absolute constants*. We write $f \leq O(1)g$, where f and g are nonnegative functions of some parameters, to express the fact that for properly selected positive absolute constant C the inequality $f \leq Cg$ holds true in the entire range of the parameters, and we write $f = O(1)g$ when both $f \leq O(1)g$ and $g \leq O(1)f$.

♣ Vectors and matrices:

- All vectors are column vectors.
- \mathbf{R}^n is the linear space of n -dimensional real vectors
- $\mathbf{R}^{m \times n}$ is the linear space of $m \times n$ matrices
- \mathbf{S}^n is the linear space of $n \times n$ real *symmetric* matrices

♣ Sometimes we use “MATLAB notation” to save space:

— $[A_1; A_2; \dots; A_k]$ is array obtained from equal width arrays A_1, \dots, A_k by writing them into column from top to bottom

— $[A_1, A_2, \dots, A_k]$ is array obtained from equal height arrays A_1, \dots, A_k by writing them into row, from left to right.

Examples:

$$\bullet A_1 = \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{bmatrix}, A_2 = \begin{bmatrix} 7 & 8 & 9 \end{bmatrix} \Rightarrow [A_1; A_2] = \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{bmatrix}$$

$$\bullet A_1 = \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix}, A_2 = \begin{bmatrix} 7 \\ 8 \end{bmatrix} \Rightarrow [A_1, A_2] = \begin{bmatrix} 1 & 2 & 7 \\ 3 & 4 & 8 \end{bmatrix}$$

$$\bullet [1, 2, 3, 4] = [1; 2; 3; 4]^T$$

$$\begin{aligned} \bullet [[1, 2; 3, 4], [5, 6; 7, 8]] &= \left[\begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix}, \begin{bmatrix} 5 & 6 \\ 7 & 8 \end{bmatrix} \right] = \begin{bmatrix} 1 & 2 & 5 & 6 \\ 3 & 4 & 7 & 8 \end{bmatrix} \\ &= [1, 2, 5, 6; 3, 4, 7, 8] \end{aligned}$$

♣ Whenever possible, we replace zero entries in a matrix with blanks.

$$\text{Say, } \begin{bmatrix} 1 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 3 \end{bmatrix} \text{ is usually written down as } \begin{bmatrix} 1 & & \\ & 2 & \\ & & 3 \end{bmatrix}$$

♣ Diag and Dg

- For vector $x \in \mathbf{R}^m$, $\text{Diag}\{x\}$ is diagonal $m \times m$ matrix with the entries in x as diagonal entries:

$$\text{Diag}\{[1; 2; 3]\} = \begin{bmatrix} 1 & & \\ & 2 & \\ & & 3 \end{bmatrix}$$

- For matrices A_1, \dots, A_k , $\text{Diag}\{A_1, \dots, A_k\}$ is block-diagonal matrix with diagonal blocks A_1, A_2, \dots, A_k :

$$\text{Diag}\{1, [2, 3], [4, 5; 6, 7]\} = \begin{bmatrix} 1 & & \\ \hline & 2 & 3 \\ \hline & & \begin{bmatrix} 4 & 5 \\ 6 & 7 \end{bmatrix} \end{bmatrix}$$

- For square matrix A , $\text{Dg}(A)$ extracts from A the vector of diagonal entries:

$$\text{Dg}\left(\begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{bmatrix}\right) = \begin{bmatrix} 1 \\ 5 \\ 9 \end{bmatrix}$$

I. FROM LINEAR TO CONIC PROGRAMMING

Linear Programming

$$\min_x \{c^T x : Ax \geq b\} \quad [x \in \mathbf{R}^n, A \in \mathbf{R}^{m \times n}]$$

♣ Aside of modelling and algorithmic issues, the most important issue in LP is *LP Duality Theory*, which, essentially, answers the following basic question:

(?) *How to certify that a system of strict and nonstrict linear inequalities*

$$\begin{cases} Px > p \\ Qx \geq q \end{cases} \quad (S)$$

has no solutions?

Certificate, *informally*, is a short and transparent proof.

◇ **Note:** it is easy to certify that (S) *has* a solution: *every solution is a certificate!*

A solution indeed is a “short and transparent proof:” given it, everybody who knows arithmetics can check that it is a solution, and thus conclude that (S) is feasible.

Illustration: *How to certify that the system of inequalities*

$$\begin{array}{rrcr} -4u & -9v & +5w & > 1.8 \\ -2u & +6v & & \geq -2 \\ 7u & & -5w & \geq 1 \end{array}$$

in variables $x = [u; v; w]$ is solvable ?

A certificate is, e.g., the vector $\bar{x} = [u; v; w] = [0.15; -0.27; 0]$. Plugging it into the inequalities, the left hand sides become $1.83 > 1.8$, $-1.92 \geq -2$, $1.05 \geq 1 \Rightarrow \bar{x}$ solves the system \Rightarrow *the system is solvable.*

General Theorem on Alternative

- Question: Given a finite system of strict and non-strict linear inequalities with n unknowns

$$\begin{cases} Px > p & (a) \\ Qx \geq q & (b) \end{cases} \quad (S)$$

how to certify that the system has *no* solutions?

Example: To certify that the system

$$\begin{array}{rrcr} -4u & -9v & +5w & > 2 \\ -2u & +6v & & \geq -2 \\ 7u & & -5w & \geq 1 \end{array}$$

has no solutions, it suffices to point out that *aggregating the inequalities of the system with weights 2,3,2, we get a contradictory inequality:*

$$\begin{array}{r|rrrr} & 2\times & -4u & -9v & +5w & > & 2 \\ + & & & & & & \\ & 3\times & -2u & +6v & & \geq & -2 \\ + & & & & & & \\ & 2\times & 7u & & -5w & \geq & 1 \\ \hline & & 0\cdot u & +0\cdot v & +0\cdot w & > & 0 \end{array}$$

By how we aggregate, every solution to the system *must* solve the aggregated inequality.

The latter has no solutions \Rightarrow so is the system.

General Theorem on Alternative

- **Question:** Given a finite system of strict and non-strict linear inequalities with n unknowns

$$\begin{cases} Px > p & (a) \\ Qx \geq q & (b) \end{cases} \quad (S)$$

How to certify that the system has no solutions?

- **Simple sufficient condition for insolvability:**

Assume that we can get, as a “linear consequence” of (S) (i.e., by multiplying inequalities (a) by nonnegative weights s_i , inequalities (b) by nonnegative weights y_j and adding the results) a **contradictory** (no solutions at all!) inequality:

*There exist **nonnegative** weight vectors s ($\dim s = \dim p$) and y ($\dim y = \dim q$) such that the inequality*

$$[s^T P + y^T Q]x \Omega s^T p + y^T q \quad \left[\Omega = \begin{cases} ">" & s \neq 0 \\ "\geq" & s = 0 \end{cases} \right] \quad (*)$$

with unknowns x has no solutions. Then (S) is infeasible.

$$\{Px > p, Qx \geq q\} \ \& \ \{s \geq 0, y \geq 0\} \Rightarrow \underbrace{[s^T P + y^T Q]x \ \Omega \ s^T p + y^T q}_{(*)} \left[\Omega = \begin{cases} ">" , & s \neq 0 \\ ">=" , & s = 0 \end{cases} \right]$$

Observation: Inequality (*) has no solutions iff $P^T s + Q^T y = 0$ and

- either $\{\Omega = ">" \text{ and } s^T p + y^T q \geq 0\}$,
- or $\{\Omega = ">=" \text{ and } s^T p + y^T q > 0\}$

We have arrived at

Proposition. *Given system of strict and nonstrict linear inequalities*

$$\begin{cases} Px > p \\ Qx \geq q \end{cases}, \quad (S)$$

let us associate with it the following two systems of linear equalities/inequalities with unknowns s, y :

$\mathcal{T}_I : \begin{cases} s, y \geq 0; \\ P^T s + Q^T y = 0; \\ p^T s + q^T y \geq 0; \\ \sum_i s_i > 0. \end{cases}$	$\mathcal{T}_{II} : \begin{cases} y \geq 0; \\ Q^T y = 0; \\ q^T y > 0. \end{cases}$
--	--

If one of the systems $\mathcal{T}_I, \mathcal{T}_{II}$ has a solution, then (S) has no solutions.

General Theorem on Alternative. *The sufficient condition for infeasibility of (S) stated by Proposition is in fact necessary and sufficient.*

$\mathcal{S} : \begin{cases} Px > p \\ Qx \geq q \end{cases}$	$\mathcal{T}_I : \begin{cases} s, y \geq 0; \\ P^T s + Q^T y = 0; \\ p^T s + q^T y \geq 0; \\ \sum_i s_i > 0. \end{cases}$	$\mathcal{T}_{II} : \begin{cases} y \geq 0; \\ Q^T y = 0; \\ q^T y > 0. \end{cases}$
---	--	--

Remark: By GTA applied to the system

$$Qx \geq q, \quad (\mathcal{S}_{NS})$$

this system is unsolvable *iff* \mathcal{T}_{II} is solvable. Thus,

- System (\mathcal{S}_{NS}) is unsolvable *iff* system \mathcal{T}_{II} is solvable;
- Assume that system (\mathcal{S}_{NS}) is solvable. Then system (\mathcal{S}) is unsolvable *iff* system \mathcal{T}_I is solvable.

Corollaries: **A.** A system of linear inequalities

$$a_i^T x \begin{matrix} > \\ \geq \\ = \\ \leq \\ < \end{matrix} b_i, \quad i = 1, \dots, m$$

is infeasible *iff* one can combine the inequalities of the system in a *legitimate linear* fashion (i.e., multiply the inequalities by weights and add the results, the sign of the weights making the summation legitimate) to get a contradictory inequality, namely, either the inequality $0^T x \geq 1$, or the inequality $0^T x > b$ with $b \geq 0$.

B. [Inhomogeneous Farkas Lemma] A scalar linear inequality $a_0^T x \leq b_0$ is a consequence of a *solvable* system of linear inequalities

$$a_i^T x \leq b_i, \quad i = 1, \dots, m$$

iff it can be obtained by taking weighted sum, with nonnegative weights, of inequalities from the system and the trivial identically true inequality $0 \leq 1$:

$$a_0 = \sum_{i=1}^m \lambda_i a_i, \quad b_0 = \lambda_0 + \sum_i \lambda_i b_i \quad \text{for some } \lambda_i \geq 0, \quad i = 0, 1, \dots, m$$

♣ GTA is a really striking fact:

$$\begin{cases} -1 \leq u \leq 1 \\ -1 \leq v \leq 1 \end{cases} \Rightarrow \begin{cases} u^2 \leq 1 \\ v^2 \leq 1 \end{cases} \Rightarrow u^2 + v^2 \leq 2$$

$$\Rightarrow u + v = 1 \times u + 1 \times v \leq \sqrt{1^2 + 1^2} \sqrt{u^2 + v^2} \leq \sqrt{2} \times \sqrt{2} = 2 \Rightarrow u + v \leq 2$$

In this “highly nonlinear” derivation, the premise is a solvable system of linear inequalities, and the conclusion is a linear inequality. How could we know in advance that *every* derivation of this type can be replaced just with linear aggregation of the inequalities in the premise and the trivial inequality $0 \leq 1$?

♣ GTA heavily exploits the fact that we are speaking about *linear* inequalities:

$$\begin{cases} u \leq 1 \\ -u \leq 1 \end{cases} \Rightarrow u^2 \leq 1 \quad \text{— definitely true!}$$

However, aggregating in a legitimate linear fashion inequalities from the premise and trivial (i.e., identically true) linear and quadratic inequalities, like

$$0 \leq 1, -u^2 \leq 0, -u^2 + 2u \leq 1, \dots$$

you *cannot* get the concluding inequality.

GTA - Sketch of the proof

♣ **Starting point: Homogeneous Farkas Lemma:** *A homogeneous linear inequality*

$$a^T x \geq 0 \quad (I)$$

is a consequence of a system of homogeneous linear inequalities

$$a_i^T x \geq 0, i = 1, \dots, m, \quad (H)$$

iff (I) can be obtained from (H) by linear aggregation:

$$\exists y \geq 0 : a = \sum_i y_i a_i,$$

that is, iff a is a conic combination (linear combination with nonnegative coefficients) of a_1, \dots, a_m .

Note: (I) being a consequence of (H) is *exactly the same* as infeasibility of the system

$$\begin{aligned} a^T x &< 0, \\ a_i^T x &\geq 0, i = 1, \dots, m. \end{aligned}$$

What GTA says in this case, is exactly HFL. Our course of actions is opposite: we will *directly prove HFL* and then derive GTA from HFL.

♣ **HFL** \Rightarrow **GTA**: Given system

$$\begin{cases} Px > p \\ Qx \geq q \end{cases} \quad (\mathcal{S})$$

in variables x , we associate with it system

$$\begin{cases} Px - tp - \epsilon \mathbf{1} \geq 0 \\ Qx - tq \geq 0 \\ t - \epsilon \geq 0 \end{cases} \quad (\mathcal{H})$$

in variables x, t, ϵ ($\mathbf{1}$: all-ones vector).

It is immediately seen that (\mathcal{S}) has no solutions iff (\mathcal{H}) has no solutions with $\epsilon > 0$, i.e., *iff the homogeneous linear inequality $-\epsilon \geq 0$ is a consequence of the system of homogeneous linear inequalities (\mathcal{H})* . HFL says *exactly* when the latter happens, and this answer turns out to be exactly the statement of GTA.

HFL – Intelligent Proof

♣ A set $X \subset \mathbf{R}^n$ is called *polyhedral*, if it is a solution set of a finite system of nonstrict linear inequalities:

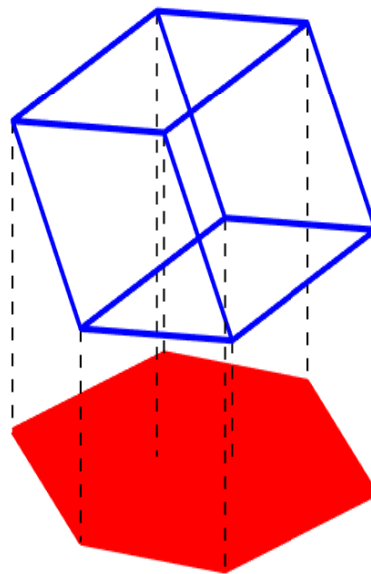
$$X \text{ is polyhedral} \Leftrightarrow \exists A, b : X = \{x \in \mathbf{R}^n : Ax \leq b\}.$$

♣ A *polyhedral representation* of a set $X \subset \mathbf{R}_x^n$ is a representation of X as the *projection* of a higher-dimensional polyhedral set

$$X^+ = \{[x; u] : Ax + Bu \leq c\} \subset \mathbf{R}_x^n \times \mathbf{R}_u^k$$

onto the x -space, that is, as the image of X^+ under the projection mapping $[x; u] \mapsto x : \mathbf{R}_x^n \times \mathbf{R}_u^k \rightarrow \mathbf{R}_x^n$:

$$X = \{x \in \mathbf{R}^n : \exists u : Ax + Bu \leq c\}$$



Rotated 3D cube and its 2D projection (hexagon)

♣ **Fact:** *A set is polyhedral iff it admits polyhedral representation, or, equivalently, the projection X of a polyhedral set*

$$X^+ = \{[x; u] : Ax + Bu \leq c\}$$

on the space of x -variables can be represented as a solution set of a finite system of nonstrict linear inequalities in x -variables only.

Proof [Fourier-Motzkin Elimination]: It suffices to consider the case when u is one-dimensional. Let us split all inequalities

$$a_i^T x + b_i u \leq c_i, \quad 1 \leq i \leq I,$$

into three groups:

- **black**: $b_i = 0$ ($i \in \text{Black}$). Black inequality says that $a_i^T x \leq c_i$;
- **red**: $b_i > 0$ ($i \in \text{Red}$). Red inequality says that $u \leq \alpha_i^T x + \beta_i$, i.e., it imposes an affine in x **upper** bound on u .
- **green**: $b_i < 0$ ($i \in \text{Green}$). Green inequality says that $u \geq \alpha_i^T x + \beta_i$, i.e., it imposes an affine in x **lower** bound on u .

♠ Observe that a vector \bar{x} belongs to the projection of X^+ on the x -plane **iff** \bar{x} satisfies all black inequalities $a_i^T \bar{x} \leq c_i \forall i \in \text{Black}$ and we can point out a real which meets all stemming from \bar{x} upper and lower bounds on u . The latter is possible **iff** every upper bound is \geq every lower bound, and we arrive at

$$X := \{x : \exists u : Ax + ub \leq c\} = \left\{ x : \begin{cases} a_i^T x \leq c_i \forall i \in \text{Black} \\ \alpha_i^T x + b_i \geq \alpha_j^T x + \beta_j \forall (i \in \text{Red}, j \in \text{Green}) \end{cases} \right\}$$

$\Rightarrow X$ indeed is polyhedral.

♣ Now we are ready to prove HFL. The only nontrivial part of the statement is *If a is not a conic combination of a_1, \dots, a_n , then $a^T d < 0$ for some d with $a_i^T d \geq 0$, $i = 1, \dots, n$.*

Proof: Let $a \notin \text{Cone}(a_1, \dots, a_n) = \left\{ \sum_{i=1}^n u_i a_i : u \geq 0 \right\}$. Observe that $\text{Cone}(a_1, \dots, a_n)$ admits polyhedral representation:

$$\text{Cone}(a_1, \dots, a_n) = \left\{ x : \exists u : \begin{array}{l} u \geq 0, \\ x - \sum_i u_i a_i = 0 \end{array} \right\}$$

By the above, $\text{Cone}(a_1, \dots, a_n)$ is polyhedral: there exists a finite system of inequalities $p_j^T x \geq b_j$, $1 \leq j \leq J$, such that

$$\text{Cone}(a_1, \dots, a_n) = \{x : p_j^T x \geq q_j\}.$$

- Since $0 \in \text{Cone}(a_1, \dots, a_n)$, we have $q_j \leq 0$ for all j ;
 - Since $a \notin \text{Cone}(a_1, \dots, a_n)$, we have $p_{j_*}^T a < q_{j_*}$ for some j_* , whence $p_{j_*}^T a < 0$;
 - since $ta_i \in \text{Cone}(a_1, \dots, a_n)$ for all i and all $t > 0$, we should have $p_{j_*}^T (ta_i) \geq q_{j_*}$ for all $t > 0$, whence $p_{j_*}^T a_i \geq 0$ for all $i = 1, \dots, n$.
- \Rightarrow with $d = p_{j_*}$ we have $a_i^T d \geq 0$ for all i and $a^T d < 0$, as required.

Dual to a Linear Programming program

- **Question:** When a real a is a lower bound on the optimal value of an LP program

$$\min_x \{c^T x : Ax - b \geq 0\} \quad ? \quad (P)$$

- **Answer:** We are asking when the linear inequality

$$c^T x \geq a$$

is a corollary of the finite system of linear inequalities

$$Ax \geq b.$$

A *sufficient* condition for this is the possibility to get the target inequality by aggregation, with nonnegative weights, of the inequalities from the system and identically true inequality $0^T x \geq -1$:

$$\exists y \geq 0 : \quad A^T y = c, \quad y^T b \geq a$$

This sufficient condition is also *necessary*, provided that (P) is feasible (Corollary B of GTA).

$$\min_x \{c^T x : Ax - b \geq 0\} \quad (P)$$

- **Conclusion:** *The optimal value in the optimization problem*

$$\max_y \{b^T y : A^T y = c, y \geq 0\} \quad (D)$$

is a lower bound on the optimal value in (P). If the optimal value in (P) is finite, then (D) is solvable, and

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

LP Duality Theorem. *Consider an LP program*

$$\min_x \{c^T x : Ax \geq b\} \quad (P)$$

(the “primal” problem) along with its dual

$$\max_y \{b^T y : A^T y = c, y \geq 0\} \quad (D)$$

Then

- *The duality is symmetric: the problem dual to dual is equivalent to the primal;*
- *The value of the dual objective at every dual feasible solution is \leq the value of the primal objective at every primal feasible solution*
- *The following 5 properties are equivalent to each other:*
 - (i) *The primal is feasible and below bounded.*
 - (ii) *The dual is feasible and above bounded.*
 - (iii) *The primal is solvable.*
 - (iv) *The dual is solvable.*
 - (v) *Both primal and dual are feasible.*

Whenever (i) \equiv (ii) \equiv (iii) \equiv (iv) \equiv (v) is the case, the optimal values in the primal and the dual problems are equal to each other:

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

$$\begin{aligned} \min_x \{c^T x : Ax \geq b\} & \quad (P) \\ \max_y \{b^T y : A^T y = c, y \geq 0\} & \quad (D) \end{aligned}$$

Corollary. [Necessary and sufficient optimality conditions in LP] *Consider an LP program (P) along with its dual (D), and let (x,y) be a pair of primal and dual feasible solutions. The pair is comprised of optimal solutions to the respective problems iff*

$$c^T x - b^T y = 0 \quad \text{[zero duality gap]}$$

as well as iff

$$y_i[Ax - b]_i = 0, \quad i = 1, \dots, m, \quad \text{[complementary slackness]}$$

Indeed, since (P) and (D) are feasible, they are solvable with equal optimal values, hence for primal-dual feasible (x,y)

$$\text{DualityGap}(x, y) \equiv c^T x - b^T y = \underbrace{c^T x - \text{Opt}(P)}_{\geq 0} + \underbrace{\text{Opt}(D) - b^T y}_{\geq 0}$$

is always nonnegative and is 0 iff x,y are optimal for the respective problems.

Next, for a primal-dual feasible (x,y) we have

$$\begin{aligned} \text{DualityGap}(x, y) &= c^T x - b^T y = (A^T y)^T x - b^T y = [Ax - b]^T y \\ &\Rightarrow c^T x - b^T y = 0 \Leftrightarrow \underbrace{[Ax - b]^T}_{\geq 0} \underbrace{y}_{\geq 0} = 0 \Leftrightarrow y_i[Ax - b]_i = 0 \quad \forall i. \end{aligned}$$

Selected Engineering Applications of LP, I

Sparsity-oriented Signal Processing and ℓ_1 minimization

♣ The basic problem of Signal Processing is as follows:

(??) “In the nature” there exists a *signal* represented by vector $x \in \mathbb{R}^n$. Given observation

$$y = Ax + \eta$$

- A : $m \times n$ sensing matrix
- η : observation noise

we want to recover x .

♠ There are many different approaches to (??), depending primarily on the relation between m and n and on a priori information on x :

Parametric case: $m \gg n$: in principle, no a priori information on x is needed. In the “no noise” case $\eta = 0$ and with a “general position” A , x is readily given by y . When $\eta \neq 0$, the challenge is to reduce the influence of the noise on the estimate. A typical estimate is the *Least Squares* one:

$$\hat{x}(y) \in \operatorname{Argmin}_{w \in \mathbb{R}^n} \|Aw - y\|_2^2.$$

Least Squares are commonly used when $\eta = \sigma\xi$, $\xi \sim \mathcal{N}(0, I_m)$.

Nonparametric case: $m \ll n$: In the “no noise” case $\eta = 0$ the equality $y = Ax$ does not define x uniquely

\Rightarrow *A priori information on x is needed!*

— In *Compressed Sensing*, a priori information is that x *is sparse* — has at most a given number $s \ll m$ of nonzero entries.

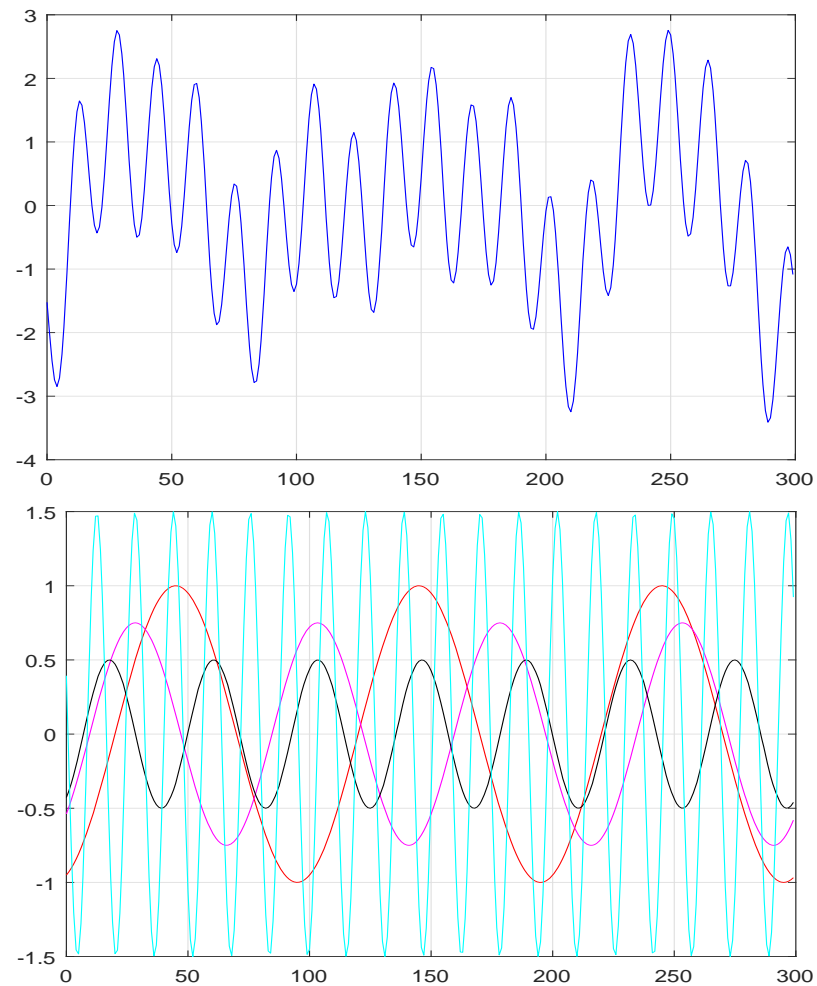
♠ **Fact:** *Many real-life signals x when presented by their coefficients in properly selected basis (“dictionary”) B :*

$$x = Bu$$

- columns of B : vectors of basis B
- u : coefficients of x in basis B

become sparse (or nearly so): u has just $s \ll n$ nonzero entries (or can be well approximated by vector with $s \ll n$ nonzero entries). We do not assume the location of “meaningful coefficients” known in advance.

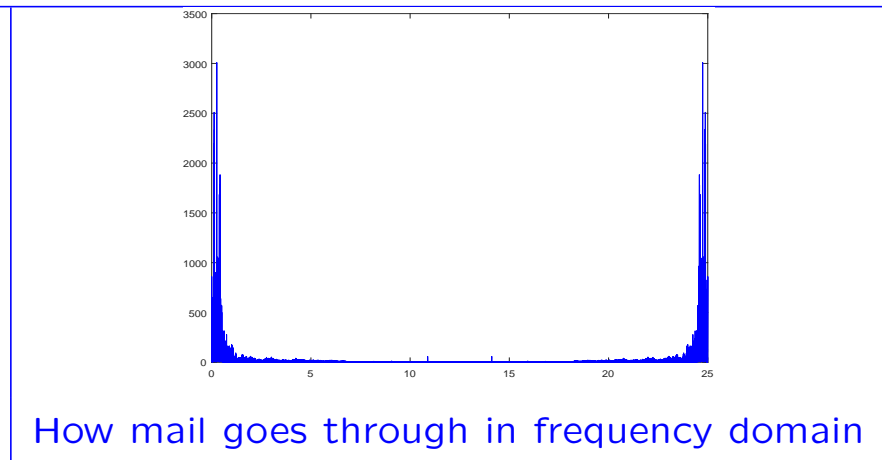
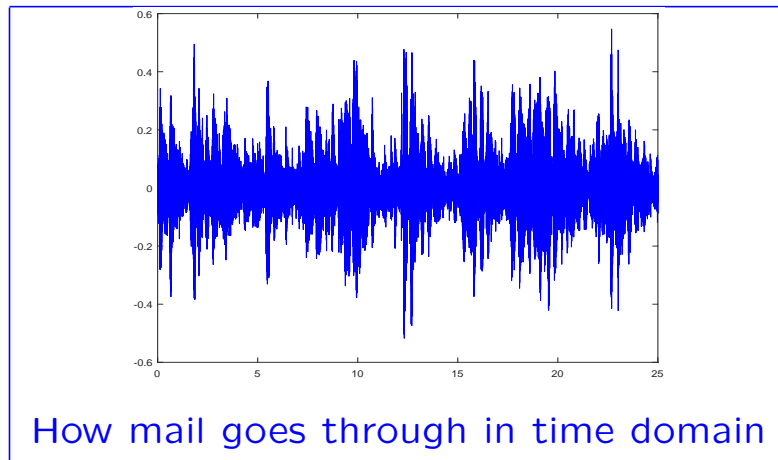
Example I: Typical audio signals become sparse (or nearly so) when representing them "in frequency domain" – as sums of harmonic oscillations of different frequencies:



Top: signal in time domain

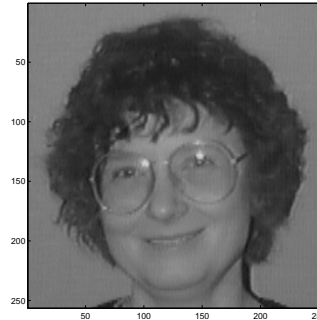
Bottom: decomposition of signal into sum of harmonic oscillations

Illustration: 25 sec fragment of audio signal “Mail must go through” (dimension 1,058,400) and its ” Fourier coefficients” – amplitudes of participating harmonic oscillations vs. the frequencies:



% of leading Fourier coefficients kept	energy
100%	100%
25%	99.8%
15%	99.6%
5%	98.2%
1%	79.0%

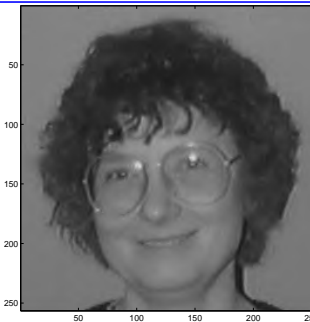
Example II: The 256×256 image



can be thought of as $256^2 = 65536$ -dimensional vector (write down the intensities of pixels column by column). This image (same as other “non-pathological” images) is nearly sparse when represented in *wavelet* basis:



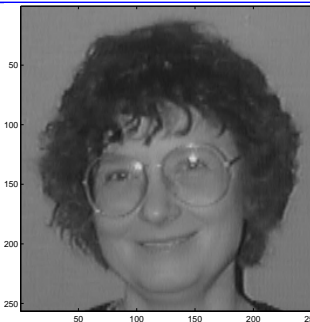
1% of leading wavelet
coefficients kept (99.70% of energy)



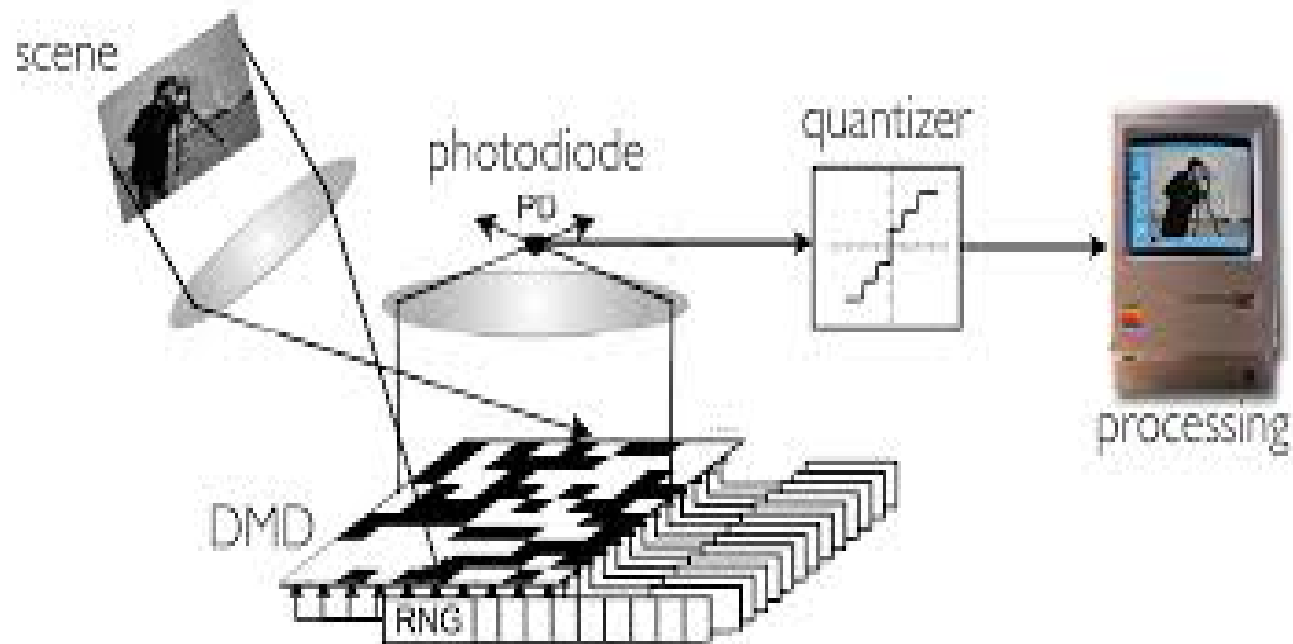
5% of leading wavelet
coefficients kept (99.93% of energy)



10% of leading wavelet
coefficients kept (99.96% of energy)



25% of leading wavelet
coefficients kept (99.99% of energy)



Single pixel camera

- David Donoho, *Compressed sensing — from blackboard to bedside*
Gauss Prize Lecture, International Congress of Mathematicians, 2018
<https://www.youtube.com/watch?v=mr-oT5gMboM>

♠ When recovering a signal x_* admitting a sparse (or nearly so) representation Bu_* in a known basis B from observations

$$y = Ax_* + \eta,$$

the situation reduces to the one when the signal to be recovered is just sparse. Indeed, we can first recover *sparse* u_* from observations

$$y = Ax_* + \eta = [AB]u_* + \eta.$$

After an estimate \hat{u} of u_* is built, we can estimate x_* by $B\hat{u}$.

⇒ In fact, sparse recovery is about how to recover a *sparse* n -dimensional signal x from $m \ll n$ observations

$$y = Ax_* + \eta.$$

$$y = Ax + \eta, \|\eta\| \leq \delta, \|x\|_0 := \text{Card}\{i : x_i \neq 0\} \leq s \quad ?? \mapsto ?? \quad \hat{x} \approx x$$

♣ **Let $\delta = 0$.** When the number s of nonzero entries in $x \in \mathbb{R}^n$ is essentially smaller than the number $m = \dim y$ of observations, the recovery problem becomes **well-posed** and can be solved by, e.g., ℓ_0 minimization:

$$\hat{x} \in \underset{w \in \mathbb{R}^n}{\text{Argmin}} \{ \|w\|_0 : Aw = y \}$$

Simple fact: Let every $m \times 2s$ submatrix of the $m \times n$ matrix A be of rank $2s$ (which is the case for a “general position” matrix A , provided that $2s \leq \min[m, n]$). Then in the noiseless case the ℓ_0 minimization recovers **exactly** every s -sparse signal x .

Indeed, x is feasible for the minimization problem $\Rightarrow \|\hat{x}\|_0 \leq \|x\|_0 \leq s \Rightarrow \|x - \hat{x}\|_0 \leq 2s$, which combines with $A(x - \hat{x}) = 0$ and the assumption that every $2s$ columns of A are linearly independent to imply $x - \hat{x} = 0$.

Bad news: ℓ_0 minimization requires to solve a disastrously complex combinatorial problem and as such is completely impractical.

A remedy: let us replace minimizing nonconvex (and even discontinuous) $\|\cdot\|_0$ with minimizing the “closest” to $\|\cdot\|_0$ convex function $\|\cdot\|_1$, thus arriving at **ℓ_1 minimization**, which in the noiseless case is

$$\hat{x}(y) \in \underset{w \in \mathbb{R}^n}{\text{Argmin}} \{ \|w\|_1 : Aw = y \}. \quad [\|z\|_1 = \sum_i |z_i|]$$

• Extensions of ℓ_1 minimization to the case of noisy observation take different forms, depending on noise’s structure. For example, in the case of *uncertain-but-bounded* noise, where all we know is that $\|\eta\| \leq \delta$, $\|\cdot\|$ and δ being given, a natural version of ℓ_1 minimization is

$$\hat{x}(y) \in \underset{w}{\text{Argmin}} \{ \|w\|_1 : \|Aw - y\| \leq \delta \}.$$

$$y = Ax + \eta, \|\eta\| \leq \delta \Rightarrow \hat{x}(y) \in \underset{w \in \mathbf{R}^n}{\text{Argmin}} \{ \|w\|_1 : \|Aw - y\| \leq \delta \}$$

Note: When $\delta = 0$, same as when $\|w\| = \|w\|_\infty := \max_i |w_i|$, ℓ_1 recovery reduces to solving an LP program!

Basic questions:

A. When A is *s-good*, that is, when ℓ_1 -recovery in the noiseless case $\delta = 0$ recovers *exactly* every s -sparse signal x ?

B. For s -good A , what are the error bounds of ℓ_1 recovery in the presence of noise?

A. When A is *s-good*, that is, when ℓ_1 -recovery in the noiseless case $\delta = 0$ recovers *exactly* every s -sparse signal x ?

Answer to A can be straightforwardly extracted from LP optimality conditions (and can be easily justified after it is guessed):

(!) A is s -good *iff* the *nullspace property* takes place: for every subset I of cardinality s of the index set $\{1, \dots, n\}$ and for every $z \in \text{Ker} A \setminus \{0\}$ one has

$$\|z_I\|_1 < \frac{1}{2} \|z\|_1.$$

where z_I is obtained from z by keeping intact all entries with indexes from I and zeroing out entries with indexes *not* in I .

Claim: A is s -good iff the *nullspace property* takes place: for every subset I of cardinality s of the index set $\{1, \dots, n\}$ and for every $z \in \text{Ker}A \setminus \{0\}$ one has

$$\|z_I\|_1 < \frac{1}{2}\|z\|_1.$$

where z_I is obtained from z by keeping intact all entries with indexes from I and zeroing out entries with indexes *not* in I .

Only if: Assume that for some I , $\text{Card}(I) \leq s$, and some nonzero $z \in \text{Ker}A$, one has $\|z_I\|_1 \geq \frac{1}{2}\|z\|_1$, or, equivalently, $\|z_I\|_1 \geq \|z_J\|_1$, $J = \{1, \dots, n\} \setminus I$, and let us prove that A is *not* s -good. Let the true signal be the s -sparse signal $x = z_I$. Then

$$\begin{aligned} Az = 0 &\Rightarrow Ax = A[-z_J] \text{ \& } \|z_J\|_1 \leq \|z_I\|_1 = \|x\|_1 \\ \Rightarrow x &\text{ is not the unique optimal solution to } \min_w \{\|w\|_1 : Aw = Ax\} \\ \Rightarrow &A \text{ is not } s\text{-good.} \end{aligned}$$

If: Let the nullspace property take place, let x be s -sparse, so that $x = x_I$ for some I , $\text{Card}(I) \leq s$, and let $\hat{x} \in \text{Argmin}_w \{\|w\|_1 : Aw = Ax\}$. Let $J = \{1, \dots, n\} \setminus I$ and $z = \hat{x} - x$. Assuming $z \neq 0$, let us lead this assumption to a contradiction. Since $0 \neq z \in \text{Ker}A$, we have by nullspace property $\|z_I\|_1 < \|z_J\|_1$, so that

$$\|x_I\|_1 - \|\hat{x}_I\|_1 \leq \|z_I\|_1 < \|z_J\|_1 = \|\hat{x}_J\|_1 \Rightarrow [\|x\|_1 =] \|x_I\|_1 < \|\hat{x}\|_1$$

and the concluding inequality contradicts the origin of \hat{x} .

B. For s -good A , what are the error bounds of ℓ_1 recovery in the presence of noise?
 Let us set

$$\|x\|_{s,1} := \max_{I: \text{Card}(I) \leq s} \|x_I\|_1 \underbrace{=}_{(!)} \max_u \{u^T x : \|u\|_\infty \leq 1, \|u\|_1 \leq s\}$$

Note: (!) is due to the evident fact that for a positive integer $s \leq n$, the extreme points of the convex polytope

$$U_s = \{u \in \mathbf{R}^n : \|u\|_\infty \leq 1, \sum_i |u_i| \leq s\}$$

are exactly the vectors with s nonzero entries equal to ± 1 .

Observation: A is s -good iff the quantity

$$\kappa_s(A) = \max_x \{\|x\|_{s,1} : Ax = 0, \|x\|_1 \leq 1\} = \max_{x,u} \{u^T x : u \in U_s, Ax = 0, \|x\|_1 \leq 1\}$$

is $< 1/2$.

Indeed, the nullspace property says that $\|x_I\|_1 < \frac{1}{2}\|x\|_1$ for all $0 \neq x \in \text{Ker}A$ and every I with $\text{Card}(I) \leq s$, which is the same as $\|x\|_{s,1} < 1/2$ whenever $x \in \text{Ker}A$ and $\|x\|_1 \leq 1$.

Observation: For every integer $s \leq n$, every $m \times n$ matrix A and every norm $\|\cdot\|$ on the image space \mathbf{R}^m of A there exists $\beta < \infty$ such that

$$\forall x \in \mathbf{R}^n : \|x\|_{s,1} \leq \beta \|Ax\| + \kappa_s(A) \|x\|_1. \quad (*)$$

The infimum of β 's satisfying this property will be denoted $\beta_s(A, \|\cdot\|)$.

Indeed, let P be orthogonal projector on $\text{Ker}A$. For some $\alpha < \infty$ and all z we have $\|(I - P)z\|_1 \leq \alpha \|A(I - P)z\|$, whence

$$\begin{aligned} \|z\|_{s,1} &\leq \|(I - P)z\|_{s,1} + \|Pz\|_{s,1} \leq \|(I - P)z\|_1 + \kappa_s(A) \|Pz\|_1 \leq \|(I - P)z\|_1 + \kappa_s(A) [\|z\|_1 + \|(I - P)z\|_1] \\ &\leq (1 + \kappa_s(A)) \|(I - P)z\|_1 + \kappa_s(A) \|z\|_1 \leq \alpha(1 + \kappa_s(A)) \|A(I - P)z\| + \kappa_s(A) \|z\|_1 \\ &= \underbrace{\alpha(1 + \kappa_s(A))}_{\beta} \|Az\| + \kappa_s(A) \|z\|_1 \end{aligned}$$

Note: (*) with $\kappa_s(A) < 1/2$ implies nullspace property.

$$\forall z \in \mathbf{R}^n : \|z\|_{s,1} \leq \beta \|Az\| + \kappa_s(A) \|z\|_1. \quad (*)$$

♣ The quantities $\kappa_s(A)$ and $\beta_s(a, \|\cdot\|)$ are responsible for the error bound in imperfect ℓ_1 recovery:

Theorem. Let A be $m \times n$ sensing matrix and s be a positive integer. Assume that

- signal $x \in \mathbf{R}^n$ is nearly s -sparse: $\|x - x^s\|_1 \leq v$ for some s -sparse vector x^s ;
- noise η in the observation $y = Ax + \eta$ satisfies $\|\eta\| \leq \delta$ for given $\delta \geq 0$ and norm $\|\cdot\|$;
- \hat{x} is obtained from A, y, δ by imperfect ℓ_1 -recovery:

$$\|\hat{x}\|_1 \leq \nu + \underbrace{\min_w \{\|w\|_1 : \|Aw - y\| \leq \delta\}}_{\text{Opt}} \quad \& \quad \|A\hat{x} - y\| \leq \delta + \epsilon.$$

Assuming $(*)$ and $\kappa_s(A) < 1/2$, the following error bound holds true:

$$\|x - \hat{x}\|_1 \leq \frac{2\beta_s(A, \|\cdot\|)[2\delta + \epsilon] + 2v + \nu}{1 - 2\kappa_s(A)}.$$

Proof. W.l.o.g. we can take $x^s = x_I$, where I is the collection of indexes of the s largest in magnitude entries in x , and x_I is obtained from x by zeroing out the entries with indexes outside of I . Let $J = \{1, \dots, n\} \setminus I$ and $z = \hat{x} - x$, so that $\|x_J\|_1 = v$. Setting $\kappa = \kappa_s(A)$, $\beta = \beta_s(A, \|\cdot\|)$, have

$$\begin{aligned}
(a) \quad & \|\hat{x}\|_1 \leq \text{Opt} + \nu \leq \|x\|_1 + \nu = \|x_I\|_1 + \|x_J\|_1 + \nu, \\
(b) \quad & \|Az\| \leq \|[A\hat{x} - y] + [y - Ax]\| \leq \|A\hat{x} - y\| + \|Ax - y\| \leq 2\delta + \epsilon, \\
(c) \quad & \|\hat{x}_J\|_1 - \|x_J\|_1 \leq \|\hat{x}\|_1 - \|\hat{x}_I\|_1 - \|x_J\|_1 \leq \nu + \|x_I\|_1 - \|\hat{x}_I\|_1 \leq \nu + \|z_I\|_1; \quad [\text{by (a)}] \\
& \|z_I\|_1 \leq \beta\|Az\| + \kappa\|z\|_1 = \beta\|Az\| + \kappa[\|z_I\|_1 + \|z_J\|_1] \\
\Rightarrow (d) \quad & \|z_I\|_1 \leq \frac{\beta\|Az\|}{1-\kappa} + \frac{\kappa}{1-\kappa}\|z_J\|_1 \leq \frac{\beta(2\delta+\epsilon)}{1-\kappa} + \frac{\kappa}{1-\kappa}\|z_J\|_1, \quad [\text{see (b)}] \\
(e) \quad & \|z\|_1 \leq \frac{\beta(2\delta+\epsilon)}{1-\kappa} + \frac{1}{1-\kappa}\|z_J\|_1. \quad [\text{by (d)}]
\end{aligned}$$

We have

$$\begin{aligned}
& \|\hat{x}_J\|_1 - \|x_J\|_1 \underbrace{\leq}_{(c)} \nu + \|z_I\|_1 \underbrace{\leq}_{(d)} \nu + \frac{\beta(2\delta+\epsilon)}{1-\kappa} + \frac{\kappa}{1-\kappa}\|z_J\|_1 \leq \nu + \frac{\beta(2\delta+\epsilon)}{1-\kappa} + \frac{\kappa}{1-\kappa}[\|x_J\|_1 + \|\hat{x}_J\|_1] \\
& \Rightarrow \frac{1-2\kappa}{1-\kappa}\|\hat{x}_J\|_1 \leq \nu + \frac{\beta(2\delta+\epsilon)}{1-\kappa} + \frac{1}{1-\kappa}\|x_J\|_1 \Rightarrow \frac{1-2\kappa}{1-\kappa}\|z_J\|_1 \leq \nu + \frac{\beta(2\delta+\epsilon)}{1-\kappa} + 2\|x_J\|_1 \\
& \Rightarrow \|z_J\|_1 \leq \frac{\nu(1-\kappa) + \beta(2\delta+\epsilon) + 2(1-\kappa)\|x_J\|_1}{1-2\kappa} \\
& \Rightarrow \|z_J\|_1 \leq \frac{\nu(1-\kappa) + \beta(2\delta+\epsilon) + 2(1-\kappa)v}{1-2\kappa}
\end{aligned}$$

Invoking (e), we arrive at the desired bound

$$\| \underbrace{x - \hat{x}}_z \|_1 \leq \frac{2\beta_s(A, \|\cdot\|)[2\delta + \epsilon] + 2v + \nu}{1 - 2\kappa_s(A)}.$$

Intermediate Summary on ℓ_1 -Recovery

♠ **Problem of interest:** to recover signal $x \in \mathbf{R}^n$ from noisy observation $y = Ax + \eta$ with “uncertain-but-bounded” observation noise: $\|\eta\| \leq \delta$ when the observations are *deficient*: $m = \dim y \ll n$ and the signal is s -sparse – has at most $s \ll m$ nonzero entries.

♠ ℓ_1 **recovery:** $\operatorname{argmin}\{\|w\|_1 : \|Aw - y\| \leq \delta\}$

♠ $\|z\|_{s,1}$: total magnitude of s largest in magnitude entries in z . A is s -good, meaning that ℓ_1 minimization recovers *exactly* all s -sparse signals *in the noiseless case*, *iff*

$$\kappa_s(A) := \max_z \{\|z\|_{s,1} : Az = 0, \|z\|_1 \leq 1\} < \frac{1}{2}$$

(“nullspace property”).

♠ Given norm $\|\cdot\|$ on \mathbf{R}^m , one has $\forall z : \|z\|_{s,1} \leq \beta_s \|Az\| + \kappa_s(A) \|z\|_1$ with properly selected $\beta_s = \beta_s(A, \|\cdot\|)$.

When $\kappa_s(A) < 1/2$, β_s and κ_s are responsible for error bounds in *imperfect ℓ_1 recovery* (nonzero noise and/or nearly s -sparse, rather than perfectly s -sparse, signal and/or imprecise ℓ_1 minimization).

Tractability Issues

♣ We have defined the quantities $\kappa_s(A)$, $\beta_s(A, \|\cdot\|)$ responsible for s -goodness of A and for the error bound for imperfect ℓ_1 recovery.

But: *It is unclear how to compute efficiently $\kappa_s(A)$. Moreover, no ways to verify the nullspace property in reasonable time are known, unless s is "very small," like 1 or 2.*

\Rightarrow We need verifiable sufficient conditions for s -goodness, or, which is basically the same, an efficiently computable upper bound $\kappa_s^+(A)$ on the quantity

$$\kappa_s(A) = \max_z \{ \|z\|_{s,1} : \|z\|_1 \leq 1, Az = 0 \}.$$

Equipped with such a bound, we could use the verifiable condition $\kappa_s^+(A) < 1/2$ as a sufficient condition for s -goodness of A .

Computationally Efficient Upper-Bounding of $\kappa_s(A)$: For $H \in \mathbb{R}^{m \times n}$ we have

$$\begin{aligned} \kappa_s(A) &:= \max_z \{ \|z\|_{s,1} : \|z\|_1 \leq 1 \text{ \& } Az = 0 \} \\ &= \max_z \{ \|[I - H^T A]z\|_{s,1} : \|z\|_1 \leq 1 \text{ \& } Az = 0 \} \leq \max_z \{ \|[I - H^T A]z\|_{s,1} : \|z\|_1 \leq 1 \} \\ &= \max_z \left\{ \left\| \sum_j z_j \text{Col}_j[I - H^T A] \right\|_{s,1} : \sum_j |z_j| \leq 1 \right\} \leq \max_z \left\{ \sum_j |z_j| \|\text{Col}_j[I - H^T A]\|_{s,1} : \sum_j |z_j| \leq 1 \right\} \\ &= \max_j \|\text{Col}_j[I - H^T A]\|_{s,1} \end{aligned}$$

\Rightarrow The efficiently computable quantity

$$\kappa_s^+(A) = \min_{H \in \mathbb{R}^{m \times n}} \max_j \|\text{Col}_j[I - H^T A]\|_{s,1}$$

is an upper bound on $\kappa_s(A)$, and thus the efficiently verifiable condition $\kappa_s^+(A) < 1/2$ is sufficient for s -goodness of A .

What is inside

Observation: $\kappa_s(A)$ is the maximum of *convex* function $\|u\|_{s,1}$ on the polytope

$$X = \text{Conv}\{\pm e_1, \dots, \pm e_n\} \cap \{x : Ax = 0\} = \{x : Ax = 0, \|x\|_1 \leq 1\}.$$

A recipe for upper-bounding a convex function $\phi(x)$ over polytope

$$X = \text{Conv}\{f_1, \dots, f_N\} \cap \{x : Ax = 0\} \quad [A \in \mathbf{R}^{m \times n}]$$

which we used is as follows: *For every $H \in \mathbf{R}^{m \times n}$ we have*

$$\begin{aligned} \phi_* &:= \max_{x \in X} \phi(x) = \max_x \{\phi(x) : x \in \text{Conv}\{f_1, \dots, f_N\}, Ax = 0\} \\ &= \max_x \left\{ \phi([I - H^T A]x) : x \in \text{Conv}\{f_1, \dots, f_N\}, Ax = 0 \right\} \leq \max_x \left\{ \phi([I - H^T A]x) : x \in \text{Conv}\{f_1, \dots, f_N\} \right\} \\ &= \max_{j \leq N} \phi([I - H^T A]f_j) \\ \Rightarrow \phi_* &\leq \phi_*^+ := \min_H \left[\max_j \phi([I - H^T A]f_j) \right], \end{aligned}$$

and ϕ_^+ is efficiently computable – this is the optimal value in explicit convex optimization problem.*

Two birds with one stone

♣ Assume that we can certify s -goodness of A by the above verifiable sufficient condition, that is, we have at our disposal a matrix H such that

$$\kappa^+ := \max_j \|\text{Col}_j[\Delta]\|_{s,1} < 1/2, \quad \Delta = I - H^T A$$

Then for every $x \in \mathbf{R}^n$ we have $x = [\Delta + H^T A]x$, whence

$$\begin{aligned} \|x\|_{s,1} &\leq \|H^T A x\|_{s,1} + \|\Delta x\|_{s,1} \leq s \|H^T A x\|_\infty + \sum_{j=1}^n |x_j| \|\text{Col}_j(\Delta)\|_{s,1} \\ &\leq \beta \|Ax\| + \kappa^+ \|x\|_1, \quad \beta = s \max_j \|\text{Col}_j[H]\|_* \\ &\quad [\|f\|_* = \max_{\|u\| \leq 1} f^T u] \end{aligned}$$

\Rightarrow We arrive at

$$\kappa_s(A) \leq \kappa^+ < \frac{1}{2} \text{ and } \beta_s(A, \|\cdot\|) \leq s \max_j \|\text{Col}_j[H]\|_*.$$

Remarks:

A. Computing $\kappa_s^+(A)$ and the associated H reduces to LP.

Indeed, for $z \in \mathbf{R}^n$ we have

$$\begin{aligned} \|z\|_{s,1} &= \max_u \{z^T u : \|u\|_\infty \leq 1, \|u\|_1 \leq s\} \\ &= \min_{y,t} \left\{ st + \sum_{j=1}^n y_j : y \geq 0, |z_j| \leq y_j + t \ \forall j \right\} \text{ [LP duality]} \\ \Rightarrow \kappa_s^+(A) &:= \min_{H,\tau} \{ \tau : \|\text{Col}_j[I - H^T A]\|_{s,1} \leq \tau \} \\ &= \min_{y^1, \dots, y^n, t_1, \dots, t_n, H, \tau} \left\{ \tau : \begin{cases} -y^j - t_j \mathbf{1} \leq \text{Col}_j[I - H^T A] \leq y^j + t_j \mathbf{1}, \ 1 \leq j \leq n \\ y^j \geq 0, \ \sum_{i=1}^n y_i^j + st_j \leq \tau, \ 1 \leq j \leq n \end{cases} \right\} \end{aligned}$$

B. One has

$$\kappa_1^+(A) = \kappa_1(A) = \max_{j \leq n} \max_x \{x_j : Ax = 0, \|x\| \leq 1\} = \min_H \max_{i,j} |[I - H^T A]_{ij}|$$

where the concluding equality is due to LP Duality Theorem.

C. It is easily seen that $\kappa_s^+(A) \leq \frac{s}{r} \kappa_r^+(A)$ when $1 \leq r \leq s$, and in particular

$$\kappa_s^+(A) \leq s \kappa_1^+(A).$$

As a result,

$$\kappa_1^+(A) < \frac{1}{2s} \Rightarrow \kappa_s^+(A) \leq s \kappa_1^+(A) < \frac{1}{2} \Rightarrow A \text{ is } s\text{-good}$$

♠ *Mutual Incoherence* of $A = [A_1, \dots, A_n]$ is defined as

$$\mu(A) = \max_{i \neq j} |A_i^T A_j| / A_j^T A_j.$$

Setting $H = \frac{1}{1+\mu(A)} [A_1/(A_1^T A_1), A_2/(A_2^T A_2), \dots, A_n/(A_n^T A_n)]$:

— diagonal entries in $H^T A$ are $\frac{1}{1+\mu(A)}$,

— magnitudes of off-diagonal entries in $H^T A$ are $\leq \frac{\mu(A)}{1+\mu(A)}$

$\Rightarrow H$ certifies that $\kappa_1^+(A) \leq \frac{\mu(A)}{\mu(A)+1} \Rightarrow A$ is s -good whenever $\frac{2s\mu(A)}{\mu(A)+1} < 1$.

Note: When entries of A are drawn at random from $\mathcal{N}(0, 1)$ or from $\text{Uniform}\{-1, 1\}$, the typical value of $\mu(A)$ is as small as $O(1)\sqrt{\ln(n)/m}$

\Rightarrow our simplified verifiable sufficient condition for s -goodness “ $\kappa_1^+(A) < \frac{1}{2s}$ ” certifies that a typical A from the random ensembles just specified is $O(\sqrt{m/\ln(n)})$ -good.

Bad news: When $A \in \mathbb{R}^{m \times n}$ is “essentially non-square,” namely, $n \geq 2m$, our verifiable sufficient condition can certify s -goodness only when $s \leq O(1)\sqrt{m}$.

Indeed, assume that $n \geq 2m$ and H certifies that $\kappa_s^+(A) < 1/2$. Setting $\bar{n} = 2m$ and denoting by D the angular $\bar{n} \times \bar{n}$ submatrix of $H^T A$, we have $\text{Rank} D \leq m$, whence $I_{\bar{n}} - D$ has at least $\bar{n} - m \geq m$ singular values ≥ 1 and thus

$$\sum_{i,j=1}^{\bar{n}} [I_{\bar{n}} - D]_{ij}^2 \geq m.$$

On the other hand, it is easily seen that

$$u \in \mathbf{R}^{\bar{n}} \Rightarrow \|u\|_2^2 \leq \max \left[\frac{\bar{n}}{s^2}, 1 \right] \|u\|_{s,1}^2,$$

and since

$$\|\text{Col}_j[I_{\bar{n}} - D]\|_{s,1} \leq \|\text{Col}_j[I_n - H^T A]\|_{s,1} \leq \kappa_s^+(A) < 1/2,$$

we get $\|\text{Col}_j[I_{\bar{n}} - D]\|_2^2 \leq \max[\frac{\bar{n}}{s^2}, 1] \cdot \frac{1}{4}$, whence

$$\sum_{i,j=1}^{\bar{n}} [I_{\bar{n}} - D]_{ij}^2 \leq \bar{n} \max \left[\frac{\bar{n}}{s^2}, 1 \right] \cdot \frac{1}{4} = \max \left[\frac{4m^2}{s^2}, 2m \right] \cdot \frac{1}{4}$$

Thus,

$$m \leq \max \left[\frac{m^2}{s^2}, \frac{m}{2} \right] \Rightarrow s \leq \sqrt{m}.$$

“True” bounds on s -goodness

♣ It is known that $m \times n$ matrices from typical *random* ensembles, e.g., *Gaussian* (i.i.d. entries $\sim \mathcal{N}(0, 1/m)$) or *Rademacher* (i.i.d. entries taking values $\pm 1/\sqrt{m}$ with probabilities $1/2$) *with probability approaching 1 as m, n grow are s -good with s as large as $O(1)m/\log(2n/m)$* , which is by far better than the maximal level of goodness $O(\sqrt{m})$ which can be certified by our verifiable sufficient conditions.

♠ Specifically, let us say that an $m \times n$ matrix A possesses **R**estricted **I**sometry **P**roperty with parameters δ, k (A is **RIP**(δ, k) for short), if *multiplying by A a k -sparse vector we nearly preserve ℓ_2 norm*:

$$(1 - \delta)\|x\|_2^2 \leq \|Ax\|_2^2 \leq (1 + \delta)\|x\|_2^2 \text{ for all } k\text{-sparse vectors } x$$

It is known that

A. *A random Gaussian/Rademacher $m \times n$ matrix is, with probability approaching 1 as m, n grow, **RIP**($0.1, k$) with k as large as $O(m/\ln(2n/m))$;*

B. *Whenever A is **RIP**($\delta, 2s$) with $\delta < 1/3$, A is s -good.*

B. Whenever A is $\text{RIP}(\delta, 2s)$ with $\delta < 1/3$, A is s -good.

Verification of B: Let A be $\text{RIP}(\delta, 2s)$, $\delta < 1/3$, and let $x \in \mathbf{R}^n$. Let x^1 be obtained from x by zeroing out all but the s largest in magnitude entries, x^2 be obtained in the same fashion from $x - x^1$, x^3 obtained in the same fashion from $x - x^1 - x^2$, etc. In other words, if i_1, i_2, \dots, i_n is the reordering of indexes such that $|x_{i_1}| \geq |x_{i_2}| \geq |x_{i_3}| \geq \dots$ and $I_p = \{i_{(p-1)s+1}, \dots, i_{ps}\}$, $1 \leq p \leq d = \lfloor n/s \rfloor$, then $x^p = x_{I_p}$.

We have $\|x^{p+1}\|_\infty \leq \|x^p\|_1/s$, $\|x^{p+1}\|_1 \leq \|x^p\|_1 \Rightarrow \|x^{p+1}\|_2 \leq \sqrt{\|x^{p+1}\|_\infty \|x^{p+1}\|_1} \leq s^{-1/2} \|x^p\|_1$. We further have

$$\|Ax^i\|_2 \|Ax\|_2 \geq [Ax^1]^T [Ax] = \sum_{p=1}^d [Ax^1]^T [Ax^p] \geq \|Ax^1\|_2^2 - \sum_{p=2}^d |[Ax^1]^T [Ax^p]| \quad (*)$$

Lemma: If A is $\text{RIP}(\delta, 2s)$ and u, v are s -sparse with non-intersecting supports, then $|u^T A^T A v| \leq \delta \|u\|_2 \|v\|_2$.

Indeed, Lemma states that if Q is symmetric matrix such that $(1 - \delta)y^T y \leq y^T Q y \leq (1 + \delta)y^T y$ for all y , then $|u^T Q v| \leq \delta \|u\|_2 \|v\|_2$ whenever $u^T v = 0$. This is evident, since from the premise it follows that the eigenvalues of Q are in-between $1 - \delta$ and $1 + \delta$, whence the spectral norm of $Q - I$ is $\leq \delta$, whence for u, v in question $|u^T Q v| = |u^T v + u^T (Q - I)v| = |u^T (Q - I)v| \leq \delta \|u\|_2 \|v\|_2$.

Applying Lemma, $(*)$ leads to

$$\begin{aligned} \|Ax^1\|_2 \|Ax\|_2 &\geq \|Ax^1\|_2^2 - \delta \sum_{p=2}^d \|x^1\|_2 \|x^p\|_2 \geq \|Ax^1\|_2^2 - \delta s^{-1/2} \|x^1\|_2 \sum_{p=1}^{d-1} \|x^p\|_1 \\ \Rightarrow \|Ax^1\|_2 &\leq \|Ax\|_2 + \delta s^{-1/2} \frac{\|x^1\|_2}{\|Ax^1\|_2} \|x\|_1 \Rightarrow \|x^1\|_2 \leq \frac{1}{\sqrt{1-\delta}} \|Ax\|_2 + s^{-1/2} \frac{\delta}{1-\delta} \|x\|_1 \end{aligned}$$

whence

$$\|x\|_{s,1} \leq s^{1/2} \|x^1\|_2 \leq \frac{s^{1/2}}{\sqrt{1-\delta}} \|Ax\|_2 + \frac{\delta}{1-\delta} \|x\|_1 \Rightarrow \kappa_s(A) \leq \frac{\delta}{1-\delta} < 1/2, \quad \beta_s(A, \|\cdot\|_2) \leq \frac{s^{1/2}}{\sqrt{1-\delta}}.$$

$$\|x^1\|_2 \leq \frac{1}{\sqrt{1-\delta}} \|Ax\|_2 + s^{-1/2} \frac{\delta}{1-\delta} \|x\|_1 \quad (!)$$

♠ Observing that $\|x^1\|_\infty \leq \|x^1\|_2$, we derive from (!) that

$$\|x\|_{1,1} \leq \frac{1}{\sqrt{1-\delta}} \|Ax\|_2 + \frac{s^{-1/2}\delta}{1-\delta} \|x\|_1,$$

meaning that *whenever A satisfies $\text{RIP}(\delta, k)$ with $\delta < 1/3$, we have $\kappa_1^+(A) \leq \frac{s^{-1/2}\delta}{1-\delta}$, and the corresponding certificate H of s -goodness can be chosen to have $\|\text{Col}_j(H)\|_2 \leq \frac{1}{\sqrt{1-\delta}}$, $1 \leq j \leq n$.*

Fact: *Our verifiable sufficient condition for s -goodness, even in its simplest form, allows to certify at least the square root of the goodness level as guaranteed by (heavily computationally intractable) RIP.*
On the other hand, whenever $n \geq 2m$, our condition for s -goodness fails to certify goodness level better than \sqrt{m} .

Numerical illustration:

Efficiently Computable Lower and Upper bounds on $s_*(A) = \max\{s : A \text{ is } s\text{-good}\}$

	m	LB I	LB II	UB
$m \times 256$ random submatrix of 256×256 Fourier matrix	128	3	5	11
	178	3	7	16
	242	5	11	26
$m \times 256$ random submatrix of 256×256 Hadamard matrix	128	2	5	7
	178	4	9	15
	242	12	26	31
$m \times 256$ Rademacher matrix	128	1	5	15
	178	2	8	24
	242	2	23	47
$m \times 256$ Gaussian matrix	128	1	5	14
	178	2	8	24
	242	2	23	47

LB I: Lower Bound on $s_*(A)$ based on
Mutual Incoherence

LB II: Lower Bound on $s_*(A)$ based on $\kappa_s^+(A)$

UB: Upper Bound on $S_*(A)$

- κ_s^+ -based goodness bounds significantly outperform bounds based on mutual incoherence
- Computability has its price: for random matrices, there is a significant gap between upper and lower goodness bounds

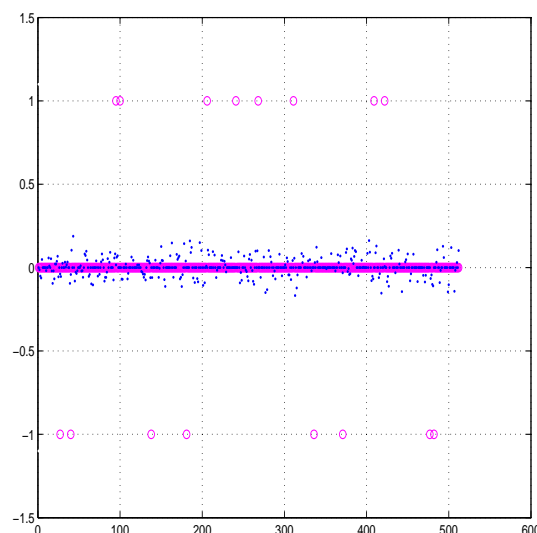
Efficiently computable goodness bounds

	m	LB I	LB II	UB
$m \times 1024$ Gaussian matrix	102	2	2	8
	204	2	4	18
	307	2	6	30
	409	3	7	44
	512	3	10	61
	614	3	12	78
	716	3	15	105
	819	4	21	135
	921	4	32	161
960×1024 convolution matrix	960	0	5	7

- *Matrices with “personal story” seem to have smaller and easier to estimate goodness than random matrices of the same sizes.*

♣ **Note:** At least in the case of random matrices A , there exists a significant gap between s -goodness (the ability of ℓ_1 recovery to recover well *all* s -sparse signals in the noiseless case) and “near s -goodness” – the ability of ℓ_1 recovery to reproduce well *with high reliability random* s -sparse signals in the noiseless case.

♠ For a randomly selected 256×512 submatrix A of the 512×512 Hadamard matrix,
— lower bound on s -goodness, as given by the condition $\kappa_s^+(A) < 0.5$, is $s = 8$
— upper bound on s -goodness is $s = 15$. Here is a badly recovered in the noiseless case 16-sparse signal:



True 16-sparse signal (magenta) and its recovery (blue)

However, in a series of 100 experiments with noiseless ℓ_1 recovery of randomly generated 81-sparse signals, not a *single* erroneous recovery was observed!

Selected Engineering Applications of LP, II

Synthesis of Linear Controllers

♣ Consider time-varying discrete time linear dynamical system

$$\begin{array}{ll}
 x_0 = z & \text{[initial state]} \\
 x_{t+1} = A_t x_t + B_t u_t + R_t d_t & \left[\begin{array}{l} \text{state equations} \\ \bullet \ x_t: \text{state} \quad \bullet \ u_t: \text{control} \\ \bullet \ d_t: \text{external disturbance} \end{array} \right] \\
 y_t = C_t x_t + D_t d_t & \text{[observed output]}
 \end{array}$$

“closed” by *affine output-based control law*

$$u_t = g_t + \sum_{\tau=0}^t G_t^\tau y_\tau. \quad (*)$$

♠ Given finite time horizon $0 \leq t \leq N$, we want to specify a control law $(*)$ which ensures that *the state-control trajectory* $w = [x_0; \dots; x_{N+1}; u_0; \dots; u_N]$ *satisfies given design specifications*

$$a_i^T w \leq b_i, \quad 1 \leq i \leq I \quad (!)$$

for all “perturbations” $\zeta = [z; d_0; \dots; d_N]$ from a given set \mathcal{Z} (equivalent wording: satisfies design specifications *robustly w.r.t.* $\zeta \in \mathcal{Z}$).

Good news: by linearity of the system and the control law, the trajectory is affine in ζ : $w = w_\gamma^0 + W_\gamma \zeta$, $\gamma = \{g_t, G_t^\tau : 0 \leq \tau \leq t \leq N\}$.

\Rightarrow The *Analysis problem: check whether a given control law $(*)$ robustly meets the design specifications* reduces to verifying whether a system of affine constraints on ζ is satisfied by all $\zeta \in \mathcal{Z}$. This is easy, provided \mathcal{Z} is “tractable.”

- System:

$$\begin{aligned}
 x_0 &= z && \text{[initial state]} \\
 x_{t+1} &= A_t x_t + B_t u_t + R_t d_t && \left[\begin{array}{l} \text{state equations} \\ \bullet x_t: \text{state} \quad \bullet u_t: \text{control} \\ \bullet d_t: \text{external disturbance} \end{array} \right] \\
 y_t &= C_t x_t + D_t d_t && \text{[observed output]}
 \end{aligned}$$

- Controller: $u_t = g_t + \sum_{\tau=0}^t G_t^\tau y_\tau$ (*)
- Trajectory: $w = [x_0; \dots x_{N+1}; u_0; \dots; u_N] = w_\gamma^0 + W_\gamma \zeta$ [$\gamma = \{g_t, G_t^\tau\}$: control law]
- Design specifications: $a_i^T w \leq b_i, 1 \leq i \leq I$ (!)

♠ From now on, assume that \mathcal{Z} is given by polyhedral representation:

$$\mathcal{Z} = \{\zeta : \exists v : P\zeta + Qv \leq r\}$$

Then to solve the *Analysis problem*: given control law, check whether (*) ensures (!) for all $\zeta \in \mathcal{Z}$ is the same as to check whether

$$b_i \geq \max_{\zeta, v} \{a_i^T [w_\gamma^0 + W_\gamma \zeta] : P\zeta + Qv \leq r\}, 1 \leq i \leq I.$$

⇒ Verification requires solving I LO programs and is therefore easy.

$$\begin{aligned}x_0 &= z \\x_{t+1} &= A_t x_t + B_t u_t + R_t d_t \\y_t &= C_t x_t + D_t d_t\end{aligned}\tag{S}$$

$$u_t = g_t + \sum_{\tau=0}^t G_t^\tau y_\tau\tag{*}$$

Bad news: the trajectory is highly nonlinear in the parameters $\gamma = \{g_t, G_t^\tau\}$ of the control law (*). Indeed

- $x_0 = z$ is independent of $\gamma \Rightarrow y_0$ is independent of $\gamma \Rightarrow u_0$ is affine in $\gamma \Rightarrow x_1$ is affine in γ
- x_1 is affine in $\gamma \Rightarrow y_1$ is affine in $\gamma \Rightarrow u_1$ is quadratic in $\gamma \Rightarrow x_2$ is quadratic in γ
- x_2 is quadratic in $\gamma \Rightarrow y_2$ is quadratic in $\gamma \Rightarrow u_2$ is cubic in $\gamma \Rightarrow x_3$ is cubic in γ

.....
 $\Rightarrow x_k$ is polynomial of degree k in γ

\Rightarrow **The Synthesis problem:** find control law (*), if it exists, which robustly meets the design specifications seems to be intractable.

$$\begin{aligned}x_0 &= z \\x_{t+1} &= A_t x_t + B_t u_t + R_t d_t \\y_t &= C_t x_t + D_t d_t\end{aligned}\tag{S}$$

$$u_t = g_t + \sum_{\tau=0}^t G_t^\tau y_\tau \tag{*}$$

Bad news: the trajectory is highly nonlinear in the parameters $\gamma = \{g_t, G_t^\tau\}$ of the control law (*).

\Rightarrow The *Synthesis problem*: find control law (*), if it exists, which robustly meets the design specifications seems to be intractable.

Remedy: pass to affine *purified*-output-based control laws.

♠ Consider, along with system (S) “closed” by some control law, its *model*

$$\begin{aligned}\hat{x}_0 &= 0 \\\hat{x}_{t+1} &= A_t \hat{x}_t + B_t u_t \\\hat{y}_t &= C_t \hat{x}_t\end{aligned}\tag{M}$$

which we “feed” by the same controls u_t as (S). We can run the model in an on-line fashion, and thus at time t , before the decision on u_t should be made, we have at our disposal *purified output* $v_t = y_t - \hat{y}_t$

Observation: *purified outputs are known in advance affine functions of ζ completely independent on the control law in use*

Indeed, setting $\Delta_t = x_t - \hat{x}_t$, we clearly have

$$v_t = C_t \Delta_t + D_t d_t \text{ with } \Delta_{t+1} = A_t \Delta_t + R_t d_t, \Delta_0 = z,$$

System:	Model:
$x_0 = z$	$\hat{x}_0 = 0$
$x_{t+1} = A_t x_t + B_t u_t + R_t d_t$	$\hat{x}_{t+1} = A_t \hat{x}_t + B_t u_t$
$y_t = C_t x_t + D_t d_t$	$\hat{y}_t = C_t \hat{x}_t$
Purified outputs: $v_t = y_t - \hat{y}_t$	
$u_t = \begin{cases} g_t + \sum_{\tau=0}^t G_t^\tau y_\tau & \text{[output-based affine law]} & (*) \\ h_t + \sum_{\tau=0}^t H_t^\tau v_\tau & \text{[purified-output-based affine law]} & (\#) \end{cases}$	

Facts:

- ♥ Affine purified-output-based and output-based controls laws are equivalent: every mapping $\zeta \rightarrow w$ which can be obtained when “closing” (S) by a law (*), can be obtained by closing (S) by a law (#), and vice versa.
- ♥ When (S) is closed by an affine purified-output-based control law (#), the trajectory $w = W[\zeta, \eta]$ becomes *bi-affine* in ζ and in the parameters $\eta = \{h_t, H_t^\tau\}$ of the control law:

$$w = w^0[\eta] + W[\eta]\zeta \text{ with } w^0[\eta], W[\eta] \text{ affine in } \eta.$$

- ... purified outputs v_t are known in advance linear functions of the external disturbances $[z; d_0; \dots; d_N]$

\Rightarrow

- $u_t = h_t + \sum_{\tau=0}^t H_t^\tau v_\tau$ is bi-affine in $[z; d_0; \dots; d_N]$ and in $\eta = \{h_t, H_t^\tau : 0 \leq \tau \leq t \leq N\}$
- By linearity of the system, the trajectory w is *linear* in the vector $[z; d_0; \dots; d_N; u_0; \dots; u_N]$, and with affine purified-output-based control. this vector is bi-affine in $[z; d_0; \dots; d_N]$ and in η

$\Rightarrow w$ is bi-affine in $[z; d_0; \dots; d_N]$ and in η ,

as claimed.

The state-control trajectory of system “closed” with affine purified-output-based control law with parameters η is bi-affine in ζ and in η :

$$w = w^0[\eta] + W[\eta]\zeta \text{ with known affine } w^0[\cdot], W[\cdot]$$

What we want:

$$Aw \leq b \quad \forall \zeta \in \mathcal{Z} = \{\zeta : \exists v : P\zeta + Qv \leq r\}$$

Facts (continued):

♥ *Sticking to purified-output-based control laws, the Synthesis problem*

Given design specifications $a_i^T w \leq b_i$, $i \leq I$, on the state-control trajectory, find a control law, if one exists, which meets these specifications robustly w.r.t. $\zeta = [z; d_0; \dots; d_N] \in \mathcal{Z}$

becomes an infinite system of linear constraints on η :

$$a_i^T [w^0[\eta] + W[\eta]\zeta] \leq b_i \quad \forall \zeta \in \mathcal{Z}, \quad 1 \leq i \leq I.$$

*which is fact is equivalent to an explicit **finite** “moderate size” system of linear constraints on ζ and additional variables.*

Question: What the infinite system of linear constraints on η :

$$\forall(\zeta : \exists v : P\zeta + Qv \leq r) : a_i^T [w^0[\eta] + W[\eta]\zeta] \leq b_i, i \leq I$$

“wants” from η ?

Answer: It wants the optimal values in I feasible parametric LP's:

$$\begin{aligned} \text{Opt}_i[\eta] &= \max_{\zeta, v} \{a_i^T W[\eta]\zeta : P\zeta + Qv \leq r\} \\ &= \min_{y^i} \{r^T y^i : P^T y^i = W^T[\eta]a_i, Q^T y^i = 0, y^i \geq 0\} \quad [\text{LP duality}] \end{aligned}$$

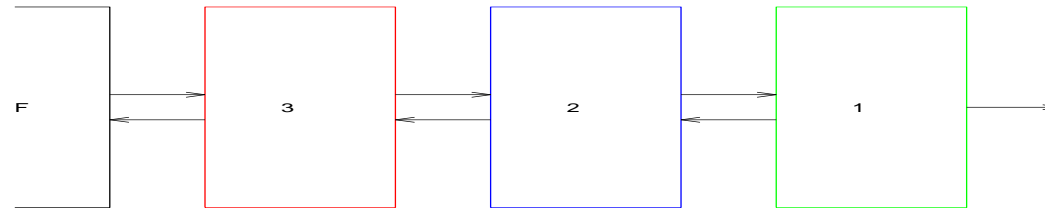
to satisfy the constraints $a_i^T w^0[\eta] + \text{Opt}_i[\eta] \leq b_i, i \leq I,$

\Rightarrow the set of desirable η admits polyhedral representation

$$\left\{ \eta : \exists y^1, \dots, y^I : \underbrace{\begin{aligned} &P^T y^i = W^T[\eta]a_i, Q^T y^i = 0, y^i \geq 0 \\ &a_i^T w^0[\eta] + r^T y^i \leq b_i \end{aligned}}_{(S)} \right\}$$

Bottom line: A purified-output-based affine control law with parameters η meets the design specifications $a_i^T w \leq b_i, 1 \leq i \leq I$, robustly in $\zeta \in \mathcal{Z}$ **iff** η can be extended by properly chosen $y^i, i \leq I$, to a feasible solution of (S).

How it Works: Controlling 3-Level Serial Inventory



3-LEVEL SERIAL INVENTORY

- Level 1 supplies external demand
- Level 2 supplies Level 1
- Level 3 supplies Level 2 and is supplied from Factory
- There is 2-period delay in executing replenishment orders

The Inventory can be modeled as the 9-state LDS

$$\begin{array}{rcll}
 x_1(t+1) & = & x_1(t) + x_{1,1}(t) & -d_t \\
 x_{1,1}(t+1) & = & x_{1,2}(t) & \\
 x_{1,2}(t+1) & = & & u_1(t) \\
 x_2(t+1) & = & x_2(t) + x_{2,1}(t) & -u_1(t) \\
 x_{2,1}(t+1) & = & x_{2,2}(t) & \\
 x_{2,2}(t+1) & = & & u_2(t) \\
 x_3(t+1) & = & x_3(t) + x_{3,1}(t) & -u_2(t) \\
 x_{3,1}(t+1) & = & x_{3,2}(t) & \\
 x_{3,2}(t+1) & = & & u_3(t) \\
 \hline
 y(t) & = & x(t) &
 \end{array}$$

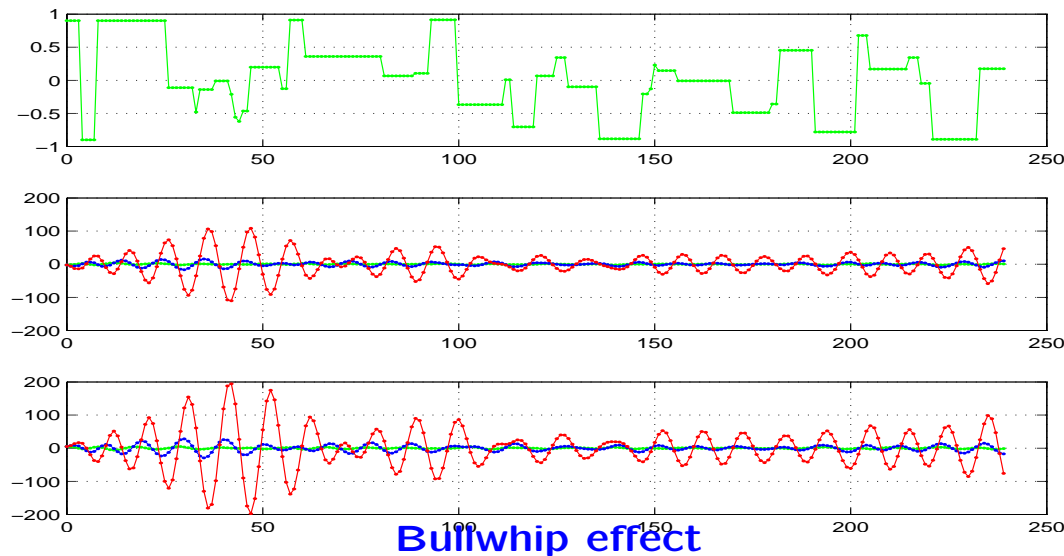
- $x_1(t)$, $x_2(t)$, $x_3(t)$ — inventory levels at the beginning of period t
- $u_1(t)$, $u_2(t)$, $u_3(t)$ — replenishment orders of period t
- $x_{p,1}(t) := u_p(t-2)$, $x_{p,2}(t) := u_p(t-1)$, $p = 1, 2, 3$
- d_t — demands



Bullwhip

♣ It is well known that serial inventories with delays (and supply chains in general) suffer from *bullwhip effect*: variations of states (e.g., inventory levels) are *severely amplified* when moving upward from external demand to production units along the supply chain. This phenomenon badly affects the production.

- This is what happens with “naive” affine controller:



Bullwhip effect

Top: time-dependent demand $d_t \in [-1, 1]$

Middle: replenishment orders $u_1(t), u_2(t), u_3(t) \in [-110, 110]$

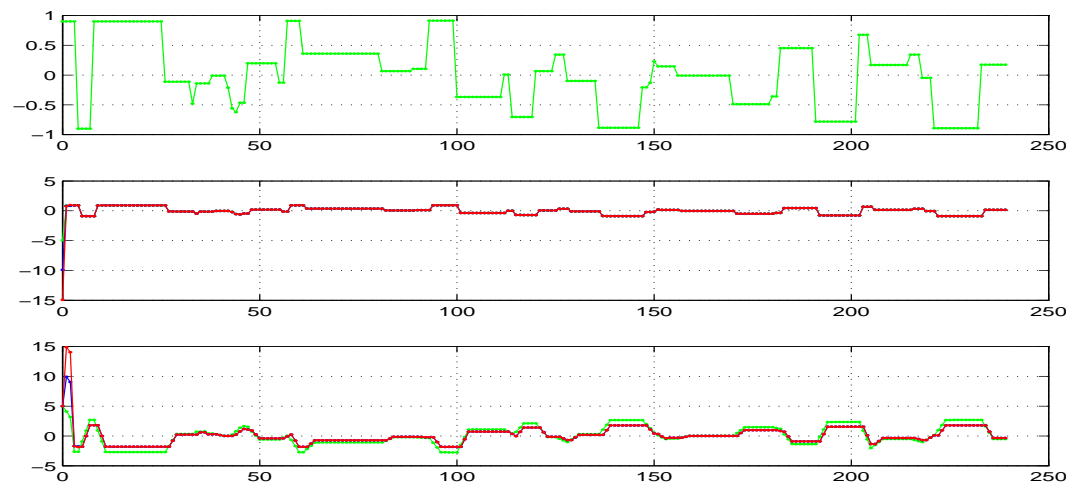
Bottom: inventory levels $x_1(t), x_2(t), x_3(t) \in [-200, 200]$

Note: variations of the demand in the range $[-1, 1]$ result in huge (hundreds!) oscillations in the level #3 and in the replenishment orders.

♥ To reduce the bullwhip effect, we can look for the best — with the largest decay rate as certified by Lyapunov Stability Certificate, whatever it means — linear feedback control law

$$u(t) = Ky(t) [= Kx(t)].$$

With this control, the picture looks much better:



Good linear feedback

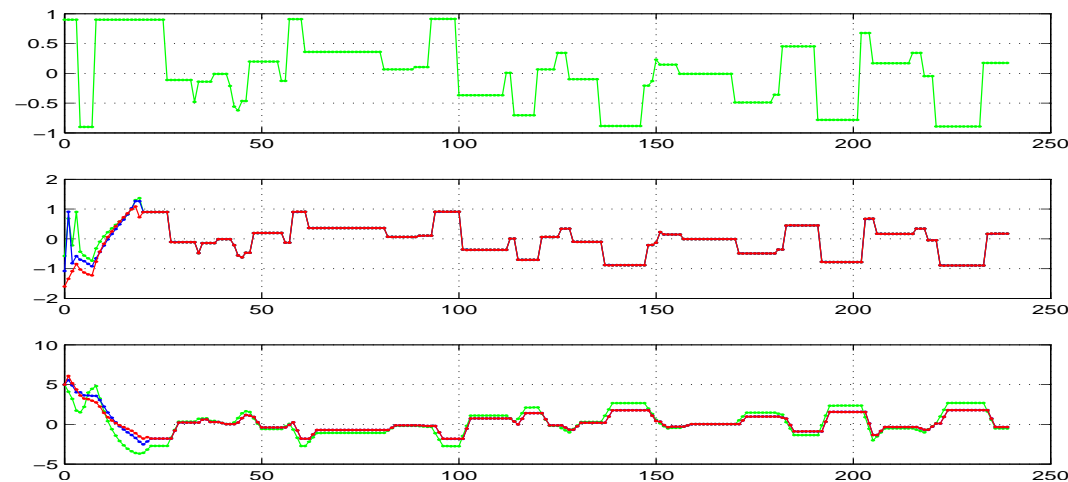
Top: time-dependent demand $d_t \in [-1, 1]$

Middle: replenishment orders $u_1(t)$, $u_2(t)$, $u_3(t) \in [-15, 5]$

Bottom: inventory levels $x_1(t)$, $x_2(t)$, $x_3(t) \in [-5, 15]$

But: At the very beginning, we still have unpleasant jumps in the inventory levels and replenishment orders.

♥ To improve the behaviour of the process in the beginning, we can use purified-output-based affine control aimed at minimizing the initial jumps and eventually switching to the above feedback control. This is what we get:



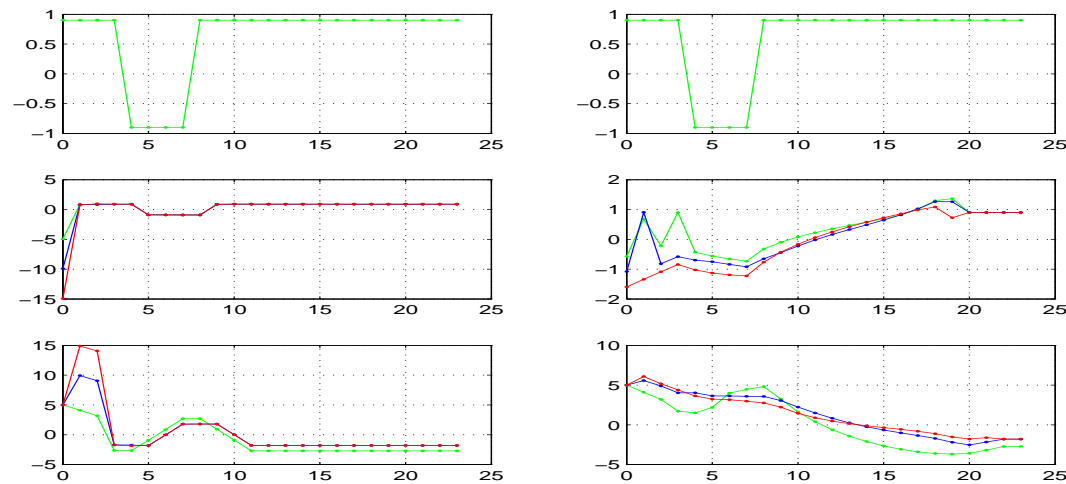
Combined p.o.b./feedback control

Top: time-dependent demand $d_t \in [-1, 1]$

Middle: replenishment orders $u_1(t)$, $u_2(t)$, $u_3(t)$ in

Bottom: inventory levels $x_1(t)$, $x_2(t)$, $x_3(t)$

♥ This is what we gain in the beginning, while loosing nothing in the long run:



Pure feedback control (left)

vs.

combined p.o.b/feedback control (right)

Top: time-dependent demand varying in $[-1, 1]$

Middle: replenishment orders $u_1(t)$, $u_2(t)$, $u_3(t)$

Bottom: inventory levels $x_1(t)$, $x_2(t)$, $x_3(t)$

From Linear to Conic Programming

♣ When passing from a generic LP problem

$$\min_x \{c^T x : Ax - b \geq 0\} \quad [A : m \times n] \quad (\text{LP})$$

to nonlinear extensions, some components of the problem become nonlinear. The traditional way is to allow nonlinearity of the objective and the constraints:

$$c^T x \mapsto c(x); a_i^T x - b_i \mapsto a_i(x)$$

and to preserve the “coordinate-wise” interpretation of the vector inequality $A(x) \geq 0$:

$$A(x) \equiv \begin{bmatrix} a_1(x) \\ \vdots \\ a_m(x) \end{bmatrix} \geq 0 \Leftrightarrow a_i(x) \geq 0, \quad i = 1, \dots, m.$$

• An alternative is to preserve the linearity of the objective and the constraint functions and to modify the interpretation of the vector inequality “ \geq ”. In Convex Programming, both approaches are equivalent.

♣ The second option turns out to be more preferable, since it “reveals the structure” of a convex program: an extremely wide variety of convex programs can be captured by vector inequalities of just 3 “standard” and well understood types.

Example: The problem with nonlinear objective and constraints

	minimize $\sum_{\ell=1}^n x_{\ell}^2$
(a)	$x \geq 0;$
(b)	$a_{\ell}^T x \leq b_{\ell}, \ell = 1, \dots, n;$
(c)	$\ Px - p\ _2 \leq c^T x + d;$
(d)	$x_{\ell}^{\frac{\ell+1}{\ell}} \leq e_{\ell}^T x + f_{\ell}, \ell = 1, \dots, n;$
(e)	$x_{\ell}^{\frac{1}{\ell+3}} x_{\ell+1}^{\frac{1}{\ell+3}} \geq g_{\ell}^T x + h_{\ell}, \ell = 1, \dots, n-1;$
(f)	$\text{Det} \left(\begin{bmatrix} x_1 & x_2 & x_3 & \cdots & x_n \\ x_2 & x_1 & x_2 & \cdots & x_{n-1} \\ x_3 & x_2 & x_1 & \cdots & x_{n-2} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ x_n & x_{n-1} & x_{n-2} & \cdots & x_1 \end{bmatrix} \right) \geq 1;$
(g)	$1 \leq \sum_{\ell=1}^n x_{\ell} \cos(\ell\omega) \leq 1 + \sin^2(5\omega) \forall \omega \in [-\frac{\pi}{7}, 1.3]$

can be converted, in a systematic way, into an equivalent problem

$$\min_x \{c^T x : Ax - b \succeq 0\},$$

" \succeq " being one of the 3 standard vector inequalities, so that seemingly highly diverse constraints of the original problem allow for unified treatment.

- A significant part of nice mathematical properties of an LP program

$$\min_x \{c^T x : Ax - b \geq 0\}$$

stems from the fact that the underlying coordinate-wise vector inequality

$$a \geq b \Leftrightarrow a_i \geq b_i, \quad i = 1, \dots, m \quad [a, b \in \mathbf{R}^m]$$

satisfies a number of quite general axioms, namely:

I. It defines a *partial ordering* of \mathbf{R}^m , i.e., is

I.a) *reflexive*: $a \geq a$ for all $a \in \mathbf{R}^m$

I.b) *anti-symmetric*: if $a \geq b$ and $b \geq a$, then $a = b$

I.c) *transitive*: if $a \geq b$ and $b \geq c$, then $a \geq c$

II. It is *compatible with linear structure of \mathbf{R}^m* , i.e., is

II.a) *additive*: if $a \geq b$ and $c \geq d$, then $a + c \geq b + d$

II.b) *homogeneous*: if $a \geq b$ and λ is nonnegative real, then $\lambda a \geq \lambda b$.

“Good” vector inequalities

- A vector inequality \succeq on \mathbf{R}^m is a **binary relation** – a set of ordered pairs (a, b) with $a, b \in \mathbf{R}^m$. The fact that a pair (a, b) belongs to this set is written down as $a \succeq b$ (“ a \succeq -dominates b ”).
- Let us call a vector inequality \succeq **good**, if it satisfies the outlined axioms, namely, is *reflexive, antisymmetric, transitive, additive, and homogeneous*.

Observation: A good vector inequality \succeq on \mathbf{R}^m is uniquely defined by the set

$$\mathbf{K} = \{a \in \mathbf{R}^m : a \succeq 0\}$$

of all $\succeq 0$ -nonnegative vectors, specifically,

$$a \succeq b \Leftrightarrow a - b \succeq 0 \Leftrightarrow a - b \in \mathbf{K}$$

A set $\mathbf{K} \subset \mathbf{R}^m$ specifies, in the above fashion, a good vector inequality **iff** \mathbf{K} is a pointed convex cone, that is,

- is nonempty,
- is conic: $a \in \mathbf{K}, \lambda \geq 0 \Rightarrow \lambda a \in \mathbf{K}$
- is convex,
- is pointed: $a \in \mathbf{K}$ and $-a \in \mathbf{K}$ iff $a = 0$,

or, equivalently, is a nonempty subset of \mathbf{R}^m closed w.r.t. taking conic combinations (linear combinations with nonnegative coefficients) of its elements and not containing lines passing through the origin.

Example: The entrywise vector inequality \geq stems from the *nonnegative orthant* \mathbf{R}_+^m :

$$a \geq b \Leftrightarrow a - b \geq 0 \Leftrightarrow a - b \in \mathbf{R}_+^m = \{x \in \mathbf{R}^m : x_i \geq 0, 1 \leq i \leq m\}.$$

The nonnegative orthant \mathbf{R}_+^m , along with being a pointed convex cone, possesses two additional properties:

- *is closed*, and
- *possesses nonempty interior*.

The first property allows to pass to termwise limits in \geq inequalities:

$$a_i \geq b_i \ \& \ a = \lim_i a_i \ \& \ b = \lim_i b_i \quad \Rightarrow \quad a \geq b.$$

The second property allows to define *strict version* $>$ of \geq :

$$a > b \Leftrightarrow a - b \in \text{int } \mathbf{R}_+^m [= \{x \in \mathbf{R}^m : x_i > 0, i \leq m\}]$$

which is stable w.r.t. small enough perturbations of a , b .

It makes sense to incorporate these useful properties into the definition of a "good" vector inequality

Bottom line: From now on, a good vector inequality on \mathbf{R}^m is the relation $\geq_{\mathbf{K}}$ specified by a *regular cone* (closed convex pointed cone with a nonempty interior) $\mathbf{K} \subset \mathbf{R}^m$ according to

$$a \geq_{\mathbf{K}} b \Leftrightarrow a - b \geq_{\mathbf{K}} 0 \Leftrightarrow a - b \in \mathbf{K}.$$

Along with $\geq_{\mathbf{K}}$, the cone \mathbf{K} specifies the strict inequality $>_{\mathbf{K}}$:

$$a >_{\mathbf{K}} b \Leftrightarrow a - b >_{\mathbf{K}} 0 \Leftrightarrow a - b \in \text{int } \mathbf{K}.$$

Note: Arithmetics and elementary topology of good vector inequalities $\geq_{\mathbf{K}}$, $>_{\mathbf{K}}$ is exactly the same as for entrywise vector inequality \geq (and the scalar \geq), e.g.

- *sum of two valid nonstrict/strict \mathbf{K} -inequalities is a valid nonstrict \mathbf{K} -inequality, and is strict, if at least one of the two inequalities we are summing up is strict;*
- *multiplying both sides of a valid nonstrict/strict \mathbf{K} -inequality by a non-negative real, we get valid nonstrict \mathbf{K} -inequality which is strict, provided that the real is positive and the original inequality was strict;*
- *small enough perturbations in both sides of a valid strict \mathbf{K} -inequality preserve inequality's validity;*
- *if left- and right hand sides in a sequence of valid \mathbf{K} -inequalities have limits, these limits are linked by valid nonstrict \mathbf{K} -inequality.*

Facts:

A. *The entrywise vector inequality*

$$a \geq b \Leftrightarrow a_i \geq b_i, i = 1, \dots, m$$

is neither the only possible, nor the only interesting good vector inequality on \mathbf{R}^m .

B. *A good vector inequality $\geq_{\mathbf{K}}$ gives rise to generic **conic program***

$$\min_x \left\{ c^T x : Ax - b \geq_{\mathbf{K}} 0 \right\},$$

and these programs inherit significant part of nice theoretical properties of LP's.

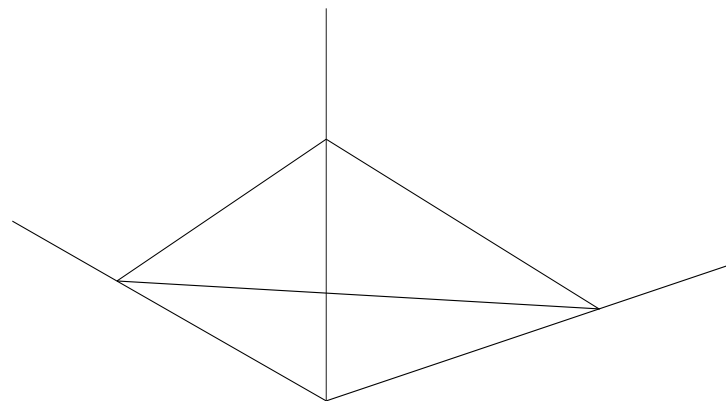
At the same time, "playing with \mathbf{K} " – working with regular cones different from nonnegative orthants – extends dramatically the scope of convex optimization problems we can handle. Moreover, for all practical purposes just three "magic" families of regular cones cover the entire Convex Programming.

Magic families of cones, I

Nonnegative Orthants

♣ **Direct products of nonnegative rays** — nonnegative orthants — give rise to the entrywise vector inequalities and thus – to generic Linear Programming problem

$$\min_{x \in \mathbf{R}^n} \{c^T x : Ax - b \geq 0\} \quad [A \in \mathbf{R}^{m \times n}]$$



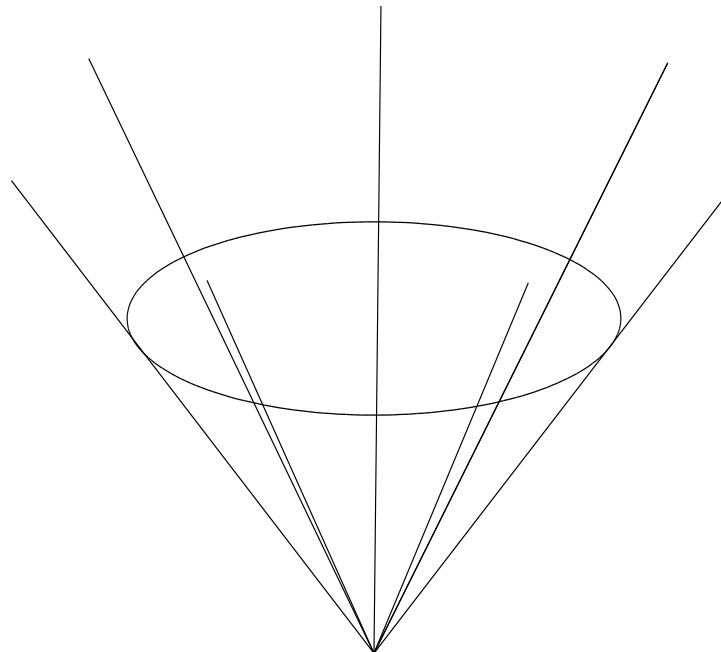
The nonnegative orthant \mathbf{R}^3

Magic families of cones, II

Direct products of Lorenz cones

♣ m -dimensional **Lorentz cone** (a.k.a. *Second Order, or Ice-Cream, cone*) is defined as

$$\mathbf{L}^m = \left\{ x = [x_1; \dots; x_m] \in \mathbf{R}^m : x_m \geq \sqrt{\sum_{i=1}^{m-1} x_i^2} \right\}$$



The ice-cream cone \mathbf{L}^3

♣ **Direct products of Lorentz cones** give rise to *Conic Quadratic* (a.k.a. Second Order Conic) programs. A generic Conic Quadratic problem is of the form

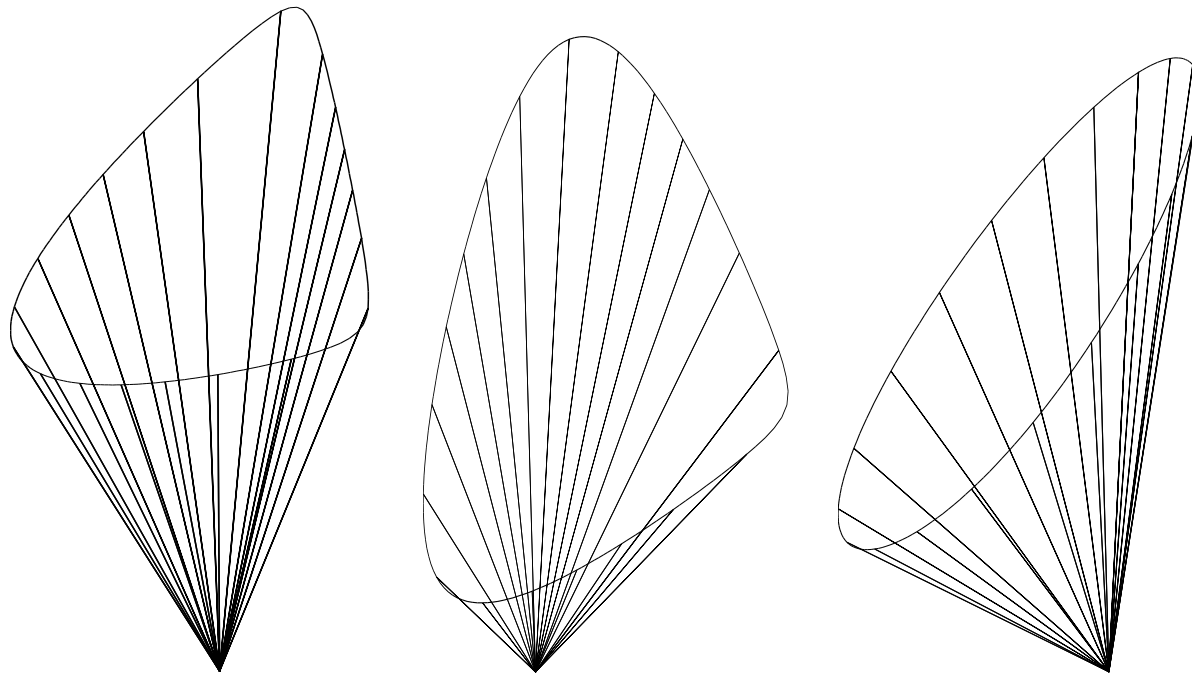
$$\begin{aligned}
 & \left[\begin{array}{c} D_i x + d_i \\ e_i^T x + f_i \end{array} \right] \in \mathbf{L}^{m_i} \\
 & \quad \quad \quad \updownarrow \\
 & \min_x \left\{ c^T x : \overbrace{\|D_i x + d_i\|_2 \leq e_i^T x + f_i, 1 \leq i \leq m} \right\} \\
 & \quad \quad \quad \updownarrow \\
 & \min_x \left\{ c^T x : Ax - b \equiv \left[\begin{array}{c} \left[\begin{array}{c} D_1 x + d_1 \\ e_1^T x + f_1 \end{array} \right] \\ \vdots \\ \left[\begin{array}{c} D_m x + d_m \\ e_m^T x + f_m \end{array} \right] \end{array} \right] \geq_{\mathbf{K}} 0 \right\}, \\
 & \quad \quad \quad \mathbf{K} = \mathbf{L}^{m_1} \times \dots \times \mathbf{L}^{m_k}
 \end{aligned}$$

is a direct product of Lorentz cones

Magic families of cones, III

Directs products of semidefinite cones

♣ **Semidefinite cone** S_+^m lives in the space S^m of real symmetric $m \times m$ matrices and is comprised of all $m \times m$ symmetric matrices A which are *positive semidefinite*, that is, produce everywhere nonnegative quadratic forms $x^T A x$ or, equivalently, have nonnegative eigenvalues.



3 random 3D cross-sections of S_+^3

♣ **Direct products of semidefinite cones** give rise to *semidefinite* programs

$$\min_x \left\{ c^T x : \mathcal{A}_i(x) := \sum_j x_j A_{ij} - B_i \succeq 0, i \leq I \right\},$$

where A_{ij} , B_i are symmetric matrices of size m_i , and $P \succeq Q$ ($\equiv Q \preceq P$) means that P, Q are symmetric matrices of the same size such that $P - Q$ is positive semidefinite.

Note: *Semidefinite program* is the program of minimizing a linear objective under the bunch of *LMI (Linear Matrix Inequality) constraints* stating each that a variable symmetric matrix with entries affine in the decision vector x should be positive semidefinite.

Note: We can always write down a semidefinite program as a program with *single* LMI constraint:

$$\min_x \{ c^T x : \mathcal{A}_i(x) \succeq 0, i \leq m \} \Leftrightarrow \min_x \{ c^T x : \mathcal{A}(x) := \text{Diag}\{\mathcal{A}_1(x), \dots, \mathcal{A}_m(x)\} \succeq 0 \}.$$

Conic Duality

- Let us look at the origin of the problem dual to an LP program

$$\min_x \{c^T x : Ax - b \geq 0\}. \quad (\text{LPr})$$

Observing that any *nonnegative* “weight vector” $y \in \mathbf{R}_+^m$ is “admissible” for the constraint-wise vector inequality on \mathbf{R}^m :

$$\forall a, b, y \in \mathbf{R}^m : a \geq b \ \& \ y \geq 0 \Rightarrow \underbrace{y^T a \geq y^T b}_{\text{usual scalar inequality}}$$

we conclude that *all scalar linear inequalities of the type*

$$[A^T y]^T x \geq b^T y \quad \text{with } y \geq 0$$

with variables x are consequences of the constraints of (LPr). Thus,

(*) *If $y \geq 0$ is such that $A^T y = c$, then $b^T y$ is a lower bound on the optimal value in (LPr).*

- The LP dual to (LPr) is exactly the problem

$$\max_y \{b^T y : A^T y = c, y \geq 0\} \quad (\text{LDI})$$

of finding the best – the largest – lower bound on the optimal value of (LPr) among those given by (*).

- Conic Duality, same as the LP one, is inspired by the desire to bound from below the optimal value in a conic program

$$\min_x \{c^T x : Ax - b \geq_{\mathbf{K}} 0\} \quad (\text{CP})$$

and follows the just outlined scheme based on “conversion” of vector inequalities into the scalar ones:

$$a \geq_{\mathbf{K}} b \Rightarrow y^T a \geq y^T b, \quad (*)$$

Crucial question is:

What are the “aggregation weights” y which make $()$ valid?*

Answer: A necessary and sufficient condition for the implication $(*)$ to be true is

$$y \in \mathbf{K}_* := \{y : y^T x \geq 0 \forall x \in \mathbf{K}\}$$

Note: \mathbf{K}_* is called the cone *dual* to \mathbf{K} . Whenever \mathbf{K} is a regular cone, so is \mathbf{K}_* , and

$$\mathbf{K} = (\mathbf{K}_*)_*.$$

♠ We are ready to build the dual of a conic program. It is convenient to start with the primal problem in the form

$$\text{Opt}(P) = \min_x \{c^T x : Ax - b \in \mathbf{K}, Rx = r\} \quad (P)$$

To build the dual, we equip the constraints of (P) with *Lagrange multipliers*

$$y \in \mathbf{K}_*, s \in \mathbf{R}^{\dim r}$$

so that $y^T[Ax - b] + s^T[Rx - r] \geq 0$ for every x feasible for (P) .

Note: the "aggregated constraint"

$$[A^T y]^T x + [R^T s]^T x \geq b^T y + r^T s,$$

by its origin is a consequence of the constraints of (P) . Consequently, Whenever $A^T y + R^T s = c$, the quantity $b^T y + r^T s$ is a lower bound on $\text{Opt}(P)$. The problem

$$\max_{y,s} \{b^T y + r^T s : y \in \mathbf{K}_*, A^T y + R^T s = c\}$$

dual to (P) is to find the best – the largest – bound of this type.

♣ "In real life" a conic problem arises as

$$\text{Opt}(P) = \min_x \{c^T x : A_i x - b_i \in \mathbf{K}^i, i \leq m, Rx = r\} \quad (P)$$

that is, the associated regular cone is the direct product $\mathbf{K} = \mathbf{K}^1 \times \dots \times \mathbf{K}^m$. We clearly have

$$\mathbf{K}_* = \mathbf{K}_*^1 \times \dots \times \mathbf{K}_*^m,$$

implying that the recipe for building the dual problem is as follows:

- we equip conic constraints $A_i x - b_i \in \mathbf{K}^i$ with Lagrange multipliers $y^i \in \mathbf{K}_*^i$, and the linear equality constraints – with Lagrange multiplier $s \in \mathbf{R}^{\dim r}$
- we induce from the constraints of (P) that $[y^i]^T [A_i x - b_i] \geq 0$ and $s^T [Rx - r] \geq 0$, so that the aggregated constraint

$$[\sum_i A_i^T y^i + R^T s]^T x \geq \sum_i b_i^T y^i + r^T s$$

is the consequence of the constraints of (P). In particular, whenever $y^i \in \mathbf{K}_*^i$ and s satisfy $\sum_i A_i^T y^i + R^T s = c$, the quantity $\sum_i b_i^T y^i + r^T s$ is a lower bound on $\text{Opt}(P)$. The dual problem

$$\text{Opt}(D) = \max_{y^i, s} \left\{ \sum_i b_i^T y^i + r^T s : y^i \in \mathbf{K}_*^i, i \leq m, \sum_i A_i^T y^i + R^T s = c \right\}$$

is to find the best – the largest – of these lower bounds on $\text{Opt}(P)$.

Note: The dual problem is conic along with the primal problem.

Note: The magic cones are self-dual, so that in this case (D) involves the same cones as (P).

$$\text{Opt}(P) = \min_x \{c^T x : Ax - b \in \mathbf{K}, Rx = r\} \quad (P)$$

$$\text{Opt}(D) = \max_{y,s} \{b^T y + r^T s : y \in \mathbf{K}_*, A^T y + R^T s = c\} \quad (D)$$

♠ The origin of the dual problem yields the

Weak Duality Theorem: $\text{Opt}(P) \geq \text{Opt}(D)$.

Equivalently: *The value of the primal objective $c^T x$ at every primal feasible solution (one feasible for (P)) is \geq the value of the dual objective $b^T y + r^T s$ at every dual feasible solution $[y; s]$ (one feasible for (D)).*

Equivalently: The *duality gap*

$$\text{DualityGap}(x; y, s) = c^T x - [b^T y + r^T s]$$

evaluated at a primal-dual feasible pair $x, [y; s]$, always is nonnegative.

Geometry of primal-dual pair of conic problems

$$\begin{aligned}\text{Opt}(P) &= \min_x \{c^T x : Ax - b \in \mathbf{K}, Rx = r\} & (P) \\ \text{Opt}(D) &= \max_{y,s} \{b^T y + r^T s : y \in \mathbf{K}_*, A^T y + R^T s = c\} & (D)\end{aligned}$$

Assumption: The systems of linear equality constraints in (P) and (D) are solvable:

$$\exists \bar{x}, [\bar{y}, \bar{s}] : R\bar{x} = r, A^T \bar{y} + R^T \bar{s} = c.$$

A: Let us pass in (P) from variable x to *primal slack* $\eta = Ax - b$. Whenever x satisfies $Rx = r$, we have

$$c^T x = [A^T \bar{y} + R^T \bar{s}]^T x = \bar{y}^T Ax + \bar{s}^T Rx = \bar{y}^T [Ax - b] + [b^T \bar{y} + r^T \bar{s}]$$

\Rightarrow (P) is equivalent to the conic problem

$$\begin{aligned}\text{Opt}(\mathcal{P}) &= \min_{\eta} \{ \bar{y}^T \eta : \eta \in [\mathcal{L} - \bar{\eta}] \cap \mathbf{K} \}, \mathcal{L} = \{Ax : Rx = 0\}, \bar{\eta} = b - A\bar{x} \\ &[\text{Opt}(\mathcal{P}) = \text{Opt}(P) - [b^T \bar{y} + r^T \bar{s}]]\end{aligned} \tag{\mathcal{P}}$$

Explanation: (P) wants of $\eta := Ax - b$ (a) to belong to \mathbf{K} , and (b) to be representable as $Ax - b$ for some x satisfying $Rx = r$. (b) says that η should belong to the *primal affine plane* $\{Ax - b : Rx = r\}$, which is the shift of the parallel linear subspace $\mathcal{L} = \{Ax : Rx = 0\}$ by a (whatever) vector from the primal affine plane, e.g., the vector $-\bar{\eta} = A\bar{x} - b$.

$$\begin{aligned}\text{Opt}(P) &= \min_x \{c^T x : Ax - b \in \mathbf{K}, Rx = r\} & (P) \\ \text{Opt}(D) &= \max_{y,s} \{b^T y + r^T s : y \in \mathbf{K}_*, A^T y + R^T s = c\} & (D)\end{aligned}$$

B. Let us pass in (D) from variables $[y; s]$ to variable y . Whenever $[y; s]$ satisfies $A^T y + R^T s = c$, we have

$$b^T y + r^T s = b^T y + \bar{x}^T R^T s = b^T y + \bar{x}^T [c - A^T y] = [b - A\bar{x}]^T y + c^T \bar{x} = \bar{\eta}^T y + c^T \bar{x},$$

\Rightarrow (D) is equivalent to the conic problem

$$\begin{aligned}\text{Opt}(\mathcal{D}) &= \max_y \{ \bar{\eta}^T y : y \in [\mathcal{L}^\perp + \bar{y}] \cap \mathbf{K}_* \} \\ &\quad \left[\text{Opt}(\mathcal{D}) = \text{Opt}(D) - c^T \bar{x} \right] & (\mathcal{D})\end{aligned}$$

Explanation: (D) wants of y (a) to belong to \mathbf{K}_* , and (b) to satisfy $A^T y = c - R^T s$ for some s . (b) says that y should belong to the *dual affine plane* $\{y : \exists s : A^T y + R^T s = c\}$, which is the shift of the parallel linear subspace $\tilde{\mathcal{L}} = \{y : \exists s : A^T y + R^T s = 0\}$ by a (whatever) vector from the dual affine plane, e.g., the vector \bar{y} .

Elementary Linear Algebra says that $\tilde{\mathcal{L}} = \mathcal{L}^\perp$. Indeed,

$$\begin{aligned}[\tilde{\mathcal{L}}]^\perp &= \{z : z^T y = 0 \forall y : \exists s : A^T y + R^T s = 0\} = \{z : z^T y + 0^T s = 0 \text{ whenever } A^T y + R^T s = 0\} \\ &= \{z : \exists x : [z^T, 0] = x^T [A^T, R^T]\} = \{z : \exists x : Ax = z, Rx = 0\} = \mathcal{L}.\end{aligned}$$

$$\begin{aligned}\text{Opt}(P) &= \min_x \{c^T x : Ax - b \in \mathbf{K}, Rx = r\} & (P) \\ \text{Opt}(D) &= \max_{y,s} \{b^T y + r^T s : y \in \mathbf{K}_*, A^T y + R^T s = c\} & (D)\end{aligned}$$

♣ **Bottom line:** Problems (P), (D) are equivalent, respectively, to

$$\begin{aligned}\text{Opt}(\mathcal{P}) &= \min_{\eta} \{\bar{y}^T \eta : \eta \in [\mathcal{L} - \bar{\eta}] \cap \mathbf{K}\} & (\mathcal{P}) \\ \text{Opt}(\mathcal{D}) &= \max_y \{\bar{\eta}^T y : y \in [\mathcal{L}^\perp + \bar{y}] \cap \mathbf{K}_*\} & (\mathcal{D}) \\ [\mathcal{L} = \{Ax : Rx = 0\}, R\bar{x} = r, \bar{\eta} = b - A\bar{x}, A^T \bar{y} + R^T \bar{s} = c]\end{aligned}$$

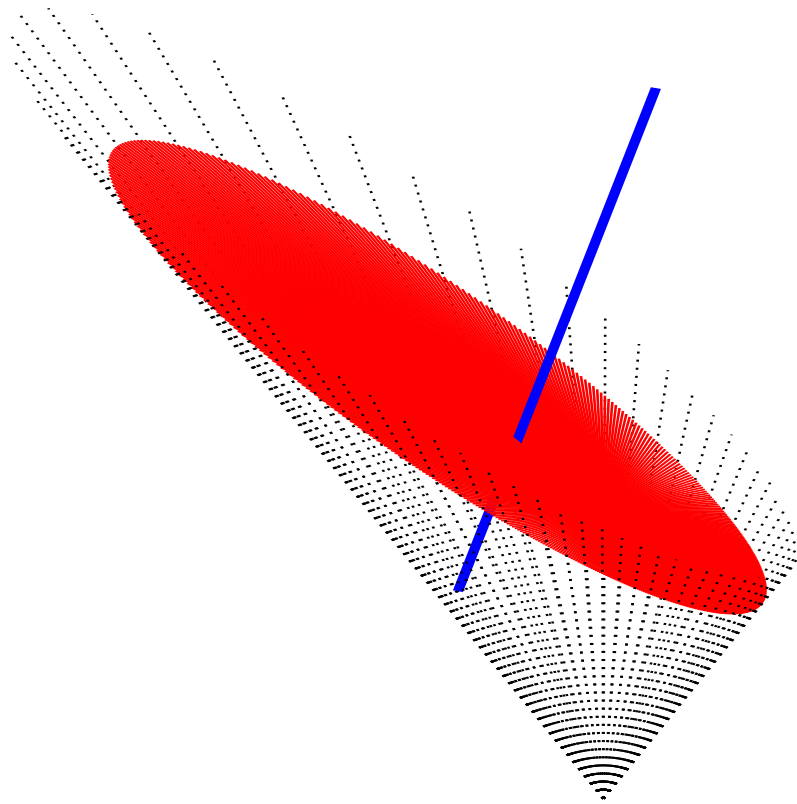
Note: When x is feasible for (P), and $[y; s]$ is feasible for (D), the vectors $\eta = Ax - b$, y are feasible for (P), resp., (D), and

$$\text{DualityGap}(x; [y; s]) = c^T x - b^T y - r^T s = [A^T y + R^T s]^T x - b^T y - r^T s = [Ax - b]^T y = \eta^T y$$

\Rightarrow Geometrically, (P), (D) are as follows: "geometric data" of the problems are the pair of linear subspaces \mathcal{L} , \mathcal{L}^\perp in the space where \mathbf{K} , \mathbf{K}_* live, the subspaces being orthogonal complements to each other, and pair of vectors $\bar{\eta}$, \bar{y} in this space.

- (P) is equivalent to minimizing $f(\eta) = \bar{y}^T \eta$ over the intersection of \mathbf{K} and the primal feasible plane \mathcal{M}_P which is the shift of \mathcal{L} by $-\bar{\eta}$
- (D) is equivalent to maximizing $g(y) = \bar{\eta}^T y$ over the intersection of \mathbf{K}_* and the dual feasible plane \mathcal{M}_D which is the shift of \mathcal{L}^\perp by \bar{y}
- taken together, (P) and (D) form the problem of minimizing the duality gap over feasible solutions to the problems, which is exactly the problem of finding pair of vectors in $\mathcal{M}_P \cap \mathbf{K}$ and $\mathcal{M}_D \cap \mathbf{K}_*$ as close to orthogonality as possible.

Pay attention to the ideal geometrical primal-dual symmetry we observe.



Primal-dual pair of conic problems on 3D Lorentz cone

Red: feasible set of (\mathcal{P}) Blue: feasible set of (\mathcal{D})

Conic Duality Theorem

♠ **Definition.** A conic problem of optimizing a linear objective under the constraints

$$Ax - b \in \mathbf{K}, Rx = r$$

is called *strictly feasible*, if there exists a feasible solution \bar{x} which *strictly* satisfies the conic constraint:

$$\exists \bar{x} : R\bar{x} = r \ \& \ A\bar{x} - b \in \text{int } \mathbf{K}.$$

● Assuming that the conic constraint is split into "general" and "polyhedral" parts, so that the feasible set is given by

$$Ax - b \in \mathbf{K}, Px - p \geq 0, Rx = r$$

the problem is called *essentially strictly feasible*, if there exists a feasible solution \bar{x} which strictly satisfies the "general" conic constraint:

$$\exists \bar{x} : R\bar{x} = r, P\bar{x} - p \geq 0, A\bar{x} - b \in \text{int } \mathbf{K}.$$

- Finally, we say that a single conic constraint

$$Bx - c \in \mathbf{M} \quad (*)$$

is *essentially strictly feasible*, if the regular cone \mathbf{M} can be represented as $\mathbf{M} = \mathbf{K} \times \mathbf{R}_+^k$ in such a way that the constraint

$$Bx - c \in [\text{int } \mathbf{K}] \times \mathbf{R}_+^k$$

is feasible.

Equivalently: Essential strict feasibility of $(*)$ means that \mathbf{M} can be decomposed as $\mathbf{M} = \mathbf{K} \times \mathbf{R}_+^k$, and the induced by this decomposition equivalent form

$$Ax - b \in \mathbf{K}, Px - p \geq 0 \quad [Bx - c \equiv [Ax - b; Px - p]]$$

of $(*)$ is such that

$$\exists \bar{x} : A\bar{x} - b \in \text{int } \mathbf{K} \ \& \ P\bar{x} - p \geq 0$$

Note: When the conic constraint in the primal problem allows for splitting into "general" and "polyhedral" parts:

$$\text{Opt}(P) = \min_x \left\{ c^T x : Ax - b \in \mathbf{K}, Px - p \geq 0, Rx = r \right\} \quad (P)$$

then the dual problem reads

$$\text{Opt}(D) = \max_{y,z,s} \left\{ b^T y + p^T z + r^T s : y \in \mathbf{K}_*, z \geq 0, A^T y + P^T z + R^T s = c \right\} \quad (D)$$

so that its conic constraint also is split into "general" and "polyhedral" parts.

♠ **Conic Duality Theorem** Consider conic program along with its dual:

$$\text{Opt}(P) = \min_x \{c^T x : Ax - b \in \mathbf{K}, Rx = r\} \quad (P)$$

$$\text{Opt}(D) = \max_{y,s} \{b^T y + r^T s : y \in \mathbf{K}_*, A^T y + R^T s = c\} \quad (D)$$

Then

♠ **Primal-Dual Symmetry:** The duality is symmetric: (D) is conic along with (P) and the problem dual to (D) is (equivalent to) (P).

♠ **Weak Duality:** One has $\text{Opt}(D) \leq \text{Opt}(P)$.

♠ **Strong Duality:** Assume that one of the problems (P), (D) is strictly feasible and bounded, boundedness meaning on the feasible set the objective is bounded from below in the minimization and from above - in the maximization case. Then the other problem in the pair is solvable, and

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

In particular, if both problems are strictly feasible (and thus both are bounded by Weak Duality), then both problems are solvable with equal optimal values.

In addition, if one of the problems is strictly feasible, then $\text{Opt}(P) = \text{Opt}(D)$.

Refinement

Let the conic constraints in (P) , (D) be split into "general" and "polyhedral" parts, so that the problems read

$$\begin{aligned}\text{Opt}(P) &= \min_x \{c^T x : Ax - b \in \mathbf{K}, Px - p \geq 0, Rx = r\} & (P) \\ \text{Opt}(D) &= \max_{y,z,s} \{b^T y + p^T z + r^T s : y \in \mathbf{K}_*, z \geq 0, A^T y + P^T z + R^T s = c\} & (D)\end{aligned}$$

Then Strong Duality can be strengthened to the following claim: *If one of the problems is essentially strictly feasible and bounded, then the other problem is solvable, and*

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

In particular, if both problems are essentially strictly feasible, both are solvable with equal optimal values.

In addition, if one of the problems is essentially strictly feasible, then $\text{Opt}(P) = \text{Opt}(D)$.

Note:

A. When no "general" conic constraint is present (i.e., in the LP situation) Refined Conic Duality Theorem is equivalent to LP Duality Theorem.

B. In general, the difference between the Strong Duality part of Conic duality Theorem and LP Duality Theorem is that the former requires (essential) *strict* feasibility, while the latter requires *just feasibility*. This difference "reflects reality" – when at least one of the primal-dual pair of problems is *not* essentially strictly feasible, various "pathologies" can arise. It can be shown by examples that it is possible that in a primal-dual pairs (P) , (D) of conic programs,

- one of the problems is strictly feasible and bounded (implying that the other problem is solvable and $\text{Opt}(P) = \text{Opt}(D)$), but is *not* solvable;
- one of the problems is solvable, and the other one is infeasible,
- both problems are solvable, but with different optimal values: $\text{Opt}(D) < \text{Opt}(P)$.

Corollary [Optimality Conditions in Conic Programming] *Consider primal-dual pair of conic problems*

$$\text{Opt}(P) = \min_x \{c^T x : Ax - b \in \mathbf{K}, Rx = r\} \quad (P)$$

$$\text{Opt}(D) = \max_{y,s} \{b^T y + r^T s : y \in \mathbf{K}_*, A^T y + R^T s = c\} \quad (D)$$

*and assume that both problems are essentially strictly feasible. A pair $x, [y; s]$ of **primal and dual feasible** solutions is comprised of optimal solutions to the respective problems*

- [Zero Duality Gap] *iff* $\text{DualityGap}(x, [y; s]) = c^T x - [b^T y + r^T s] = 0$, and
- [Complementary Slackness] *iff* $y^T [Ax - b] = 0$.

Proof: We are in the situation when $\text{Opt}(P) = \text{Opt}(D)$ by Strong Duality part of Conic Duality Theorem. Consequently, for primal-dual feasible $x, [y; s]$ it holds

$$\text{DualityGap}(x, [y; s]) = [c^T x - \text{Opt}(P)] + [\text{Opt}(D) - b^T y - r^T s]$$

By primal-dual feasibility, both brackets are nonnegative, and their sum can be 0 iff $c^T x = \text{Opt}(P)$ and $b^T y + r^T s = \text{Opt}(D)$, as claimed in Zero Duality Gap. Next, we have

$$\begin{aligned} \text{DualityGap}(x, [y; s]) &= c^T x - b^T y - r^T s = [A^T y + R^T s]^T x - b^T y - r^T s \\ &= [Ax - b]^T y + [Rx - r]^T s = [Ax - b]^T y, \end{aligned}$$

implying that Zero Duality Gap is equivalent, for primal-dual feasible $x, [y; s]$, to Complementary Slackness.

Example: Dual to the Steiner sum problem

♣ Steiner sum problem:

$$\min_{x \in \mathbb{R}^n} \sum_{i=1}^m \|x - a_i\|_2. \quad [m > 1, a_1, \dots, a_m \text{ are distinct points in } \mathbb{R}^n]$$

“Cover story” ($n = 2$): There are m oil wells located at points $a_1, \dots, a_m \in \mathbb{R}^2$. Where should one place an oil collector in order to minimize the total length of pipelines connecting the wells to the collector?

♣ The problem can be reformulated as conic:

$$\min_{t_1, \dots, t_m, x} \left\{ \sum_{i=1}^m t_i : \underbrace{[x - a_i; t_i]}_{\Leftrightarrow \|x - a_i\|_2 \leq t_i} \in \mathbf{L}^{n+1}, i = 1, \dots, m \right\} \quad (P)$$

Lorentz cones are self-dual, so that the problem dual to (S) is obtained by
— assigning the constraints $[x - a_i; t_i]$ by Lagrange multipliers $[y_i; z_i] \in \mathbf{L}^{n+1}$ giving rise to the aggregated constraint

$$[\sum_i y_i^T]x + \sum_i z_i t_i \geq \sum_i y_i^T a_i$$

— imposing on the multipliers the restriction that the left hand side in the aggregated constraint is, identically in the primal variables x, t_i , equal to the primal objective $\sum_i t_i$, which amounts to

$$\sum_i y_i = 0, z_1 = \dots = z_m = 1$$

and maximizing under this restriction the right hand side of the aggregated constraint. Thus, the dual problem reads

$$\max_{y_1, \dots, y_m} \left\{ \sum_i a_i^T y_i : \sum_i y_i = 0, \|y_i\|_2 \leq 1, i \leq m \right\} \quad (D)$$

$$\text{Opt}(P) = \min_{t_1, \dots, t_m, x} \left\{ \sum_{i=1}^m t_i : [x - a_i; t_i] \in \mathbf{L}^{n+1}, i = 1, \dots, m \right\} \quad (P)$$

$$\text{Opt}(D) = \max_{y_1, \dots, y_m} \left\{ \sum_i a_i^T y_i : \sum_i y_i = 0, \|y_i\|_2 \leq 1, i \leq m \right\} \quad (D)$$

- (P) clearly is solvable and strictly feasible $\Rightarrow (D)$ is solvable and $\text{Opt}(P) = \text{Opt}(D)$.

- From optimality conditions it is easily seen that

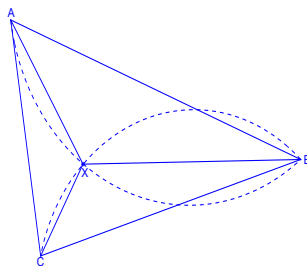
— A point x distinct from a_1, \dots, a_m is an optimal solution to the Steiner sum problem *iff*

$$\sum_i \frac{a_i - x}{\|a_i - x\|_2} = 0.$$

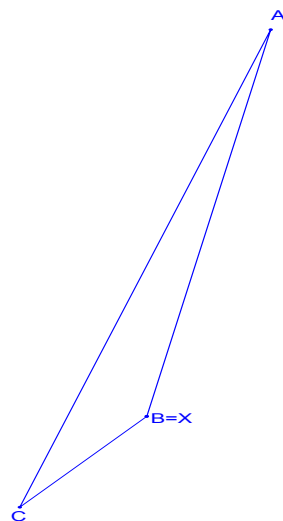
— point $x = a_\ell$ is an optimal solution *iff*

$$\left\| \sum_{i \neq \ell} \frac{a_i - x}{\|a_i - x\|_2} \right\|_2 \leq 1.$$

♠ In the simplest case of 3 points $a_1 = A, a_2 = B, a_3 = C$ in 2D plane, the optimal solution is
— either the point from which all 3 sides of the triangle $\triangle ABC$ are seen at the angle 120°



(such a point exists if angles of the triangle are $< 120^\circ$),
— or the vertex of the triangle corresponding to the angle $\geq 120^\circ$, if such an angle is present:



Proof of Conic Duality Theorem

$$\text{Opt}(P) = \min_x \{c^T x : Ax - b \in \mathbf{K}, Rx = r\} \quad (P)$$

$$\text{Opt}(D) = \max_{y,s} \{b^T y + r^T s : y \in \mathbf{K}_*, A^T y + R^T s = c\} \quad (D)$$

Primal-Dual Symmetry: (D) is a conic problem. To write down its dual, we rewrite it as a minimization problem

$$-\text{Opt}(D) = \min_{y,s} \{-b^T y - r^T s : y \in \mathbf{K}_*, A^T y + R^T s = c\}$$

denoting the Lagrange multipliers for the constraints $y \in \mathbf{K}_*$ and $A^T y + R^T s = c$ by z and $-x$, the dual to dual problem reads

$$\max_{z,x} \left\{ -c^T x : \underbrace{-Ax + z = -b, z \in (\mathbf{K}_*)_* [= \mathbf{K}]}_{\text{says that } Ax - b \in \mathbf{K}}, -Rx = -r \right\}.$$

Eliminating z , we arrive at (P) . □

Weak Duality: By construction of the dual. □

$$\begin{aligned}\text{Opt}(P) &= \min_x \{c^T x : Ax - b \in \mathbf{K}, Rx = r\} & (P) \\ \text{Opt}(D) &= \max_{y,s} \{b^T y + r^T s : y \in \mathbf{K}_*, A^T y + R^T s = c\} & (D)\end{aligned}$$

Strong Duality [under strict, rather than essentially strict, feasibility] We should prove that if one of the problems (P) , (D) is strictly feasible and bounded, then the other problem is solvable with $\text{Opt}(P) = \text{Opt}(D)$, or, which is the same by Weak Duality, with $\text{Opt}(D) \geq \text{Opt}(P)$. By Primal-Dual Symmetry, we lose nothing when assuming that (P) is strictly feasible and bounded.

Step 0: Let us reduce the situation to the one when a strictly feasible solution to (P) is the origin. Specifically, denoting by \bar{x} a strictly feasible solution to (P) and passing in P from variable x to $z = x - \bar{x}$, we arrive at the problem

$$[\text{Opt}(P) - c^T \bar{x} =] \text{Opt}(P') = \min_z \{c^T z : Az - [b - A\bar{x}] \in \mathbf{K}, Rz = 0\} \quad (P')$$

with strictly feasible solution 0 and with the dual problem

$$\text{Opt}(D') = \max_{y,s} \{[b - A\bar{x}]^T y : y \in \mathbf{K}_*, A^T y + R^T s = c\} \quad (D')$$

Note that the feasible sets of (D) and (D') are the same, and on this feasible set, due to $R\bar{x} = r$, we have

$$[b - A\bar{x}]^T y = b^T y + r^T s - \bar{x}^T [A^T y + R^T s] = b^T y + r^T s - c^T \bar{x},$$

implying that (D) and (D') simultaneously are solvable/unsolvable, and their optimal values, same as those of (P) and (P') , differ by $c^T \bar{x}$, so that $\text{Opt}(P) = \text{Opt}(D)$ is equivalent to $\text{Opt}(P') = \text{Opt}(D')$.

Thus, it suffices to prove Strong Duality in the case when $\bar{x} = 0$.

$$\begin{aligned}\text{Opt}(P) &= \min_x \{c^T x : Ax - b \in \mathbf{K}, Rx = 0\} & (P) \\ \text{Opt}(D) &= \max_{y,s} \{b^T y : y \in \mathbf{K}_*, A^T y + R^T s = c\} & (D)\end{aligned}$$

$x = 0$ is strictly feasible solution to (P) , that is

$$-b \in \text{int } \mathbf{K}.$$

Step 1. Let $L = \{x : Rx = 0\}$. It may happen that c is orthogonal to L ("trivial case"). In this case the primal objective vanishes on the primal feasible set, that is, $\text{Opt}(P) = 0$, and $c = A^T s_*$ for some s_* , implying that $[y = 0; s_*]$ is a feasible solution to (D) with zero value of the dual objective. Thus, $\text{Opt}(D) \geq 0 = \text{Opt}(P)$, implying that $\text{Opt}(D) = \text{Opt}(P)$ and the solution $[0; s_*]$ is optimal for (D) , so that Strong Duality holds true in the trivial case.

Step 2. Now let the projection \bar{c} of c on L be nonzero, implying that the set

$$L_- = \{x \in L : \bar{c}^T x < \text{Opt}(P)\} = \{x \in L : c^T x < \text{Opt}(P)\}$$

is nonempty. Note that the convex set $M = \{Ax - b : x \in L_-\}$ is nonempty and does *not* intersect \mathbf{K} . Consequently, M and \mathbf{K} can be separated:

$$\exists f \neq 0 : \inf_{z \in \mathbf{K}} f^T z \geq \sup_{z \in M} f^T z.$$

$$\bar{c}^T x \text{ is nonconstant on } L = \{x : Rx = 0\} \quad (a)$$

$$f \neq 0 \quad (b)$$

$$\inf_{z \in \mathbf{K}} f^T z \geq \sup_x \{f^T [Ax - b] : Rx = 0, \bar{c}^T x < \text{Opt}(P)\} \quad (c)$$

• \mathbf{K} is a cone and inf in (c) is finite \Rightarrow this inf is zero and $f \in \mathbf{K}_*$

\Rightarrow sup in (b) is ≤ 0 , so that (b) reads

$$0 \geq \sup_x \{[A^T f]^T x : Rx = 0, \bar{c}^T x < \text{Opt}(P)\} - f^T b. \quad (d)$$

The maximization domain here is cut off linear space $L = \{x : Rx = 0\}$ by strict linear inequality $\bar{c}^T x < \text{Opt}(P)$ with nonconstant on L left hand side

\Rightarrow (d) implies that the orthogonal projection of $A^T f$ onto L is $\alpha \bar{c}$ with some $\alpha \geq 0$

\Rightarrow (d) reads

$$0 \geq \sup_x \{\alpha \bar{c}^T x : Rx = 0, \bar{c}^T x < \text{Opt}\} - f^T b = \alpha \text{Opt}(P) - f^T b. \quad (e)$$

Now, we have seen that $f \in \mathbf{K}_*$ and $f \neq 0$ by (b), while $-b \in \text{int } \mathbf{K} \Rightarrow f^T b > 0$, implying by (e) that $\alpha > 0$.

Setting $y = \alpha^{-1} f$, we get $y \in \mathbf{K}_*$, and (e) reads $y^T b \geq \text{Opt}(P)$. Besides this, the orthogonal projection of $A^T y$ onto L is exactly the orthogonal projection \bar{c} of e onto L

$\Rightarrow A^T y - \bar{c}$ is orthogonal to $L = \{x : Rx = 0\} \Rightarrow A^T y + R^T s = \bar{c}$ for properly selected s

$\Rightarrow [y; s]$ is dual feasible with the value of dual objective $\text{Opt}(D) = \text{Opt}(P)$.

- It remains to prove the if one of the problems (P) , (D) is strictly feasible, then

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

Indeed, by Primal-Dual Symmetry we lose nothing when assuming that (P) is strictly feasible. The case when (P) is also bounded has been considered; when (P) is unbounded, (D) is infeasible by Weak Duality; thus, in this case $\text{Opt}(P) = \text{Opt}(D) = -\infty$.

□

Consequences of Conic Duality Theorem

Question: When a linear vector inequality

$$Ax \geq_{\mathbf{K}} b \quad (I)$$

has no solutions?

Note: For $\lambda \in \mathbf{K}_*$, the scalar inequality $[A^T \lambda]^T x \geq b^T \lambda$ is a consequence of (I).

\Rightarrow **Sufficient condition for infeasibility:** *If by “admissible aggregation” of (I) one can obtain a contradictory scalar inequality:*

$$\exists \lambda \geq_{\mathbf{K}_*} 0 : \quad A^T \lambda = 0, \quad \lambda^T b > 0. \quad (II)$$

then (I) has no solutions.

$$Ax \geq_{\mathbf{K}} b \quad (\text{I})$$

$$\lambda \geq_{\mathbf{K}_*} 0, A^T \lambda = 0, \lambda^T b > 0 \quad (\text{II})$$

Conic Theorem on Alternative:

A. If (II) has a solution, then (I) has no solutions.

B. If (II) has no solutions, then (I) is "almost solvable," meaning that for every $\epsilon > 0$, you may perturb b by no more than ϵ to get a solvable system (I):

$$\forall \epsilon > 0 \exists (b', x) : \|b - b'\| \leq \epsilon \ \& \ Ax \geq_{\mathbf{K}} b'.$$

C. (II) has no solutions *iff* (I) is almost solvable.

$$\begin{aligned} Ax &\geq_{\mathbf{K}} b & (I) \\ \lambda &\geq_{\mathbf{K}_*} 0, A^T \lambda = 0, \lambda^T b > 0 & (II) \end{aligned}$$

Proof of CTA: Let us fix $f \succ_{\mathbf{K}} 0$, and consider the conic program

$$\text{Opt} = \min_{t,x} \{t : Ax \geq_{\mathbf{K}} b - tf\} \quad (P)$$

Since $f \succ_{\mathbf{K}} 0$, all pairs $[x = 0; t]$ with large enough positive t are strictly feasible solutions to (P) (since for large $t > 0$ we have $tf - b = t(f - t^{-1}b) \succ_{\mathbf{K}} 0$).

Claim: (I) is almost solvable iff $\text{Opt} \leq 0$.

One direction: If $\text{Opt} \leq 0$, then for every $\delta > 0$ (P) has a feasible solution with $t \leq \delta$, and, in addition, (P) has a feasible solution with some nonnegative t . Since the feasible set of (P) is convex, for every $\delta > 0$ (P) has a feasible solution x_δ, t_δ with $t_\delta \in [0, 2\delta] \Rightarrow b_\delta := b - t_\delta f$ is such that $Ax_\delta \geq_{\mathbf{K}} b_\delta$. Since $\|b_\delta - b\| = t_\delta \|f\| \leq 2\delta \|f\|$ and δ can be made arbitrarily small, (I) is almost solvable.

Opposite direction: If (I) is almost solvable, then for every $\delta > 0$ there exist b_δ, x_δ such that $Ax_\delta \geq_{\mathbf{K}} b_\delta$ and $\|b - b_\delta\| \leq \delta$. Since $f \succ_{\mathbf{K}} 0$, \mathbf{K} contains a ball of radius $r > 0$ centered at f , or, equivalently,

$$\frac{\|d\|}{r} f \geq_{\mathbf{K}} d \forall d.$$

In particular, $Ax_\delta \geq_{\mathbf{K}} b_\delta \Rightarrow Ax_\delta \geq_{\mathbf{K}} b + [b_\delta - b] \geq_{\mathbf{K}} b - \frac{\|b - b_\delta\|}{r} f \geq_{\mathbf{K}} b - \frac{\delta}{r} f$, whence $\text{Opt} \leq \delta/r$ for all $\delta > 0$, that is, $\text{Opt} \leq 0$.

Claim \Rightarrow CTA: (P) is strictly feasible, so that by Conic Duality Theorem $\text{Opt} \leq 0$ iff the optimal value in the problem

$$\max_{\lambda} \{b^T \lambda : A\lambda = 0, \lambda \in \mathbf{K}_*, f^T \lambda = 1\} \quad (D)$$

dual to (P) is ≤ 0 . The latter is the case iff $b^T \lambda \leq 0$ for every *nonzero* $\lambda \in \mathbf{K}_*$ such that $A\lambda = 0$ (since for such λ it holds $f^T \lambda > 0$, so that after multiplying y by a positive scalar it becomes feasible for (D)), which is exactly the same as to say that (II) has no solutions. \square

$$Ax \geq_{\mathbf{K}} b \quad (\text{I})$$

$$\lambda \geq_{\mathbf{K}_*} 0, A^T \lambda = 0, \lambda^T b > 0 \quad (\text{II})$$

CTA vs. GTA: "Polyhedral analogy" of CTA is General Theorem on Alternative restricted to the situation where the system of (scalar) linear inequalities for which we want to certify unsolvability contains nonstrict inequalities only. In this situation GTA is stronger than item **C** in CTA – in GTA "almost solvable" is simply "solvable."

♠ In the general conic case, "almost solvable" cannot be strengthened to "solvable," as is seen from the following example: the linear vector inequality with one variable

$$Ax - b := [3; 4; 5] \cdot x - [4; -3; 0] = [3x - 4; 4x + 3; 5x] \geq_{\mathbf{L}^3} 0 \quad (I)$$

reads

$$25x^2 \geq \underbrace{(3x - 4)^2 + (4x + 3)^2}_{25x^2 + 25} \ \& \ 5x \geq 0$$

and has no solutions. However, with $b_\epsilon = [4; -3 + \epsilon; 0]$, the inequality $Ax - b_\epsilon \geq_{\mathbf{L}^3} 0$ reads

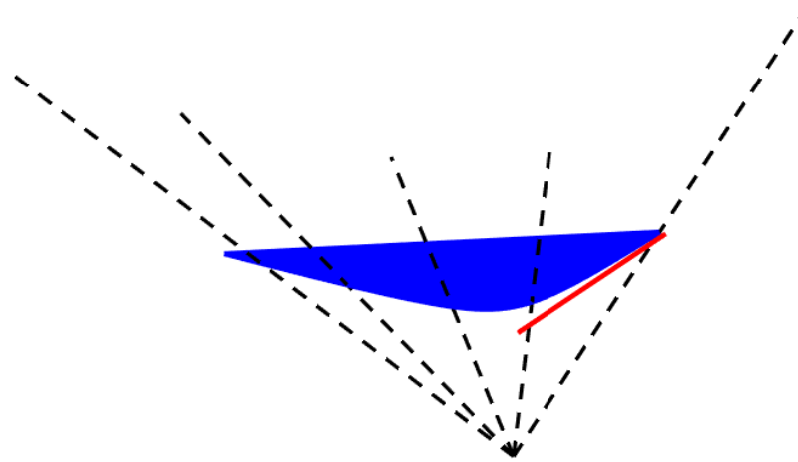
$$25x^2 \geq \underbrace{(3x - 4)^2 + (4x + 3 - \epsilon)^2}_{25x^2 - 8\epsilon x + 16 + (3 - \epsilon)^2} \ \& \ x \geq 0$$

and becomes solvable whenever $\epsilon > 0$.

⇒ (I) is unsolvable and almost solvable, implying that the "alternative" to (I) system

$$\lambda \geq_{\mathbf{L}^3} 0, A^T \lambda = 0, b^T \lambda > 0 \quad (II)$$

has no solutions.



Geometrically: Consider $2D$ plane which intersects the boundary of \mathbf{L}^3 along (branch of) hyperbola. Let $x \in \mathbf{R}$, and let the line $\ell = \{Ax - b : x \in \mathbf{R}\}$ in \mathbf{R}^3 be the asymptote of the hyperbola. Then ℓ does not intersect the cone; however, since the hyperbola is in the cone and ℓ is the asymptote of the hyperbola, appropriate, whatever small, shifts of ℓ make it intersecting the cone.

$$\begin{array}{ll} Ax \geq_{\mathbf{K}} b & \text{(I)} \\ \lambda \geq_{\mathbf{K}^*} 0, A^T \lambda = 0, \lambda^T b > 0 & \text{(II)} \end{array}$$

What is going on: The set of those b 's for which (I) is solvable is the convex set

$$B = \{b = Ax - u, x \in \mathbf{R}^n, u \in \mathbf{K}\},$$

and the set B_* of those b 's for which (I) is almost solvable is the set of b 's which can be approximated to whatever high accuracy by points from B , that is, B_* is the *closure* of B .

By item **C** of CTA, (II) is solvable whenever b is outside of B_* . *When B is closed*, to be outside of B and of B_* is one and the same

\Rightarrow *When the set of those b 's for which (I) is solvable is closed, (II) is solvable whenever (I) is **unsolvable**.*

However, B is not necessarily closed, so that *in general solvability of (II) is only sufficient, but not necessary, condition for insolvability of (I).*

When \mathbf{K} is a polyhedral cone, B is polyhedral (as the arithmetic sum of two polyhedral sets, B admits an immediate polyhedral representation)

$\Rightarrow B$ is automatically closed.

Question: When a scalar inequality

$$c^T x \geq d \quad (S)$$

is a consequence of a vector inequality

$$Ax \geq_{\mathbf{K}} b \quad ? \quad (V)$$

Answer: **A.** *If (S) can be obtained from (V) and the trivial inequality $0 \geq -1$ by "admissible linear aggregation:"*

$$\exists y \geq_{\mathbf{K}_*} 0 : A^T y = c \ \& \ y^T b \geq d, \quad (*)$$

then (S) is a consequence of (V).

B. *If (S) is a consequence of (V) and (V) is essentially strictly feasible, then (S) can be obtained from (V) by admissible linear aggregation.*

Both claims are immediate consequences of the Refined Conic Duality Theorem as applied to the conic problem

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

— (S) is exactly the same as $\text{Opt}(P) \geq d$, and **A**, **B** is what Weak, respectively, Strong, Duality says.

II. CONIC QUADRATIC PROGRAMMING

♣ The m -dimensional Lorentz cone is

$$\mathbf{L}^m = \{x = [x_1; \dots; x_m] \in \mathbf{R}^m : x_m \geq \sqrt{x_1^2 + \dots + x_{m-1}^2}\}$$

By definition, $\mathbf{L}^1 = \mathbf{R}_+$ ("empty sum equals zero").

A *conic quadratic problem* is a conic problem

$$\min_x \{c^T x : Ax - b \geq_{\mathbf{K}} 0\} \quad (\text{CP})$$

for which the cone \mathbf{K} is a direct product of Lorentz cones:

$$\mathbf{K} = \mathbf{L}^{m_1} \times \mathbf{L}^{m_2} \times \dots \times \mathbf{L}^{m_k} = \left\{ y = \begin{bmatrix} y[1] \\ y[2] \\ \vdots \\ y[k] \end{bmatrix} : y[i] \in \mathbf{L}^{m_i}, i = 1, \dots, k \right\}.$$

• Thus, a conic quadratic problem is an optimization problem with linear objective and finitely many "conic quadratic constraints":

$$\min_x \{c^T x : A_i x - b_i \geq_{\mathbf{L}^{m_i}} 0, i = 1, \dots, k\}. \quad (*)$$

$$\min_x \{c^T x : A_i x - b_i \geq_{\mathbf{L}^{m_i}} 0, i = 1, \dots, k\}. \quad (*)$$

Representing

$$[A_i, b_i] = \left[\begin{array}{c|c} D_i & d_i \\ \hline p_i^T & q_i \end{array} \right]$$

(q_i is a real), we may rewrite (*) as

$$\min_x \left\{ c^T x : \underbrace{\|D_i x - d_i\|_2}_{\substack{\updownarrow \\ A_i x - b_i \geq_{\mathbf{L}^{m_i}} 0}} \leq p_i^T x - q_i, i = 1, \dots, k \right\}. \quad (\text{CQ})$$

- A scalar linear inequality $a^T x - b \geq 0$ is the same as the conic quadratic inequality $a^T x - b \in \mathbf{L}^1$, so that adding to (CQ) finitely many scalar linear inequalities, we do not vary the structure of the problem.

Problem dual to Conic Quadratic Problem

$$\min_x \left\{ c^T x : \underbrace{\|D_i x - d_i\|_2 \leq p_i^T x - q_i}_{\substack{\updownarrow \\ [D_i; p_i^T]x - [d_i; q_i] \geq_{\mathbf{L}^{m_i}} 0}}, i = 1, \dots, k \right\}. \quad (\text{CQ})$$

Fact: Lorentz cones are self-dual: $(\mathbf{L}^m)_* = \mathbf{L}^m$.

Indeed,

$$\begin{aligned} (\mathbf{L}^m)^* &= \{[y; s] : [y; s]^T [x; t] \geq 0 \forall (x, t : \|x\|_2 \leq t)\} = \{[y; s] : [y; s]^T [x; 1] \geq 0 \forall (x : \|x\|_2 \leq 1)\} \\ &= \{[y; s] : s \geq \max_{\|x\|_2 \leq 1} [-y^T x]\} = \{[y; s] : s \geq \|y\|_2\}. \end{aligned}$$

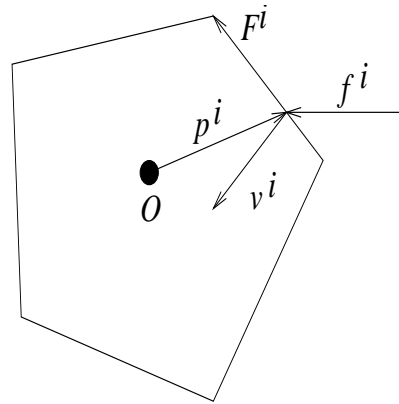
\Rightarrow The problem dual to (CQ) reads

$$\max_{[y_i; s_i], i \leq k} \left\{ \sum_i [y_i^T d_i + s_i q_i] : \|y_i\|_2 \leq s_i, i \leq k, \sum_i [D_i^T y_i + s_i p_i] = c \right\}$$

Examples of CQP's, I

Stable Grasp

♣ When an N -finger robot is capable to hold a rigid body?
This is what happens at i -th contact point:



p^i : the contact point; f^i : the contact force; ν^i : the unit inward normal to body's surface

♣ [Coulomb's Law] *The friction force F^i caused by the contact force f^i is tangent to the surface of the body at p^i :*

$$(F^i)^T \nu^i = 0,$$

and its magnitude is bounded by constant times the normal component of the external force:

$$\|F^i\|_2 \leq \mu (f^i)^T \nu^i$$

$[\mu > 0$: friction coefficient]

♣ Assume that the body is affected by additional external forces (e.g., the gravity ones). From the viewpoint of Mechanics, all these forces can be represented by a single external force F^{ext} (the sum of actual external forces) – and a *torque* T^{ext} (the sum of vector products of the actual external forces and the points where the forces are applied).

The body can be in static equilibrium iff the total force acting at the body and the total torque are zero:

$$\begin{aligned}\sum_{i=1}^N (f^i + F^{\text{ext}}) &= 0 \\ \sum_{i=1}^N p^i \times (f^i + F^{\text{ext}}) + T^{\text{ext}} &= 0\end{aligned}\tag{1}$$

$u \times v$: vector product of $u, v \in \mathbf{R}^3$

♣ Assume $f^i, F^{\text{ext}}, T^{\text{ext}}$ are given. The nature will try to adjust the friction forces F^i to satisfy the equilibrium constraints (1) along with the "friction constraints"

$$[\nu^i]^T F^i = 0, \quad \|F^i\|_2 \leq \mu [\nu^i]^T f^i, \quad i = 1, \dots, N\tag{2}$$

If it is possible, the body will be held by the robot ("stable grasp"), otherwise it will somehow move.

Conclusion: *Possibility of stable grasp is equivalent to solvability of system of conic quadratic constraints*

$$\left. \begin{aligned} \sum_{i=1}^N (f^i + F^i) + F^{\text{ext}} &= 0, \\ \sum_{i=1}^N p^i \times (f^i + F^i) + T^{\text{ext}} &= 0, \\ [\nu^i]^T F^i &= 0, \quad \|F^i\|_2 \leq \mu [\nu^i]^T f^i \end{aligned} \right\}, \quad i = 1, \dots, N$$

with variables F^i , $i = 1, \dots, N$.

⇒ Various grasp-related optimization problems, like

Given

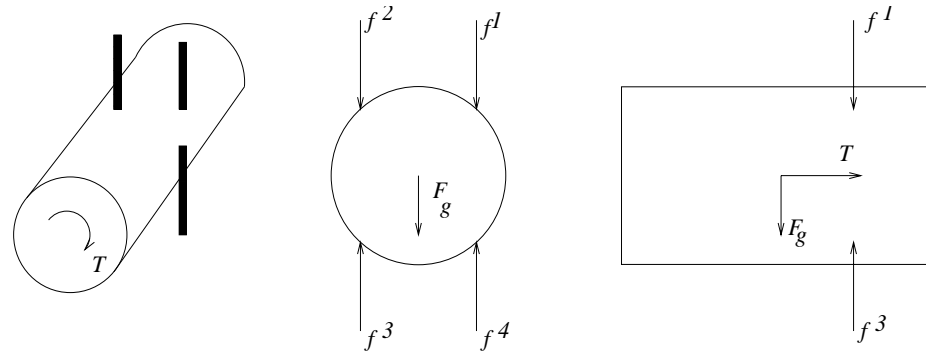
- external force F^{ext} ,
- the direction e^{ext} of external torque,
- the directions u^i of forces exerted by robot's fingers,
- ranges $[0, f_{\text{max}}^i]$ of magnitudes of the forces exerted by robot's fingers:

$$f^i = \lambda_i u^i, \quad \lambda_i \in [0, f_{\text{max}}^i],$$

find the largest possible magnitude T of the external torque still allowing for stable grasp.

can be posed as conic quadratic problems.

Example. A 4-finger robot should hold a cylinder:



Perspective, front and side views

The external torque is directed along the cylinder axis. What is the largest magnitude of the torque still allowing for stable grasp?

This is the conic quadratic problem

$$\max_{T, F^i, \lambda_i} \left\{ T : \begin{array}{l} \sum_i (\lambda_i u^i + F^i) + F^{\text{ext}} = 0 \\ \sum_i p^i \times (\lambda_i u^i + F^i) + T e^{\text{ext}} = 0 \\ \|F^i\|_2 \leq \mu [\nu^i]^T u^i \lambda_i, [\nu^i]^T F^i = 0, i \leq N \\ 0 \leq \lambda_i \leq f_{\max}^i, i \leq N \end{array} \right\}.$$

Examples of CQP's, II

Estimating state of Linear Dynamical System

♣ Consider discrete time Linear Dynamical System with linear output-based feedback

$$\begin{array}{rcl} x_0 & = & z \\ x_{t+1} & = & Ax_t + Bu_t + D\delta_t \\ y_t & = & Cx_t + \epsilon_t \\ \hline u_t & = & Ky_t \end{array}$$

$$\left[\begin{array}{lll} \bullet x_t: \text{states} & \bullet u_t: \text{controls;} & \bullet y_t: \text{observed outputs} \\ \bullet \delta_t: \text{disturbances} & \bullet \epsilon_t: \text{observation errors} & \end{array} \right]$$

on time horizon $0 \leq t \leq N$.

Assuming the dynamics A, B, C, D and feedback K known in advance, and observation errors, and disturbances bounded:

$$\begin{array}{l} \|\epsilon_t\|_2 \leq \epsilon, \|\delta_t\|_2 \leq \delta, \quad 0 \leq t \leq T \\ [\epsilon, \delta: \text{known bounds}] \end{array}$$

we want given y_0, \dots, y_N , to localize x_{N+1} in the smallest possible box $\{x : a \leq x \leq b\}$.

$$\left. \begin{array}{rcl} x_0 & = & z \\ x_{t+1} & = & Ax_t + Bu_t + D\delta_t, \\ y_t & = & Cx_t + \epsilon_t \\ \hline u_t & = & Ky_t \end{array} \right\} (*)$$

$$\left[\begin{array}{l} \bullet x_t: \text{states} \quad \bullet u_t: \text{controls}; \quad \bullet y_t: \text{observed outputs} \\ \bullet \delta_t, \|\delta_t\|_2 \leq \delta: \text{disturbances} \quad \bullet \epsilon_t, \|\epsilon_t\|_2 \leq \epsilon: \text{observation errors} \end{array} \right]$$

Given A, B, C, D, K , observations y_t , $0 \leq t \leq N$, and δ, ϵ , find the smallest box $\{x : a \leq x \leq b\}$ localizing x_{N+1} .

♠ **Solution:** Finding the smallest box localizing x_{N+1} reduces to minimizing several linear forms $e^T x_{N+1}$ over vectors x_{N+1} compatible with observations y_t and bounds on observation errors and disturbances, or, equivalently, to solving several conic quadratic problems of the form

$$\min_{\substack{z, x_t, u_t \\ \epsilon_t, \delta_t}} \left\{ e^T x_{N+1} \text{ s.t. } (*) \ \& \ \|\delta_t\|_2 \leq \delta, 0 \leq t \leq N \ \& \ \|\epsilon_t\|_2 \leq \epsilon, 0 \leq t \leq N \right\}$$

Note: In the problems, y_t 's are *not* optimization variables, they are given data!

What can be expressed via conic quadratic constraints?

♣ Normally, an initial form of an optimization model is

$$\min\{f(x) : x \in X\}, \quad X = \bigcap_{i=1}^m X_i \quad [\text{usually } X_i = \{x : g_i(x) \leq 0\}]$$

We can always make the objective linear:

$$\min_{x \in X} f(x) \Leftrightarrow \min_{y=[x;t] \in Y} t \quad [Y = \{[x;t] : x \in X, t \geq f(x)\}]$$

From now on, assume that the objective is linear, so that the original problem is

$$\min_x \{c^T x : x \in X\} \quad \left[X = \bigcap_{i=1}^m X_i \right] \quad (\text{Ini})$$

♣ **Question:** *When (Ini) can be reformulated as a conic quadratic problem?*

$$\min_x \{c^T x : x \in X\} \quad \left[X = \bigcap_{i=1}^m X_i \right] \quad (\text{Ini})$$

Question: When (Ini) can be reformulated as a conic quadratic problem?

♣ **Answer:** This is the case when X is a *Conic Quadratic representable* (CQr) set.

Definition. Let $X \subset \mathbf{R}^n$. We say that X is CQr, if X admits *Conic Quadratic Representation* (CQR)

$$X = \{x \in \mathbf{R}^n : \exists u \in \mathbf{R}^m : Px + Qu - r \in \mathbf{K}\}, \quad (\text{CQR})$$

where \mathbf{K} is a direct product of Lorentz cones, that is, X can be represented as a projection onto the plane of x -variables of the solution set of a conic constraint in (x, u) -variables, the cone being a direct product of Lorentz cones.

Equivalently: $X \subset \mathbf{R}^n$ is CQr $\Leftrightarrow x \in X$ if and only if x can be extended, by properly selected "certificate" $u \in \mathbf{R}^m$, to a solution to a system of conic quadratic inequalities in variables x, u . Every system with this property is a Conic Quadratic Representation of X .

$$X = \{x \in \mathbf{R}^n : \exists u \in \mathbf{R}^m : Px + Qu - r \in \mathbf{K}\}, \quad (\text{CQR})$$

Immediate observation: Given Conic Quadratic Representation (CQR) of X , the problem $\min_{x \in X} c^T x$ is equivalent to the conic quadratic program

$$\min_{x,u} \left\{ c^T x : Px + Qu - r \in \mathbf{K} \right\},$$

equivalence meaning that x is feasible for the former problem iff x can be extended to a feasible solution to the latter problem. Note that this extension preserves the value of the objective.

Example: Consider the program

$$\min_x \{x : x^2 + 2x^4 \leq 1\} \quad (\text{Ini})$$

A CQR for $X = \{x : x^2 + 2x^4 \leq 1\}$ can be obtained as follows:

$$x^2 + 2x^4 \leq 1 \Leftrightarrow \exists t_1, t_2 : \begin{cases} x^2 & \leq t_1 \\ t_1^2 & \leq t_2 \\ t_1 + 2t_2 & \leq 1 \end{cases}$$

and

$$s^2 \leq r \Leftrightarrow 4s^2 + (r - 1)^2 \leq (r + 1)^2 \Leftrightarrow \begin{bmatrix} 2s \\ r - 1 \\ r + 1 \end{bmatrix} \geq_{\mathbf{L}^3} 0,$$

$$\Rightarrow X = \left\{ x : \exists t_1, t_2 : \underbrace{\begin{bmatrix} 2x \\ t_1 - 1 \\ t_1 + 1 \end{bmatrix} \geq_{\mathbf{L}^3} 0}_{\text{"says" that } x^2 \leq t_1}, \underbrace{\begin{bmatrix} 2t_1 \\ t_2 - 1 \\ t_2 + 1 \end{bmatrix} \geq_{\mathbf{L}^3} 0}_{\text{"says" that } t_1^2 \leq t_2}, t_1 + 2t_2 \leq 1 \right\},$$

and (Ini) is the conic quadratic program

$$\min_{x, t_1, t_2} \left\{ x : \begin{bmatrix} 2x \\ t_1 - 1 \\ t_1 + 1 \end{bmatrix} \geq_{\mathbf{L}^3} 0, \begin{bmatrix} 2t_1 \\ t_2 - 1 \\ t_2 + 1 \end{bmatrix} \geq_{\mathbf{L}^3} 0, t_1 + 2t_2 \leq 1 \right\}.$$

Definition. Let $f : \mathbf{R}^n \rightarrow \mathbf{R} \cup \{+\infty\}$ be a function. We say that f is *Conic Quadratic representable* (CQr), if its epigraph

$$\text{Epi}\{f\} = \{[x; t] \in \mathbf{R}^n \times \mathbf{R} : f(x) \leq t\}$$

is a CQR set. Every CQR of $\text{Epi}\{f\}$ is called a Conic Quadratic Representation (CQR) of f .

Thus, CQR of f is the equivalence

$$t \geq f(x) \Leftrightarrow \exists u : Px + tp + Qu - r \in \mathbf{K},$$

where \mathbf{K} is a direct product of Lorentz cones.

Example: The function $f(x) = x^2 + 2x^4 : \mathbf{R} \rightarrow \mathbf{R}$ is CQR:

$$t \geq x^2 + 2x^4 \Leftrightarrow \exists t_1, t_2 : \begin{bmatrix} 2x \\ t_1 - 1 \\ t_1 + 1 \end{bmatrix} \geq_{\mathbf{L}^3} 0, \begin{bmatrix} 2t_1 \\ t_2 - 1 \\ t_2 + 1 \end{bmatrix} \geq_{\mathbf{L}^3} 0, t_1 + 2t_2 \leq t$$

Immediate Observation: Level sets $\{x : f(x) \leq a\}$ of a CQR function $f : \mathbf{R}^n \rightarrow \mathbf{R}$ are CQR sets with CQR's readily given by a CQR of f :

$$\underbrace{t \geq f(x) \Leftrightarrow \exists u : Px + pt + Qu - r \in \mathbf{K}}_{\Downarrow} \\ \{x : f(x) \leq a\} = \{x : \exists u : Px + pa + Qu - r \in \mathbf{K}\}$$

Immediate Observation: Given CQR's of a CQR function f and a CQR set X , minimization of f over X reduces straightforwardly to a conic quadratic problem:

$$\underbrace{\left[\begin{array}{l} t \geq f(x) \Leftrightarrow \exists u : P_f x + t p_f + Q_f u - r_f \in \mathbf{K}_f \\ x \in X \Leftrightarrow \exists v : P_X x + Q_X v - r_X \in \mathbf{K}_X \end{array} \right]}_{\Downarrow}$$

$$\min_{x \in X} f(x) \Leftrightarrow \min_{t, x, u, v} \left\{ t : \begin{array}{l} P_f x + t p_f + Q_f u - r_f \in \mathbf{K}_f \\ P_X x + Q_X v - r_X \in \mathbf{K}_X \end{array} \right\}$$

- **Polyhedral** representation of a set X :

$$X = \{x \in \mathbf{R}^n : \exists u \in \mathbf{R}^k : a_i^T x + b_i^T u - c_i \geq 0, i \leq m\} \quad (P)$$

- **Conic Quadratic** representation of a set X :

$$X = \{x \in \mathbf{R}^n : \exists u \in \mathbf{R}^k : d_i^T x + e_i^T u + f_i - \|A_i^T x + B_i^T u - c_i\|_2 \geq 0, i \leq m\} \quad (CQ)$$

Note: Representations have similar structure, with scalar linear inequalities in (P) substituted with conic quadratic inequalities in (CQ) .

♠ **But:**

- **By Fourier-Motzkin**, linear inequalities in variables x, u in (P) can be replaced with (perhaps, a much larger set of) linear inequalities *in variables x only*, so that the only (important!) role of polyhedral representability, as opposed to plain polyhedrality, is in allowing for more flexible and compact representations of polyhedral sets.
- **There is no analogy of Fourier-Motzkin** in the case of conic quadratic inequalities – *a set X admitting representation (C) **not always** can be described by finite system of conic quadratic inequalities in x -variables only*, so that allowing for u -variables extends dramatically the scope of Conic Quadratic Programming.

Calculus of CQr functions/sets

Fact: CQr functions/sets admit a fully algorithmic calculus: basic *convexity-preserving* operations with functions/sets as applied to CQr operands, produce CQr results, and CQR's of these results are readily given by CQR's of the operands.

Note: "Convexity-preserving" is crucial here: convexity is built-in property of CQr functions/sets, so that operations which do not preserve convexity (like taking union of two sets) do not preserve, in general, conic quadratic representability.

Calculus of CQR's: Raw Materials. The following functions/sets are CQR with explicit CQR's:

1. Closed half-spaces and affine functions

$$X = \{x : a^T x - b \geq 0\} \text{ — this is CQR}$$

$$\text{Epi}\{a^T x + b\} = \{[x; t] : t - a^T x - b \geq 0\} \text{ — this is CQR}$$

2. Lorentz cone \mathbf{L}^{n+1} and Euclidean norm $f(x) = \|x\|_2 : \mathbf{R}^n \rightarrow \mathbf{R}$:

$$\mathbf{L}^{n+1} = \{y \in \mathbf{R}^{n+1} : y \in \mathbf{L}^{n+1}\} \quad [\text{tautology!}]$$

$$\text{Epi}\{f\} := \{[x; t] : t \geq \|x\|_2\} = \{[x; t] \in \mathbf{L}^{n+1}\}$$

3. Squared Euclidean norm $f(x) = x^T x : \mathbf{R}^n \rightarrow \mathbf{R}$:

$$t \geq x^T x \Leftrightarrow (t+1)^2 \geq (t-1)^2 + 4x^T x \Leftrightarrow [2x; t-1; t+1] \in \mathbf{L}^{n+2}$$

[Note useful identity: $4rs = (r+s)^2 - (r-s)^2 \forall r, s \in \mathbf{R}$]

4. Fractional-quadratic function $f(x, s) = \begin{cases} \frac{x^T x}{s}, & s > 0 \\ 0, & x = 0, s = 0 \\ +\infty, & \text{all remaining cases} \end{cases}$

$[x \in \mathbf{R}^n, s \in \mathbf{R}]$: $f(x, s) \leq t \Leftrightarrow \{x^T x \leq ts \ \& \ s \geq 0, t \geq 0\}$

$$\Rightarrow \quad \text{Epi}\{f\} = \{[x; s; t] : [2x; t - s; t + s] \in \mathbf{L}^{n+2}\}$$

5. Branch of hyperbola $\{(t, s) \in \mathbf{R}^2 : ts \geq 1, t, s \geq 0\}$:

$$\{(t, s) : ts \geq 1, t, s \geq 0\} = \{(t, s) : [2; t - s; t + s] \in \mathbf{L}^3\}$$

6. Rotated Lorentz cone $X = \{[x; t; s] : x^T x \leq ts, t, s \geq 0\} \subset \mathbf{R}^n \times \mathbf{R} \times \mathbf{R}$:

$$\{[x; t; s] : x^T x \leq ts, t, s \geq 0\} = \{[x; t; s] : [2x; t - s; t + s] \in \mathbf{L}^{n+2}\}$$

(cf. item 4: Rotated Lorentz cone is the epigraph of fractional-quadratic function)

Operations preserving CQ-representability of sets

S.A. Taking finite intersections: Intersection of CQr sets X_i , $i \leq N$, is CQr:

$$\underbrace{X_i = \{x \in \mathbf{R}^n : \exists u^i : P_i x + Q_i u^i - r_i \in \mathbf{K}_i\}, i \leq N}_{\Downarrow}$$

$$\bigcap_{i \leq N} X_i = \{x : \exists u = [u^1; \dots; u^N] : P_i x + Q_i u^i - r_i \in \mathbf{K}_i, i \leq N\}$$

In particular, a *polyhedral set* $\{x : Ax - b \geq 0\}$ is CQr (as the intersection of closed half-spaces, which are CQr), and *intersecting a CQr set with the solution set of a finite system of nonstrict linear inequalities preserves CQ-representability*.

S.B. Taking direct products. Direct product of CQr sets $X_i \subset \mathbf{R}^{n_i}$, $i \leq N$, is CQr:

$$\underbrace{X_i = \{x^i \in \mathbf{R}^{n_i} : \exists u^i : P_i x^i + Q_i u^i - r_i \in \mathbf{K}_i\}, i \leq N}_{\Downarrow}$$

$$X_1 \times \dots \times X_N := \{[x^1; \dots; x^N] : x^i \in X_i\} = \{[x^1; \dots; x^N] : \exists u = [u^1; \dots; u^N] : P_i x^i + Q_i u^i - r_i \in \mathbf{K}_i, i \leq N\}$$

S.C. Taking affine images: If $X \subset \mathbf{R}^n$ is CQr and $x \mapsto Ax + b : \mathbf{R}^n \rightarrow \mathbf{R}^k$ is an affine mapping, then the set $AX + b := \{y = Ax + b : x \in X\}$ is CQr:

$$\begin{aligned}
 X &= \underbrace{\{x : \exists u : Px + Qu - r \in \mathbf{K}\}} \\
 &\quad \downarrow \\
 AX + b &= \{y : \exists [x; u] : \underbrace{y = Ax + b}_{\substack{\updownarrow \\ y - [Ax + b] \in \mathbf{R}_+^k, \\ [Ax + b] - y \in \mathbf{R}_+^k}}, Px + Qu - r \in \mathbf{K}\}
 \end{aligned}$$

and all cones involved are direct products of Lorentz cones.

Corollary: *Let \mathcal{S} be a finite system of conic quadratic inequalities in variables (x, u) . Then the set*

$$X = \{x : \exists u : (x, u) \text{ solves } \mathcal{S}\}$$

is CQr.

Indeed, the solution set Y of (\mathcal{S}) clearly is CQr with CQR given by (\mathcal{S}) , and X is the linear image of Y .

S.D. Taking inverse affine images. If $X \subset \mathbf{R}^n$ is CQr and $y \mapsto \mathcal{A}(y) = Ay + b : \mathbf{R}^k \rightarrow \mathbf{R}^n$ is an affine mapping, then the set $\mathcal{A}^{-1}(X) := \{y : Ay + b \in X\}$ is CQr:

$$\begin{aligned}
 X &= \underbrace{\{x : \exists u : Px + Qu - r \in \mathbf{K}\}} \\
 &\quad \downarrow \\
 \mathcal{A}^{-1}(X) &= \{y : \exists u : P[Ay + b] + Qu - r \in \mathbf{K}\}
 \end{aligned}$$

S.E. Taking arithmetic sums: If sets $X_i \subset \mathbf{R}^n$, $i = 1, \dots, N$, are CQr, so is their arithmetic sum $X = X_1 + \dots + X_N := \{x = x_1 + \dots + x_N : x_i \in X_i, i = 1, \dots, N\}$:

$$\underbrace{X_i = \{x : \exists u^i : P_i x + Q_i u^i - r_i \in \mathbf{K}_i\}, i \leq N}_{\downarrow}$$

$$X_1 + \dots + X_N = \{x : \exists x^i, u^i, i \leq N : P_i x^i + Q_i u^i - r_i \in \mathbf{K}_i, i \leq N, x = \sum_i x^i\}$$

Alternatively: X is the image of the direct product $Y = X_1 \times \dots \times X_N$ under the linear mapping

$$y \equiv (x_1, \dots, x_N) \mapsto x_1 + \dots + x_N,$$

and both operations preserve CQ representability.

♣ Several more advanced convexity-preserving operations "behave well" on CQR sets under mild regularity assumptions:

S.F*. **Passing from a set to its support function and polar.** Let $X \subset \mathbf{R}^n$ be a nonempty convex set. Its *support function* is defined as

$$\phi_X(y) = \sup_x \{y^T x : x \in X\} : \mathbf{R}^n \rightarrow \mathbf{R} \cup \{+\infty\}.$$

The support function of X is the same as the support function of the closure of X , and the function "remembers" this closure: if X, X' are nonempty convex sets, then $\phi_X \equiv \phi_{X'}$ iff $\text{cl}X = \text{cl}X'$.

Fact: If $X \subset \mathbf{R}^n$ is a nonempty convex set given by *essentially strictly feasible* CQR, then $\phi_X(\cdot)$ is CQR:

$$\begin{aligned} X &= \underbrace{\{x : \exists u : Px + Qu - r \in \mathbf{K}\}} \\ &\quad \downarrow \\ t \geq \phi_X(y) &\Leftrightarrow t \geq \sup_{x,u} \{y^T x : Px + Qu \geq_{\mathbf{K}} r\} \\ &\Leftrightarrow t \geq \min_{\lambda} \{-r^T \lambda : P^T \lambda + y = 0, Q^T \lambda = 0, \lambda \in \mathbf{K}_*\} \\ &\Leftrightarrow \{[y; t] : \exists \lambda : P^T \lambda + y = 0, Q^T \lambda = 0, t + r^T \lambda \geq 0, \lambda \in \mathbf{K}_* [= \mathbf{K}]\} \end{aligned}$$

where the second and the third \Leftrightarrow are due to (refined) Strong Duality.

Corollary: *When X is CQr with essentially strictly feasible CQR, the polar of X*

$$\text{Polar}(X) = \{y : y^T x \leq 1 \forall x \in X\}$$

is CQr.

Indeed, $\text{Polar}(X) = \{y : \phi_X(y) \leq 1\}$, and a level set of CQr function is CQr with CQR readily given by a CQR of the function.

Fact: *$\text{Polar}(X)$ always is closed, convex, and contains the origin.*

Fact: *When X is a closed convex set containing the origin, so is $\text{Polar}(X)$, and the polar of the polar is X .*

Fact: *The larger is a set, the smaller is its polar:*

$$X \subset Y \Rightarrow 0 \in \text{Polar}(Y) \subset \text{Polar}(X).$$

S.G*. **Passing from a set to its recessive cone.** Let X be a nonempty closed convex set. Its *recessive cone* is defined as

$$\text{Rec}(X) = \{d : \exists \bar{x} \in X : \bar{x} + td \in X \forall t \geq 0\}.$$

i.e., $\text{Rec}(X)$ is comprised of directions d of all rays (treating a point as a ray with zero direction) contained in X . It is easily seen that

- If X contains a ray, directed by d , then the parallel ray emanating from *whatever* point of X , is contained in X :

$$X = X + \text{Rec}(X)$$

- $\text{Rec}(X)$ is closed convex cone.
- $\text{Rec}(X) = \{0\}$ iff X is bounded.
- For a polyhedral set $X = \{x : Ax \leq b\}$ it holds

$$\text{Rec}(X) = \{x : Ax \leq 0\}.$$

Fact: Let a CQr set $X = \{x : \exists u : Px + Qu - r \in \mathbf{K}\}$ be nonempty. Then

A. The CQr set $R = \{x : \exists u : Px + Qu \in \mathbf{K}\}$ is a convex cone contained in the recessive cone of $\text{cl}X$.

B. Let the intersection of the image space of Q and \mathbf{K} be trivial – the origin: $Qu \in \mathbf{K} \Rightarrow Qu = 0$. Then X is closed and $R = \text{Rec}(X)$.

Proof. A is evident:

$$\bar{x} \in X \ \& \ d \in R \Leftrightarrow \exists u, v : P\bar{x} + Q\bar{u} - r \in \mathbf{K} \ \& \ Pd + Qv \in \mathbf{K} \Rightarrow \\ \forall t \geq 0 : P(\bar{x} + td) + Q(u + tv) - r \in \mathbf{K} \Rightarrow \{\bar{x} + td : t \geq 0\} \subset X \Rightarrow d \in \text{Rec}(\text{cl}X).$$

To prove **B**, we need

Lemma. *Under the premise of **B** there exists $C < \infty$ such that*

$$Qu + z \in \mathbf{K} \Rightarrow \exists u_z : Qu_z + z \in \mathbf{K} \ \& \ \|u_z\|_2 \leq C\|z\|_2.$$

Lemma \Rightarrow B: Let $X \ni x_i \rightarrow \bar{x}, i \rightarrow \infty$. By Lemma, the sequence $u = u_{x_i}$ is bounded; passing to subsequence, we can assume that $u_i \rightarrow u, i \rightarrow \infty$. Since $Px_i + Qu_i - r \in \mathbf{K}$, we get $Px + Qu - r \in \mathbf{K}$, that is, $x \in X$. Thus, X is closed. Next, $d \in \text{Rec}(X) \ \& \ \bar{x} \in X \ \& \ t > 1 \Rightarrow \exists u^t : P(x + td) + Qu^t - r \in \mathbf{K} \Rightarrow P[x + td] + Qu_t - r \in \mathbf{K}$ with $u_t = u_{P[x+td]-r} \Rightarrow Pd + Qt^{-1}u_t + [Px - r]/t \in \mathbf{K}$, and $v_t = t^{-1}u_t$ remain bounded as $t \rightarrow \infty$ by Lemma. Selecting $t_j \rightarrow \infty, j \rightarrow \infty$, such that $v_{t_j} \rightarrow \bar{v}$ as $j \rightarrow \infty$, we have

$$Pd + Qv = \lim_{j \rightarrow \infty} [Pd + Qt^{-1}v_{t_j} + [Px - r]/t_j] \in \mathbf{K},$$

Thus, $d \in R$, and therefore $\text{Rec}(X) \subset R$, which combines with **A** to imply $R = \text{Rec}(X)$. \square

Proof of Lemma. Let $Z = \{z : \exists u : Qu + z \in \mathbf{K}\}$. For $z \in Z$, let u_z be the $\|\cdot\|_2$ -smallest vector u such that $Qu + z \in \mathbf{K}$; clearly, u_z exists, $u_0 = 0$, $u_z \in [\text{Ker}Q]^\perp$, and $u_{tz} = tu_z$ when $t > 0$. It suffices to prove that $\|u_z\|_2 \leq C\|z\|_2$ for some $C < \infty$. Assuming the opposite, there exists a sequence $z_i \in Z$ such that $\|u_{z_i}\|_2 > i\|z_i\|_2 \Rightarrow u_{z_i} \neq 0$. Setting $\zeta_i = z_i/\|u_{z_i}\|_2$, $u_i = u_{\zeta_i} = u_{z_i}/\|u_{z_i}\|_2$, we get $u_i \in [\text{Ker}Q]^\perp$, $\|u_i\|_2 = 1$, $Qu_i + \zeta_i \in \mathbf{K}$ and $\zeta_i \rightarrow 0, i \rightarrow \infty$. For properly selected $i_1 < i_2 < \dots$ we have $u_{i_j} \rightarrow u, j \rightarrow \infty$, implying $\|u\|_2 = 1$, $u \in [\text{Ker}Q]^\perp$ and $Qu \in \mathbf{K}$. Since $0 \neq u \in [\text{Ker}Q]^\perp$, we have also $Qu \neq 0$, which under the premise of **B** is impossible. \square

Note: When our sufficient condition $Qu \geq_{\mathbf{K}} 0 \Rightarrow Qu = 0$ for the validity of the implication

$$X = \{x : \exists u : Px + Qu - r \in \mathbf{K}\} \Rightarrow X \text{ is closed} \quad \& \quad \text{Rec}(X) = R := \{d : \exists v : Pd + Qv \in \mathbf{K}\}$$

is violated, the implication may fail to be true.

However: when the condition is "severely violated:" $\exists u : Qu >_{\mathbf{K}} 0$, the implication holds true by trivial reasons – in this case $X = R$ is the entire space!

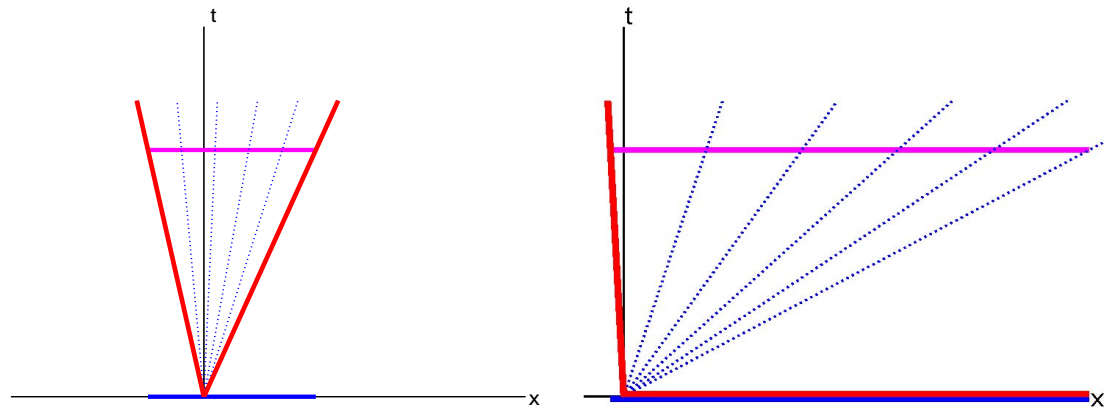
S.H*. **Taking conic hull.** The *conic hull* of a nonempty convex set $X \subset \mathbf{R}^n$ is CQr is defined as

$$X^+ := \{[x; t] : t > 0, x/t \in X\}$$

To get X^+ , we lift $X \subset \mathbf{R}^n$ to get the set $X_+ = \{[x; 1] : x \in X\} \subset \mathbf{R}^{n+1}$; X^+ is the union of all (open) rays in \mathbf{R}^{n+1} emanating from the origin and crossing X_+ , i.e., $X^+ \cup \{0\}$ is the smallest cone containing X_+ .

We can “see” X in X^+ : $X = \{x : [x; 1] \in X^+\}$

Along with (never closed!) conic hull of a convex set, we are interested in the closure of this hull, called *closed conic hull*.



Closed conic hulls of two *closed* convex sets shown in blue: segment (left) and ray (right). Magenta sets are liftings of the blue ones; red angles are the closed conic hulls of blue sets. To get from closed conic hulls the conic hulls *per se*, you should eliminate from the red angles their parts on the x -axis. When the original set is bounded (left picture), all you need to eliminate is the origin; when it is unbounded (right picture), you need to eliminate much more.

The conic hull of a nonempty convex set $X \subset \mathbf{R}^n$ is CQR is defined as

$$X^+ := \{[x; t] : t > 0, x/t \in X\}$$

Fact: *The conic hull X^+ of CQR set X is CQR:*

$$\underbrace{X = \{x : \exists u : Px + Qu - r \in \mathbf{K}\}, X^+ = \{[x; t] : t > 0, x/t \in X\}}_{\Downarrow}$$

$$X^+ = \{[x; t] : \exists u, s : Px + Qu - tr \in \mathbf{K}, \underbrace{t \geq 0, s \geq 0, ts \geq 1}_{\equiv [2; t-s; t+s] \in \mathbf{L}^3}\}$$

Indeed, $\{[x; t] : t > 0, x/t \in X\} = \{[x; t] : \exists u : t > 0, P[x/t] + Qu - r \in \mathbf{K}\} = \{[x; t] : \exists u : t > 0, Px + Qu - tr \in \mathbf{K}\} = \{[x; t] : \exists u, s; Px + Qu - tr \in \mathbf{K}, s \geq 0, t \geq 0, st \geq 1\}.$

$$X^+ = \{[x; t] : t > 0, t^{-1}x \in X\} \text{ [conic hull of } X\text{]}$$

Note: If nonempty CQr set $X = \{x : \exists u : Px + Qu - r \in \mathbf{K}\}$ is closed, then the CQr set

$$\widehat{X}^+ = \{[x; t] : \exists u : Px + Qu - tr \in \mathbf{K}, t \geq 0\}$$

is "in-between" the complete conic hull $\bar{X}^+ = X^+ \cup \{0\}$ of X and the closed conic hull $\text{cl}X^+ = \text{cl}\bar{X}^+$ of X :

$$\bar{X}^+ := X^+ \cup \{0\} \subset \widehat{X}^+ \subset \text{cl}X^+ = \text{cl}\bar{X}^+.$$

If X is closed and bounded, then \bar{X}^+ is closed, so that in this case

$$\bar{X}^+ = \widehat{X}^+ = \text{cl}\bar{X}^+$$

is CQr.

Proof. \widehat{X}^+ clearly contains the origin and we already know that it contains the conic hull $X^+ = \{[x; t] \in \bar{X}^+ : t > 0\}$ of $X \Rightarrow \bar{X}^+ \subset \widehat{X}^+$. On the other hand, let $[x; t] \in \widehat{X}^+$ and $\bar{x} \in X$, so that $t \geq 0$, $Px + Qu - tr \in \mathbf{K}$, and $P\bar{x} + Qv - r \in \mathbf{K}$ for some u, v . Then for every $\epsilon \in (0, 1)$ we have

$$P \underbrace{[x + \epsilon \bar{x}]}_{x_t} + Q[u + \epsilon v] - \underbrace{[t + \epsilon]}_{=: t_\epsilon} r \in \mathbf{K} \Rightarrow [x_\epsilon; t_\epsilon] \in X^+.$$

Since $[x_\epsilon; t_\epsilon] \rightarrow [x; t]$ as $\epsilon \rightarrow +0$, we get $[x; t] \in \text{cl}X^+$. Thus, $\widehat{X}^+ \subset \text{cl}X^+$.

The fact that \bar{X}^+ is closed whenever X is bounded and closed is immediate. Let $\bar{X}^+ \ni [x_i; t_i] \rightarrow [x; t]$, $i \rightarrow \infty$; we should prove that $[x; t] \in \bar{X}^+$. If infinitely many of t_i are zeros, then $[x; t]$ is the origin (since $[x; 0] \in \bar{X}^+$ iff $x = 0$), and the origin does belong to \bar{X}^+ . When only finitely many of t_i are zeros, then the vectors $y_i = x_i/t_i$ are well defined for all large enough i and belong to X , and thus form a bounded sequence. Passing to a subsequence, we can assume that $y_i \rightarrow y$ as $i \rightarrow \infty$, and $y \in X$ since X is closed. We see that $[x_i; t_i] = t_i[y_i; 1]$ with $y_i \rightarrow y \in X$, $i \rightarrow \infty$, implying that $[x; t] = \lim_{i \rightarrow \infty} [x_i; t_i] = \lim_{i \rightarrow \infty} t_i[y_i; 1] = t[y; 1]$. Since $t \geq 0$ and $y \in X$, we see that $[x; t] \in \bar{X}^+$. \square

S.I*. Taking convex hulls of finite unions. Let $X_i \subset \mathbb{R}^n$, $i = 1, \dots, N$, be nonempty **closed** CQr sets: $X_i = \{x : \exists u^i : P_i x + Q_i u^i - r_i \in \mathbf{K}_i\}$, and \widehat{X} be the convex hull of their union:

$$\widehat{X} = \text{Conv}(X_1 \cup \dots \cup X_N).$$

Then the CQr set

$$\widetilde{X} = \left\{ x : \exists y^i, u^i, \lambda_i, i \leq N : \begin{array}{l} \lambda_i \geq 0, \sum_i \lambda_i = 1, x = \sum_i y^i \\ P_i y^i + Q_i u^i - \lambda_i r_i \in \mathbf{K}_i, i \leq N \end{array} \right\}$$

is in-between \widehat{X} and $\text{cl}\widehat{X}$: $\widehat{X} \subset \widetilde{X} \subset \text{cl}\widehat{X}$. In particular, when \widehat{X} is closed (which definitely is the case, e.g., when all X_i are bounded), then $\widehat{X} = \widetilde{X}$ is CQr.

Proof. When $x \in \widehat{X}$, we have $x = \sum_i \lambda_i x^i$ with $\lambda_i \geq 0$, $\sum_i \lambda_i = 1$ and $x^i \in X_i$, that is, $P_i x^i + Q_i v^i - r_i \in \mathbf{K}_i$ for some v^i . Setting $y_i = \lambda_i x^i$, $u^i = \lambda_i v^i$, we get $P_i y^i + Q_i u^i - \lambda_i r_i \in \mathbf{K}_i$ and $x = \sum_i y^i$, whence $x \in \widetilde{X}$. Thus, $\widehat{X} \subset \widetilde{X}$. Now let $x \in \widetilde{X}$ and \bar{y}^i be such that $N\bar{y}^i \in X_i$, so that

$$\exists(y^i, u^i, \bar{u}_i, \lambda_i) : \lambda_i \geq 0, \sum_i \lambda_i = 1, x = \sum_i y^i, P_i y^i + Q_i u^i - \lambda_i r_i \in \mathbf{K}_i, P_i \bar{y}^i + Q_i \bar{u}^i - N^{-1} p_i \in \mathbf{K}_i.$$

For $\epsilon \in (0, 1]$ it holds

$$P_i \underbrace{[(1 - \epsilon)y^i + \epsilon \bar{y}^i]}_{y_\epsilon^i} + Q_i \underbrace{[(1 - \epsilon)u^i + \epsilon \bar{u}^i]}_{u_\epsilon^i} - \underbrace{[(1 - \epsilon)\lambda_i + \epsilon N^{-1}]}_{\lambda_{i,\epsilon} > 0} r_i \in \mathbf{K}_i, \quad i \leq N,$$

whence $z_\epsilon^i := y_\epsilon^i / \lambda_{i,\epsilon} \in X_i$, $i \leq N$, and since $\sum_i \lambda_{i,\epsilon} = 1$ and $\lambda_{i,\epsilon} \geq 0$, we get

$$x_\epsilon := \sum_i y_\epsilon^i = \sum_i \lambda_{i,\epsilon} z_\epsilon^i \in \widehat{X}.$$

When $\epsilon \rightarrow +0$, $x_\epsilon \rightarrow x = \sum_i y^i$, whence $x \in \text{cl} \widehat{X}$. Thus, $\widetilde{X} \subset \text{cl} \widehat{X}$. □

Operations preserving CQ-representability of functions

F.A. Restricting onto CQr set. If $f(x) : \mathbf{R}^n \rightarrow \mathbf{R} \cup \{+\infty\}$ is CQr function and $X \subset \mathbf{R}^n$ is CQr set, then the restriction $f_X(x) = \begin{cases} f(x), & x \in X \\ +\infty, & \text{otherwise} \end{cases}$ is CQr:

$$\underbrace{\left[\begin{array}{l} t \geq f(x) \Leftrightarrow \exists u : P_f x + tp + Q_f u - r_f \in \mathbf{K}_f \\ X = \{x : \exists v : P_X x + Q_X v - r_X \in \mathbf{K}_X\} \end{array} \right]} \Downarrow$$

$$t \geq f_X(x) \Leftrightarrow \exists u, v : P_f x + tp + Q_f u - r_f \in \mathbf{K}_f, P_X x + Q_X v - r_X \in \mathbf{K}_X$$

F.B. Taking finite maxima. If $f_i : \mathbf{R}^n \rightarrow \mathbf{R} \cup \{+\infty\}$, $i = 1, \dots, N$, are CQr, then so is their maximum $f(x) = \max_i f_i(x)$.

Indeed, $\text{Epi}\{f\} = \bigcap_i \text{Epi}\{f_i\}$, and intersection of finitely many CQr sets is CQr.

F.C. Summation with nonnegative weights. If functions $f_i : \mathbf{R}^n \rightarrow \mathbf{R} \cup \{+\infty\}$, $i = 1, \dots, N$, are CQr and $\alpha_i \geq 0$, then the function

$$f(x) = \sum_{i=1}^n \alpha_i f_i(x)$$

is CQr. Indeed, assuming w.l.o.g. that $\alpha_i > 0$, $i \leq N$, we have

$$\underbrace{t \geq f_i(x) \Leftrightarrow \exists u^i : P_i x + t p_i + Q_i u^i - r_i \in \mathbf{K}_i, i \leq N}_{\Downarrow}$$

$$t \geq \sum_i \alpha_i f_i(x) \Leftrightarrow \exists t_i, u_i, i \leq N : P_i x + t_i p_i + Q_i u_i - r_i \in \mathbf{K}_i \forall i, t \geq \sum_i \alpha_i t_i.$$

F.D. Direct summation. If $f_i : \mathbf{R}^{n_i} \rightarrow \mathbf{R} \cup \{+\infty\}$, $i = 1, \dots, N$, are CQr, so is

$$f(x^1, \dots, x^N) = \sum_{i=1}^N f_i(x^i) : \mathbf{R}_{x^1}^{n_1} \times \dots \times \mathbf{R}_{x^N}^{n_N} \rightarrow \mathbf{R} \cup \{+\infty\} :$$

$$\underbrace{t \geq f_i(x^i) \Leftrightarrow \exists u^i : P_i x^i + t p_i + Q_i u^i - r_i \in \mathbf{K}_i, i \leq N}_{\Downarrow}$$

$$t \geq \sum_i f_i(x^i) \Leftrightarrow \exists t_i, u_i, i \leq N : P_i x^i + t_i p_i + Q_i u_i - r_i \in \mathbf{K}_i \forall i, t \geq \sum_i t_i.$$

F.E. Affine substitution of argument. If $f : \mathbf{R}^n \rightarrow \mathbf{R} \cup \{+\infty\}$ is CQr and $y \mapsto Ay + b : \mathbf{R}^k \rightarrow \mathbf{R}^n$ is an affine mapping, then the superposition

$$g(y) = f(Ay + b)$$

is CQr:

$$\underbrace{t \geq f(x) \Leftrightarrow \exists u : Px + tp + Qu - r \in \mathbf{K}}_{\Downarrow} \\ t \geq g(y) \Leftrightarrow \exists u : P[Ay + b] + tp + Qu - r \in \mathbf{K}$$

F.F. Taking superposition. Let $F(y) : \mathbb{R}^m \rightarrow \mathbb{R} \cup \{+\infty\}$ and $f_i(x) : \mathbb{R}^n \rightarrow \mathbb{R} \cup \{+\infty\}$, $i = 1, \dots, m$, be CQr. Assume that $F(y)$ is nondecreasing in every one of y_i . Then the superposition

$$G(x) = \begin{cases} F(f_1(x), \dots, f_m(x)), & f_i(x) < +\infty, i \leq m \\ +\infty, & \text{otherwise} \end{cases}$$

is CQr:

$$\underbrace{\begin{bmatrix} t \geq F(y) \Leftrightarrow \exists u : Py + tp + Qu - r \in \mathbf{K} \\ t \geq f_i(x) \Leftrightarrow \exists u^i : P_i x + tp_i + Q_i u^i - r_i \in \mathbf{K}_i, i \leq N \end{bmatrix}}_{\Downarrow} \\ t \geq G(x) \Leftrightarrow \exists \tau = [\tau_1; \dots; \tau_m], v^i : \underbrace{P\tau + tp + Qu - r \in \mathbf{K}}_{\text{says that } t \geq F(\tau)}, \underbrace{P_i x + \tau_i p_i + Q_i u^i - r_i \in \mathbf{K}_i, i \leq N}_{\text{say that } \tau_i \geq f_i(x)}$$

Refinement I. Let f_1, \dots, f_k be affine. Then the conclusion of Superposition Theorem remains true when F is nondecreasing in arguments y_{k+1}, \dots, y_m , CQr of G being

$$t \geq G(x) \Leftrightarrow \exists u, \tau = [\tau_1; \dots; \tau_m], v^i : P\tau + Qu - r \in \mathbf{K}, P_i x + \tau_i p_i + Q_i u^i - r_i \in \mathbf{K}_i, i \leq N, \tau_i = f_i(x), i \leq k$$

Illustration: The functions $F(y) = y^2$ and $f(x) = x^2 - 1$ are CQr; however, $F(f(x)) = (x^2 - 1)^2$ is nonconvex and thus is not CQr. In contrast, square of *affine* function is CQr.

Refinement II: Let $F(y) : \mathbf{R}^m \rightarrow \mathbf{R} \cup \{+\infty\}$ and $f_i(x) : \mathbf{R}^n \rightarrow \mathbf{R} \cup \{+\infty\}$, $i = 1, \dots, m$, be CQr, with f_1, \dots, f_k affine. Assume that for some CQr set $Y \subset \mathbf{R}^m$ F is nondecreasing in $y_{k+1}, y_{k+2}, \dots, y_m$ on Y :

$$\forall (y' \in Y, y \in Y, y' \geq y \ \& \ y_i = y'_i, i \leq k) : F(y') \geq F(y)$$

and let for every x such that $f_i(x) < +\infty, i \leq m$, it holds $f(x) := [f_1(x); \dots; f_m(x)] \in Y$. Then the superposition

$$G(x) = \begin{cases} F(f_1(x), \dots, f_m(x)), & f_i(x) < +\infty, i \leq m \\ +\infty, & \text{otherwise} \end{cases}$$

is CQr:

$$\left[\begin{array}{l} t \geq F(y) \Leftrightarrow \exists u : Py + tp + Qu - r \in \mathbf{K} \\ \quad \quad \quad f_i \text{ affine}, 1 \leq i \leq k \\ t \geq f_i(x) \Leftrightarrow \exists u^i : P_i x + tp_i + Q_i u^i - r_i \in \mathbf{K}_i, k < i \leq m \\ Y = \{y : \exists w : Ry + Sw - s \in \mathbf{K}_Y\}, f(x) \in \mathbf{R}^m \Rightarrow f(x) \in Y \end{array} \right]$$

\Downarrow

$$t \geq G(x) \Leftrightarrow \exists u, \tau = [\tau_1; \dots; \tau_m], v^i, w : \begin{cases} P\tau + tp + Qu - r \in \mathbf{K} \ [\Rightarrow F(\tau) \leq t] \\ \tau_i = f_i(x), 1 \leq i \leq k \\ P_i x + \tau_i p_i + Q_i v^i - r_i \in \mathbf{K}_i, k < i \leq m \ [\Rightarrow \tau_i \geq f_i(x), k < i \leq m] \\ R\tau + Sw - s \in \mathbf{K}_Y \ [\Rightarrow \tau \in Y] \end{cases}$$

Illustration: The functions $F(y) = y^2$ and $f(x) = x^2$ are CQr, and F is nondecreasing on the CQr set $Y = \mathbf{R}_+$ where f takes its values $\Rightarrow F(f(x)) = x^4$ is CQr.

F.G. Projective transformation. Let $f(x) : \mathbf{R}^n \rightarrow \mathbf{R} \cup \{+\infty\}$ be a convex function. It is known that then the *projective transformation*

$$F(x, \alpha) = \begin{cases} \alpha f(x/\alpha), & \alpha > 0 \\ +\infty, & \text{otherwise} \end{cases}$$

is convex as well. *When f is CQR, so is its projective transformation:*

$$\underbrace{t \geq f(x) \Leftrightarrow \exists u : Px + tp + Qu - r \in \mathbf{K}}_{\Downarrow}$$

$$t \geq F(x, \alpha) \Leftrightarrow \exists u, s : \begin{cases} Px + tp + Qu - \alpha r \in \mathbf{K} & [\text{when } \alpha > 0, \text{ says that } t/\alpha \geq f(x/\alpha)] \\ [2; \alpha - s; \alpha + s] \in \mathbf{L}^3 & [\text{enforces } \alpha > 0] \end{cases}$$

♣ Several more advanced convexity-preserving operations "behave well" on CQr functions under mild regularity assumptions:

F.H*. Partial minimization. Let $f(x, y) : \mathbf{R}^{n_x} \times \mathbf{R}^{n_y} \rightarrow \mathbf{R} \cup \{+\infty\}$ be CQr, $X \in \mathbf{R}^{n_x}$ be a CQr set, and let parametric problem

$$\min_y f(x, y)$$

with $x \in X$ be solvable whenever it is feasible. Then the function

$$g(x) = \begin{cases} \min_y f(x, y), & x \in X \\ +\infty, & x \notin X \end{cases}$$

is CQr:

$$\left[\begin{array}{l} t \geq f(x, y) \Leftrightarrow \exists u : P_f[x; y] + tp_f + Q_f u - r_f \in \mathbf{K}_f \\ X = \{x : \exists v : P_X x + Q_X v - r_X \in \mathbf{K}_X\} \text{ \& \; } \min_y f(x, y) \text{ is achieved whenever it is } < +\infty \end{array} \right]$$

$$\downarrow$$

$$t \geq g(x) \Leftrightarrow \exists y, u, v : \underbrace{P_f[x; y] + tp_f + Q_f u - r_f \in \mathbf{K}_f}_{\text{says that } t \geq f(x, y)}, \underbrace{P_X x + Q_X v - r_X \in \mathbf{K}_X}_{\text{says that } x \in X}$$

F.I*. Taking Legendre transformation: If $f : \mathbf{R}^n \rightarrow \mathbf{R} \cup \{+\infty\}$ is CQr with an *essentially strictly feasible* CQR

$$\{(t, x) : t \geq f(x)\} = \{(t, x) : \exists u : Px + tp + Qu - r \in \mathbf{K}\}$$

then the Legendre transformation of f

$$f_*(\xi) = \sup_x [\xi^T x - f(x)]$$

is CQr:

$$\begin{aligned} \{[\xi, \tau] : \tau \geq f_*(\xi)\} &= \{[\xi; \tau] : \tau \geq \xi^T x - t \forall (t, x) \in \text{Epi}\{f\}\} \\ &= \{[\xi; \tau] : \tau \geq \sup_{(t, x) \in \text{Epi}\{f\}} [\xi^T x - t]\} \\ &= \left\{ [\xi; \tau] : \tau \geq \sup_{x, t, u} \{\xi^T x - t : Px + tp + Qu - r \in \mathbf{K}\} \right\} \\ &= \left\{ [\xi; \tau] : \tau \geq \min_y \{-r^T y : P^T y + \xi = 0, Q^T y = 0, p^T y = 1, y \in \mathbf{K}_* [= \mathbf{K}]\} \right\} \quad (a) \\ &= \{[\xi; \tau] : \exists y : p^T y = 1, P^T y + \xi = 0, Q^T y = 0, \tau + r^T y \geq 0, y \in \mathbf{K}_*\} \quad (b) \end{aligned}$$

where (a), (b) are due to Strong Duality.

More examples of CQr functions/sets

7. Convex quadratic form $f(x) = x^T Q^T Q x + q^T x + r$ is CQr, since it can be obtained from the squared Euclidean norm and affine function (both are CQr) by affine substitution of argument and addition. Here is an explicit CQR for f :

$$\text{Epi}\{f\} = \{(x, t) : \begin{bmatrix} 2Qx \\ t - q^T x - r - 1 \\ t - q^T x - r + 1 \end{bmatrix} \geq_{\mathbf{L}^{m+2}} 0\} \quad [Q : m \times n]$$

8. Power functions.

Observation: Let m be nonnegative integer, and let $M = 2^m$. The set

$$\begin{aligned}\mathcal{X}_m &= \left\{ (t, x_1, x_2, \dots, x_M) \in \mathbf{R}_+^{M+1} : t^M \leq x_1 \dots x_M \right\} \\ &= \left\{ (t, x_1, x_2, \dots, x_{2^m}) \geq 0 : t \leq \sqrt[2^m]{x_1 x_2 \dots x_{2^m}} \right\}\end{aligned}$$

is CQR with explicit CQR. In particular, so are the sets of *nonnegative* t and x 's given by

- $t \leq \sqrt{x_1 x_2}$
- $t \leq [x_1 x_x x_3 x_4]^{1/4}$
- $t \leq [x_1 x_2 \dots x_{1024}]^{1/1024}$

Indeed,

[illegible]

and the set $\{(u, v, w) \geq 0 : u \leq \sqrt{vw}\}$ is CQR - it is the intersection of the rotated ice-cream cone $\{(u, v, w) : v \geq 0, w \geq 0, u^2 \leq vw\}$ and the half-space $\{(u, v, w) : u \geq 0\}$, and both these sets are CQr.

♠ Observation implies CQr's of convex power functions.

8.1. *Convex increasing power function $f(x) = (x)_+^\pi$, $x_+ = \max[x, 0]$, with rational degree $\pi = \frac{p}{q} \geq 1$ is CQr.*

Indeed, let $\mu \in \mathbb{N}$ be such that $M \equiv 2^\mu \geq p + q$. We have

$$\begin{aligned}
 & Y \equiv \{(\tau, x_1, \dots, x_M) \geq 0 : \tau^M \leq x_1 \dots x_M\} \text{ is CQr} \\
 \Rightarrow & \{(t, \xi) \geq 0 : \xi^M \leq t^q \xi^{M-p} \mathbf{1}^{p-q}\} = \{(t, \xi) : \mathcal{A}(t, \xi) \in Y\} \text{ is CQr} \\
 & \text{[affine substitution of variables } \mathcal{A}((t, \xi) = (\xi, \underbrace{t, \dots, t}_q, \underbrace{\xi, \dots, \xi}_{M-p}, \underbrace{1, \dots, 1}_{p-q})] \\
 & \Rightarrow \{(t, \xi) \geq 0 : t \geq \xi^{\frac{p}{q}}\} \text{ is CQr} \\
 \Rightarrow & \text{Epi}\{f\} = \{(t, x) : \exists \xi : (t, \xi) \geq 0, t \geq \xi^{\frac{p}{q}}, \xi \geq x\} \text{ is CQr}
 \end{aligned}$$

♠ **Illustration. Why x_+^3 is CQr:**

- First, the set $\{t \geq y^3 \ \& \ y \geq 0\}$ is CQr:

$$\{t \geq y^3 \ \& \ y \geq 0\} = \{(y, t) : y \geq 0, t \geq 0, y^4 \leq t \cdot y \cdot 1 \cdot 1\}$$

and the right hand side set is obtained from CQr set

$$\{[\tau; x_1; x_2; x_3; x_4] \geq 0 : \tau^4 \leq x_1 x_2 x_3 x_4\}$$

by taking inverse affine image.

- Second, we have

$$\{t \geq x_+^3\} \Leftrightarrow \{[x; t] : \exists y : y \geq x, y \geq 0, t \geq y^3\}$$

and the right hand side set is the inverse affine image, under the mapping $(x, t, y) \mapsto (x, t)$, of the CQr set $\{(x, t, y) : y \geq x, y \geq 0, t \geq y^3\}$.

- Here is a complete CQR of $f(x) = x_+^3$:

$$t \geq x_+^3 \Leftrightarrow \exists y, u : y \geq x, y \geq 0, u \geq 0, \underbrace{[2u; t - y; t + y] \in \mathbf{L}^3}_{\Leftrightarrow \{u^2 \leq ty \ \& \ t \geq 0\} \text{ when } y \geq 0}, \underbrace{[2y; u - 1; u + 1] \in \mathbf{L}^3}_{\Leftrightarrow y^2 \leq u}$$

♠ Illustration. Why $x_+^{7/3}$ is CQr:

$$\begin{aligned}
t \geq (x)_+^{7/3} &\Leftrightarrow \exists(z : z \geq 0, z \geq x) : t \geq z^{7/3} \\
&\Leftrightarrow \exists(z : z \geq 0, z \geq x) : t \geq 0, z^{16} \leq t^3 z^9 1^4 = t \cdot t \cdot t \cdot z \cdot z \cdot z \cdot z \cdot z \cdot z \cdot z \cdot z \cdot z \cdot z \cdot 1 \cdot 1 \cdot 1 \cdot 1 \\
&\Leftrightarrow \exists(z, u_i : z \geq 0, z \geq x, u_i \geq 0) : \begin{cases} u_1^2 \leq t^2, u_2^2 \leq tz, u_3^2 \leq z^2, u_4^2 \leq z^2, u_5^2 \leq z^2, u_6^2 \leq z^2, u_7^2 \leq 1, u_8^2 \leq 1 \\ z^8 \leq u_1 u_2 u_3 u_4 u_5 u_6 u_7 u_8, t \geq 0 \end{cases} \\
&\Leftrightarrow \exists(z, u_i, v_i : z \geq 0, z \geq x, u_i \geq 0, v_i \geq 0) : \begin{cases} u_1^2 \leq t^2, u_2^2 \leq tx, u_3^2 \leq x^2, u_4^2 \leq x^2, u_5^2 \leq x^2, u_6^2 \leq x^2, u_7^2 \leq 1, u_8^2 \leq 1 \\ v_1^2 \leq u_1 u_2, v_2^2 \leq u_3 u_4, v_3^2 \leq u_5 u_6, v_4^2 \leq u_7 u_8 \\ z^4 \leq v_1 v_2 v_3 v_4, t \geq 0 \end{cases} \\
&\Leftrightarrow \exists \left(z, u_i, v_i, w_i : \begin{array}{l} z \geq 0, z \geq x, \\ u_i \geq 0, v_i \geq 0, \\ w_i \geq 0 \end{array} \right) : t \geq 0, \begin{cases} u_1^2 \leq t^2, u_2^2 \leq tx, u_3^2 \leq x^2, u_4^2 \leq x^2, u_5^2 \leq x^2, u_6^2 \leq x^2, u_7^2 \leq 1, u_8^2 \leq 1 \\ v_1^2 \leq u_1 u_2, v_2^2 \leq u_3 u_4, v_3^2 \leq u_5 u_6, v_4^2 \leq u_7 u_8 \\ w_1^2 \leq v_1 v_2, w_2^2 \leq v_3 v_4 \\ z^2 \leq w_1 w_2, t \geq 0 \end{cases}
\end{aligned}$$

8.2. *Convex piecewise power function* $f(x) = \begin{cases} x^{\pi+}, & x \geq 0 \\ |x|^{\pi-}, & x \leq 0 \end{cases}$ with rational degrees $\pi_{\pm} \geq 1$ is CQr.

Indeed, the function is obtained from CQR function $(x)_+^{\pi_+}$ by summation and affine substitution of variables:

$$f(x) = (x_+)^{\pi_+} + (-x)_+^{\pi_-}$$

8.3. *Decreasing power function* $f(x) = \begin{cases} x^{-\pi} & , x > 0 \\ +\infty & , x \leq 0 \end{cases}$ of rational degree $-\pi < 0$ is CQr.

Indeed, when $\pi = p/q$ with positive integers p, q and $\mu \in \mathbf{N}$ is such that $M = 2^\mu \geq p + q$ we have

$$\text{Epi}\{f\} = \{(t, x) : t \geq 0, x \geq 0, x^p t^q \geq 1\} = \{(t, x) : 1 \leq x^p t^q 1^{M-p-q}\},$$

which is the inverse affine image of the CQr set

$$\{(\tau, x_1, \dots, x_m) \geq 0 : \tau^M \leq x_1 \cdot \dots \cdot x_M\}$$

under the affine mapping $(t, x) \mapsto (1, \underbrace{x, \dots, x}_p, \underbrace{t, \dots, t}_q, \underbrace{1, \dots, 1}_{M-p-q})$

8.4. The hypograph of a concave power monomial. When $\pi_i > 0$ are rational and $\sum_i \pi_i \leq 1$, the convex monomial

$$f(x) = \begin{cases} -x_1^{\pi_1} \dots x_m^{\pi_m}, & x \geq 0 \\ +\infty, & \text{otherwise} \end{cases}$$

is CQr.

Indeed, let $\pi_i = p_i/q$ with positive integers p_i and positive integer q , and let $\mu \in \mathbb{N}$ be such that $M = 2^\mu \geq q$. Then

$$\begin{aligned} \text{Epi}\{f\} &= \{(x, t) : \exists \tau : \tau \geq 0, t + \tau \geq 0, (\tau, x) \in \mathcal{M}\}, \\ \mathcal{M} &= \{(\tau, x_1, \dots, x_m) \geq 0 : \tau^q \leq x_1^{p_1} x_2^{p_2} \dots x_m^{p_m}\} \\ &= \{(\tau, x_1, \dots, x_m) \geq 0 : \tau^M \leq x_1^{p_1} x_2^{p_2} \dots x_m^{p_m} \tau^{M-q} \mathbf{1}^{q-\sum_i p_i}\} \end{aligned}$$

that is, $\text{Epi}\{f\}$ is the intersection of a polyhedral set and the inverse image of the CQr set

$$\{(s, y_1, \dots, y_M) \geq 0 : s^M \leq y_1 \dots y_M\}$$

under the affine mapping

$$(\tau, x_1, \dots, x_m) \mapsto (\tau, \underbrace{x_1, \dots, x_1}_{p_1}, \dots, \underbrace{x_m, \dots, x_m}_{p_m}, \underbrace{\tau, \dots, \tau}_{M-q}, \underbrace{1, \dots, 1}_{q-\sum_i p_i}).$$

8.5. The epigraph of a convex power monomial. When $\pi_i > 0$ are rational, the function

$$f(x) = \begin{cases} x_1^{-\pi_1} \dots x_m^{-\pi_m}, & x > 0 \\ +\infty, & \text{otherwise} \end{cases}$$

is CQr.

Indeed, when p_1, \dots, p_m, q are positive integers such that $\pi_i = p_i/q$ and $\mu \in \mathbb{N}$ is such that $M = 2^\mu \geq p_1 + \dots + p_m + q$, we have

$$\text{Epi}\{f\} = \{(t, x_1, \dots, x_m) \geq 0 : t^q x_1^{p_1} \dots x_m^{p_m} \geq 1\},$$

that is, $\text{Epi}\{f\}$ is the intersection of a polyhedral set and the inverse image of the CQr set

$$\{(s, y_1, \dots, y_M) \geq 0 : s^M \leq y_1 \dots y_M\}$$

under the affine mapping

$$(t, x_1, \dots, x_m) \mapsto (1, \underbrace{t, \dots, t}_q, \underbrace{x_1, \dots, x_1}_{p_1}, \dots, \underbrace{x_m, \dots, x_m}_{p_m}, \underbrace{1, \dots, 1}_{M-q \sum_i p_i}).$$

8.6. The epigraph of the $\|\cdot\|_\pi$ -norm. When $\pi \geq 1$ is rational (or $\pi = \infty$), the function $f(x) = \|x\|_\pi : \mathbf{R}^m \rightarrow \mathbf{R}$ is CQr. For example, the sets

$$\{[x; t] : t \geq [\sum_i |x_i|^{3/2}]^{2/3}\} \quad \text{or} \quad \{[x; t] : t \geq [\sum_i |x_i|^5]^{1/5}\}$$

are CQr.

Indeed, the case of $\pi = \infty$ is trivial – in this case $\text{Epi}\{f\}$ is a polyhedral set. Now let $\pi = p/q$ with positive integer $p \geq q$. It is immediately seen that

$$\|x\|_p \leq t \Leftrightarrow t \geq 0 \ \& \ \exists v_1, \dots, v_m \geq 0 : |x_i| \leq t^{(\pi-1)/\pi} v_i^{1/\pi}, \ i = 1, \dots, m, \sum_{i=1}^n v_i \leq t. \quad (*)$$

As we have seen in **8.5**, the set $Z = \{(\tau, \xi, \sigma) : \tau \geq 0, \sigma \geq 0, \xi \leq \tau^{\frac{p-q}{p}} \sigma^{\frac{q}{p}}\}$ is CQr. Consequently, so are the sets

$$X_i = \{(x, v, t) \in \mathbf{R}^{2m+1} : t \geq 0, v \geq 0, |x_i| \leq t^{(\pi-1)/\pi} v_i^{1/\pi}\} = \{(x, v, t) \in \mathbf{R}^{2m+1} : t \geq 0, v \geq 0, \pm x_i \leq t^{p-q/p} v_i^{q/p}\}$$

– each of these sets is the intersection of two inverse affine images of Z under affine mappings. By (*), $\text{Epi}\{f\}$ is the image, under the linear mapping $(x, t, v) \mapsto (x, t)$, of the CQr set

$$\{(x, t, v) : \sum_i v_i \leq t\} \cap [\cap_i X_i],$$

so that $\text{Epi}\{f\}$ is a CQr set, $\Rightarrow f$ is CQr.

8.7. The hypograph of the concave function $(\sum_i x_i^\pi)^{1/\pi} : \mathbf{R}_+^m \rightarrow \mathbf{R}$, where $\pi \in (0, 1]$ is rational, is CQr. For example, the sets

$$\{[x; t] : x \geq 0, t \leq \left[\sum_i x_i^{2/3}\right]^{3/2}\} \quad \text{or} \quad \{[x; t] : x \geq 0, t \leq \left[\sum_i x_i^{1/5}\right]^5\}$$

are CQr.

Indeed, the case of $\pi = 1$ is trivial. When $0 < \pi < 1$, it is immediately seen that

$$\begin{aligned} x \geq 0 \ \& \ t \leq (\sum_i x_i^\pi)^{1/\pi} &\Leftrightarrow x \geq 0 \ \& \ \exists \tau : 0 \leq \tau, t \leq \tau, \tau \leq (\sum_i x_i^\pi)^{1/\pi} \\ &\Leftrightarrow x \geq 0 \ \& \ \exists \tau \geq 0, v_i \geq 0 : 0 \leq \tau, t \leq \tau, \underbrace{v_i \leq x_i^\pi \tau^{1-\pi}}_{\text{CQr}}, \sum_i v_i \geq \tau \end{aligned}$$

Fast CQr approximations of exponent and logarithm

♠ Exponent $\exp\{x\}$ which lives in our mind is defined on the entire real axis and rapidly goes to 0 as $x \rightarrow -\infty$ and to $+\infty$ as $x \rightarrow \infty$. Exponent which lives in a computer is a different beast: if you ask a computer what is $\exp\{-750\}$ or $\exp\{750\}$, it will return 0 in the first, and $+\infty$ in the second case.

⇒ For all practical purposes, we can restrict the domain of the exponent — pass from $\exp\{x\}$ to

$$\text{Exp}_R(x) = \begin{cases} \exp\{x\}, & |x| \leq R \\ +\infty, & \text{otherwise} \end{cases}$$

with once for ever fixed moderate (few hundreds) R .

Fact: For all practical purposes, $\text{Exp}_R(\cdot)$ is CQr. Rigorously speaking: for every $\epsilon \in (0, 0.1)$ we can point out a CQr function $E_{R,\epsilon}$ with domain $[-R, R]$ and with explicit CQR (involving $O(1)\ln(R/\epsilon)$ variables and conic quadratic constraints) such that

$$\forall x \in [-R, R] : (1 - \epsilon) \exp\{x\} \leq E_{R,\epsilon}(x) \leq \exp\{x\}.$$

The idea: As you hopefully remember, *by definition*

$$\exp\{x\} = \lim_{n \rightarrow \infty} (1 + x/n)^n.$$

Direct computation shows that *when $n = 1024$ and $|x| \leq 4$, one has*

$$0.992 \exp\{x\} \leq f_{1024}(x) := (1 + x/1024)^{1024} \leq \exp\{x\}.$$

On the segment $|x| \leq 4$, the function $f_{1024}(x)$ admits short CQR:

$$|x| \leq 4, t \geq f_{1024}(x) \leq t \Leftrightarrow \{-4 \leq x \leq 4 \ \& \ \exists u_0, \dots, u_9 : 1 + x/1024 \leq u_0, u_0^2 \leq u_1, u_1^2 \leq u_2, \dots, u_8^2 \leq u_9, u_9^2 \leq t\}$$

It is clear how to proceed: *a tight CQR approximation of $\exp\{x\}$ on a segment $|x| \leq R$ is $(1 + x/2^k)^{(2^k)}$ with properly selected integer k .*

$$\text{Exp}_R(x) = \begin{cases} \exp\{x\}, & |x| \leq R \\ +\infty, & \text{otherwise} \end{cases}$$

The construction. Given R and $\epsilon \in (0, 0.1)$, let k be positive integer such that $2^k > 2R$. For $x \in [-R, R]$, setting $y = 2^{-k}x$, we have $|y| \leq \frac{1}{2} \Rightarrow$

$$\exp\{y - 4y^2\} \leq 1 + y \leq \exp\{y\} \quad \& \quad \exp\{x\} = \exp\{2^k y\},$$

whence

$$\exp\{x\} \exp\{-2^{k+2}y^2\} \leq [1 + y]^{(2^k)} \leq \exp\{x\}$$

We have $2^{k+2}y^2 = 2^{k+2}2^{-2k}x^2 \leq 2^{2-k}R^2$

\Rightarrow with properly chosen $O(1)$ and $k = \lceil O(1) \ln(R/\epsilon) \rceil$, we have

$$(1 - \epsilon) \exp\{x\} \leq [1 + 2^{-k}x]^{(2^k)} \leq \exp\{x\} \quad \forall x \in [-R, R]$$

\Rightarrow With the just defined k , the CQr function given by

$$t \geq E_{R,\epsilon}(x) \Leftrightarrow \{|x| \leq R, \exists u_0, \dots, u_{k-1} : 1 + 2^{-k}x \leq u_0, u_0^2 \leq u_1, u_1^2 \leq u_2, \dots, u_{k-2}^2 \leq u_{k-1}, u_{k-1}^2 \leq t\}$$

is the required tight CQr approximation of $\text{Exp}_R(\cdot)$.

♣ Tight CQr approximation of “computer exponent” $\text{Exp}_R(\cdot)$ yields tight CQr approximation of the (minus) “computer logarithm.” The construction is as follows:

- Given $\epsilon \in (0, 0.1)$ and R , we have built a CQr set $\mathcal{Q} = \mathcal{Q}_{R,\epsilon} \subset \mathbf{R}^2$ with “short” (of size $O(1) \ln(R/\epsilon)$) explicit CQR

$$(x, t) \in \mathcal{Q} \Leftrightarrow \left\{ |x| \leq R, \exists u_0, \dots, u_{k-1} : 1 + 2^{-k}x \leq u_0, u_0^2 \leq u_1, u_1^2 \leq u_2, \dots, u_{k-2}^2 \leq u_{k-1}, u_{k-1}^2 \leq t \right\}$$

and have ensured that

A. If $(x, t) \in \mathcal{Q}$ and $t' \geq t$, then $(x, t') \in \mathcal{Q}$;

B. If $(x, t) \in \mathcal{Q}$, then $|x| \leq R$ and $t \geq (1 - \epsilon) \exp\{x\}$

C. If $|x| \leq R$, then there exists t such that $(x, t) \in \mathcal{Q}$ and $t \leq (1 + \epsilon) \exp\{x\}$

Now let

$$\Delta = \Delta_{R,\epsilon} = (1 + \epsilon)[\exp\{-R\}, \exp\{R\}]$$

(with R like 700, Δ , “for all practical purposes,” is the entire positive ray), and let

$$\overline{\mathcal{Q}} = \overline{\mathcal{Q}}_{R,\epsilon} = \{(x, t) \in \mathcal{Q}_{R,\epsilon} : t \in \Delta\}$$

Note that $\overline{\mathcal{Q}}$ is CQr with explicit and short CQR readily given by the CQR of \mathcal{Q} . Let function $\text{Ln}(t) := \text{Ln}_{R,\epsilon}(t) : \mathbf{R} \rightarrow \mathbf{R} \cup \{-\infty\}$ be defined by the relation

$$z \leq \text{Ln}(t) \Leftrightarrow \exists x : z \leq x \ \& \ (x, t) \in \overline{\mathcal{Q}}$$

From A, B, C it immediately follows that

$\text{Ln}(t)$ is a concave function with hypograph given by explicit CQR which approximates $\ln(t)$ on Δ within accuracy $O(\epsilon)$:

$$t \notin \Delta \Rightarrow \text{Ln}(t) = -\infty \ \& \ t \in \Delta \Rightarrow -\ln(1 + \epsilon) \leq \text{Ln}(t) - \ln(t) \leq \ln\left(\frac{1}{1 - \epsilon}\right)$$

$$z \leq \text{Ln}(t) \Leftrightarrow \exists x : z \leq x \ \& \ (x, t) \in \overline{\mathcal{Q}} \quad (*)$$

Claim: $t \notin \Delta \Rightarrow \text{Ln}(t) = -\infty$ & $t \in \Delta \Rightarrow -\ln(1 + \epsilon) \leq \text{Ln}(t) - \ln(t) \leq -\ln(1 - \epsilon)$

Verification is immediate. When $t \notin \Delta$, the right hand side condition in (*) never takes place (since $(x, t) \in \overline{\mathcal{Q}}$ implies $t \in \Delta$) $\Rightarrow \text{Ln}(t) = -\infty$ outside of Δ , as claimed.

Now let $t \in \Delta$. If $z \leq \text{Ln}(t)$, then there exists $x \geq z$ such that $(x, t) \in \overline{\mathcal{Q}} \subset \mathcal{Q}$, whence $\exp\{x\}(1 - \epsilon) \leq t$ by B, that is, $z \leq x \leq \ln(t) - \ln(1 - \epsilon)$. Since this relation holds true for every $z \leq \text{Ln}(t)$, we get

$$\text{Ln}(t) \leq \ln(t) - \ln(1 - \epsilon),$$

as claimed. On the other hand, let $x_t = \ln(t) - \ln(1 + \epsilon)$, that is, $\exp\{x_t\}(1 + \epsilon) = t$. Since $t \in \Delta$, we have $|x_t| \leq R$, which, by C, implies that there exists t' such that $(x_t, t') \in \mathcal{Q}$ and $t' \leq (1 + \epsilon) \exp\{x_t\} = t$. By A it follows that $(x_t, t) \in \mathcal{Q}$, and since $t \in \Delta$, we have also $(x_t, t) \in \overline{\mathcal{Q}}$. Setting $z = x_t$, we get $z \leq x_t$ and $(x_t, t) \in \overline{\mathcal{Q}} \Rightarrow z = x_t \leq \text{Ln}(t)$ by (*). Thus, $\text{Ln}(t) \geq x_t = \ln(t) - \ln(1 + \epsilon)$ \square

♠ Our construction has two components:

- Computing $\exp\{x\}$ for large x reduces to computing $\exp\{2^{-k}x\}$ and squaring the result k times
- For small y , $\exp\{y\} \approx 1 + y$, and this simplest approximation is accurate enough for our purposes.

Note: The second component can be improved: we can approximate $\exp\{y\}$ by a larger part of its Taylor expansion, provided that the epigraph of this part is CQr. For example,

$$g_6(y) = 1 + y + \frac{y^2}{2} + \frac{y^3}{6} + \frac{y^4}{24} + \frac{y^5}{120} + \frac{y^6}{720}$$

for small y approximates $\exp\{y\}$ much better than $g_1(y) = 1 + y$ and happens to be convex function of y representable as

$$g_6(y) = c_0 + c_2(\alpha_2 + y)^2 + c_4(\alpha_4 + y)^4 + c_6(\alpha_6 + y)^6 \quad [c_i > 0 \forall i]$$

$\Rightarrow g_6$ is CQr with simple CQR. As a result, the CQr function $E(x)$ with the CQR

$$t \geq E(x) \Leftrightarrow \left\{ |x| \leq R, \exists u_0, \dots, u_{k-1} : \underbrace{g_6(2^{-k}x) \leq u_0}_{\text{CQR}}, u_0^2 \leq u_1, u_1^2 \leq u_2, \dots, u_{k-2}^2 \leq u_{k-1}, u_{k-1}^2 \leq t \right\}$$

ensures the target relation

$$|x| \leq R \Rightarrow (1 - \epsilon) \exp\{x\} \leq E(x) \leq (1 + \epsilon) \exp\{x\}$$

with smaller k than in our initial construction. For example, with g_6 in the role of g_1 , $R = 700$, $k = 15$ we ensure $\epsilon = 3.0\text{e-}11$.

Note: Our result is “honest” – this is what happens on a real computer. In our previous considerations there was a slight cheating: by reasons similar to those which make “computer exponent” of 750 equal $+\infty$, with standard floating point arithmetic, operating with numbers like $1 + y$ for “very small” y leads to significant loss of accuracy. As a result, with our initial construction and $R = 700$ the best achievable ϵ is as “large” as $1.13\text{e-}5$ (corresponds to $k = 35$).

Robust Linear Programming: motivation

♣ Consider an LP program

$$\min_x \{c^T x : Ax + b \geq 0\} \quad (\text{LP})$$

In applications, the *data* (c, A, b) of the program not always are known exactly, reasons being *at least*:

- *measurement errors* in data entries like characteristics of devices, durations and outcomes of technological processes, etc. Not only data of this type are obtained by imprecise measurements — in reality the very values of these data, rather than being fixed reals, slightly fluctuate in time
- *prediction errors* in data entries related to remote, not directly accessible, locations in time and space (future demands, temperatures, etc.)
- *implementation errors*. Some of the design variables x_j may represent characteristics of physical processes and/or devices and as such cannot be implemented exactly as computed: the implemented value x_j of a variable and its computed value x_j^* usually are linked by $x_j = (1 + \epsilon_j)x_j^*$ with unknown ϵ_j from a known small range. The effect of implementation errors on a linear constraint $\sum_j a_j x_j \leq b$ is exactly *as if* there were no implementation errors, but the data a_j were subject to perturbation $a_j \mapsto a_j(1 + \epsilon_j)$.

♠ In LP practice small data uncertainties (like 0.1% or less) are usually ignored, and the problem is processed as if the data were exact.

(!) It turns out that ignoring small data uncertainties can make the resulting nominal optimal solution meaningless.

Example 1: Synthesis of Antenna array

♣ **The diagram of an antenna.** Consider a (monochromatic) antenna placed at the origin. The electric field generated by the antenna at a remote point $r\delta$ (δ is a unit direction) is

$$E = a(\delta)r^{-1} \cos(\phi(\delta) + t\omega - 2\pi r/\lambda) + o(r^{-1})$$

- t : time • ω : frequency • λ : wavelength
- It is convenient to aggregate $a(\delta)$ and $\phi(\delta)$ into a single complex-valued function – the *diagram* of the antenna

$$D(\delta) = a(\delta)(\cos(\phi(\delta)) + i \sin(\phi(\delta))).$$

- The directional density of the energy sent by the antenna is proportional to $|D(\cdot)|^2$
- The diagram $D(\cdot)$ of a complex antenna comprised of several antenna elements is the sum of the diagrams $D_i(\cdot)$ of the elements:

$$D(\delta) = D_1(\delta) + \dots + D_N(\delta)$$

♣ **Synthesis of Array of Antennae:** Given a target diagram $D_*(\cdot)$ along with N “building blocks” – antenna elements with diagrams $D_1(\cdot), \dots, D_N(\cdot)$ – find “weights” $z_j \in \mathbb{C}$ such that the function

$$\sum_{j=1}^N z_j D_j(\cdot)$$

is as close as possible to the target diagram $D_*(\cdot)$.

- Physically, multiplication of a diagram $D_j(\cdot)$ by a complex weight z_j means that the corresponding standard “building block” is preceded by appropriate amplification and delay devices.
- Choosing a fine grid Δ of directions δ , we may pose the Antenna Synthesis problem as a discrete approximation problem with complex-valued data and design variables:

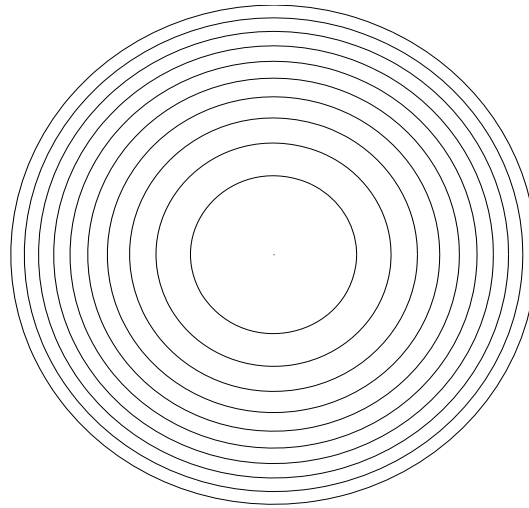
$$\min_{\tau, z} \left\{ \tau : \left| D_*(\delta) - \sum_{j=1}^N z_j D_j(\delta) \right| \leq \tau \quad \forall \delta \in \Delta \right\},$$

which is a CQP.

- Sometimes the diagrams of the elements and the target diagram are real-valued. In this case, we lose nothing when restricting z_j to be real, and thus end up with an LP program.

Antenna synthesis: Example

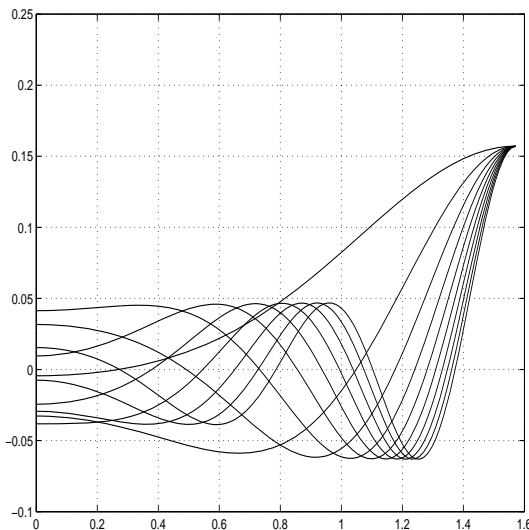
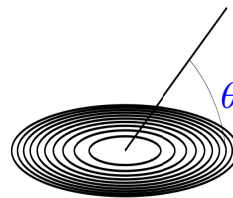
- ♣ Let a planar antenna be comprised of a central circle and 9 concentric rings of the same area placed in the XY -plane (“Earth’s surface”):



The radius of the antenna is 1m

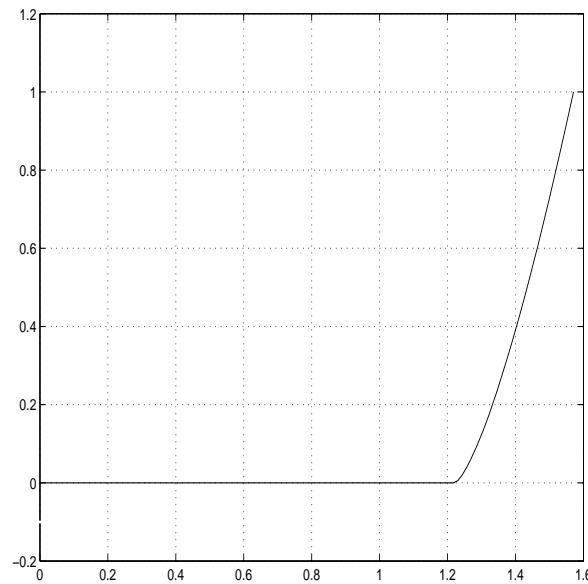
- The diagram of a ring $\{a \leq r \leq b\}$ in the XY -plane is real-valued and depends on direction's altitude angle θ only:

$$D_{a,b}(\theta) = \frac{1}{2} \int_a^b \left[\int_0^{2\pi} \rho \cos(2\pi \rho \lambda^{-1} \cos(\theta) \cos(\phi)) d\phi \right] d\rho.$$



Diagrams of 10 rings as functions
of altitude angle $\theta \in [0, \pi/2]$, $\lambda = 0.5\text{m}$

- Assume the target diagram to be real-valued function of the altitude angle “concentrated” in the segment $\frac{\pi}{2} - \frac{\pi}{12} \leq \theta \leq \frac{\pi}{2}$:

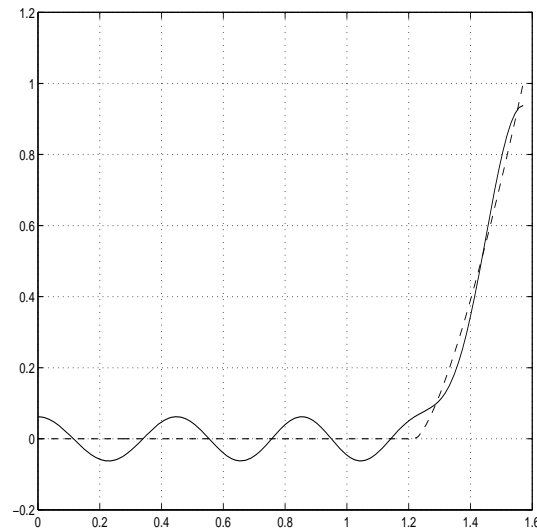


The target diagram

- With 120-point discretization of altitudes, the Antenna Synthesis problem becomes an LP program with 11 variables and 240 linear constraints:

$$\min_{x, \tau} \left\{ \tau : -\tau \leq D_*(\theta_\ell) - \sum_{j=1}^{10} x_j D_j(\theta_\ell) \leq \tau, \theta_\ell = \frac{\pi}{2\ell}, 1 \leq \ell \leq 120 \right\}$$

- The resulting diagram approximates the target within absolute inaccuracy 0.0621:



The target diagram (dashed) and the synthesised diagram (solid)

- The optimal weights (rounded to 5 digits) are

element #	1	2	3	4	5	6	7	8	9	10
weight	1624.4	-14700	55383	-107247	95468	19221	-138620	144870	-69303	13311

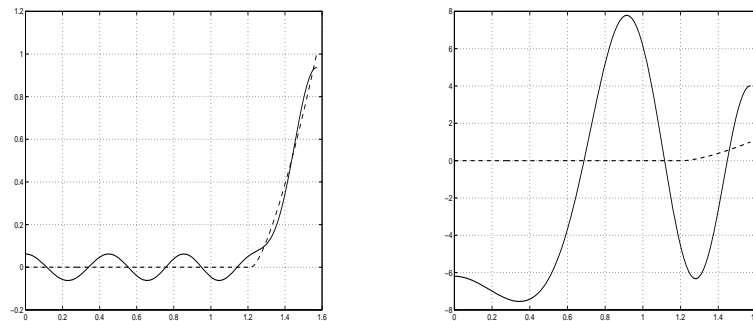
♣ The optimal weights x_j^* , $j = 1, \dots, 10$, are characteristics of physical devices. In reality, they somehow drift around their computed values. What happens when the weights are affected by small (just 0.1%) random perturbations:

$$x_j = (1 + \epsilon_j)x_j^*$$

$$\left[\left\{ \epsilon_j \sim \text{Uniform}[-0.001, 0.001] \right\}_{j=1}^{10} \right] \quad ?$$

♣ The results of 0.1% “implementation errors” are disastrous:

“Dream and reality”



“Nominal” diagram Actual diagram
[dashed: the target diagram]

- The target diagram is of the uniform norm 1, and its uniform distance from the nominal diagram is ≈ 0.06 .
- The realization of “actual diagram” shown on the picture is at the uniform distance 7.8 from the target diagram!

Example 2: NETLIB Case Study: Diagnosis

♣ NETLIB is a collection of about 100 not very large LPs, mostly of real-world origin. To motivate the methodology of our “case study”, here is constraint # 372 of the NETLIB problem PIL0T4:

$$\begin{aligned}
 a^T x &\equiv -15.79081x_{826} - 8.598819x_{827} - 1.88789x_{828} - 1.362417x_{829} - 1.526049x_{830} \\
 &\quad -0.031883x_{849} - 28.725555x_{850} - 10.792065x_{851} - 0.19004x_{852} - 2.757176x_{853} \\
 &\quad -12.290832x_{854} + 717.562256x_{855} - 0.057865x_{856} - 3.785417x_{857} - 78.30661x_{858} \\
 &\quad -122.163055x_{859} - 6.46609x_{860} - 0.48371x_{861} - 0.615264x_{862} - 1.353783x_{863} \\
 &\quad -84.644257x_{864} - 122.459045x_{865} - 43.15593x_{866} - 1.712592x_{870} - 0.401597x_{871} \\
 &\quad +x_{880} - 0.946049x_{898} - 0.946049x_{916} \\
 &\geq b \equiv 23.387405
 \end{aligned} \tag{C}$$

The related *nonzero* coordinates in the optimal solution x^* of the problem, as reported by CPLEX, are:

$$\begin{array}{lll}
 x_{826}^* = 255.6112787181108 & x_{827}^* = 6240.488912232100 & x_{828}^* = 3624.613324098961 \\
 x_{829}^* = 18.20205065283259 & x_{849}^* = 174397.0389573037 & x_{870}^* = 14250.00176680900 \\
 x_{871}^* = 25910.00731692178 & x_{880}^* = 104958.3199274139 &
 \end{array}$$

This solution makes (C) an equality within machine precision.

♣ Most of the coefficients in (C) are “ugly reals” like -15.79081 or -84.644257. We definitely may believe that these coefficients characterize technological devices/processes, and as such *hardly are known to high accuracy*. Thus, “ugly coefficients” may be assumed to be uncertain and to coincide with the “true” data within accuracy of 3-4 digits. The only exception is the coefficient 1 of x_{880} , which perhaps reflects the structure of the problem and is exact.

$$\begin{aligned}
a^T x &\equiv -15.79081x_{826} - 8.598819x_{827} - 1.88789x_{828} - 1.362417x_{829} - 1.526049x_{830} \\
&\quad -0.031883x_{849} - 28.725555x_{850} - 10.792065x_{851} - 0.19004x_{852} - 2.757176x_{853} \\
&\quad -12.290832x_{854} + 717.562256x_{855} - 0.057865x_{856} - 3.785417x_{857} - 78.30661x_{858} \\
&\quad -122.163055x_{859} - 6.46609x_{860} - 0.48371x_{861} - 0.615264x_{862} - 1.353783x_{863} \\
&\quad -84.644257x_{864} - 122.459045x_{865} - 43.15593x_{866} - 1.712592x_{870} - 0.401597x_{871} \\
&\quad +x_{880} - 0.946049x_{898} - 0.946049x_{916} \\
&\geq b \equiv 23.387405
\end{aligned} \tag{C}$$

$$\begin{array}{lll}
x_{826}^* = 255.6112787181108 & x_{827}^* = 6240.488912232100 & x_{828}^* = 3624.613324098961 \\
x_{829}^* = 18.20205065283259 & x_{849}^* = 174397.0389573037 & x_{870}^* = 14250.00176680900 \\
x_{871}^* = 25910.00731692178 & x_{880}^* = 104958.3199274139 &
\end{array}$$

♣ Assume that the uncertain entries of a are 0.1%-accurate approximations of unknown entries in the “true” data \tilde{a} , how would this uncertainty affect the validity of the constraint *evaluated at the nominal solution* x^* ?

- The worst case, over all 0.1%-perturbations of uncertain data, violation of the constraint is as large as 450% of the right hand side!
- In the case of *random* and *independent* 0.1% perturbations of the uncertain coefficients, the statistics of the “relative constraint violation”

$$V = \frac{\max[b - \tilde{a}^T x^*, 0]}{b} \times 100\%$$

also is disastrous:

Prob{ $V > 0$ }	Prob{ $V > 150\%$ }	Mean(V)
0.50	0.18	125%

Relative violation of constraint # 372 in PILOT4

(1,000-element sample of 0.1% perturbations of the uncertain data)

♣ We see that *quite small (just 0.1%) perturbations of “obviously uncertain” data coefficients can make the “nominal” optimal solution x^* heavily infeasible and thus – practically meaningless.*

♣ In our Case Study, we choose a “perturbation level” ϵ (taking values 1%, 0.1%, 0.01%), and, for every one of the NETLIB problems, measure the “reliability index” of the nominal solution at this perturbation level, specifically, as follows.

- We compute the optimal solution x^* of the program by CPLEX.
- For every one of the *inequality* constraints

$$a^T x \leq b \quad (*)$$

— we split the right hand side coefficients a_j into “certain” (rational fractions p/q with $|q| \leq 100$) and “uncertain” (all the rest). Let J be the set of all uncertain coefficients of $(*)$.

— we define the *reliability index* of $(*)$ as

$$\frac{a^T x^* + \epsilon \sqrt{\sum_{j \in J} a_j^2 (x_j^*)^2} - b}{\max[1, |b|]} \times 100\%$$

Note that *the reliability index is of order of typical violation* (measured in percents of the right hand side) *of the constraint, as evaluated at x^* , under independent random perturbations, of relative magnitude ϵ , of the uncertain coefficients.*

- We treat the nominal solution as *unreliable*, and the problem - as *bad*, the level of perturbations being ϵ , if the worst, over the inequality constraints, reliability index is worse than 5%.

♣ **The results** of the Diagnosis phase of our Case Study are as follows.

From the total of 90 NETLIB problems we have processed,

- in 27 problems the nominal solution turned out to be unreliable at the largest ($\epsilon = 1\%$) level of uncertainty;
- 19 of these 27 problems are already bad at the 0.01%-level of uncertainty, and in 13 of these 19 problems, 0.01% perturbations of the uncertain data can make the nominal solution more than 50%-infeasible for some of the constraints.

Problem	Size ^{a)}	$\epsilon = 0.01\%$		$\epsilon = 0.1\%$		$\epsilon = 1\%$	
		#bad ^{b)}	Index ^{c)}	#bad	Index	#bad	Index
80BAU3B	2263 × 9799	37	84	177	842	364	8,420
25FV47	822 × 1571	14	16	28	162	35	1,620
ADLITTLE	57 × 97			2	6	7	58
AFIRO	28 × 32			1	5	2	50
BNL2	2325 × 3489					24	34
BRANDY	221 × 249					1	5
CAPRI	272 × 353			10	39	14	390
CYCLE	1904 × 2857	2	110	5	1,100	6	11,000
D2Q06C	2172 × 5167	107	1,150	134	11,500	168	115,000
E226	224 × 282					2	15
FFFFFF800	525 × 854					6	8
FINNIS	498 × 614	12	10	63	104	97	1,040
GREENBEA	2393 × 5405	13	116	30	1,160	37	11,600
KB2	44 × 41	5	27	6	268	10	2,680
MAROS	847 × 1443	3	6	38	57	73	566
NESM	751 × 2923					37	20
PEROLD	626 × 1376	6	34	26	339	58	3,390
PILOT	1442 × 3652	16	50	185	498	379	4,980
PILOT4	411 × 1000	42	210,000	63	2,100,000	75	21,000,000
PILOT87	2031 × 4883	86	130	433	1,300	990	13,000
PILOTJA	941 × 1988	4	46	20	463	59	4,630
PILOTNOV	976 × 2172	4	69	13	694	47	6,940
PILOTWE	723 × 2789	61	12,200	69	122,000	69	1,220,000
SCFXM1	331 × 457	1	95	3	946	11	9,460
SCFXM2	661 × 914	2	95	6	946	21	9,460
SCFXM3	991 × 1371	3	95	9	946	32	9,460
SHARE1B	118 × 225	1	257	1	2,570	1	25,700

- a) # of linear constraints (excluding the box ones) plus 1 and # of variables
b) # of constraints with index > 5%
c) The worst, over the constraints, reliability index, in %

♣ Conclusions:

◇ *In real-world applications of Linear Programming one cannot ignore the possibility that a small uncertainty in the data (intrinsic for the majority of real-world LP programs) can make the usual optimal solution of the problem completely meaningless from a practical viewpoint.*

Consequently,

◇ *In applications of LP, there exists a real need of a technique capable of detecting cases when data uncertainty can heavily affect the quality of the nominal solution, and in these cases to generate a “reliable” solution, one which is immune against uncertainty.*

Robust Linear Programming: the paradigm

♣ Consider an LP program

$$\min_x \{c^T x : Ax + b \geq 0\} \quad (\text{LP})$$

Assume that the data (c, A, b) of the program are not known exactly; all we know is an *uncertainty set* \mathcal{U} the “true data” belong to.

♣ A natural way to process an LP program with uncertain data is to build the *robust counterpart* of the program, where we impose on candidate solutions the requirement to be *robust feasible* – to satisfy *all* realizations of the inequality constraints. Among these robust feasible solutions, we are seeking for the “best” – with the smallest possible *guaranteed* value of the objective. Thus, the robust counterpart of (LP) is the problem

$$\min_x \left\{ f(x) = \max_{c \in \mathcal{U}_{\text{obj}}} c^T x : Ax + b \geq 0 \ \forall (A, b) \in \mathcal{U}_{\text{cons}} \right\} \quad (\text{RC})$$

where

$$\begin{aligned} \mathcal{U}_{\text{obj}} &= \{c : \exists (A, b) : (c, A, b) \in \mathcal{U}\}, \\ \mathcal{U}_{\text{cons}} &= \{(A, b) : \exists c : (c, A, b) \in \mathcal{U}\} \end{aligned}$$

are the projections of the uncertainty set on the spaces of the data of the objective and the constraints, respectively.

$$\min_x \left\{ c^T x : Ax + b \geq 0 \right\}, \quad (c, A, b) \in \mathcal{U} \quad (\text{ULP})$$

\Downarrow

$$\min_x \left\{ f(x) = \max_{c \in \mathcal{U}_{\text{obj}}} c^T x : Ax + b \geq 0 \ \forall (A, b) \in \mathcal{U}_{\text{cons}} \right\}$$

\Updownarrow

$$\min_{t, x} \left\{ t : \begin{array}{l} c^T x \leq t, \\ Ax + b \geq 0 \end{array} \ \forall (c, A, b) \in \mathcal{U} \right\} \quad (\text{RC})$$

♣ Robust counterpart is a *semi-infinite* convex optimization program – one with infinitely many linear inequality constraints. Possibilities to process such a problem depend on the geometry of the uncertainty set \mathcal{U} .

♣ *If the uncertainty set \mathcal{U} is an ellipsoid (or an intersection of ellipsoids), or, more generally, is CQr, (RC) can be converted to a conic quadratic program.*

Theorem. Consider an uncertain LP

$$\left\{ \min_x \left\{ c^T x : Ax \geq b \right\} : (c, A, b) \in \mathcal{U} \right\} \quad (\text{ULP})$$

and assume that the uncertainty set \mathcal{U} is CQR with an essentially strictly feasible CQR. Then the set of robust feasible solutions to (ULP) is CQR with explicitly given CQR, so that the Robust Counterpart of (ULP) is (equivalent to) an explicit conic quadratic problem.

If \mathcal{U} is polyhedrally representable, then the RC of (ULP) is (equivalent to) an explicit LP problem.

Theorem is an immediate consequence of the following

Observation: Let $\mathcal{Z} \subset \mathbb{R}^{n+1}$ be a nonempty CQR set given by essentially strictly feasible CQR. Then the set

$$\mathcal{X} = \{x : z^T [x; 1] \leq 0 \forall z \in \mathcal{Z}\}$$

is CQR with explicitly given CQR.

Observation: Let $\mathcal{Z} \subset \mathbf{R}^{n+1}$ be a nonempty CQR set given by essentially strictly feasible CQR. Then the set

$$\mathcal{X} = \{x : z^T[x; 1] \leq 0 \forall z \in \mathcal{Z}\}$$

is CQR with explicitly given CQR.

• **Observation \Rightarrow Theorem:** The data a, b of a single uncertain constraint

$$0 \geq a^T x + b \equiv [a; b]^T [x; 1] \quad (*)$$

is linear image of the full data: $z := [a; b] = P\zeta$, $\zeta := (c, A, b)$.

\Rightarrow The RC of $(*)$ reads

$$z^T[x; 1] \leq 0 \forall z \in \mathcal{Z} = P\mathcal{U} := \{z : \exists u \in \mathcal{U} : z = Pu\} \quad (!)$$

The set \mathcal{Z} admits essentially strictly feasible CQR (inherited from essentially strictly feasible CQR of \mathcal{U}), implying by Observation that the feasible set of $(!)$, or, which is the same, of $(*)$, is CQR with CQR readily given by CQR of \mathcal{U} .

\Rightarrow The feasible set of the RC of every uncertain constraint in our uncertain LP is CQR with explicit CQR, so that the feasible set of problem's RC admits explicit CQR.

Observation: Let $\mathcal{Z} \subset \mathbf{R}^{n+1}$ be a nonempty CQR set given by essentially strictly feasible CQR:

$$\mathcal{Z} = \left\{ z \in \mathbf{R}^n : \exists u : \begin{cases} Pz + Qu - r \in \mathbf{K} \\ Rz + Su - s = 0 \end{cases} \right\}$$

$$\exists(\bar{z}, \bar{u}) : P\bar{z} + Q\bar{u} - r \in \text{int } \mathbf{K}, R\bar{z} + S\bar{u} = s$$

(\mathbf{K} : direct product of Lorentz cones). Then the set

$$\mathcal{X} = \{x : z^T[x; 1] \leq 0 \forall z \in \mathcal{Z}\}$$

is CQR with explicitly given CQR.

• The claim is readily given by the Calculus rule on CQ representability of the support function of a CQR set. Here is direct demonstration:

$$\begin{aligned} x \in \mathcal{X} &\Leftrightarrow \sup_{z \in \mathcal{Z}} [x; 1]^T z \leq 0 \Leftrightarrow 0 \geq \sup_{z, u} \{ [x; 1]^T z : Pz + Qu - r \in \mathbf{K}, Rz + Su = s \} \\ &\underbrace{\Leftrightarrow}_{(a)} 0 \geq \min_{y, v} \{ -r^T y - s^T v : y \in \mathbf{K}_* [= \mathbf{K}], P^T y + R^T v + [x; 1] = 0, Q^T y + S^T v = 0 \} \\ &\underbrace{\Leftrightarrow}_{(b)} \exists y, v : y \in \mathbf{K}_* [= \mathbf{K}], P^T y + R^T v + [x; 1] = 0, Q^T y + S^T v = 0, r^T y + s^T v \geq 0 \end{aligned}$$

with (a), (b) given by Strong Duality.

♠ In simple cases we can write the RC of an uncertain LP straightforwardly, without using CQR machinery.

Example: The Robust Counterpart of uncertain LP with *interval uncertainty*:

$$\begin{aligned}\mathcal{U}_{\text{obj}} &= \{c : |c_j - c_j^0| \leq \delta c_j, j = 1, \dots, n\} \\ \mathcal{U}_i &= \{(a_{i1}, \dots, a_{in}, b_i) : |a_{ij} - a_{ij}^0| \leq \delta a_{ij}, |b_i - b_i^0| \leq \delta b_i\}\end{aligned}$$

is the LP program

$$\min_{x,y,t} \left\{ t : \begin{aligned} \sum_j c_j^0 x_j + \sum_j \delta c_j y_j &\leq t \\ \sum_j a_{ij}^0 x_j + \sum_j \delta a_{ij} y_j &\leq b_i - \delta b_i^0 \\ -y_j &\leq x_j \leq y_j \end{aligned} \right\}$$

How it works? – Antenna Example

$$\min_{x,\tau} \left\{ \tau : -\tau \leq D_*(\theta_\ell) - \sum_{j=1}^{10} x_j D_j(\theta_\ell) \leq \tau, \ell = 1, \dots, L \right\}$$

$$\Leftrightarrow \boxed{\min_{x,\tau} \{ \tau : Ax + \tau a + b \geq 0 \}} \quad (\text{LP})$$

- The influence of “implementation errors”

$$x_j \mapsto (1 + \epsilon_j)x_j, |\epsilon_j| \leq \rho,$$

is *as if* there were no implementation errors, but the part A of the constraint matrix were uncertain and known “up to multiplication by a diagonal matrix with diagonal entries from $[1 - \rho, 1 + \rho]$ ”:

$$\mathcal{U}_{\text{ini}} = \left\{ A = A^{\text{nom}} \text{Diag}\{1 + \epsilon_1, \dots, 1 + \epsilon_{10}\} : |\epsilon_j| \leq \rho \right\} \quad (\text{U})$$

Note that *as far as a particular constraint is concerned, the uncertainty is an interval one with $\delta A_{ij} = \rho |A_{ij}|$. The remaining coefficients (and the objective) are certain.*

♣ To improve reliability of our design, we could replace the uncertain LP program (LP), (U) with its robust counterpart, which is nothing but an explicit LP program.

♠ However, to work with interval uncertainty set \mathcal{U}_{ini} would be “too conservative” – the implementation errors are random and independent \Rightarrow the probability for all of them to take simultaneously the “most unfavourable” values is negligibly small.

Let us try to define the uncertainty set in a smarter way.

♣ Consider a linear constraint

$$\sum_{j=1}^n a_j x_j + b \geq 0 \quad (\text{L})$$

and let a_j be randomly perturbed: $a_j \mapsto (1 + \epsilon_j)a_j$ ϵ_j being independent symmetrically distributed and bounded random variables:

$$\epsilon_j \sim -\epsilon_j \text{ and } |\epsilon_j| \leq \sigma_j.$$

What is a “reliable version” of (L)?

Note: When assuming a_j fixed and x_j randomly perturbed: $x_j \mapsto (1 + \epsilon_j)x_j$, we are in exactly the same situation as when a_j are randomly perturbed and x_j are fixed!

$$\sum_{j=1}^n a_j x_j + b \geq 0 \quad (\text{L})$$

- With randomly perturbed a_j , the left hand side in (L) becomes a random variable:

$$\zeta = \sum_{j=1}^n a_j (1 + \epsilon_j) x_j + b$$

$$\left[\begin{array}{l} \text{Mean}\{\zeta\} \equiv \mathcal{E}\{\zeta\} = \sum_{j=1}^n a_j x_j + b, \\ \text{StD}\{\zeta\} \equiv (\mathcal{E}\{(\zeta - \text{Mean}\{\zeta\})^2\})^{1/2} \leq \sqrt{\sum_{j=1}^n \sigma_j^2 a_j^2 x_j^2}. \end{array} \right]$$

- Let us choose a “safety parameter” κ and ignore all events where

$$\zeta < \text{Mean}\{\zeta\} - \kappa \text{StD}\{\zeta\},$$

taking full responsibility for all remaining events.

With this “common sense” approach, a “reliable” version of (L) becomes the conic quadratic inequality

$$\sum_{j=1}^n a_j x_j + b - \kappa \sqrt{\sum_{j=1}^n \sigma_j^2 a_j^2 x_j^2} \geq 0 \quad (\text{L}_{\text{rel}})$$

$$\boxed{\begin{aligned} \sum_{j=1}^n a_j(1 + \epsilon_j)x_j + b &\geq 0 & (L) \\ \mathcal{E}\{\epsilon_j\} &= 0; \quad |\epsilon_j| \leq \sigma_j \end{aligned}}$$

\Downarrow

$$\boxed{\sum_{j=1}^n a_j x_j + b - \kappa \sqrt{\sum_{j=1}^n \sigma_j^2 a_j^2 x_j^2} \geq 0 \quad (L_{\text{rel}})}$$

- Note: (L_{rel}) is exactly the robust counterpart of (L) associated with the *ellipsoidal* uncertainty set

$$\begin{aligned} \mathcal{U}_\kappa &= \left\{ a' = a + \kappa \text{Diag}(\sigma_1 a_1, \dots, \sigma_n a_n) u : u^T u \leq 1 \right\} \\ &= \left\{ a' : \sum_{j=1}^n \frac{(a'_j - a_j)^2}{\sigma_j^2 a_j^2} \leq \kappa^2 \right\} \end{aligned} \quad (\text{EII})$$

Thus, ignoring “rare events” is equivalent to replacing the actual box

$$\mathcal{U}_{\text{true}} = \left\{ a' : \frac{(a'_j - a_j)^2}{\sigma_j^2 a_j^2} \leq 1, j = 1, \dots, n \right\}$$

of values of the perturbed coefficient vector

$$a' = ((1 + \epsilon_1)a_1, \dots, (1 + \epsilon_n)a_n)^T$$

with ellipsoid (EII).

- It is easily seen that

$$\text{Prob} \left\{ \zeta < \text{Mean}\{\zeta\} - \kappa \sqrt{\sum_{j=1}^n \sigma_j^2 a_j^2 x_j^2} \right\} \leq \exp \left\{ -\frac{\kappa^2}{2} \right\}$$

The probability of the “rare event” we are ignoring when replacing $\mathcal{U}_{\text{true}}$ with $\mathcal{U}_{5.26}$ is $< 10^{-6}$. Note that for n large and all σ_j are of the same order of magnitude, the ellipsoid $\mathcal{U}_{5.26}$ is a “negligible part” of the box $\mathcal{U}_{\text{true}}$!

Proof of the Probability Bound

Theorem [Hoeffding's Inequality] *Let c_i, σ_i be deterministic reals, and ξ_i be independent random variables with zero mean such that $|\xi_i| \leq \sigma_i$. Then for every $\kappa > 0$ one has*

$$p(\kappa) = \text{Prob} \left\{ \sum_i c_i \xi_i > \kappa \underbrace{\sqrt{\sum_i c_i^2 \sigma_i^2}}_{\sigma} \right\} \leq \exp \left\{ -\kappa^2 / 2 \right\}.$$

Proof. For $\gamma > 0$ we have

$$\begin{aligned} \exp\{\gamma\kappa\sigma\}p(\kappa) &\leq \mathbf{E} \left\{ \exp\left\{\gamma \sum_i c_i \xi_i\right\} \right\} = \prod_i \mathbf{E} \left\{ \exp\{\gamma c_i \xi_i\} \right\} \\ &= \prod_i \mathbf{E} \left\{ \exp\{\gamma c_i \xi_i\} - \sinh(\gamma c_i \sigma_i) \sigma_i^{-1} \xi_i \right\} \quad [\text{since } \mathbf{E}\{\xi_i\} = 0] \\ &\leq \prod_i \max_{-\sigma_i \leq s_i \leq \sigma_i} \underbrace{\left[\exp\{\gamma c_i s_i\} - \sinh(\gamma c_i \sigma_i) \sigma_i^{-1} s_i \right]}_{\substack{g_i(s_i), g_i(\cdot): \text{convex} \\ g_i(\pm\sigma_i) = \cosh(\gamma c_i \sigma_i)}} \\ &= \prod_i \cosh(\gamma c_i \sigma_i) = \prod_i \left[\sum_{k=0}^{\infty} \frac{[\gamma^2 c_i^2 \sigma_i^2]^k}{(2k)!} \right] \leq \prod_i \left[\sum_{k=0}^{\infty} \frac{[\gamma^2 c_i^2 \sigma_i^2]^k}{2^k k!} \right] \\ &= \prod_i \exp\left\{ \frac{\gamma^2 c_i^2 \sigma_i^2}{2} \right\} = \exp\{\gamma^2 \sigma^2\}. \end{aligned}$$

Thus,

$$p(\kappa) \leq \min_{\gamma > 0} \exp\left\{ \frac{\gamma^2 \sigma^2}{2} - \gamma \kappa \sigma \right\} = \exp \left\{ -\kappa^2 / 2 \right\}.$$

♣ Applying the outlined methodology to our Antenna example:

$$\min_{x, \tau} \left\{ \tau : -\tau \leq D_*(\theta_\ell) - \sum_{j=1}^{10} x_j D_j(\theta_\ell) \leq \tau, 1 \leq \ell \leq 120 \right\} \quad (\text{LP})$$

\Downarrow

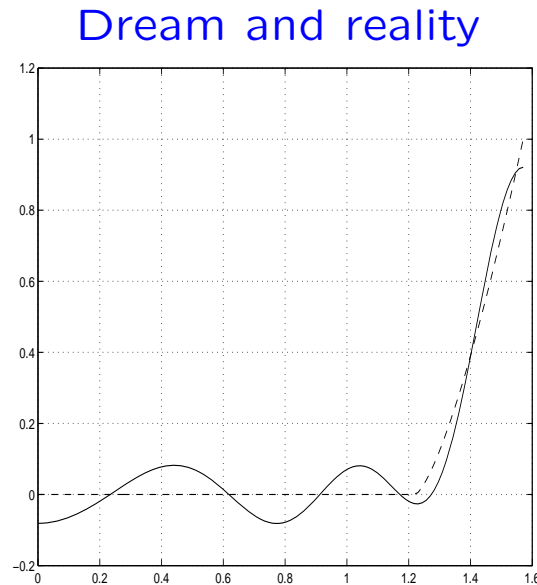
$$\begin{array}{ll} \min_{x, \tau} & \tau \\ D_*(\theta_\ell) - \sum_{j=1}^{10} x_j D_j(\theta_\ell) + \kappa \sigma \sqrt{\sum_{j=1}^{10} x_j^2 D_j^2(\theta_\ell)} & \leq \tau \\ D_*(\theta_\ell) - \sum_{j=1}^{10} x_j D_j(\theta_\ell) - \kappa \sigma \sqrt{\sum_{j=1}^{10} x_j^2 D_j^2(\theta_\ell)} & \geq -\tau \\ & 1 \leq \ell \leq 120 \end{array}$$

(RC)

$$[\sigma = 0.001]$$

we get a *robust design*.

- The results of “Robust Antenna Design” ($\kappa = 1$) are as follows:



A typical “robust” diagram

- The diagram shown on the picture is at uniform distance 0.0822 from the target (just by 30% larger than the “nominal optimal value” is 0.0622 given by “nominal design” which ignores the implementation errors)
- As a compensation, robust design is incomparably more stable than the nominal one: in a sample of 40 realizations of “robust diagrams”, the uniform distance to the target varies from 0.0814 to 0.0830.
- When implementation errors become 10 times larger (1% instead of 0.1%), the “robust design” remains nearly as good as in the case of 0.1%-perturbations: now in a sample of 40 realizations of “robust diagrams”, the uniform distance to the target varies from 0.0834 to 0.116.

♣ Why the “nominal design” is that sensitive to implementation errors? The basic diagrams $D_j(\cdot)$ are “nearly linearly dependent”. As a result, the nominal problem is “ill-posed” – it possesses a huge domain comprised of “nearly optimal” solutions. Indeed, look what are the optimal values in the nominal Antenna Design LP with added box constraints $|x_j| \leq L$ on the variables:

L	1	10	10^2	10^3	10^4	10^5	10^6	10^7
Opt_Val	0.09449	0.07994	0.07358	0.06955	0.06588	0.06272	0.06215	0.06215

The “exactly optimal” solution to the nominal problem is very large, and therefore even small *relative* implementation errors may completely destroy the corresponding design.

In the robust counterpart, magnitudes of candidate solutions are penalized, so that RC implements a smart trade-off between the optimality and the magnitude (i.e., the stability) of the solution.

j	1	2	3	4	5	6	7	8	9	10
x_j^{nom}	1.6e3	-1.4e4	5.5e4	-1.1e5	9.5e4	1.9e4	-1.3e5	1.4e6	-6.9e4	1.3e4
x_j^{rob}	-0.30	5.0	-3.4	-5.1	6.9	5.5	5.3	-7.5	-8.9	13

How it works? NETLIB Case Study

- ♣ We solved the Robust Counterparts of the bad NETLIB problems, assuming interval uncertainty in “ugly coefficients” of *inequality* constraints and *no uncertainty in equations*. It turns out that
 - Reliable solutions do exist, except for 4 cases corresponding to the highest ($\epsilon = 1\%$) perturbation level.
 - The “price of immunization” in terms of the objective value is surprisingly low: when $\epsilon \leq 0.1\%$, it never exceeds 1% and it is less than 0.1% in 13 of 23 cases. Thus, *passing to the robust solutions, we gain a lot in the ability of the solution to withstand data uncertainty, while losing nearly nothing in optimality.*

Problem	Nominal optimal value	Objective at robust solution		
		$\epsilon = 0.01\%$	$\epsilon = 0.1\%$	$\epsilon = 1\%$
80BAU3B	987224.2	987311.8 (+ 0.01%)	989084.7 (+ 0.19%)	1009229 (+ 2.23%)
25FV47	5501.846	5501.862 (+ 0.00%)	5502.191 (+ 0.01%)	5505.653 (+ 0.07%)
ADLITTLE	225495.0		225594.2 (+ 0.04%)	228061.3 (+ 1.14%)
AFIRO	-464.7531		-464.7500 (+ 0.00%)	-464.2613 (+ 0.11%)
BNL2	1811.237		1811.237 (+ 0.00%)	1811.338 (+ 0.01%)
BRANDY	1518.511			1518.581 (+ 0.00%)
CAPRI	1912.621		1912.738 (+ 0.01%)	1913.958 (+ 0.07%)
CYCLE	1913.958	1913.958 (+ 0.00%)	1913.958 (+ 0.00%)	1913.958 (+ 0.00%)
D2Q06C	122784.2	122793.1 (+ 0.01%)	122893.8 (+ 0.09%)	Infeasible
E226	-18.75193			-18.75173 (+ 0.00%)
FFFFF800	555679.6			555715.2 (+ 0.01%)
FINNIS	172791.1	172808.8 (+ 0.01%)	173269.4 (+ 0.28%)	178448.7 (+ 3.27%)
GREENBEA	-72555250	-72526140 (+ 0.04%)	-72192920 (+ 0.50%)	-68869430 (+ 5.08%)
KB2	-1749.900	-1749.877 (+ 0.00%)	-1749.638 (+ 0.01%)	-1746.613 (+ 0.19%)
MAROS	-58063.74	-58063.45 (+ 0.00%)	-58011.14 (+ 0.09%)	-57312.23 (+ 1.29%)
NESM	14076040			14172030 (+ 0.68%)
PEROLD	-9380.755	-9380.755 (+ 0.00%)	-9362.653 (+ 0.19%)	Infeasible
PILOT	-557.4875	-557.4538 (+ 0.01%)	-555.3021 (+ 0.39%)	Infeasible
PILOT4	-64195.51	-64149.13 (+ 0.07%)	-63584.16 (+ 0.95%)	-58113.67 (+ 9.47%)
PILOT87	301.7109	301.7188 (+ 0.00%)	302.2191 (+ 0.17%)	Infeasible
PILOTJA	-6113.136	-6113.059 (+ 0.00%)	-6104.153 (+ 0.15%)	-5943.937 (+ 2.77%)
PILOTNOV	-4497.276	-4496.421 (+ 0.02%)	-4488.072 (+ 0.20%)	-4405.665 (+ 2.04%)
PILOTWE	-2720108	-2719502 (+ 0.02%)	-2713356 (+ 0.25%)	-2651786 (+ 2.51%)
SCFXM1	18416.76	18417.09 (+ 0.00%)	18420.66 (+ 0.02%)	18470.51 (+ 0.29%)
SCFXM2	36660.26	36660.82 (+ 0.00%)	36666.86 (+ 0.02%)	36764.43 (+ 0.28%)
SCFXM3	54901.25	54902.03 (+ 0.00%)	54910.49 (+ 0.02%)	55055.51 (+ 0.28%)
SHARE1B	-76589.32	-76589.32 (+ 0.00%)	-76589.32 (+ 0.00%)	-76589.29 (+ 0.00%)

Objective values for nominal and robust solutions to bad NETLIB problems.

More on Robust LP: Affinely Adjustable Robust Counterpart

♣ The rationale behind the Robust Optimization paradigm as applied to LP is based on two assumptions:

1. Constraints are a “must”: a meaningful solution should satisfy all realizations of the constraints allowed by the uncertainty set.
2. All decision variables should be specified (get numeric values) before the true data becomes known and thus should be independent of the true data.

♣ In many cases, Assumption 2 is too conservative:

A. In dynamical decision-making, only part of decision variables correspond to “here and now” decisions, while the remaining variables represent “wait and see” decisions which are to be made when certain part of the true data is already revealed. A “wait and see” decision can – and should! – depend on the corresponding part of the true data.

B. Some of the decision variables do not correspond to actual decisions at all; they are artificial “analysis variables” introduced to convert the problem into the LP form. These variables can – and should! – depend on the entire true data.

Example: Consider the problem of finding the best $\|\cdot\|_1$ -approximation

$$\min_{x,t} \left\{ t : \sum_i |b_i - \sum_j a_{ij}x_j| \leq t \right\}. \quad (\text{P})$$

When the data are certain, this problem is equivalent to the LP program

$$\min_{x,y,t} \left\{ t : \sum_i y_i \leq t, -y_i \leq b_i - \sum_j a_{ij}x_j \leq y_i \forall i \right\}. \quad (\text{LP})$$

With uncertain data, the Robust Counterpart of (P) becomes the semi-infinite problem

$$\min_{x,t} \left\{ t : \sum_i |b_i - \sum_j a_{ij}x_j| \leq t \forall (b_i, a_{ij}) \in \mathcal{U} \right\},$$

or, which is the same, the problem

$$\min_{x,t} \left\{ t : \forall (b_i, a_{ij}) \in \mathcal{U} \exists y : \sum_i y_i \leq t, -y_i \leq b_i - \sum_j a_{ij}x_j \leq y_i \right\}, \quad (\text{RCP})$$

while the RC of (LP) is the much more conservative problem

$$\min_{x,t} \left\{ t : \exists y : \forall (b_i, a_{ij}) \in \mathcal{U} : \sum_i y_i \leq t, -y_i \leq b_i - \sum_j a_{ij}x_j \leq y_i \right\}. \quad (\text{RCLP})$$

Adjustable Robust Counterpart of an Uncertain LP

♣ Consider an uncertain LP. W.l.o.g., we may assume that the data of this LP are affinely parameterized by a “perturbation vector” ζ running through a given *perturbation set* \mathcal{Z} :

$$\mathcal{LP} = \left\{ \min_x \left\{ c^T[\zeta]x : A[\zeta]x - b[\zeta] \geq 0 \right\} : \zeta \in \mathcal{Z} \right\} \\ \left[c_j[\zeta], A_{ij}[\zeta], b_i[\zeta] \text{ are affine in } \zeta \right]$$

♣ Assume that every decision variable may depend on a given “portion” of the true data. Since the latter is affine in ζ , this assumption says that x_j may depend on $P_j\zeta$, where P_j are given matrices.

- $P_j = 0 \Rightarrow x_j$ is *non-adjustable*: this is an independent of the true data “here and now” decision;
- $P_j \neq 0 \Rightarrow x_j$ is *adjustable*: this is a “wait and see” decision or an analysis variable which may adjust itself – fully or partially, depending on P_j – to the true data.

$$\mathcal{LP} = \left\{ \min_x \{c^T[\zeta]x : A[\zeta]x - b[\zeta] \geq 0\} : \zeta \in \mathcal{Z} \right\}$$

$[c_j[\zeta], A_{ij}[\zeta], b_i[\zeta]]$ are affine in ζ

♣ In our now circumstances, a natural Robust Counterpart of \mathcal{LP} is the problem

Find t and **functions** $\phi_j(\cdot)$ such that the **decision rules** $x_j = \phi_j(P_j\zeta)$ make all the constraints feasible for all perturbations $\zeta \in \mathcal{Z}$, while minimizing the guaranteed value t of the objective:

$$\min_{t, \{\phi_i(\cdot)\}} \left\{ t : \begin{array}{l} \sum_j c_j[\zeta] \phi_j(P_j\zeta) \leq t \quad \forall \zeta \in \mathcal{Z} \\ \sum_j \phi_j(P_j\zeta) A_j[\zeta] - b[\zeta] \geq 0 \quad \forall \zeta \in \mathcal{Z} \end{array} \right\} \quad (\text{ARC})$$

♣ **Very bad news:** The *Adjustable Robust Counterpart*

$$\min_{t, \{\phi_i(\cdot)\}} \left\{ t : \begin{array}{l} \sum_j c_j[\zeta] \phi_j(P_j \zeta) \leq t \forall \zeta \in \mathcal{Z} \\ \sum_j \phi_j(P_j \zeta) A_j[\zeta] - b[\zeta] \geq 0 \forall \zeta \in \mathcal{Z} \end{array} \right\} \quad (\text{ARC})$$

of uncertain LP is an *infinite-dimensional* optimization program and as such typically is absolutely intractable: How could we represent efficiently general-type functions of many variables, not speaking about how to optimize with respect to these functions?

♣ **Remedy (perhaps?):** Let us restrict the decision rules $x_j = \phi_j(P_j \zeta)$ to be easily representable – specifically, *affine* – functions:

$$\phi_j(P_j \zeta) \equiv \mu_j + \nu_j^T P_j \zeta.$$

With this dramatic simplification, (ARC) becomes a *finite-dimensional* (still semi-infinite) *optimization problem in new non-adjustable variables* μ_j, ν_j

$$\min_{t, \{\mu_j, \nu_j\}} \left\{ t : \begin{array}{l} \sum_j c_j[\zeta] (\mu_j + \nu_j^T P_j \zeta) \leq t \forall \zeta \in \mathcal{Z} \\ \sum_j (\mu_j + \nu_j^T P_j \zeta) A_j[\zeta] - b[\zeta] \geq 0 \forall \zeta \in \mathcal{Z} \end{array} \right\} \quad (\text{AARC})$$

♣ We have associated with uncertain LP

$$\mathcal{LP} = \left\{ \min_x \left\{ c^T[\zeta]x : A[\zeta]x - b[\zeta] \geq 0 \right\} : \zeta \in \mathcal{Z} \right\} \\ \left[c_j[\zeta], A_{ij}[\zeta], b_i[\zeta] \text{ are affine in } \zeta \right]$$

and the “information matrices” P_1, \dots, P_n the *Affinely Adjustable Robust Counterpart*

$$\min_{t, \{\mu_j, \nu_j\}} \left\{ t : \begin{array}{l} \sum_j c_j[\zeta](\mu_j + \nu_j^T P_j \zeta) \leq t \forall \zeta \in \mathcal{Z} \\ \sum_j (\mu_j + \nu_j^T P_j \zeta) A_j[\zeta] - b[\zeta] \geq 0 \forall \zeta \in \mathcal{Z} \end{array} \right\} \quad (\text{AARC})$$

♠ **Relatively good news:**

A. AARC is by far more flexible than the usual (non-adjustable) RC of \mathcal{LP} .

B. As compared to ARC, AARC has much more chances to be computationally tractable:

— *With “fixed recourse”*, where the coefficients of adjustable variables are certain, *AARC has the same tractability properties as RC: If the perturbation set \mathcal{Z} is CQR (or polyhedrally representable), (AARC) is equivalent to an explicit CQ (resp., LP) program.*

— *In the general case*, (AARC) may be computationally intractable; however, under mild assumptions on the perturbation set, (AARC) admits “tight” computationally tractable approximation.

Example: Simple Inventory Model. There is a single-product inventory system with

- a single warehouse which should at any time store at least V_{\min} and at most V_{\max} units of the product;
- *uncertain* demands d_t of periods $t = 1, \dots, T$ known to vary within given bounds:

$$d_t \in [d_t^*(1 - \theta), d_t^*(1 + \theta)], t = 1, \dots, T$$

(θ is the uncertainty level). *No backlogged demand is allowed!*

- I factories from which the warehouse can be replenished:
 - at the beginning of period t , you may order $p_{t,i}$ units of product from factory i . Your orders should satisfy the constraints

$$0 \leq p_{t,i} \leq P_i(t) \quad \text{[bounds on orders per period]}$$

$$\sum_t p_{t,i} \leq Q_i \quad \text{[bounds on cumulative orders]}$$

- there is no delivery delay
- order $p_{t,i}$ costs you $c_i(t)p_{t,i}$.

The goal is *to minimize the total cost of the orders*.

♠ *With certain demand*, the problem can be modelled as the LP program

$$\min_{\substack{p_{t,i}, i \leq I, t \leq T, \\ v_t, 2 \leq t \leq T+1}} \sum_{t,i} c_i(t) p_{t,i} \quad \text{[total cost]}$$

s.t.

$$\begin{aligned} v_{t+1} - v_t - \sum_i p_{t,i} &= -d_t, \quad t = 1, \dots, T && \left[\begin{array}{l} \text{state equations. } v_t: \text{ inventory level} \\ \text{at the beginning of day } t \text{ (} v_1 \text{ is given)} \end{array} \right] \\ V_{\min} \leq v_t \leq V_{\max}, \quad 2 \leq t \leq T+1 &&& \text{[bounds on states]} \\ 0 \leq p_{t,i} \leq P_i(t), \quad i \leq I, t \leq T &&& \text{[bounds on orders]} \\ \sum_t p_{t,i} \leq Q_i, \quad i \leq I &&& \left[\begin{array}{l} \text{cumulative bounds} \\ \text{on orders} \end{array} \right] \end{aligned}$$

♠ *With uncertain demand*, it is natural to assume that the orders $p_{t,i}$ may depend on the demands of the preceding periods $1, \dots, t-1$. The *analysis variables* v_t are allowed to depend on the entire true data; in fact, it suffices to allow for v_t to depend on d_1, \dots, d_{t-1} .

• Applying the AARC methodology, we make $p_{t,i}$ and v_t affine functions of past demands:

$$\begin{aligned} p_{t,i} &= \phi_{t,i}^0 + \sum_{1 \leq \tau < t} \phi_{t,i}^\tau d_\tau \\ v_t &= \psi_t^0 + \sum_{1 \leq \tau < t} \psi_t^\tau d_\tau \end{aligned}$$

♣ The AARC is the following *semi-infinite* LP in *non-adjustable* design variables ϕ 's and ψ 's:

$$\begin{aligned}
 & \min_{C, \phi_{t,i}^\tau, \psi_t^\tau} C \\
 \text{s.t.} \quad & \sum_{t,i} c_i(t) [\phi_{t,i}^0 + \sum_{1 \leq \tau < t} \phi_{t,i}^\tau d_\tau] \leq C \\
 & [\psi_{t+1}^0 + \sum_{\tau=1}^t \psi_{t+1}^\tau d_\tau] - [\psi_t^0 + \sum_{\tau=1}^{t-1} \psi_t^\tau d_\tau] - \sum_i [\phi_{t,i}^0 + \sum_{\tau=1}^{t-1} \phi_{t,i}^\tau d_\tau] = -d_t \\
 & V_{\min} \leq [\psi_t^0 + \sum_{\tau=1}^{t-1} \psi_t^\tau d_\tau] \leq V_{\max} \\
 & 0 \leq [\phi_{t,i}^0 + \sum_{\tau=1}^{t-1} \phi_{t,i}^\tau d_\tau] \leq P_i(t) \\
 & \sum_t [\phi_{t,i}^0 + \sum_{\tau=1}^{t-1} \phi_{t,i}^\tau d_\tau] \leq Q_i
 \end{aligned}$$

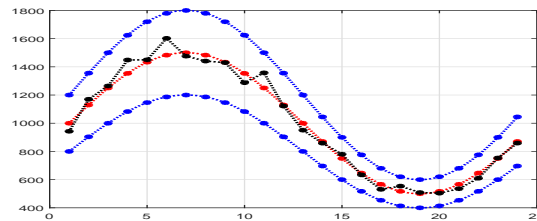
The constraints should be valid for all values of “free” indices and all demand realizations $d = \{d_t\}_{t=1}^T$ from the “demand uncertainty box”

$$\mathcal{D} = \{d : d_t^*(1 - \theta) \leq d_t \leq d_t^*(1 + \theta), 1 \leq t \leq T\}.$$

♣ The AARC can be straightforwardly converted to a usual LP and easily solved.

♣ In the numerical illustration to follow:

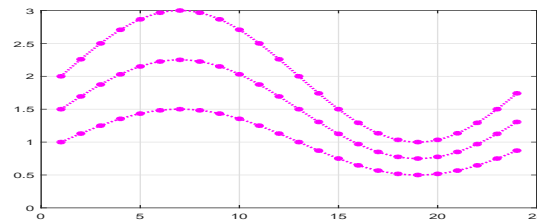
- the planning horizon is $T = 24$
- there are $I = 3$ factories with per period capacities $P_i(t) = 567$ and cumulative capacities $Q_i = 13600$
- the nominal demand d_t^* is seasonal:



$$d_t^* = 1000 \left(1 + 0.5 \sin \left(\frac{\pi(t-1)}{12} \right) \right)$$

demand trajectories: nominal, extreme, sample

- the per-unit ordering costs $c_i(t)$ also are seasonal:



per-unit ordering costs of factories vs. t

$$c_i(t) = c_i \left(1 + 0.5 \sin \left(\frac{\pi(t-1)}{12} \right) \right), \quad c_1 = 1, c_2 = 1.5, c_3 = 2$$

- $v_1 = V_{\min} = 500, V_{\max} = 2000$
- demand uncertainty $\theta = 20\%$

♣ *Results:*

- The AARC optimal value is 35542.

Note: The *non-adjustable* RC is infeasible even at 5% uncertainty level!

- With uniformly distributed in the range $\pm 20\%$ demand perturbations, the average, over 100 simulations, AARC management cost is 35121.

Note: Over the same 100 simulations, the average “utopian” management cost (optimal for *a priori known* demand trajectories) is 33958, i.e., is by just 3.5% (!) less than the average AARC management cost.

Comparison with Dynamic Programming. *When applicable, DP is **the** technique for dynamical decision-making under uncertainty – in (worst-case-oriented) DP, one solves the Adjustable Robust Counterpart of uncertain LP in question, with no ad hoc simplifications like “let us restrict ourselves with affine decision rules”.*

Unfortunately, DP suffers from “curse of dimensionality” – with DP, the computational effort blows up rapidly as the state dimension of the dynamical process grows. Usually state dimension 4 is already “too big”.

Note: There is no “curse of dimensionality” in AARC!

- In our toy Inventory model, the state dimension is 4 (what matters for the future, is the current amount of product at the warehouse and 3 remaining cumulative capacities of the 3 factories). Thus, DP is hardly applicable.

- However, reducing the number of factories to 1, increasing the per period capacity of the remaining factory to 1800 and making its cumulative capacity $+\infty$, we reduce the state dimension to 1 and make DP easily implementable. With this setup,

- the DP (that is, the “absolutely best”) optimal value is 31270

- the *computed* AARC optimal value is 31514 – just by 0.8% worse! In fact, 0.8% is due to rounding errors — it was shown [Bertsimas,Iancu,Parrilo’09] *that in the case in question the ARC and the AARC optimal values are the same!*

Whether Conic Quadratic Programming exists? Fast Polyhedral Approximation of Lorentz Cone

♠ **Fact:** The canonical polyhedral representation $X = \{x \in \mathbf{R}^n : Ax \leq b\}$ of the projection

$$X = \{x : \exists u : Px + Qu \leq r\}$$

of a polyhedral set $X^+ = \{[x; u] : Px + Qu \leq r\}$ given by a moderate number of linear inequalities in variables x, u can require a huge number of linear inequalities in variables x .

Question: Can we use this phenomenon in order to *approximate* to high accuracy a non-polyhedral set $X \subset \mathbf{R}^n$ by projecting onto \mathbf{R}^n a higher-dimensional *polyhedral and simple* (given by a moderate number of linear inequalities) set X^+ ?

♠ The outlined possibility does exist when X is the Lorentz cone.

Theorem: For every n and every ϵ , $0 < \epsilon < 1/2$, one can point out a polyhedral set \mathbf{L}^+ given by an explicit system of homogeneous linear inequalities in variables $x \in \mathbf{R}^n$, $t \in \mathbf{R}$, $w \in \mathbf{R}^k$:

$$\mathbf{L}^+ = \{[x; t; w] : Px + tp + Qw \leq 0\} \quad (!)$$

such that

- the number of inequalities in the system ($\approx 2n \ln(1/\epsilon)$) and the dimension of the slack vector w ($\approx 0.7n \ln(1/\epsilon)$) do not exceed $O(1)n \ln(1/\epsilon)$
- the projection

$$\mathbf{L} = \{[x; t] : \exists w : Px + tp + Qw \leq 0\}$$

of \mathbf{L}^+ on the space of x, t -variables is in-between the Second Order Cone and $(1 + \epsilon)$ -extension of this cone:

$$\mathbf{L}^{n+1} := \{[x; t] \in \mathbf{R}^{n+1} : \|x\|_2 \leq t\} \subset \mathbf{L} \subset \mathbf{L}_\epsilon^{n+1} := \{[x; t] \in \mathbf{R}^{n+1} : \|x\|_2 \leq (1 + \epsilon)t\}.$$

In particular, we have

$$B_n^1 \subset \{x : \exists w : Px + p + Qw \leq 0\} \subset B_n^{1+\epsilon}$$

$$B_n^r = \{x \in \mathbf{R}^n : \|x\|_2 \leq r\}$$

Note: When $\epsilon = 1.e-17$, a usual computer does not distinguish between $r = 1$ and $r = 1 + \epsilon$. Thus, *for all practical purposes*, the n -dimensional Euclidean ball admits polyhedral representation with $\approx 28n$ variables w and $\approx 79n$ linear inequality constraints.

Note: A straightforward representation $X = \{x : Ax \leq b\}$ of a polyhedral set X satisfying

$$B_n^1 \subset X \subset B_n^{1+\epsilon}$$

requires at least $N = O(1)\epsilon^{-\frac{n-1}{2}}$ linear inequalities. With $n = 100$, $\epsilon = 0.01$, we get

$$N \geq 3.0e85 \approx 300,000 \times [\# \text{ of atoms in universe}]$$

With “fast polyhedral approximation” of B_n^1 , a 0.01-approximation of B_{100} requires just 922 linear inequalities on 100 original and 325 additional variables.

♣ With fast polyhedral approximation of the cone $\mathbf{L}^{n+1} = \{[x; t] \in \mathbf{R}^{n+1} : \|x\|_2 \leq t\}$, Conic Quadratic Optimization programs “for all practical purposes” become LO programs. For example, by what we know about CQr functions/sets, the program

$$\begin{aligned} & \text{minimize } c^T x \text{ subject to} \\ & Ax = b \\ & x \geq 0 \\ & \left(\sum_{i=1}^8 |x_i|^3 \right)^{1/3} \leq x_2^{1/7} x_3^{2/7} x_4^{3/7} + 2x_1^{1/5} x_5^{2/5} x_6^{1/5} \\ & 5x_2 \geq \frac{1}{x_1^{1/2} x_2} + \frac{2}{x_2^{1/3} x_3^3 x_4^{5/8}} \end{aligned}$$

can be *in a systematic fashion* converted to Conic Quadratic Programming and thus “for all practical purposes” is just an LP program.

Building Fast Polyhedral Approximation

♣ **Goal:** To *nearly* represent by linear inequalities the set

$$\mathbf{L}^{n+1} = \{[x_1; \dots; x_n; t] : \sqrt{x_1^2 + \dots + x_n^2} \leq t\}$$

that is, to find a polyhedrally represented set

$$\hat{\mathbf{L}} = \{[x = [x_1; \dots; x_n; t] : \exists w : Px + tp + Qw \leq 0\}$$

such that

$$\mathbf{L}^{n+1} \subset \hat{\mathbf{L}} \subset \mathbf{L}_\epsilon^{n+1},$$

$$\mathbf{L}_\epsilon^{n+1} = \{[x_1; \dots; x_n; t] : \sqrt{x_1^2 + \dots + x_n^2} \leq (1 + \epsilon)t\}$$

• $\epsilon > 0$: given tolerance.

♠ **Observation:** *It suffices to solve our problem when $n = 2$.*

Reason: Inequality $\sqrt{x_1^2 + \dots + x_n^2} \leq t$ can be represented by a system of similar inequalities with 3 variables in each.

Example: To represent the set

$$\mathbf{L}^6 = \{[x; t] \in \mathbf{R}^6 : \sqrt{x_1^2 + x_2^2 + \dots + x_5^2} \leq t\},$$

by a system of constraints of the form $\sqrt{p^2 + q^2} \leq r$, we

♠ add to x, t variable w_1 and write down the system

$$\sqrt{x_4^2 + x_5^2} \leq w_1, \sqrt{x_1^2 + x_2^2 + x_3^2 + w_1^2} \leq t$$

- the system does represent \mathbf{L}^6 – the projection of its solution set on the space of x, t -variables is exactly \mathbf{L}^6

- the “sizes” (# of variables involved) of the constraints in the system are ≤ 5 , while the size of the constraint in the original description of \mathbf{L}^6 was 6.

♠ add to x, t, w_1 variable w_2 and write down the system

$$\sqrt{x_4^2 + x_5^2} \leq w_1, \sqrt{x_3^2 + w_1^2} \leq w_2, \sqrt{x_1^2 + x_2^2 + w_2^2} \leq t$$

This system still represents \mathbf{L}^6 , and the maximal size of its constraints is 4.

♠ add to x, t, w_1, w_2 variable w_3 and write down the system

$$\sqrt{x_4^2 + x_5^2} \leq w_1, \sqrt{x_3^2 + w_1^2} \leq w_2, \sqrt{x_2^2 + w_2^2} \leq w_3, \sqrt{x_1^2 + w_3^2} \leq t$$

This system represents \mathbf{L}^6 , and all its constraints are of the form $\sqrt{p^2 + q^2} \leq r$. We are done.

Note: The above recipe clearly extends from the 6-dimensional case to the general one. Representing \mathbf{L}^{n+1} via constraints of the form $\sqrt{p^2 + q^2} \leq r$ requires $n - 2$ additional variables and $n - 1$ constraints.

Note: The number of steps in the latter procedure can be reduced from $n - 2$ to $\text{Ceil}(\log_2(n)) - 1$ by using the same construction as when building CQR of the set $\{(t, x_1, \dots, x_{2^\mu}) \geq 0 : t \leq (x_1, \dots, x_{2^\mu})^{1/2^\mu}\}$; the resulting number of constraints of the form $\sqrt{p^2 + q^2} \leq r$ and of additional variables still are (at most) $n - 1$ and $n - 2$ respectively.

♠ **Conclusion:** In order to find a tight polyhedral approximation of

$$\mathbf{L}^{n+1} = \{[x_1; \dots; x_n; t] : \sqrt{x_1^2 + \dots + x_n^2} \leq t\},$$

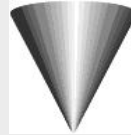
we can

- represent the constraint $\sqrt{x_1^2 + \dots + x_n^2} \leq t$ by a system of inequalities of the form $\sqrt{p^2 + q^2} \leq r$
- replace every one of the resulting constraints by its tight polyhedral approximation.

Note: We should account for “accumulation of errors.” This is an easy task...

Fast polyhedral approximation of

$$\mathbf{L}^3 = \{[p; q; r] : \sqrt{p^2 + q^2} \leq r\}$$



“Ice-cream” cone \mathbf{L}^3

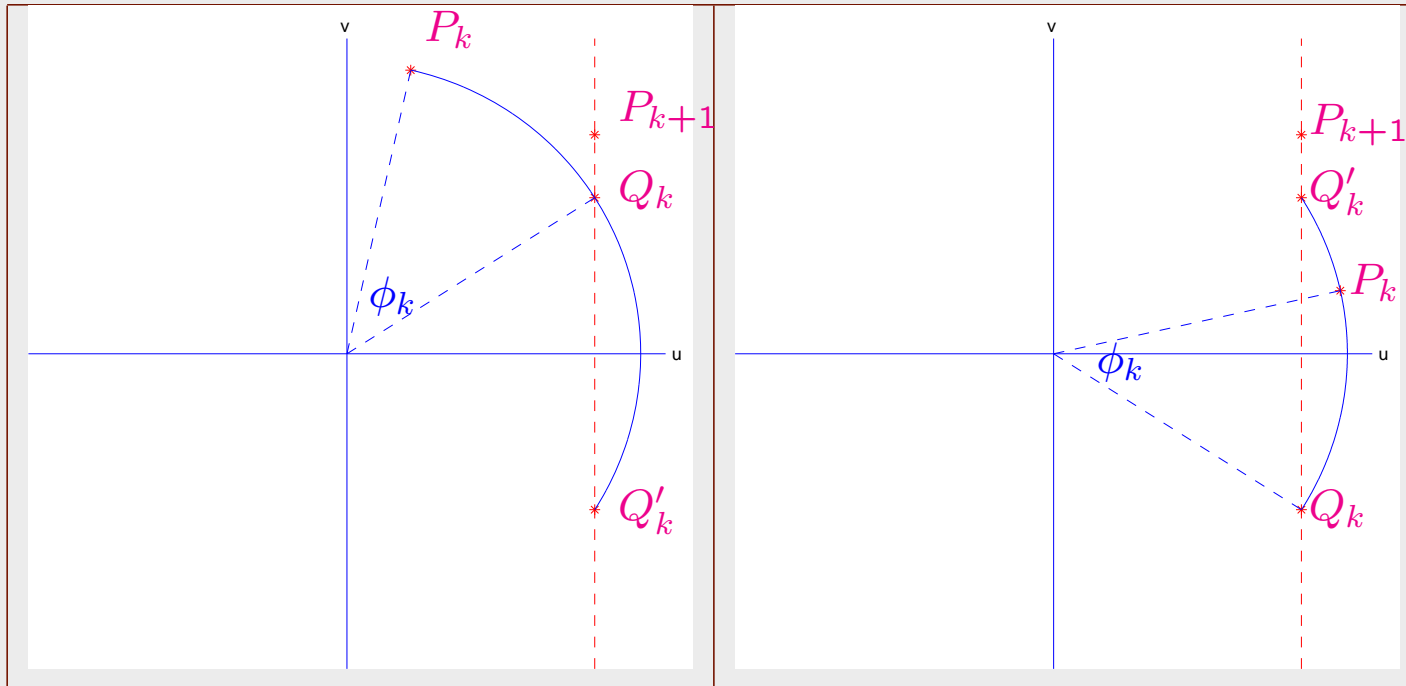
♠ Given variables p, q, r , we choose a positive integer K , and consider $K + 1$ points P_1, \dots, P_{K+1} on the 2D plane as follows.

- The first point $P_1 = [u_1; v_1]$ satisfies

$$u_1 \geq |p|, v_1 \geq |q|$$

which can be represented by a system of 4 linear constraints in variables p, q, u_1, v_1 .

- The relation between $P_k = [u_k; v_k]$ and $P_{k+1} = [u_{k+1}; v_{k+1}]$ is as follows.
 - we rotate P_k clockwise by the angle $\phi_k = \pi/2^{k+1}$, thus getting a point Q_k .
 - we reflect Q_k w.r.t. the u -axis, thus getting point Q'_k .
 - we impose on $P_{k+1} = [u_{k+1}; v_{k+1}]$ the restriction to belong to the vertical line passing through Q_k and Q'_k and to be not lower than Q_k and Q'_k .



♠ **Note:** Relations between $P_k = [u_k; v_k]$ and $P_{k+1} = [u_{k+1}; v_{k+1}]$ amount to a system of linear constraints

$$u_{k+1} = \cos(\phi_k)u_k + \sin(\phi_k)v_k$$

right hand side: u -coordinate of Q_k and Q'_k

$$v_{k+1} \geq -\sin(\phi_k)u_k + \cos(\phi_k)v_k$$

right hand side: v -coordinate of Q_k

$$v_{k+1} \geq \sin(\phi_k)u_k - \cos(\phi_k)v_k$$

right hand side: v -coordinate of Q'_k

in variables $u_k, v_k, u_{k+1}, v_{k+1}$.

♠ Let us write down all built so far constraints on original and additional variables

u_1	\geq	p
u_1	\geq	$-p$
v_1	\geq	q
v_2	\geq	$-q$
u_{k+1}	$=$	$\cos(\phi_k)u_k + \sin(\phi_k)v_k$
v_{k+1}	\geq	$-\sin(\phi_k)u_k + \cos(\phi_k)v_k$
v_{k+1}	\geq	$\sin(\phi_k)u_k - \cos(\phi_k)v_k$
$k = 1, \dots, K$		

and augment this system by the requirement for P_{K+1} to be close to the segment $[0, r]$ of the u -axis:

$$0 \leq u_{K+1} \leq r, \quad 0 \leq v_{K+1} \leq \tan(\phi_K) \cdot r$$

Observation 1: *When p, q, r can be augmented by properly selected u 's and v 's to satisfy the above constraints, we have*

$$\sqrt{p^2 + q^2} \leq r\sqrt{1 + \tan^2(\phi_K)}$$

Indeed, by the above constraints on p, q, r and the additional variables, the points $P_k = [u_k; v_k]$ satisfy

$$\|[p; q]\|_2 \leq \|P_1\|_2 \leq \dots \leq \|P_{K+1}\|_2 = \sqrt{u_{K+1}^2 + v_{K+1}^2} \leq r\sqrt{1 + \tan^2(\phi_K)}.$$

u_1	\geq	p
u_1	\geq	$-p$
v_1	\geq	q
v_2	\geq	$-q$
u_{k+1}	$=$	$\cos(\phi_k)u_k + \sin(\phi_k)v_k$
v_{k+1}	\geq	$-\sin(\phi_k)u_k + \cos(\phi_k)v_k$
v_{k+1}	\geq	$\sin(\phi_k)u_k - \cos(\phi_k)v_k$
		$k = 1, \dots, K$
$0 \leq u_{K+1} \leq r, 0 \leq v_{K+1} \leq \tan(\phi_K) \cdot r$		

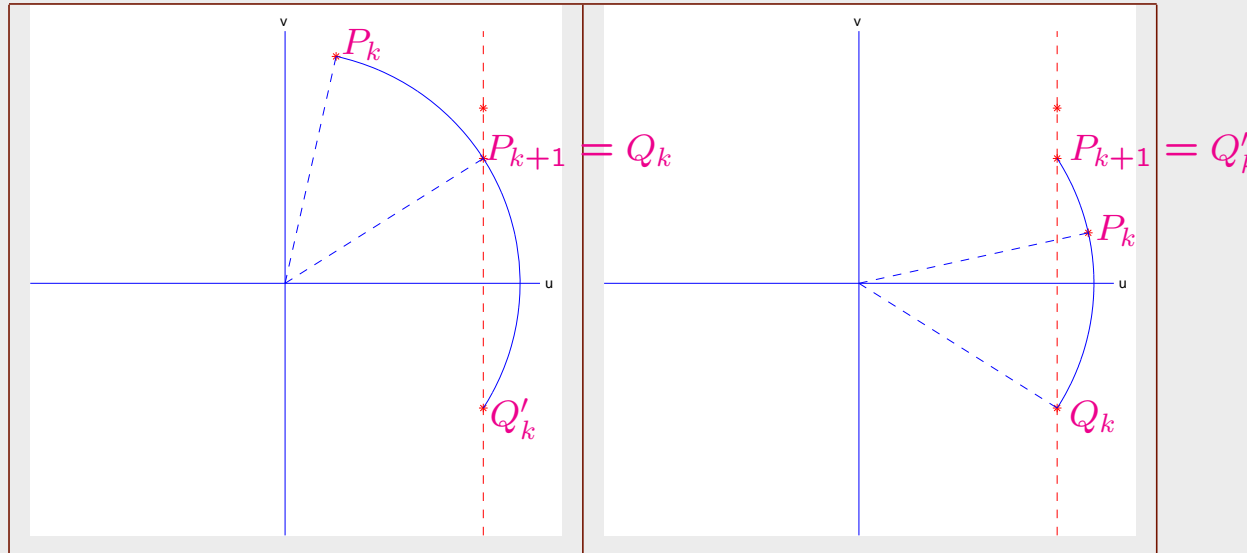
Observation 2: When $\sqrt{p^2 + q^2} \leq r$, p, q, r indeed can be augmented by u 's and v 's to satisfy our constraints.

This combines with Observation 1 to imply that *the projection of the polyhedral set given by our constraints onto the space of p, q, r variables is in-between the \mathbf{L}^3 and $\mathbf{L}_{\delta_K}^3$, with*

$$\begin{aligned}\delta_K &= \sqrt{1 + \tan^2(\phi_K)} - 1 \\ &= \sqrt{1 + \tan^2\left(\frac{\pi}{2^{K+1}}\right)} - 1 \leq \frac{\pi^2}{2^{2K+2}}.\end{aligned}$$

\Rightarrow To make $\delta_K \leq \epsilon$, we need just $O(1)\ln(1/\epsilon)$ additional variables and linear constraints!

♠ To justify Observation 2, let us augment p , q with u 's and v 's which “rigidly” satisfy the magenta constraints, specifically, let us set $u_1 = |p|$, $v_1 = |q|$, and let P_{k+1} be the “highest” of the points Q_k, Q'_k :



Then

$$r \geq \sqrt{p^2 + q^2} = \|[p; q]\|_2 = \|P_1\|_2 = \dots = \|P_{K+1}\|_2$$

and *the angle between P_{k+1} and the nonnegative ray of the u -axis does not exceed $\phi_k = \frac{\pi}{2^{k+1}}$.*

$\Rightarrow P_{K+1} = [u_{K+1}, v_{K+1}]$ indeed satisfies

$$0 \leq u_{K+1} \leq r \text{ and } 0 \leq v_{K+1} \leq \tan(\phi_K) \cdot r.$$

♥ To justify the claim on the angles, observe that with our “rigid” construction of P_1, \dots, P_{K+1} ,

- P_1 lives in the first quadrant, and P_2 is obtained from P_1 by rotating clockwise by the angle $\phi_1 = \pi/4$ (and, perhaps, reflecting the result w.r.t. the u -axis to bring it to the first quadrant).

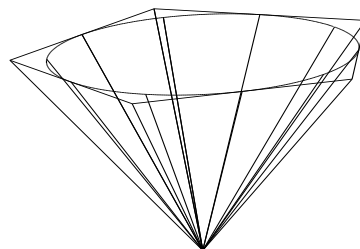
After rotation, the angle between the point and the u -axis does not exceed $\pi/4$, and reflection, if any, keeps this angle intact

$\Rightarrow P_2$ lives in the first quadrant and makes angle at most $\phi_1 = \pi/4$ with the u -axis

$\Rightarrow P_3$, which is obtained from P_2 by rotating clockwise by the angle $\phi_2 = \pi/8$ (and, perhaps, reflecting the result w.r.t. u -axis to bring it to the first quadrant), lives in the first quadrant and makes the angle at most $\phi_2 = \pi/8$ with the u -axis

$\Rightarrow \dots \Rightarrow P_{K+1}$ lives in the first quadrant and makes angle at most $\phi_K = \frac{\pi}{2^{K+1}}$ with the u -axis.

♣ The simplest way to build a polyhedral approximation of the Lorentz cone is to take the tangent planes along a “fine” finite grid of generators and to use, as the approximation, the resulting polyhedral cone:



This approach is a complete failure: the number of tangent planes required to get an 0.5-approximation of \mathbf{L}^m is at least

$$N = \sqrt{2\pi(m-2)} \exp\{m/6\},$$

which is $> 429,481,377$ for $m = 100$.

♣ With our approach, we approximate \mathbf{L}^m by a *projection of a higher-dimensional polyhedron*. When projecting an N -dimensional polyhedron onto a plane of dimension $\ll N$, the number of facets may grow up exponentially, so that a low-dimensional projection of a “simple” high-dimensional polyhedron may have astronomically many facets. With our approach, we build a family of polyhedral cones $P^{m,k} \subset \mathbf{R}^{O(mk)}$ given by just $O(mk)$ linear inequalities, while their projections $\hat{P}^{m,k}$ on \mathbf{R}^m have enough facets to approximate \mathbf{L}^m within accuracy $\exp\{-O(k)\}$:

- ♣ Approximating sets by *projections of higher-dimensional polyhedral sets* we can dramatically reduce the “size” of approximation. For example,
- *When approximating the unit 2D circle by a projection of a higher-dimensional polytope P , we can get approximations as follows:*
 - with P given by 12 inequalities in 10 variables – accuracy $5.e-3$, as good as circumscribed polygon with 16 sides
 - with P given by 18 inequalities in 13 variables – accuracy $3.e-4$, as good as circumscribed polygon with 127 sides
 - with P given by 30 inequalities in 19 variables – accuracy $7.e-8$, as good as circumscribed polygon with 8,192 sides
 - with P given by 54 inequalities in 31 variables – accuracy $4.e-15$, as good as circumscribed polygon with 34,200,933 sides

♠ Polyhedral approximation of \mathbf{L}^m is basically the same as polyhedral approximation of m -dimensional Euclidean ball

$$\mathbf{B}_m = \{x \in \mathbf{R}^m : \|x\|_2 \leq 1\}.$$

There is a less sophisticated way to approximate Euclidean balls by projections of polyhedral sets:

Theorem [Lindenstrauss-Johnson]: *For two positive integers N, n with $N \geq 10n$, random n -dimensional projection of N -dimensional unit box – the set*

$$B = \{x \in \mathbf{R}^n : \exists y \in \mathbf{R}^N : x = Ay, -1 \leq y_1, \dots, y_N \leq 1\}$$

[A : drawn at random from Gaussian distribution]

with probability approaching one as N, n grow, is in-between two n -dimensional Euclidean balls with the ratio of radii $(1 + O(\sqrt{n/N}))$.

This result has tremendous theoretical implications. However,
— no *individual* matrices A yielding “nearly round” B are known (pity! these matrices would be ideally suited for Compressed Sensing)

Note: *Our fast polyhedral approximation is explicit!*

— to make B an ϵ -approximation of B_n , you need $N = O(1/\epsilon^2)n$

Note: With fast polyhedral approximation, you need much smaller N : $N = O(\ln(1/\epsilon))n$

♠ **Open question:** With fast polyhedral approximation, *centrally symmetric* ball \mathbf{B}_n is ϵ -approximated by the projection of a *highly asymmetric* polyhedron of dimension $N = O(\ln(1/\epsilon))n$ given by $M = O(N)$ linear inequalities. *Is it possible to make this higher-dimensional polyhedron centrally symmetric, preserving the type of dependence of N, M on n and ϵ ?*

III. SEMIDEFINITE PROGRAMMING

Preliminaries

- As a linear space, the space $\mathbf{R}^{m \times n}$ of $m \times n$ matrices can be identified with \mathbf{R}^{mn} by writing columns of a matrix one beneath another:

$$A = [a_{ij}]_{\substack{i=1,\dots,m \\ j=1,\dots,n}} \leftrightarrow \text{Vec}(A) = [a_{11}; a_{21}; \dots; a_{m1}; a_{12}; \dots; a_{m2}; \dots; a_{1n}; \dots; a_{mn}]$$

The inner product of matrices induced by this representation (a.k.a. *Frobenius* inner product) is

$$\langle A, B \rangle \equiv \sum_{i,j} A_{ij} B_{ij} = \text{Tr}(A^T B) = \text{Tr}(AB^T) \quad [A, B \in \mathbf{R}^{m \times n}]$$
$$\left[\text{Tr}(C) = \sum_{i=1}^n C_{ii}, C \in \mathbf{R}^{n \times n}, \text{ is the trace of } C \right]$$

- In particular, the space \mathbf{S}^m of $m \times m$ *symmetric* matrices equipped with the inner product inherited from $\mathbf{R}^{m \times m}$:

$$\langle A, B \rangle \equiv \sum_{i,j} A_{ij} B_{ij} = \text{Tr}(A^T B) = \text{Tr}(AB)$$

is a Euclidean space ($\dim \mathbf{S}^m = \frac{m(m+1)}{2}$).

- A matrix $A \in \mathbf{S}^m$ is called *positive semidefinite* (notation: $A \succeq 0$) if the quadratic form associated with A is nonnegative everywhere:

$$A \succeq 0 \Leftrightarrow \xi^T A \xi \geq 0 \quad \forall \xi \in \mathbf{R}^m,$$

When $\xi^T A \xi > 0$ whenever $\xi \neq 0$, A is called *positive definite* ($A \succ 0$).

- The *positive semidefinite* $m \times m$ matrices form a cone \mathbf{S}_+^m (closed, convex, pointed and with a nonempty interior) in \mathbf{S}^m :

$$\mathbf{S}_+^m = \{A \in \mathbf{S}^m : \xi^T A \xi \geq 0 \quad \forall \xi \in \mathbf{R}^m\}$$

The interior of this cone is comprised of positive definite matrices:

$$\text{int } \mathbf{S}_+^m = \{A \in \mathbf{S}^m : A \succ 0\}$$

$$\mathbf{S}_+^m = \left\{ A \in \mathbf{S}^m : \xi^T A \xi \geq 0 \ \forall \xi \in \mathbf{R}^m \right\}$$

• **Equivalent descriptions of \mathbf{S}_+^m :** an $m \times m$ matrix A is positive semidefinite

— iff A is symmetric ($A = A^T$) and all its eigenvalues are nonnegative;

— iff A can be decomposed as $A = D^T D$

— iff A can be represented as a sum of symmetric dyadic matrices:

$$A = \sum_j d_j d_j^T;$$

— iff $A = U^T \Lambda U$ with orthogonal U and diagonal Λ , the diagonal entries of Λ being nonnegative;

— iff A is symmetric ($A = A^T$) and all principal minors of A are nonnegative. In particular,

$$0 \preceq \begin{bmatrix} a & b \\ b & c \end{bmatrix} \in \mathbf{S}^2 \Leftrightarrow a \geq 0 \ \& \ c \geq 0 \ \& \ ac - b^2 \geq 0.$$

• As every regular cone, \mathbf{S}_+^m defines a “good” partial ordering on \mathbf{S}^m :

$$\begin{aligned} A \succeq B &\Leftrightarrow A - B \succeq 0 \Leftrightarrow \xi^T A \xi \geq \xi^T B \xi \quad \forall \xi \\ &\left[A = A^T, B = B^T \text{ are of the same size} \right] \end{aligned}$$

- **Useful observation:** *Validity of \succeq inequality is preserved when multiplying both sides by a matrix Q from the left and by Q^T from the right:*

$$A \succeq B \Rightarrow Q^T A Q \succeq Q^T B Q \quad \left[A, B \in \mathbf{S}^m, Q \in \mathbf{R}^{m \times k} \right]$$

Indeed,

$$\left\{ \xi^T A \xi \geq \xi^T B \xi \quad \forall \xi \right\} \Rightarrow \left\{ \eta^T Q^T A \underbrace{Q \eta}_{\xi} \geq \eta^T Q^T B Q \eta \quad \forall \eta \right\}$$

- **Useful observation:** *When A and B are rectangular matrices such that $\text{Tr}(AB)$ is well defined (i.e., AB is well defined and square), we have*

$$\text{Tr}(AB) = \text{Tr}(BA).$$

Warning: *The above observation does **not** mean that the trace of the product of several matrices is independent of the order of factors! In general, $\text{Tr}(ABC)$ is **not** the same as $\text{Tr}(BAC)$. The above observation says only that *if $\text{Tr}(ABC)$ makes sense* (i.e., ABC is a square matrix), *then $\text{Tr}(ABC) = \text{Tr}(BCA) = \text{Tr}(CAB)$.**

- **Observation:** The semidefinite cone is self-dual:

$$(\mathbf{S}_+^m)_* \equiv \left\{ A \in \mathbf{S}^m : \text{Tr}(AB) \geq 0 \ \forall B \in \mathbf{S}_+^m \right\} = \mathbf{S}_+^m.$$

Indeed,

$$\xi^T A \xi = \text{Tr}(\xi^T A \xi) = \text{Tr}(A \xi \xi^T)$$

It follows that *if* $A \in \mathbf{S}^m$ is such that $\text{Tr}(AB) \geq 0$ for all $B \succeq 0$, *then* $A \succeq 0$:

$$\xi \in \mathbf{R}^m \Rightarrow B = \xi \xi^T \succeq 0 \Rightarrow \text{Tr}(AB) = \xi^T A \xi \geq 0$$

Vice versa, *if* $A \succeq 0$, *then* $\text{Tr}(AB) \geq 0$ for all $B \succeq 0$:

$$B \succeq 0 \Rightarrow B = \sum_j d_j d_j^T \Rightarrow \text{Tr}(AB) = \sum_j \text{Tr}(A d_j d_j^T) = \sum_j d_j^T A d_j \geq 0.$$

Semidefinite program

- A *semidefinite program* is a conic program associated with the semidefinite cone:

$$\min_{x \in \mathbf{R}^n} \left\{ c^T x : \mathcal{A}x - B \succeq 0 \quad \left[\Leftrightarrow \mathcal{A}x - B \succeq_{\mathbf{S}_+^m} 0 \right] \right\}$$
$$\left[\mathcal{A}x = \sum_{i=1}^n x_i A_i, \quad A_i \in \mathbf{S}^m \right]$$

A constraint of the type

$$x_1 A_1 + \dots + x_n A_n \succeq B$$

with variables x_1, \dots, x_n is called an *LMI* – Linear Matrix Inequality. Thus, a semidefinite program is to minimize a linear objective under an LMI constraint.

- **Observation:** A *system* of LMI constraints

$$\mathcal{A}_i(x) := \sum_j x_j A_{ij} - B_i \succeq 0, \quad i = 1, \dots, m$$

is equivalent to *single* LMI constraint

$$\text{Diag}\{\mathcal{A}_1(x), \dots, \mathcal{A}_m(x)\} \succeq 0.$$

- For notation $\text{Diag}\{\dots\}$, see slide 0.3

Program dual to an SDP program

$$\min_x \left\{ c^T x : \mathcal{A}x - B \equiv \sum_{j=1}^n x_j A_j - B \succeq 0 \right\} \quad (\text{SDPr})$$

According to our general scheme, the problem dual to (SDPr) is

$$\max_Y \{ \langle B, Y \rangle : \mathcal{A}^* Y = c, Y \succeq 0 \} \quad (\text{SDDI})$$

(recall that \mathbf{S}_+^m is self-dual!).

It is easily seen that the operator \mathcal{A}^* conjugate to \mathcal{A} :

$$\forall (x \in \mathbf{R}^n, Y \in \mathbf{S}^m) : \langle \mathcal{A}x, Y \rangle = x^T [\mathcal{A}^* Y]$$

is given by

$$\mathcal{A}^* Y = (\text{Tr}(Y A_1), \dots, \text{Tr}(Y A_n))^T : \mathbf{S}^m \rightarrow \mathbf{R}^n.$$

Consequently, the dual problem is

$$\max_Y \{ \text{Tr}(BY) : \text{Tr}(Y A_i) = c_i, i = 1, \dots, n, Y \succeq 0 \} \quad (\text{SDDI})$$

SDP optimality conditions

$$\min_x \left\{ c^T x : \mathcal{A}x - B \equiv \sum_{j=1}^n x_j A_j - B \succeq 0 \right\} \quad (\text{SDPr})$$

$$\max_Y \left\{ \text{Tr}(BY) : \text{Tr}(A_j Y) = c_j, \ j = 1, \dots, n; \ Y \succeq 0 \right\} \quad (\text{SDDI})$$

- Assume that

(!) both (SDPr) and (SDDI) are essentially strictly feasible,

so that by Conic Duality Theorem both problems are solvable with equal optimal values.

By Conic Duality, the necessary and sufficient condition for a primal-dual feasible pair (x, Y) to be primal-dual optimal is that

$$\text{Tr} \left(\underbrace{[\mathcal{A}x - B]}_{\text{"primal slack" } X} \ Y \right) = 0$$

- For a pair of symmetric positive semidefinite matrices X and Y , one has

$$\text{Tr}(XY) = 0 \Leftrightarrow XY = YX = 0.$$

$$\min_x \left\{ c^T x : \mathcal{A}x - B \equiv \sum_{j=1}^n x_j A_j - B \succeq 0 \right\} \quad (\text{SDPr})$$

$$\max_Y \left\{ \text{Tr}(BY) : \text{Tr}(A_j Y) = c_j, j = 1, \dots, n; Y \succeq 0 \right\} \quad (\text{SDDI})$$

(!) both (SDPr) and (SDDI) are essentially strictly feasible,

- Thus, *under assumption (!) a primal-dual feasible pair (x, Y) is primal-dual optimal iff*

$$[\mathcal{A}x - B]Y = Y[\mathcal{A}x - B] = 0$$

Cf. Linear Programming:

$$(\text{P}): \quad \min_x \left\{ c^T x : Ax - b \geq 0 \right\}$$

$$(\text{D}): \quad \max_y \left\{ b^T y : A^T y = c, y \geq 0 \right\}$$

(x, y) primal-dual optimal



(x, y) primal-dual feasible and $y_j[Ax - b]_j = 0 \quad \forall j$

What can be expressed via SDP?

$$\min_x \{c^T x : x \in X\} \quad (\text{Ini})$$

- A sufficient condition for (Ini) to be equivalent to an SD program is that X is a SDR (“SemiDefinite-representable”) set:

Definition. A set $X \subset \mathbf{R}^n$ is called SDR, if it admits SDR (“SemiDefinite Representation”)

$$X = \{x : \exists u : \mathcal{A}(x, u) \succeq 0\}$$
$$\left[\mathcal{A}(x, u) = \sum_j x_j A_j + \sum_\ell u_\ell B_\ell + C : \mathbf{R}_x^n \times \mathbf{R}_u^k \rightarrow \mathbf{S}^m \right]$$

- Given a SDR of X , we can write down (Ini) equivalently as the semidefinite program

$$\min_{x, u} \{c^T x : \mathcal{A}(x, u) \succeq 0\}.$$

- ♠ Same as in the case of Conic Quadratic Programming, we can
- Define the notion of a SDr function

$$f : \mathbf{R}^n \rightarrow \mathbf{R} \cup \{\infty\}$$

as a function with SDr epigraph:

$$\{(t, x) : t \geq f(x)\} = \left\{ (t, x) : \exists u : \underbrace{\mathcal{A}(t, x, u) \succeq 0}_{\text{LMI}} \right\}$$

and verify that if f is a SDr function, then all its level sets

$$\{x : f(x) \leq a\}$$

are SDr;

- Develop a “calculus” of SDr functions/sets with *exactly* the same combination rules as for CQ-representability.

♠ **Note:** The calculus of CQR's and SDR's is fully algorithmic and can be built into a compiler. This fact is used in CVX

Michael Grant and Stephen Boyd. CVX: Matlab software for disciplined convex programming <http://cvxr.com/cvx>

- CVX is second-to-none in terms of its scope and user-friendliness “go-between” for processing well-structured convex problems reduced to (or well approximated by) SDP.
- CVX gets on input high level MATLAB description of objective and constraints and uses calculus of CQR's and SDR's to recognize that subsequent steps in this description are covered by calculus (this is where *disciplined* comes from). If it is the case, CVX automatically applies calculus rules to end up with SDR's of objective and constraints, and sends the resulting “standard form” SDP to SDP solver.
- The solution found by the solver is then “transformed back” to the original “problem language” and returned to the user.

♠ CVX is extremely convenient. Consider, e.g., the problem of inscribing the largest volume ellipsoid into a polytope $\{x \in \mathbb{R}^n : Ax \leq b\}$.

• **Human formulation:** Given $m \times n$ matrix A with rows a_i^T and $b \in \mathbb{R}^m$, maximize $\text{Det}(X)$ over $X \in \mathbb{S}_+^n$ and $c \in \mathbb{R}^n$ such that $\|Xa_i\|_2 \leq b_i - a_i^T c$, $i \leq i \leq m$.

Explanation: We represent a candidate ellipsoid as $E = \{c + Xu : \|u\|_2 \leq 1\}$ with $X \succeq 0$. The constraints on X and c state that $\max_{\|u\|_2 \leq 1} a_i^T (Xu + c) \leq b_i$ for all i , i.e., that

$E \subset \{x : Ax \leq b\}$, and $\text{Det}(X)$ is proportional to the volume of E .

• **CVX formulation:**

```
[m,n]=size(A)
cvx_begin
variable c(n,1)
variable X(n,n) symmetric
X == semidefinite(n)
for i=1:n
    ai=A(i,:)
    norm(ai*X)+ai*c <= b(i)
end
maximize det_rootn(X)
cvx_end
```

Note: CVX is enough intelligent to know SDR of the SDR function $-\text{Det}^{1/n}(X)$ of $X \in \mathbb{S}_+^n$ ($\text{Det}^{1/n}(X)$ is `det_rootn(X)` in CVX); it knows SDR's of tens of useful SDR functions.

When a function/set is SDr?

Proposition. *Every CQr set/function is SDr as well.*

Proof:

Lemma. *Every direct product of Lorentz cones is SDr.*

Lemma⇒**Proposition:** Let $X \subset \mathbb{R}^n$ be CQr:

$$X = \{x \mid \exists u : A(x, u) \in \mathbf{K}\},$$

\mathbf{K} being a direct product of Lorentz cones and $A(x, u)$ being affine.
By Lemma,

$$\mathbf{K} = \{y : \exists v : \mathcal{B}(y, v) \succeq 0\}$$

with affine $\mathcal{B}(\cdot, \cdot)$. It follows that

$$X = \left\{ x : \exists u, v : \underbrace{\mathcal{B}(A(x, u), v)}_{\text{LMI}} \succeq 0 \right\},$$

which is a SDR for X .

Lemma. *Every direct product of Lorentz cones is SDr.*

Proof. It suffices to prove that a Lorentz cone \mathbf{L}^m is a SDr set (since SD-representability is preserved when taking direct products).

To prove that \mathbf{L}^m is SDr, let us make use of the following

Lemma on Schur Complement. *A symmetric block matrix*

$$A = \left(\begin{array}{c|c} P & Q^T \\ \hline Q & R \end{array} \right)$$

with positive definite R is positive (semi)definite iff the matrix

$$P - Q^T R^{-1} Q$$

is positive (semi)definite.

LSC \Rightarrow Lemma: Consider the linear mapping

$$\begin{bmatrix} x_1 \\ x_2 \\ \dots \\ x_m \end{bmatrix} \mapsto \mathcal{A}x = \left(\begin{array}{c|cccc} x_m & x_1 & x_2 & x_3 & \dots & x_{m-1} \\ \hline x_1 & x_m & & & & \\ x_2 & & x_m & & & \\ x_3 & & & x_m & & \\ \vdots & & & & \ddots & \\ x_{m-1} & & & & & x_m \end{array} \right)$$

We claim that

$$\mathbf{L}^m = \{x : \mathcal{A}(x) \succeq 0\}.$$

Indeed,

$$\mathbf{L}^m = \left\{ x \in \mathbf{R}^m : x_m \geq \sqrt{x_1^2 + \dots + x_{m-1}^2} \right\}$$

and therefore

- if $x \in \mathbf{L}^m$ is nonzero, then $x_m > 0$ and

$$x_m - (x_1^2 + x_2^2 + \dots + x_{m-1}^2)/x_m \geq 0$$

so that $\mathcal{A}(x) \succeq 0$ by LSC. If $x = 0$, then $\mathcal{A}(x) = 0 \succeq 0$.

- if $\mathcal{A}(x) \succeq 0$ and $\mathcal{A}(x) \neq 0$, then $x_m > 0$ and, by LSC,

$$x_m - (x_1^2 + x_2^2 + \dots + x_{m-1}^2)/x_m \geq 0 \Rightarrow x \in \mathbf{L}^m.$$

And if $\mathcal{A}(x) = 0$, then $x = 0 \in \mathbf{L}^m$.

Lemma on Schur Complement. *A symmetric block matrix*

$$A = \left[\begin{array}{c|c} P & Q^T \\ \hline Q & R \end{array} \right]$$

with positive definite R is positive (semi)definite iff the matrix

$$P - Q^T R^{-1} Q$$

is positive (semi)definite.

Proof. A is $\succeq 0$ if and only if

$$\inf_v \begin{bmatrix} u \\ v \end{bmatrix}^T \left[\begin{array}{c|c} P & Q^T \\ \hline Q & R \end{array} \right] \begin{bmatrix} u \\ v \end{bmatrix} \geq 0 \quad \forall u. \quad (*)$$

When $R \succ 0$, the left hand side inf can be easily computed and turns to be

$$u^T (P - Q^T R^{-1} Q) u.$$

Thus, $(*)$ is valid if and only if

$$u^T (P - Q^T R^{-1} Q) u \geq 0 \quad \forall u,$$

i.e., iff

$$P - Q^T R^{-1} Q \succeq 0.$$

♠ **Convention:** For a symmetric $m \times m$ matrix X ,

$$\lambda(X) = [\lambda_1(x); \lambda_2(X); \dots; \lambda_m(X)] \in \mathbf{R}^m$$

stands for the vector of eigenvalues of X *written down with their multiplicities in the non-ascending order:*

$$\lambda_1(X) \geq \lambda_2(X) \geq \dots \geq \lambda_m(X)$$

Examples:

$$\lambda(I_m) = [1; \dots; 1] \in \mathbf{R}^m; \quad \lambda \left(\begin{bmatrix} 2 & & \\ & 1 & \\ & & 2 \end{bmatrix} \right) = [2; 2; 1].$$

More examples of SD-representable functions/sets

- **The largest eigenvalue** $\lambda_{\max}(X)$ regarded as a function of $m \times m$ symmetric matrix X is SDr:

$$\lambda_{\max}(X) \leq t \quad \Leftrightarrow \quad tI_m - X \succeq 0,$$

I_k being the unit $k \times k$ matrix.

- **The largest eigenvalue of a matrix pencil.** Let $M, A \in \mathbf{S}^m$ be such that $M \succ 0$.

The eigenvalues of the *pencil* $[M, A]$ are reals λ such that the matrix $\lambda M - A$ is singular, or, equivalently, such that

$$\exists e \neq 0 : \quad Ae = \lambda Me.$$

The eigenvalues of the pencil $[M, A]$ are the usual eigenvalues of the symmetric matrix $D^{-1}AD^{-T}$, where D is such that $M = DD^T$.

The largest eigenvalue $\lambda_{\max}(X : M)$ of a pencil $[M, X]$ with $M \succ 0$, regarded as a function of X , is SDr:

$$\lambda_{\max}(X : M) \leq t \quad \Leftrightarrow \quad tM - X \succeq 0.$$

- **Sum of k largest eigenvalues.** For a symmetric $m \times m$ matrix X , let $\lambda(X)$ be the vector of eigenvalues of X taken with their multiplicities in the non-ascending order:

$$\lambda_1(X) \geq \lambda_2(X) \geq \dots \geq \lambda_m(X),$$

and let $S_k(X)$ be the sum of k largest eigenvalues of X :

$$S_k(X) = \sum_{i=1}^k \lambda_i(X) \quad [1 \leq k \leq m]$$

$$[S_1(X) = \lambda_{\max}(X); S_m(X) = \text{Tr}(X)]$$

The functions $S_k(X)$ are SDr. We shall see that *this fact is crucial when building SDR's of a wide family of useful for applications functions of eigenvalues.*

- $\lambda_1(X) \geq \lambda_2(X) \geq \dots \geq \lambda_m(x)$: eigenvalues of $X \in \mathbf{S}^m$
- $S_k(X) = \sum_{i=1}^k \lambda_i(X)$

♠ **Towards SDR for $S_k(X)$: vector case.** The vector analogy of $S_k(x)$ is the sum $s_k(x)$ of k largest entries in $x \in \mathbf{R}^m$.

- Recalling that the extreme points of the polytope $\{y \in \mathbf{R}^m : 0 \leq y_i \leq 1 \forall i, \sum_i y_i = k\}$ are zero-one vectors with exactly k entries equal to 1, we have

$$\begin{aligned} s_k(x) &= \max_y \{x^T y : 0 \leq y_i \leq 1 \forall i, \sum_i y_i = k\} \\ &= \min_{z,s} \{\sum_i z_i + ks : x \leq z + s[1; \dots; 1], z \geq 0\} \text{ [LP Duality]} \end{aligned}$$

We arrive at polyhedral representation of $s_k(\cdot)$:

$$s_k(x) \leq t \Leftrightarrow \exists s \in \mathbf{R}, z \in \mathbf{R}^m : \sum_i z_i + ks \leq t, z \geq 0, x \leq z + s[1; \dots; 1]$$

- $S_k(x)$: the sum of k largest eigenvalues of $X \in \mathbf{S}^m$
- $s_k(x)$: the sum of k largest entries of $x \in \mathbf{R}^m$

$$s_k(x) \leq t \Leftrightarrow \exists s \in \mathbf{R}, z \in \mathbf{R}^m : \sum_i z_i + ks \leq t, z \geq 0, x \leq z + s[1; \dots; 1]$$

This *suggests* (suggests, not implies!) the SDR of $S_k(x)$ as follows:

$$S_k(X) \leq t \Leftrightarrow \exists s, Z : \begin{cases} (a) & ks + \text{Tr}(Z) \leq t \\ (b) & Z \succeq 0 \\ (c) & X \preceq Z + sI_m \end{cases}$$

Proof. We should prove that

- (i) If a pair X, t can be extended, by properly chosen s, Z , to a solution of (a) – (c), then $S_k(X) \leq t$;
- (ii) If $S_k(X) \leq t$, then the pair X, t can be extended by properly chosen s, Z , to a solution of (a) – (c).

$$S_k(X) \leq t \Leftrightarrow \exists s, Z : \begin{cases} (a) & ks + \text{Tr}(Z) \leq t \\ (b) & Z \succeq 0 \\ (c) & X \preceq Z + sI_m \end{cases}$$

“(i) If a pair X, t can be extended, by properly chosen s, Z , to a solution of (a) – (c), then $S_k(X) \leq t$ ”

(i): We use the following

Basic Fact: *The vector $\lambda(X)$ is a \succeq -monotone function of $X \in \mathbf{S}^m$: $X \succeq X' \Rightarrow \lambda(X) \geq \lambda(X')$.*

Let (X, t, s, Z) solve (a) – (c). Then

$$\begin{aligned} & X \preceq Z + sI_m && \text{[by (c)]} \\ \Rightarrow & \lambda(X) \leq \lambda(Z + sI_m) = \lambda(Z) + s \begin{bmatrix} 1 \\ \vdots \\ 1 \end{bmatrix} && \text{[by Basic Fact]} \\ \Rightarrow & S_k(X) \leq S_k(Z) + sk \\ \Rightarrow & S_k(X) \leq \text{Tr}(Z) + sk && \left[\begin{array}{l} \text{since } S_k(Z) \leq \text{Tr}(Z) \\ \text{due to (b)} \end{array} \right] \\ \Rightarrow & S_k(X) \leq t && \text{[by (a)]} \end{aligned}$$

(ii): Let $S_k(X) \leq t$, and let $X = U \text{Diag}\{\lambda\} U^T$, $\lambda = \lambda(X)$, be the eigenvalue decomposition of X .

$$s = \lambda_k, \quad Z = U \underbrace{\begin{bmatrix} \lambda_1 - \lambda_k & & & & \\ & \ddots & & & \\ & & \lambda_{k-1} - \lambda_k & & \\ & & & 0 & \\ & & & & \ddots \\ & & & & & 0 \end{bmatrix}}_{\text{Diag}\{\lambda(Z)\}} U^T,$$

we have

$$\begin{aligned} & Z \succeq 0, \\ \text{Diag}\{\lambda(X)\} & \leq \text{Diag} \left\{ \lambda(Z) + s \begin{bmatrix} 1 \\ \vdots \\ 1 \end{bmatrix} \right\} \Rightarrow X \preceq Z + sI_m, \\ & t \geq S_k(X) = ks + \text{Tr}(Z), \end{aligned}$$

so that (t, X, s, Z) solves the system of LMIs

$$\begin{aligned} (a) \quad & ks + \text{Tr}(Z) \leq t \\ (b) \quad & Z \succeq 0 \\ (c) \quad & X \preceq Z + sI_m \end{aligned}$$

Basic Fact: The vector $\lambda(X)$ is a \succeq -monotone function of $X \in \mathbf{S}^m$: $X \succeq X' \Rightarrow \lambda(X) \geq \lambda(X')$.

Here is a hint to explanation:

♠ **Question:** Given a finite collection of numbers, how to find the 3-rd largest of them?

♠ **An instructive answer:**

- somehow throw away 2=3-1 numbers from the collection and find the maximum of the remaining ones; this maximum depends on the pair of numbers thrown away.

- minimize this maximum over all pairs of elements you can throw away. This minimum will be exactly the 3-rd largest number in the collection.

♠ **Corollary:** *When increasing somehow every number in the original collection, the third largest number cannot decrease.*

Indeed, when increasing the numbers in the collection, every one of the above maxima cannot decrease, and thus the minimum of these maxima cannot decrease as well.

Basic Fact: The vector $\lambda(X)$ is a \succeq -monotone function of $X \in \mathbf{S}^m$: $X \succeq X' \Rightarrow \lambda(X) \geq \lambda(X')$.

This is an immediate corollary of the following matrix analogy of the above recipe for finding k -th largest number in a collection:

Variational Characterization of Eigenvalues: For an $m \times m$ symmetric matrix A , one has

$$\lambda_k(A) = \min_{E \in \mathcal{E}_k} \max_{e \in E: e^T e = 1} e^T A e, \quad (*)$$

where \mathcal{E}_k is the collection of all linear subspaces of \mathbf{R}^m of the dimension $m - k + 1$ (“subspaces obtained from \mathbf{R}^m by throwing $k - 1$ dimensions away”)

In particular,

$$\begin{aligned} \lambda_1(A) &= \max_{e: e^T e = 1} e^T A e \\ \lambda_m(A) &= \min_{e: e^T e = 1} e^T A e \end{aligned}$$

Note: When A \succeq -grows, the right hand side in $(*)$ grows or remains the same, implying Basic Fact.

- VCE has a lot of important consequences, e.g, the following one:

Eigenvalue Interlacement Theorem: Let A be a symmetric $m \times m$ matrix, and \hat{A} be a $(m - k) \times (m - k)$ principal submatrix of A . Then

$$\lambda_i(A) \geq \lambda_i(\hat{A}) \geq \lambda_{i+k}(A).$$

Proof of VCE. Let $\lambda_k = \lambda_k(A)$, and let

$$\mu_k = \min_{E: \dim E = m - k + 1} \max_{e \in E: e^T e = 1} e^T A e;$$

we should prove that $\mu_k = \lambda_k(A)$.

Both μ_k and λ_k remain invariant when A is replaced with UAU^T with orthogonal U
 \Rightarrow It suffices to consider the case of $A = \text{Diag}\{\lambda(A)\}$.

$\lambda_k \geq \mu_k$: Let $E = \{x : x_1 = \dots = x_{k-1} = 0\}$. Then

$$\begin{aligned} \dim E &= m - k + 1 \Rightarrow \\ \mu_k &\leq \max_{e \in E: e^T e = 1} e^T A e = \max_{\substack{e_k, \dots, e_m, \\ e_k^2 + \dots + e_m^2 = 1}} \sum_{i=k}^m \lambda_i e_i^2 = \lambda_k. \end{aligned}$$

$\lambda_k \leq \mu_k$: Let $F = \{x : x_{k+1} = \dots = x_m = 0\}$, so that $\dim F = k$. For every subspace E with $\dim E = m - k + 1$, we have $\dim E + \dim F > m$, so that there exists a unit vector $f \in F \cap E$. We have

$$\max_{e \in E: e^T e = 1} e^T A e \geq f^T A f = \sum_{i=1}^k \lambda_i f_i^2 \geq \lambda_k \sum_{i=1}^k f_i^2 = \lambda_k.$$

Thus, $\mu_k \equiv \min_{E: \dim E = m - k + 1} \max_{e \in E: e^T e = 1} e^T A e \geq \lambda_k$.

- To proceed, we need the following

Birkhoff Theorem: Let P_m be the set of double-stochastic $m \times m$ matrices, that is, matrices $[p_{ij}]_{i,j=1}^m$ such that

$$p_{ij} \geq 0; \quad \sum_i p_{ij} = 1 \forall j; \quad \sum_j p_{ij} = 1 \forall i.$$

The vertices of the polytope P_m are exactly the permutation matrices, so that every double stochastic matrix is a convex combination of permutation matrices.

Sketch of the proof: The only nontrivial claim is that *an extreme point p of P_m is a Boolean (\equiv with entries 0/1) matrix.*

P_m is cut off \mathbb{R}^{m^2} by m^2 inequalities $p_{ij} \geq 0$ and $2m - 1$ linearly independent linear equalities ("if all row sums and *all but one* column sums in a square matrix are equal to 1, than all row and column sums are equal to 1").

\Rightarrow extreme point p should make $m^2 - (2m - 1)$ of the bounds $p_{ij} \geq 0$ active

\Rightarrow there is a column in p with at most one nonzero

$\Rightarrow p$ has an entry equal to 1, and all remaining entries in the row and the column of this entry are zeros.

Eliminating from p the row and the column of an entry equal to 1, we get a (clearly extreme) point of P_{m-1}

\Rightarrow The claim can be proved by induction in m .

Definition: A function $f : \mathbf{R}^n \rightarrow \mathbf{R} \cup \{+\infty\}$ is called *symmetric*, if $f(x)$ remains intact when permuting the entries of x , as is the case, e.g., when $f(x) = \sum_i x_i$ or $f(x) = x_1.x_2...x_n$, or $f(x) = \sum \ln(x_i)$.

♠ **Corollary of Birkhoff's Theorem:** *Let $f(x)$ be a symmetric convex function on \mathbf{R}^m , and let π be a double-stochastic $m \times m$ matrix. Then*

$$f(\pi x) \leq f(x) \quad \forall x \in \mathbf{R}^m.$$

Proof. By Birkhoff Theorem, πx is a convex combination of permutations x^i of x . Therefore, by Jensen's Inequality, $f(\pi x)$ is not greater than $\max_i f(x^i)$, and this is exactly $f(x)$ due to the symmetry of f .

♠ **Corollary of Corollary:** *Let $f(x)$ be a symmetric convex function on \mathbb{R}^m . Then the function*

$$F(X) = f(\lambda(X))$$

is convex on \mathbf{S}^m , and, moreover,

$$F(X) = \max_{U: U^T U = I} f(\text{Dg}(UXU^T)). \quad (*)$$

Reason (to be explained at the next slide): $\text{Dg}(UXU^T)$ is the image of $\lambda(X)$ under multiplication by a double-stochastic matrix.

- For notation $\text{Dg}(\cdot)$, see slide 0.3

Corollary of Corollary: Let $f(x)$ be a symmetric convex function on \mathbf{R}^m . Then the function

$$F(X) = f(\lambda(X))$$

is convex on \mathbf{S}^m , and, moreover,

$$F(X) = \max_{U: U^T U = I} f(\text{Dg}(UXU^T)). \quad (*)$$

Proof: It suffices to verify (*); indeed, given (*), $F(\cdot)$ is convex as the upper bound, w.r.t. orthogonal U , of the family of (clearly convex) functions $f_U(\cdot)$.

For properly chosen orthogonal U we have

$$UXU^T = \text{Diag}\{\lambda(X)\} \Rightarrow \max_{U: U^T U = I} f(\text{Dg}(UXU^T)) \geq f(\lambda(X)).$$

To prove the opposite inequality, observe that every matrix of the form UXU^T with orthogonal U is of the form $V\text{Diag}\{\lambda(X)\}V^T$ with orthogonal V as well. Now,

$$[\text{Dg}(UXU^T)]_i = [V\text{Diag}\{\lambda(X)\}V^T]_{ii} = \sum_j V_{ij}^2 \lambda_j(X),$$

that is, $\text{Dg}(UXU^T) = \pi\lambda(X)$ for the *double stochastic* matrix $\pi = [V_{ij}^2]_{i,j}$. Therefore

$$f(\text{Dg}(UXU^T)) = f(\pi\lambda(X)) \leq f(\lambda(X)).$$

♠ **Corollary of Corollary of Corollary:** *Let f be a convex symmetric function on \mathbb{R}^m . Then*

$$f(\text{Dg}(X)) \leq f(\lambda(X))$$

for every symmetric matrix X .

For example, for every symmetric matrix X with the vector of eigenvalues λ one has

- The sum of k largest diagonal entries of X does not exceed $S_k(X) = \lambda_1 + \dots + \lambda_k$

$[f(x) = \max_{i_1 < i_2 < \dots < i_k} [x_{i_1} + \dots + x_{i_k}]$ is the sum of k largest entries in $x]$

- The sum of k smallest diagonal entries in X is at least the sum of k smallest of λ_i 's
- If $X \succ 0$, then the product of the k smallest diagonal entries in X is at least the product of the k smallest of λ_i 's. In particular, the product of all diagonal entries in X is $\geq \text{Det}(X)$.

$[g(x) = \min_{i_1 < i_2 < \dots < i_k} [\ln x_{i_1} + \dots + \ln x_{i_k}]$ is the sum of logs of k smallest entries in $x > 0$, $f(x) = -g(x)]$

♣ For $z \in \mathbf{R}^m$, let $s_k(z)$ be the sum of k largest entries in z .

● **Majorization Principle:** *Let $x \in \mathbf{R}^m$. A point y can be represented as πx with a double stochastic matrix π if and only if*

$$s_k(y) \leq s_k(x), \quad k < m, \quad \text{and} \quad s_m(y) = s_m(x)$$

Remark: *For $x, y \in \mathbf{R}^m$, the condition $s_k(y) \leq s_k(x)$, $k \leq m$, is necessary and sufficient for existence of double-stochastic matrix π such that $y \leq \pi x$.*

Corollary: SD-representability of symmetric SDr functions of eigenvalues. *Let $f(x)$ be a SDr symmetric function on \mathbf{R}^m . Then the function*

$$F(X) = f(\lambda(X)) : \mathbf{S}^m \rightarrow \mathbf{R} \cup \{+\infty\}$$

is SDr with SDR readily given by SDR of f . In particular, the following functions are SDr with explicit SDR's:

- $-\text{Det}^\pi(X)$, $X \in \mathbf{S}_+^m$ ($\pi \in (0, \frac{1}{m}]$ is rational);
- $\text{Det}^{-\pi}(X)$, $X \succ 0$ ($\pi > 0$ is rational);
- $|X|_\pi = \|\lambda(X)\|_\pi$, $X \in \mathbf{S}^m$ ($\pi \in [1, \infty)$ is rational or $\pi = \infty$).

Proof. Let $t \geq f(x) \Leftrightarrow \exists u : \mathcal{A}(t, x, u) \succeq 0$. Then

$$\begin{aligned}
 t \geq F(X) &\Leftrightarrow t \geq f(\lambda(X)) \Leftrightarrow \exists (y \in \mathbf{R}^m, \pi \in P_m) : \begin{cases} y_1 \geq y_2 \geq \dots \geq y_m \\ f(y) \leq t \\ \lambda(X) = \pi y \\ [\text{since } f(\pi y) \leq f(y)] \end{cases} \\
 \Rightarrow t \geq F(X) &\Leftrightarrow \exists y \in \mathbf{R}^m : \begin{cases} y_1 \geq y_2 \geq \dots \geq y_m, f(y) \leq t \\ s_k(\lambda(X)) \leq y_1 + \dots + y_k, k < m \\ s_m(\lambda(X)) = y_1 + \dots + y_m \\ [\text{by Majorization Principle}] \end{cases} \\
 \Rightarrow t \geq F(X) &\Leftrightarrow \exists (y \in \mathbf{R}^m, u) : \begin{cases} y_1 \geq y_2 \geq \dots \geq y_m, \mathcal{A}(y, t, u) \succeq 0 \\ \underbrace{S_k(X) \leq y_1 + \dots + y_k, k < m}_{\text{SD-representable!}} \\ \text{Tr}(X) = y_1 + \dots + y_m \end{cases}
 \end{aligned}$$

Majorization Principle: *Let $x \in \mathbb{R}^m$. A point y can be represented as πx with a double stochastic matrix π if and only if*

$$s_k(y) \leq s_k(x), k < m, \text{ and } s_m(y) = s_m(x) \quad (*)$$

Proof, “only if” part: If $y = \pi x$ with double stochastic π , then $s_k(y) \leq s_k(x)$ by Corollary of the Birkhoff Theorem ($s_k(\cdot)$ are convex symmetric functions!), and of course $s_m(y) = s_m(x)$.

Majorization Principle: Let $x \in \mathbb{R}^m$. A point y can be represented as πx with a double stochastic matrix π if and only if

$$s_k(y) \leq s_k(x), \quad k < m, \quad \text{and} \quad s_m(y) = s_m(x) \quad (*)$$

Proof, “if” part: Let x and y satisfy $(*)$; we should prove that $y = \pi x$ for a double stochastic matrix π . By “permutational symmetry” of the claim, we may assume that

$$x_1 \geq x_2 \geq \dots \geq x_m, \quad y_1 \geq y_2 \geq \dots \geq y_m.$$

Let X be the set of all permutations of x ; by Birkhoff Theorem, $y = \pi x$ for certain double stochastic π **iff** $y \in \text{Conv}(X)$, thus all we should prove is that $y \in \text{Conv}(X)$. Assume that $y \notin \text{Conv}(X)$. Then there exists e such that

$$e^T y > \max_{x' \in X} e^T x'. \quad (**)$$

Permuting the entries in e , we do not vary the right hand side in $(**)$. If $e_i < e_j$ for a pair i, j with $i > j$, then, swapping e_i and e_j , we do not decrease $e^T y$ (since $y_1 \geq y_2 \geq \dots \geq y_m$). Thus, we may assume that e in $(*)$ satisfies $e_1 \geq e_2 \geq \dots \geq e_m$. Then

$$\begin{aligned} e^T y &= e_1 y_1 + e_2 y_2 + \dots + e_m y_m \\ &= e_m (y_1 + \dots + y_m) + (e_{m-1} - e_m)(y_1 + \dots + y_{m-1}) \\ &\quad + (e_{m-2} - e_{m-1})(y_1 + \dots + y_{m-2}) + \dots + (e_1 - e_2)y_1 \\ &= e_m s_m(y) + \underbrace{(e_{m-1} - e_m)}_{\geq 0} s_{m-1}(y) \\ &\quad + \underbrace{(e_{m-2} - e_{m-1})}_{\geq 0} s_{m-2}(y) + \dots + \underbrace{(e_1 - e_2)}_{\geq 0} s_1(y) \\ &\leq e_m s_m(x) + (e_{m-1} - e_m) s_{m-1}(x) \\ &\quad + (e_{m-2} - e_{m-1}) s_{m-2}(x) + \dots + (e_1 - e_2) s_1(x) \quad [\text{by } (*)] \\ &= e^T x - \text{contradicts } (**)! \end{aligned}$$

Remark: For $x, y \in \mathbf{R}^m$, the condition $s_k(y) \leq s_k(x)$, $k \leq m$, is necessary and sufficient for existence of double-stochastic matrix π such that $y \leq \pi x$.

Proof: If part: The functions $s_k(\cdot)$ clearly are monotone, so that when $y \leq \pi x$ with double-stochastic π , we have $s_k(y) \leq s_k(\pi x)$, and the latter quantity, as we know, is $\leq s_k(x)$.

Only if part: Let $s_k(y) \leq s_k(x)$, $k \leq m$. Let x_t be obtained from x by decreasing by t the smallest entry in x and keeping the remaining entries intact. We have $s_k(x_t) = s_k(x)$, $k < m$, and $s_m(x_t) = s_m(x) - t$. Setting $t = s_m(x) - s_m(y)$, we get $s_k(x_t) \geq s_k(y)$, $k < m$, and $s_m(x_t) = s_m(y)$. By Majorization principle, $y = \pi x_t$ for some double-stochastic matrix π , and $\pi x_t \leq \pi x$ since $x_t \leq x \Rightarrow y \leq \pi x$.

- **The function** $-\sqrt[m]{\mathbf{Det}(X)} : \mathbf{S}_+^m \rightarrow \mathbf{R}$ is SDr.

This is a particular case of our general result on SD-representability of symmetric SDr functions of eigenvalues as applied to the CQr function $-\sqrt[m]{x_1 \dots x_m} : \mathbf{R}_+^m \rightarrow \mathbf{R}$.

Due to its importance in various volume-related problems, we present a “customized” SDR for $-\mathbf{Det}^{1/m}(X)$ which is shorter than the one given by our general theory:

Fact: The set $\mathcal{X} = \{(X, t) : X \in \mathbf{S}_+^m, t \leq \mathbf{Det}^{1/m}(X)\}$ admits the SDR readily given by the following representation:

$$X \succeq 0, t \leq \mathbf{Det}^{1/m}(X) \Leftrightarrow \exists D, \tau : \left\{ \begin{array}{l} X \succeq 0, D : \text{lower triangular with nonnegative diagonal} \\ \left[\begin{array}{c|c} X & D \\ \hline D^T & \tau I_m \end{array} \right] \succeq 0 \\ \underbrace{\tau \leq (D_{11}D_{22}\dots D_{mm})^{1/m}}_{\text{CQr}}, t \leq \tau \end{array} \right.$$

Claim: The set $\mathcal{X} := \{(X, t) : X \in \mathbf{S}_+^m, t \leq \text{Det}^{1/m}(X)\}$ admits representation

$$X \succeq 0, t \leq \text{Det}^{1/m}(X) \Leftrightarrow \left\{ (\exists D, \tau : \left\{ \begin{array}{ll} D : \text{lower triangular with nonnegative diagonal} & (a) \\ \left[\begin{array}{c|c} X & D \\ \hline D^T & \tau I_m \end{array} \right] \succeq 0 & (b) \\ \tau \leq (D_{11}D_{22}\dots D_{mm})^{1/m} & (c) \\ t \leq \tau & (d) \end{array} \right\} \right\}$$

• **In one direction:** *Let $(X, t) \in \mathcal{X}$.* When $X \succ 0$, let the lower triangular Δ with nonnegative diagonal be given by Choleski decomposition of X : $X = \Delta\Delta^T$, and let $\tau = \text{Det}^{1/m}(X)$. Setting $D = \sqrt{\tau}\Delta$, we meet (b) (by Schur Complement Lemma), (a), and (d) (since $t \leq \text{Det}^{1/m}(X)$). In addition, $\text{Det}(X) = \text{Det}^2(\Delta) \Rightarrow \tau = (\Delta_{11}\Delta_{22}\dots\Delta_{mm})^{2/m} \Rightarrow (D_{11}D_{22}\dots D_{mm})^{1/m} = \sqrt{\tau}(\Delta_{11}\Delta_{22}\dots\Delta_{mm})^{1/m} = \tau$, implying (c). We have augmented (X, t) to a solution of (a-d). When $X \succeq 0$ is singular, the required augmentation is given by $D = 0, \tau = 0$.

• **In the opposite direction:** *Let X, t, D, τ solve (a-d).* Then $X \succeq 0$ by (b). If $\tau = 0$, then $t \leq 0$ by (d), and thus $t \leq \text{Det}^{1/m}(X)$. If $\tau > 0$, then $X \succeq \tau^{-1}DD^T$ by (b) and Schur Complement Lemma $\Rightarrow \text{Det}(X) \geq \tau^{-m}\text{Det}^2(D) \Rightarrow \text{Det}^{1/m}(X) \geq \tau^{-1}(D_{11}D_{22}\dots D_{mm})^{2/m} \geq \tau$, with concluding inequality given by (c), $\Rightarrow t \leq \text{Det}^{1/m}(X)$ by (d) $\Rightarrow (X, t) \in \mathcal{X}$.

- **Norm of rectangular matrix.** Let X be a $m \times n$ matrix. Its spectral norm

$$\|X\| = \max_{\|\xi\|_2 \leq 1} \|X\xi\|_2$$

is SDr:

$$t \geq \|X\| \quad \Leftrightarrow \quad \begin{bmatrix} tI_n & X^T \\ X & tI_m \end{bmatrix} \succeq 0.$$

More generally, let

$$\sigma_i(X) = \sqrt{\lambda_i(X^T X)}, \quad 1 \leq i \leq n,$$

be the singular values of an $m \times n$ matrix X . It is known that $\sigma_i(x) = 0$ when $i > \min[m, n]$.

Fact: $X \in \mathbf{R}^{m \times n}$ admits *singular value decomposition*:

$$X = UDV^T$$

with $m \times m$ orthogonal U , $n \times n$ orthogonal V , and $m \times n$ matrix D with diagonal entries $D_{ii} = \sigma_i(X)$, $1 \leq i \leq \min[m, n]$, and all remaining entries equal to 0.

- **The sum of k largest singular values** $\Sigma_k(X) = \sum_{i=1}^k \sigma_i(X)$ is a SDr function of $X \in \mathbf{R}^{m \times n}$.

Indeed, it suffices to consider the case of $k \leq \min[m, n]$. It is easily seen that the eigenvalues of *linearly depending on X symmetric* matrix

$$A(X) = \left[\begin{array}{c|c} & X \\ \hline X^T & \end{array} \right]$$

are nonzero singular values of X , minus nonzero singular values of X , and perhaps a number of zeros. As a result,

$$\Sigma_k(X) = S_k(A(X))$$

Since S_k is SDr and this property is preserved by affine substitution of argument, Σ_k is SDr.

- **SDr of symmetric monotone SDr function of singular values.** Let $f(\lambda) : \mathbf{R}_+^n \rightarrow \mathbf{R} \cup \{\infty\}$ be a symmetric w.r.t. permutations of coordinates and \geq -nondecreasing SDr function. Then the function

$$F(X) = f(\sigma(X)) : \mathbf{R}^{m \times n} \rightarrow \mathbf{R} \cup \{\infty\}$$

is SDr:

$$t \geq F(X) \Leftrightarrow \exists z : \begin{cases} z_1 \geq z_2 \geq \dots \geq z_n \geq 0, \underbrace{f(z) \leq t}_{\text{SDr}} \\ \underbrace{z_1 + \dots + z_k \geq S_k \left(\left[\begin{array}{c|c} & X \\ \hline X^T & \end{array} \right] \right)}_{\text{SDr}}, k \leq n \end{cases}$$

[recall that $y \in \mathbf{R}^n$ is $\leq \pi x$ for some double-stochastic π iff $s_k(y) \leq s_k(x)$ for $k \leq n$]

Corollary: *The Schatten norms – the functions*

$$|X|_\pi = \|\sigma(X)\|_\pi : \mathbf{R}^{m \times n} \rightarrow \mathbf{R}_+$$

with rational $\pi \in [1, \infty)$ are SDR with explicit SDR's.

Note: Nuclear norm $|X|_1 = \sum_i \sigma_i(X)$ is what is replacing ℓ_1 -norm when passing from recovering *sparse vectors* to recovering *low rank matrices*. A very popular problem of the latter form is *Matrix Completion*: recovery of low rank matrix from noisy measurements of *part* of entries of the matrix.

♠ Due to importance of nuclear norm, we present a “customized” SDR for $|X|_1$; this SDR is shorter than the one given by our general theory.

Fact: For $m \times n$ matrices X , one has

$$|X|_1 \leq t \Leftrightarrow \exists P, Q : \left[\begin{array}{c|c} P & X \\ \hline X^T & Q \end{array} \right] \succeq 0 \text{ \& } \text{Tr}(P) + \text{Tr}(Q) \leq 2t$$

Indeed, given X , let $X = UDV^T$ be the svd of X . The matrices

$$\left[\begin{array}{c|c} P & X \\ \hline X^T & Q \end{array} \right], \left[\begin{array}{c|c} U^T P U & D \\ \hline D^T & V^T Q V \end{array} \right] = \left[\begin{array}{c|c} U^T & \\ \hline & V^T \end{array} \right] \left[\begin{array}{c|c} P & X \\ \hline X^T & Q \end{array} \right] \left[\begin{array}{c|c} U & \\ \hline & V \end{array} \right]$$

are rotations of each other and therefore simultaneously are/are not $\succeq 0$, and $\text{Tr}(P) = \text{Tr}(U^T P U)$, $\text{Tr}(Q) = \text{Tr}(V^T Q V)$

\Rightarrow It suffices to verify that if D is $m \times n$ matrix with zeros outside of the diagonal and with diagonal entries $D_{ii} = \sigma_i \geq 0$, $1 \leq i \leq k := \min[m, n]$, then

$$\sum_i \sigma_i \leq t \Leftrightarrow \exists P, Q : \underbrace{\left[\begin{array}{c|c} P & D \\ \hline D^T & Q \end{array} \right]}_A \succeq 0 \text{ \& } \text{Tr}(P) + \text{Tr}(Q) \leq 2t. \quad (*)$$

- In one direction: when $\sum_i \sigma_i \leq t$, specifying P, Q as diagonal $m \times m$, resp., $n \times n$ matrices with σ_i , $1 \leq i \leq k$, as the first k diagonal entries and zero remaining diagonal entries, we ensure the validity of the right hand side requirements in $(*)$. \square

- In the opposite direction: when P, Q, t satisfy the right hand side requirements in $(*)$, we have $\sigma_i \leq \sqrt{P_{ii}Q_{ii}}$, $1 \leq i \leq k$ (look at principal 2×2 minors of A) $\Rightarrow \sum_i \sigma_i \leq \sum_{i \leq k} \sqrt{P_{ii}Q_{ii}} \leq \frac{1}{2} \sum_{i \leq k} (P_{ii} + Q_{ii}) \leq \frac{1}{2} [\text{Tr}(P) + \text{Tr}(Q)]$ (since clearly $P \succeq 0$, $Q \succeq 0$) $\Rightarrow \sum_{i \leq k} \sigma_i \leq t$. \square

- “ \succeq -convex quadratic matrix function”

$$F(X) = (AXB)(AXB)^T + CXD + (CXD)^T + E$$

$$\left[F : \mathbf{R}^{p \times q} \rightarrow \mathbf{S}^m \right]$$

($A, B, C, D, E = E^T$ are constant matrices such that $F(\cdot)$ makes sense and takes its values in \mathbf{S}^m) is SDr in the sense that its “ \succeq graph”

$$\text{Epi}\{F\} = \{(X, Y) \in \mathbf{R}^{p \times q} \times \mathbf{S}^m : F(X) \preceq Y\}$$

is an SDr set:

$$Y \succeq F(X)$$

$$\Updownarrow [\text{LSC}]$$

$$\left[\begin{array}{c|c} \frac{Y - E - CXD - (CXD)^T}{(AXB)^T} & AXB \\ \hline & I_r \end{array} \right] \succeq 0 \quad [B : q \times r]$$

(by the Schur Complement Lemma).

- “ \succeq -convex fractional-quadratic function”. Let X be a rectangular $p \times q$ matrix, and V be a positive definite symmetric $q \times q$ matrix. Consider the matrix-valued function

$$F(X, V) = XV^{-1}X^T : \mathbf{R}^{p \times q} \times \text{int } \mathbf{S}_+^q \rightarrow \mathbf{S}^p$$

The closure of the “ \succeq graph” of $F(X, V)$ – the set

$$\mathcal{G} \equiv \text{cl} \left\{ (X, V, Y) \in \mathbf{R}^{p \times q} \times \text{int } \mathbf{S}_+^q \times \mathbf{S}^p : F(X, V) \preceq Y \right\}$$

is SDr:

$$\mathcal{G} = \left\{ (X, V, Y) \in \mathbf{R}^{p \times q} \times \mathbf{S}^q \times \mathbf{S}^p \mid \left[\begin{array}{c|c} Y & X \\ \hline X^T & V \end{array} \right] \succeq 0 \right\}.$$

(by the Schur Complement Lemma).

♠ **Matrix square root.** For $X \succeq 0$, the matrix $X^{1/2}$ is, *by definition* symmetric matrix such that

$$X^{1/2} \succeq 0 \text{ \& } [X^{1/2}]^2 = X.$$

It is known that these requirements *uniquely* define $X^{1/2}$.

- $X^{1/2}$ is readily given by eigenvalue decomposition if $X \succeq 0$:

$$X = U \text{Diag}\{\lambda_1, \dots, \lambda_m\} U^T \Rightarrow X^{1/2} = U \text{Diag}\{\sqrt{\lambda_1}, \dots, \sqrt{\lambda_m}\} U^T$$

$$[U^T U = I_m]$$

- Surprisingly, considered as a function of $X \succeq 0$, $X^{1/2}$ is \succeq -monotone and \succeq -concave:

$$0 \preceq X \preceq Y \Rightarrow 0 \preceq X^{1/2} \preceq Y^{1/2} \text{ \& } \{(X, Z) : X \succeq 0, Z \preceq X^{1/2}\} \text{ is convex set}$$

- “ \succeq -hypograph of the matrix square root.” The sets

$$\{(X, Y) \in \mathbf{S}_+^m \times \mathbf{S}_+^m : X^2 \preceq Y\} = \{(X, Y) : X \succeq 0, \left[\begin{array}{c|c} Y & X \\ \hline X & I \end{array} \right] \succeq 0\}$$

and

$$\{(X, Y) \in \mathbf{S}_+^m \times \mathbf{S}_+^m : X \preceq Y^{1/2}\} = \{(X, Y) : \exists Z : 0 \preceq X \preceq Z, \left[\begin{array}{c|c} Y & Z \\ \hline Z & I \end{array} \right] \succeq 0\}$$

both are SDr. These sets are different:

$$0 \preceq X, X^2 \preceq Y \Rightarrow X \preceq Y^{1/2}, \text{ but } 0 \preceq X \preceq Y^{1/2} \not\Rightarrow X^2 \preceq Y$$

$$\left[0 \preceq \underbrace{\begin{bmatrix} 6 & 0 \\ 0 & 1 \end{bmatrix}}_X \preceq \underbrace{\begin{bmatrix} 12 & 8 \\ 8 & 12 \end{bmatrix}}_{Y^{1/2}}, \text{ but } \text{Det}\left(\underbrace{\begin{bmatrix} 172 & 192 \\ 192 & 207 \end{bmatrix}}_{Y-X^2}\right) = -1260 < 0! \right]$$

Reason for “pathology”: When $m \geq 2$, the mapping $X \mapsto X^{1/2}$ is \succeq -monotone on \mathbf{S}_+^m , while the mapping $X \mapsto X^2$ is not \succeq -monotone on \mathbf{S}_+^m .

Sums-of-Squares

Situation: We are given real-valued functions $\phi_0(x) \equiv 1, \phi_1(x), \dots, \phi_d(x)$ on some set X . These data specify the linear space Φ of functions $\phi(\cdot)$ which can be represented as linear combinations of $\phi_i(\cdot)$ and their pairwise products, or, which is the same due to $\phi_0(\cdot) \equiv 1$, as linear combinations of the pairwise products $\phi_i\phi_j$:

$$\Phi = \{f(\cdot) = \sum_{i,j=0}^d c_{ij}\phi_i(\cdot)\phi_j(\cdot)\}$$

W.l.o.g. we can assume that $c_{ij} = c_{ji}$. Note that Φ is the image of \mathbf{S}^{d+1} under the linear mapping

$$\mathbf{S}^{d+1} \ni C = [c_{ij}]_{0 \leq i,j \leq d} \mapsto \mathcal{A}[C](\cdot) = \sum_{i,j} c_{ij}\phi_i(\cdot)\phi_j(\cdot)$$

$$\mathbf{S}^{d+1} \ni C = [c_{ij}]_{0 \leq i, j \leq d} \mapsto \mathcal{A}[C](\cdot) = \sum_{i, j} c_{ij} \phi_i(\cdot) \phi_j(\cdot) \text{ \& } \Phi = \mathcal{A}[\mathbf{S}^{d+1}]$$

Observation: Sums of squares of linear combinations of functions ϕ_0, \dots, ϕ_d are exactly the elements of the image of the positive semidefinite cone \mathbf{S}_+^{d+1} under the mapping \mathcal{A} .

Indeed, $[\sum_i \lambda_i \phi_i(\cdot)]^2 = \mathcal{A}[\lambda \lambda^T]$, and the matrices from \mathbf{S}_+^{d+1} are nothing but sums of dyadic matrices.

Corollary: The set of (arrays of coefficients of) algebraic polynomials which are sums of squares of linear combinations of given algebraic polynomials $\phi_0(\cdot) \equiv 1, \phi_1(\cdot), \dots, \phi_d(\cdot)$ on \mathbf{R}^n is SDr.

Indeed, this set is the image of \mathbf{S}_+^{d+1} under linear mapping $\mathcal{A}[\cdot]$.

Conclusion: A *sufficient* condition for a function $f \in \Phi$ to be nonnegative on X is the possibility to find a $C \in \mathbf{S}^{d+1}$ such that

$$\mathcal{A}[C] = f \text{ \& } C \succeq 0. \quad (!)$$

When $X = \mathbf{R}^n$ and all ϕ_i are polynomials, (!) is a semidefinite feasibility problem.

Nonnegative polynomials

♣ For every positive integer k , the following sets are SDr:

— The set $P_{2k}^+(\mathbf{R})$ of coefficients of algebraic polynomials of degree $\leq 2k$ which are nonnegative on the entire axis:

$$P_{2k}^+ = \left\{ p = (p_0, \dots, p_{2k})^T : \exists Q = [Q_{ij}]_{i,j=0}^k \in \mathbf{S}_+^{k+1} : p_\ell = \sum_{i+j=\ell} Q_{ij}, \ell = 0, 1, \dots, 2k \right\}$$

Equivalently: A polynomial $p(t)$ of degree $\leq 2k$ is nonnegative on \mathbf{R} *iff* it can be obtained from $Q \in \mathbf{S}_+^{k+1}$ according to

$$p(t) = [1; t; t^2; \dots; t^k]^T Q [1; t; t^2; \dots; t^k]$$

— The set $P_k^+(\mathbf{R}_+)$ of coefficients of algebraic polynomials of degree $\leq k$ which are nonnegative on the nonnegative ray \mathbf{R}_+

— The set $P_k^+([0, 1])$ of coefficients of algebraic polynomials of degree $\leq k$ which are nonnegative on the segment $[0, 1]$

— The set $T_k^+(\Delta)$ of coefficients of trigonometric polynomials of degree $\leq k$, $p(\phi) = p_0 + \sum_{\ell=1}^k [p_{\ell,c} \cos(\ell\phi) + p_{\ell,s} \sin(\ell\phi)]$, which are nonnegative on a given segment $\Delta \in [-\pi, \pi]$.

♣ As a corollary, for every segment $\Delta \subset \mathbb{R}$ and every positive integer k , the function

$$f(p) = \max_{t \in \Delta} p(t)$$

of the vector p of coefficients of an algebraic (or a trigonometric) polynomial $p(\cdot)$ of degree $\leq k$ is SDr.

Indeed, $\tau \geq f(p)$ if and only if the polynomial $q_{p,\tau}(t) = \tau - p(t)$ of t is non-negative on Δ , and the coefficients of q are affine in τ and the coefficients of p .

- **SDR of the cone $P_{2k}^+(\mathbf{R})$:** Consider the linear mapping Π from the space S^{k+1} to the space of polynomials of degree $\leq 2k$:

$$\Pi([a_{ij}]_{i,j=0}^k) = \sum_{i,j=0}^k a_{ij} t^{i+j}$$

Observation: The images of dyadic matrices aa^T under the mapping Π are exactly squares of polynomials of degree $\leq k$:

$$\Pi(aa^T) = \sum_{i,j=0}^k a_i a_j t^{i+j} = \left(\sum_{i=0}^k a_i t^i \right)^2.$$

- The positive semidefinite cone is exactly the set of sums of dyadic matrices. Therefore, by Observation, the image of positive semidefinite cone under the mapping Π is exactly the set of polynomials of degree $\leq 2k$ which are *sums of squares*. It remains to note that *A univariate polynomial is nonnegative on the entire axis iff it is sum of squares*, whence

$$P_{2k}^+(\mathbf{R}) = \Pi(S_+^{k+1}),$$

and thus P_{2k}^+ is SDr.

- SDR of $P_{2k}^+(\mathbf{R})$ induces all other SDRs we need, namely
 - SDR of $P_k^+(\mathbf{R}_+)$ due to

$$p(t) \in P_k^+(\mathbf{R}_+) \Leftrightarrow \pi[p](t) \equiv p(t^2) \in P_{2k}^+(\mathbf{R}),$$

- SDR of $P_k^+([0, 1])$ due to

$$p(t) \in P_k^+([0, 1]) \Leftrightarrow \psi[p](t) \equiv (1 + t^2)^k p\left(\frac{t^2}{1 + t^2}\right) \in P_{2k}^+(\mathbf{R})$$

- SDR of $T_k(\Delta)$ due to

$$p(\phi) \in T_k(\Delta) \Leftrightarrow \theta[p](t) \equiv (1 + t^2)^k p(2 \operatorname{atan}(t)) \in P_{2k}^+(\widehat{\Delta}),$$

$$\widehat{\Delta} = \{t = \tan(\phi/2), \phi \in \Delta\}$$

and the coefficients of $\pi[p]$, $\psi[p]$, $\theta[p]$ are affine in the coefficients of p .

- Why a nonnegative on the axis polynomial is a sum of squares?
Assume a polynomial

$$p(t) = a(t - s_1) \dots (t - s_n)$$

of certain degree n is nonnegative on the entire axis. Then

- the degree is even,
- the leading coefficient a is positive,
- all real roots, if any, are of even multiplicities.

If z, z^* is a conjugate pair of complex roots, then the corresponding factor $(t - z)(t - z^*)$ in p is a sum of squares of a linear function and a real.

Thus, p is *the product of sums of squares of polynomials*, and such a product again is a sum of squares of polynomials.

- In fact, our reasoning says that p is a product of factors which are sums of *at most two squares* each. As a result, p itself is a sum of *just two* squares, due to the identity

$$(a^2 + b^2)(c^2 + d^2) = (ac - bd)^2 + (ad + bc)^2.$$

SDP models in Engineering

A. Dynamic Stability in Mechanics. The “free” (when no external forces are applied) motions of linearly elastic mechanical systems (buildings, bridges, masts, etc.) are governed by the Newton Law in the form:

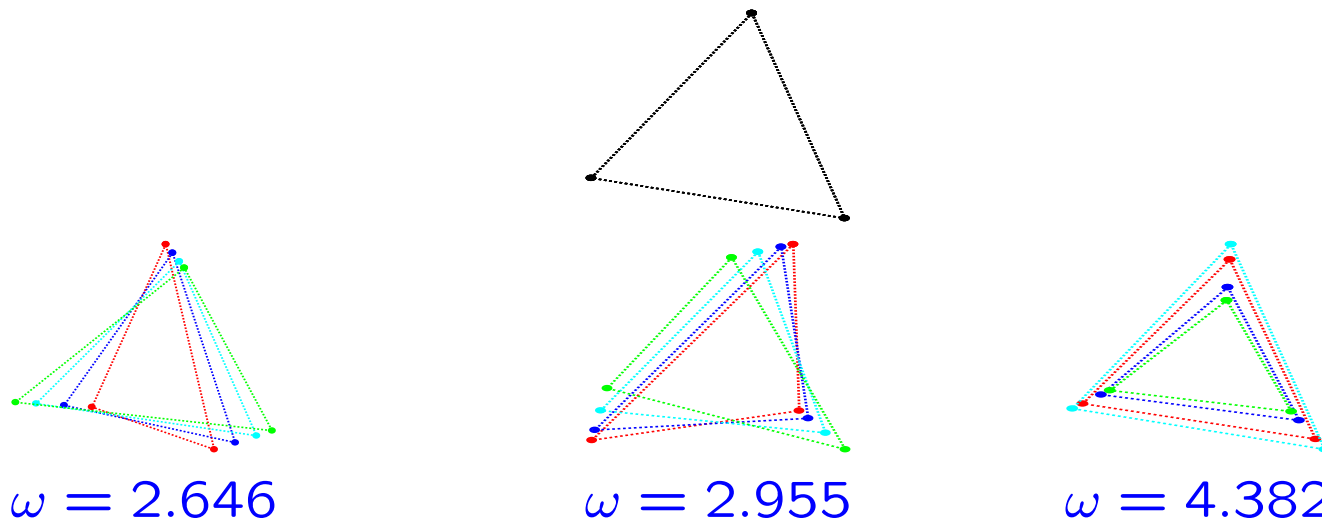
$$M \frac{d^2}{dt^2} x(t) = -Ax(t) \quad (\text{NL})$$

where

- $x(t)$ is the state of the system — block-vector of deviations of system’s “atoms” from their equilibrium positions;
- $M \succ 0$ is the *mass* matrix;
- $A \succeq 0$ is the *stiffness* matrix; $\frac{1}{2}x^T Ax$ is the potential energy of the system at state x .
- It is easily seen that every solution to (NL) is linear combination of basic harmonic oscillations (“modes”)

$$\cos(\omega_\ell t) \vec{f}_\ell, \sin(\omega_\ell t) \vec{f}_\ell$$

where the eigenfrequencies ω_ℓ are square roots of the eigenvalues $\lambda(A : M)$ of the matrix pencil $[M, A]$, and f_ℓ are eigenvectors of the pencil.



Top: equilibrium position of spring triangle (3 unit masses linked by springs)
 Bottom: “nontrivial” modes of the triangle (positions at 4 time instants)
 There are 3 modes more with $\omega = 0$ (coming from shifts and rotation)

- A typical Dynamic Stability specification is a lower bound on the eigenfrequencies:

$$\lambda_{\min}(A : M) \geq \lambda_*,$$

which is the matrix inequality

$$A \succeq \lambda_* M. \tag{S}$$

- When A and M are affine in the design variables, (S) is an LMI!

B. Structural Design. Consider a linearly elastic mechanical system \mathcal{S} with stiffness matrix $A \succ 0$ loaded by an external load f (block-vector of external physical forces acting at system's “atoms”). Under the load, the system deforms until the tensions caused by the deformation compensate the external forces. The corresponding *equilibrium displacement* x_f solves the *equilibrium equation*

$$Ax = f \quad [\Rightarrow x_f = A^{-1}f]$$

The *compliance* of \mathcal{S} w.r.t. load f is the potential energy

$$\text{Compl}_f = \frac{1}{2}x_f^T Ax_f = \frac{1}{2}f^T A^{-1}f$$

stored in the system in the corresponding equilibrium. The compliance quantifies the “rigidity” of \mathcal{S} w.r.t. f : the less is the compliance, the better \mathcal{S} withstands the load.

♣ In a typical Structural Design problem, we are given

- a stiffness matrix $A = A(t)$ *affinely* depending on a vector t of design parameters,
- a collection f_1, \dots, f_k of “loading scenarios”,
- a set \mathcal{T} of allowed values of t

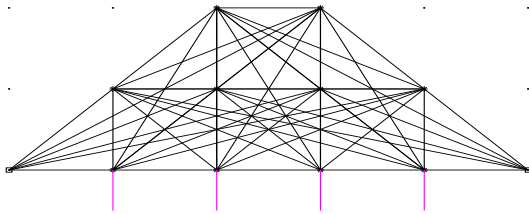
and are seeking for the design $t \in \mathcal{T}$ which results in the smallest possible worst-case, w.r.t. the scenarios, compliance, thus arriving at the optimization problem

$$\min_{t \in \mathcal{T}} \max_{\ell=1, \dots, k} \frac{1}{2} f_{\ell}^T A^{-1}(t) f_{\ell}.$$

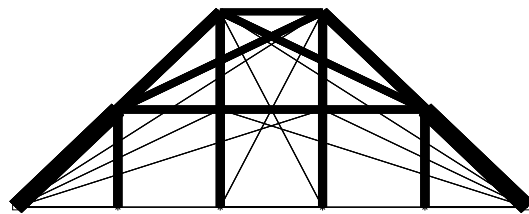
$$\min_{t \in \mathcal{T}} \max_{\ell=1, \dots, k} \frac{1}{2} f_{\ell}^T A^{-1}(t) f_{\ell}. \quad (\text{SD})$$

- When \mathcal{T} is SDr, problem (SD) becomes the semidefinite program

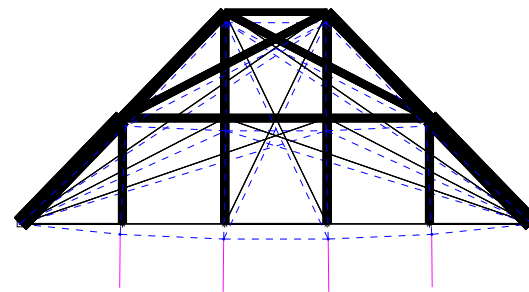
$$\min_{t, \tau} \left\{ \tau : \begin{bmatrix} 2\tau & f_{\ell}^T \\ f_{\ell} & A(t) \end{bmatrix} \succeq 0, \ell = 1, \dots, k, t \in \mathcal{T} \right\}$$



Data for Bridge Design problem [12 nodes, 51 tentative bars, 4-force load]

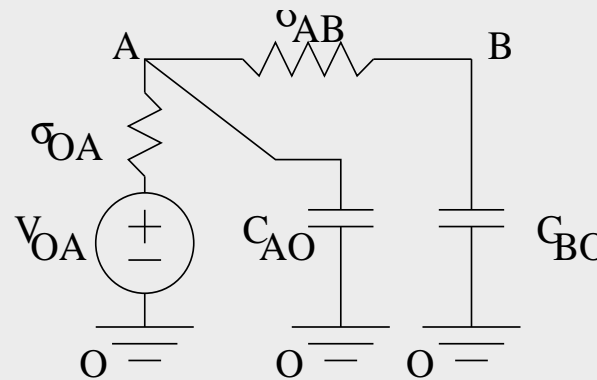


Optimal bridge (29 bars)



Equilibrium displacement

C. Boyd's Time Constant of an RC circuit. Consider a circuit comprised of (a) resistors, (b) capacitors, and (c) resistors in serial connection with outer voltages:



A simple circuit

Element OA: outer supply of voltage V_{OA} and resistor with conductance σ_{OA}

Element AO: capacitor with capacitance C_{AO}

Element AB: resistor with conductance σ_{AB}

Element BO: capacitor with capacitance C_{BO}

♣ A chip is a complicated RC circuit where the outer voltages are switching, at certain frequency, between several constant values. In order for chip to work reliably, the time of transition to the steady-state corresponding to given outer voltages should be much less than the time between switches of the voltages. How to model this crucial requirement?

- In an RC circuit, the transition period is governed by the Kirchhoff laws which result in the dynamics

$$C\dot{w} = -Rw \quad (\text{H})$$

where

- w is the difference between the current state of the circuit and its steady state;
- $C \succ 0$ is given by circuit's topology and the capacitances of the capacitors and is affine in the capacitances;
- $R \succ 0$ is given by circuit's topology and the conductances of the resistors and is affine in the conductances.

The space of solutions to (H) is spanned by functions

$$w_\ell(t) = \exp\{-\lambda_\ell t\} f_\ell,$$

where λ_ℓ are the eigenvalues of the matrix pencil $[C, R]$.

- $\lambda_{\min}(R : C)$ can be viewed as the “decay rate” for (H): the “duration” of the transition period is of order of $\lambda_{\min}^{-1}(R : C)$.

S. Boyd has proposed to use $\lambda_{\min}^{-1}(R : C)$ as a “time constant” for an RC circuit and to model a lower bound on the speed of the circuit (\equiv an upper bound on the duration of the transition period) as a lower bound on $\lambda_{\min}(R : C)$, i.e., as the matrix inequality

$$R \succeq \lambda_* C. \quad (\text{B})$$

When R and C are affine in the design variables, (B) becomes an LMI, which allows to pose numerous circuit design problems with bounds on the speed as SDPs.

D. Lyapunov Stability Analysis. Consider an uncertain time varying linear dynamical system

$$\dot{x}(t) = A(t)x(t) \quad (\text{ULS})$$

where

- $x(t) \in \mathbf{R}^n$ is the state vector at time t
- $A(t)$ takes values in a given *uncertainty set* $\mathcal{U} \subset \mathbf{R}^{n \times n}$
- ♣ (ULS) is called *stable*, if all trajectories of the system converge to 0 as $t \rightarrow \infty$:

$$A(t) \in \mathcal{U} \ \forall t \geq 0, \ \dot{x}(t) = A(t)x(t) \Rightarrow \lim_{t \rightarrow \infty} x(t) = 0.$$

How to certify stability?

- **Standard sufficient stability condition** is the existence of *Lyapunov Stability Certificate* – a matrix $X \succ 0$ such that the function $L(x) = x^T X x$ decreases exponentially along the trajectories:

$$\begin{aligned} \exists \alpha > 0 : \quad & \frac{d}{dt} L(x(t)) \leq -\alpha L(x(t)) \text{ for all trajectories} \\ \Big[\Rightarrow L(x(t)) \leq \exp\{-\alpha t\} L(x(0)) \Rightarrow x(t) \rightarrow 0, \ t \rightarrow \infty \Big] \end{aligned}$$

For a time-invariant system, this condition is necessary and sufficient for stability.

♣ **Question:** When $\alpha > 0$ is such that

$$\frac{d}{dt}L(x(t)) \leq -\alpha L(x(t)) \text{ for all trajectories } \dot{x}(t) = A(t)x(t), A(t) \in \mathcal{U} \quad ?$$

♣ **Answer:**

$$\begin{aligned} \frac{d}{dt} (x^T(t)Xx(t)) &= (\dot{x}(t))^T Xx(t) + x^T(t)X\dot{x}(t) \\ &= x^T(t)A^T(t)Xx(t) + x^T(t)XA(t)x(t) \\ &= x^T(t) [A^T(t)X + XA(t)] x(t) \end{aligned}$$

Thus,

$$\begin{aligned} \frac{d}{dt}L(x(t)) &\leq -\alpha L(x(t)) \text{ for all trajectories} \\ \Leftrightarrow x^T(t) [A^T(t)X + XA(t)] x(t) &\leq -\alpha x^T(t)Xx(t) \text{ for all trajectories} \\ \Leftrightarrow A^T X + XA &\preceq -\alpha X \quad \forall A \in \mathcal{U} \end{aligned}$$

♣ Thus,

$$\begin{aligned} \exists(\alpha > 0, X \succ 0) : \frac{d}{dt} (x^T(t)Xx(t)) &\leq -\alpha (x^T(t)Xx(t)) \text{ for all trajectories} \\ \Leftrightarrow \exists(\alpha > 0, X \succ 0) : A^T X + XA &\preceq -\alpha X \quad \forall A \in \mathcal{U} \\ \Leftrightarrow \exists X : X \succeq I, A^T X + XA &\preceq -I \quad \forall A \in \mathcal{U} \end{aligned}$$

- The existence of a Lyapunov Stability Certificate is equivalent to solvability of the *semi-infinite* system of LMIs in matrix variable X :

$$X \succeq I; \quad A^T X + X A \preceq -I \quad \forall (A \in \mathcal{U}) \quad (\text{L})$$

- Every solution to (L) is a Lyapunov Stability Certificate for the uncertain dynamical system

$$\dot{x}(t) = A(t)x(t) \quad [A(t) \in \mathcal{U} \forall t]$$

- In some cases, the semi-infinite system of LMIs is equivalent to a usual system of LMIs, so that search for a Lyapunov Stability Certificate reduces to solving an SDP.

Example 1: Polytopic uncertainty


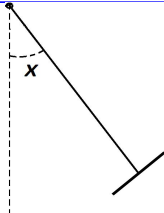
$$\mathcal{U} = \text{Conv}\{A_1, \dots, A_L\}.$$

In this case (L) clearly is equivalent to the finite system of LMIs

$$X \succeq I; \quad A_\ell^T X + X A_\ell \preceq -I, \ell = 1, \dots, L.$$

Illustration: Why can we swing on a swing, or Parametric Resonance

♠ Free motion of swing with friction is

$$\begin{aligned} \dot{x} &= v \\ \dot{v} &= -\alpha x - \beta v \end{aligned} \Leftrightarrow \begin{bmatrix} \dot{x} \\ \dot{v} \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -\alpha & -\beta \end{bmatrix} \cdot \begin{bmatrix} x \\ v \end{bmatrix}$$

• x : deviation of swing from equilibrium • $\alpha > 0$: elasticity • $\beta > 0$: friction

- This system is stable: all trajectories tend to 0 as $t \rightarrow \infty$.

Question: How can we swing on swing without external assistance and make oscillations larger and larger?

Answer: We do not sit in a fixed position: moving our body, we make the effective length of the rope, and thus the elasticity α , periodic function of time. As a result, time-invariant system becomes time-varying one and loses stability, provided the uncertainty range is not too small.

◇ A smart policy of making swing unstable is

- to reduce elasticity as much as possible when moving away from equilibrium, when elasticity slows us down
- to increase elasticity as much as possible when moving towards equilibrium, when elasticity accelerates us

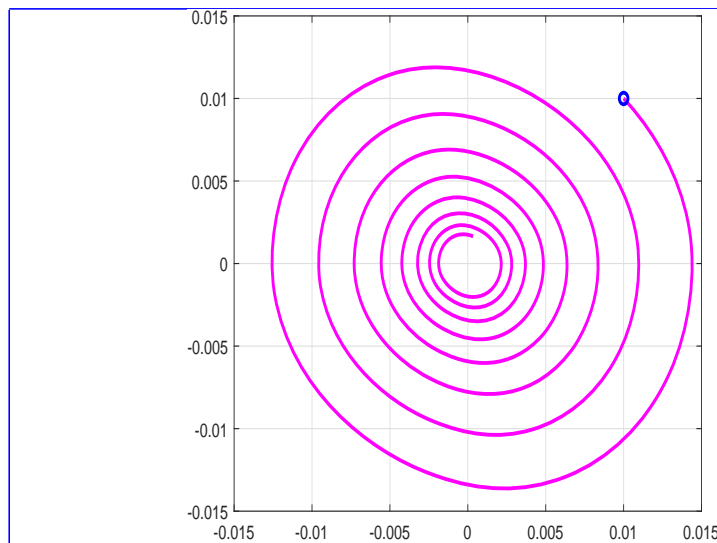
♠ The observed phenomenon – instability of uncertain dynamical system with all instances certain – is called *parametric resonance*.

Numerical illustration: The nominal system is

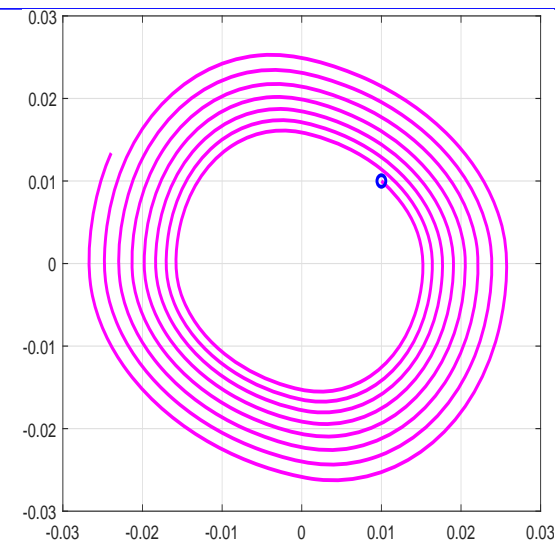
$$\begin{bmatrix} \dot{x} \\ \dot{v} \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -1 & -0.2 \end{bmatrix} \cdot \begin{bmatrix} x \\ v \end{bmatrix}$$

- Assuming that the only uncertain element in the matrix of the system is the elasticity (polytopic uncertainty!), Lyapunov Stability Analysis shows that the largest range of elasticity around its nominal value -1 allowing for Lyapunov Stability Certificate is $\{\alpha : |\alpha + 1| \leq \Delta_* = 0.198\}$.

This is what may happen with time-varying system:



range of elasticity: $|\alpha + 1| \leq 0.9\Delta_*$
 magnitude of oscillations goes to 0 as $t \rightarrow \infty$



range of elasticity: $|\alpha + 1| \leq 1.75\Delta_*$
 magnitude of oscillations goes to ∞ as $t \rightarrow \infty$

Phase portraits $[x(t); v(t)]$ of time-varying swing. \circ : starting point

- **Example 2: Norm-bounded uncertainty**

$$\mathcal{U} = \{A = A_0 + P\Delta Q : \Delta \in \mathbb{R}^{p \times q}, \|\Delta\| \leq 1\} \quad (\text{NB})$$

- **Illustration:** Consider a controlled linear time-invariant dynamical system

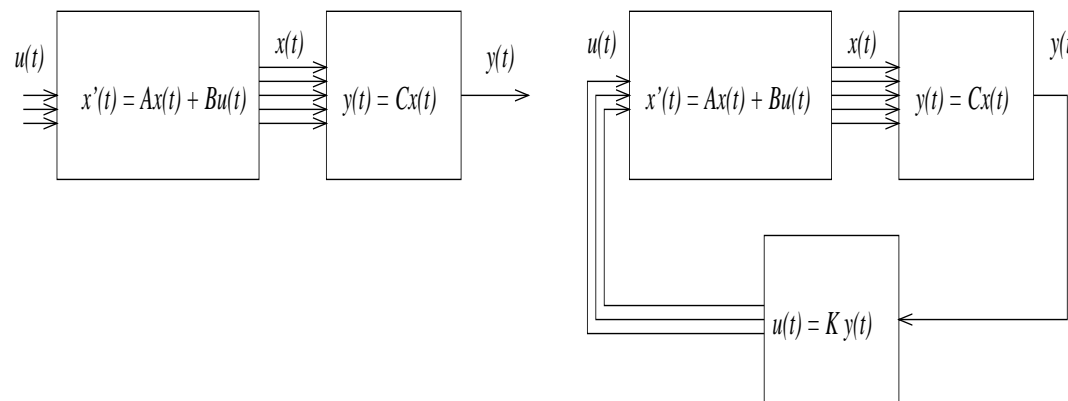
$$\dot{x}(t) = Ax(t) + Bu(t)$$

$$y(t) = Cx(t)$$

- x : state ● u : control ● y : observed output

“closed” by a feedback

$$u(t) = Ky(t).$$



Open loop (left) and closed loop (right) systems

$$\begin{aligned}\dot{x}(t) &= Ax(t) + Bu(t) \\ y(t) &= Cx(t) \\ u(t) &= Ky(t)\end{aligned}$$

- The resulting *closed loop* system is given by

$$\dot{x}(t) = \hat{A}x(t), \quad \hat{A} = A + BKC \quad (1)$$

Assuming that A , B , C are certain, and feedback matrix K is drifting around nominal feedback K^* :

$$K = K^* + \Delta,$$

where $\|\Delta\|$ does not exceed a given level, \hat{A} runs through uncertainty set of the form

$$\mathcal{U} = \{A = A_0 + P\Delta Q : \Delta \in \mathbf{R}^{p \times q}, \|\Delta\| \leq 1\} \quad (\text{NB})$$

$$\mathcal{U} = \{A = A_0 + P\Delta Q : \Delta \in \mathbf{R}^{p \times q}, \|\Delta\| \leq 1\} \quad (\text{NB})$$

Proposition. *With the uncertainty set (NB), the Lyapunov Stability Certificate semi-infinite system of LMIs*

$$X \succeq I; \quad A^T X + X A \preceq -I \quad \forall (A \in \mathcal{U}) \quad (L)$$

is equivalent to the LMIs

$$X \succeq I, \quad \left[\begin{array}{c|c} -I - A_0^T X - X A_0 - \lambda Q^T Q & -X P \\ \hline -P^T X & \lambda I \end{array} \right] \succeq 0$$

in variables X, λ .

- An instrumental role in the proof of Proposition is played by the following statement which is extremely useful by its own right:

S-Lemma: Consider a homogeneous quadratic inequality

$$x^T A x \geq 0 \tag{A}$$

which is strictly feasible: $\bar{x}^T A \bar{x} > 0$ for certain \bar{x} .

A homogeneous quadratic inequality

$$x^T B x \geq 0 \tag{B}$$

is a consequence of (A) *iff* it is a “linear” consequence of (A), i.e., *iff* (B) can be obtained by summing up a nonnegative multiple of (A) and identically true homogeneous quadratic inequality, or, which is the same, *iff*

$$\exists(\lambda \geq 0) : \quad B \succeq \lambda A.$$

♠ S-Lemma admits inhomogeneous version:

Inhomogeneous S-Lemma: Consider a quadratic inequality

$$x^T A x + 2a^T x + \alpha \geq 0 \quad (A)$$

which is strictly feasible: $\bar{x}^T A \bar{x} + 2a^T \bar{x} + \alpha > 0$ for certain \bar{x} .

A quadratic inequality

$$x^T B x + 2b^T x + \beta \geq 0 \quad (B)$$

is a consequence of (A) *iff* the homogenized version

$$x^T B x + 2tb^T x + \beta t^2 \geq 0 \quad (B_h)$$

of (B) is a consequence of the homogenized version

$$x^T A x + 2ta^T x + \alpha t^2 \geq 0 \quad (A_h)$$

of (A), or, which is the same by Homogeneous S-Lemma, *iff*

$$\exists(\lambda \geq 0) : \left[\begin{array}{c|c} B - \lambda A & b - \lambda a \\ \hline b^T - \lambda a^T & \beta - \lambda \alpha \end{array} \right] \succeq 0.$$

Proof of Proposition is given by the following fact:

(!) Assume that $E \neq 0$. Then

$$\begin{aligned} C + D^T \Delta E + E^T \Delta^T D &\succeq 0 \quad \forall(\Delta, \|\Delta\| \leq 1) \\ \Leftrightarrow \exists \lambda : \left[\begin{array}{c|c} C - \lambda E^T E & D^T \\ \hline D & \lambda I \end{array} \right] &\succeq 0 \end{aligned}$$

In particular, when $Q \neq 0$, one has

$$\begin{aligned} &= [-I - A_0^T X - X A_0] + [-P^T X]^T \Delta Q + Q^T \Delta^T [-P^T X] \\ &\quad \overbrace{-I - [A_0 + P \Delta Q]^T X - X [A_0 + P \Delta Q]} \succeq 0 \quad \forall(\Delta, \|\Delta\| \leq 1) \\ &\Leftrightarrow \exists \lambda : \left[\begin{array}{c|c} -I - A_0^T X - X A_0 - \lambda Q^T Q & -X P \\ \hline -P^T X & \lambda I \end{array} \right] \succeq 0 \end{aligned}$$

Proof of (!):

$$\begin{aligned} &C + D^T \Delta E + E^T \Delta^T D \succeq 0 \quad \forall(\Delta, \|\Delta\| \leq 1) \\ &\Leftrightarrow \xi^T C \xi + 2\xi^T D^T \underbrace{[\Delta E \xi]}_{\eta} \geq 0 \quad \forall \xi \forall(\Delta, \|\Delta\| \leq 1) \\ &\Leftrightarrow \xi^T C \xi + 2\xi^T D^T \eta \geq 0 \quad \forall \xi \forall(\eta, \|\eta\|_2 \leq \|E\xi\|_2) \\ &\Leftrightarrow \xi^T C \xi + 2\xi^T D^T \eta \geq 0 \quad \forall(\xi, \eta : \xi^T E^T E \xi - \eta^T \eta \geq 0) \\ &\underbrace{\Leftrightarrow}_{[\mathcal{S}\text{-Lemma}]} \exists \lambda \geq 0 : \left[\begin{array}{c|c} C & D^T \\ \hline D & \end{array} \right] \succeq \lambda \left[\begin{array}{c|c} E^T E & \\ \hline & -I \end{array} \right] \end{aligned}$$

SDP approximations of computationally intractable problems

A. SDP relaxations in Combinatorics. In a typical combinatorial problem, we are interested to minimize a “simple” function over a *discrete* set, e.g.

- **Shortest Path:** Given a graph with arcs assigned nonnegative integer lengths and two nodes a, b , find the shortest path from a to b or detect that no path exists.

- **Integer Linear Programming:**

$$\min_x \{c^T x : Ax \leq b, x \in \mathbb{Z}^n\}$$

$[\mathbb{Z}^n : n\text{-dimensional integral vectors}]$

(all entries in A, b, c are integral)

- **Boolean Programming:**

$$\min_x \{c^T x : Ax \leq b, x \in \mathbb{B}^n\}$$

$[\mathbb{B}^n : n\text{-dimensional 0-1 vectors}]$

(all entries in A, b, c are integral)

- **Knapsack problem:**

$$\max_x \left\{ \sum_{i=1}^n c_i x_i : \sum_{i=1}^n a_i x_i \leq b, x_i \in \{0; 1\} \right\}$$

(c_i, a_i, b are positive integers)

- **“Stones”:** Given n stones of positive integer weights a_1, \dots, a_n , check whether you can partition them into two groups of equal weight, i.e., check whether the linear equation

$$\sum_{i=1}^n a_i x_i = 0$$

has a solution with $x_i = \pm 1$.

- ♣ As far as solution methods are concerned, the majority of generic combinatorial problems
 - are reducible to each other and are therefore of basically the same complexity
 - are **NP-complete** – “as difficult as a problem can be”.
- In the above list the only “easy” – known to be efficiently solvable – problem is Shortest Path, while all other problems are of basically the same “maximal possible” complexity.

- Most of solution methods for difficult combinatorial problems heavily use *bounding*. Bounding techniques are aimed at building “efficiently computable” lower bounds for the optimal value in combinatorial problem

$$\min_x \{f(x) : x \in \mathcal{X}\}. \quad (\text{Ini})$$

A typical way to find such a bound is given by *relaxation*: we replace \mathcal{X} with a *larger* set \mathcal{X}^+ such that the problem

$$\min_x \{f(x) : x \in \mathcal{X}^+\} \quad (\text{Rel})$$

is efficiently solvable, and use the optimal value of (Rel) as a lower bound on the optimal value of (Ini):

$$\mathcal{X} \subset \mathcal{X}^+ \Rightarrow \text{Opt}(\text{Rel}) \leq \text{Opt}(\text{Ini}).$$

♣ **Generic Example:** Let (Ini) be quadratic quadratically constrained problem:

$$\text{Opt} = \min_x \left\{ x^T Q_0 x + 2b_0^T x + c_0 : \begin{array}{l} f_i(x) = x^T Q_i x + 2b_i^T x + c_i \leq 0, i = 1, \dots, m \\ h_\ell(x) = x^T R_\ell x + 2d_\ell^T x + e_\ell = 0, \ell = 1, \dots, k \end{array} \right\} \quad (\text{Ini})$$

♠ **Note:** The scope of quadratic quadratically constrained problems is really huge:

- quadratic constraints allow to model discrete feasible sets:

$$x_i^2 = x_i \Leftrightarrow x_i \text{ is } 0 \text{ or } 1, \quad x_i^2 = 1 \Leftrightarrow x_i \text{ is } 1 \text{ or } -1$$

- problems with polynomial objective and constraints can be reduced to problems with *quadratic* objective and constraints, since polynomial monomials can be expressed by quadratic equalities and two-term products.

For example, given variables x, y, z , introducing variables \overline{xy} , \overline{xyy} , \overline{xyyz} and subject them to quadratic constraints

$$\overline{xy} = x \cdot y, \quad \overline{xyy} = \overline{xy} \cdot y, \quad \overline{xyyz} = \overline{xyy} \cdot z,$$

the monomial $x^2 y^2 z$ becomes just the product $x \cdot \overline{xyyz}$.

♠ Let (Ini) be quadratic quadratically constrained problem:

$$\text{Opt} = \min_x \left\{ x^T Q_0 x + 2b_0^T x + c_0 : \begin{array}{l} f_i(x) = x^T Q_i x + 2b_i^T x + c_i \leq 0, i = 1, \dots, m \\ h_\ell(x) = x^T R_\ell x + 2d_\ell^T x + e_\ell = 0, \ell = 1, \dots, k \end{array} \right\} \quad (\text{Ini})$$

$$X(x) = [x; 1][x; 1]^T = \left[\begin{array}{c|c} xx^T & x \\ \hline x^T & 1 \end{array} \right], \quad A_i = \left[\begin{array}{c|c} Q_i & b_i \\ \hline b_i^T & c_i \end{array} \right], \quad 0 \leq i \leq m, \quad B_\ell = \left[\begin{array}{c|c} R_\ell & d_\ell \\ \hline d_\ell^T & e_\ell \end{array} \right], \quad 1 \leq \ell \leq k,$$

we can write down (Ini) equivalently as

$$\min_X \left\{ \begin{array}{l} \text{Tr}(A_0 X) : \\ \text{Tr}(A_i X) \leq 0, i = 1, \dots, m, \\ \text{Tr}(B_\ell X) = 0, \ell = 1, \dots, k, \\ X \in \mathcal{X} \end{array} \right\}, \quad \mathcal{X} = \{X = X(x) : x \in \mathbb{R}^n\}. \quad (\text{Med})$$

• Matrices of the form $X(x) = [x; 1][x; 1]^T$ with $x \in \mathbb{R}^n$ are exactly $X \in \mathbb{S}^{n+1}$ satisfying the constraints

$$(a) \quad X \succeq 0 \quad (b) \quad X_{n+1, n+1} = 1 \quad (c) \quad \text{Rank}(X) = 1$$

(c) makes \mathcal{X} difficult nonconvex set. Dropping (c) — extending \mathcal{X} to the set

$$\mathcal{X}^+ = \{X \in \mathbb{S}^{n+1} : X \succeq 0, X_{n+1, n+1} = 1\}, \quad [\supset \mathcal{X}]$$

the semidefinite program

$$\text{Opt}_{\text{Rel}} = \min_X \left\{ \begin{array}{l} \text{Tr}(A_0 X) : \\ \text{Tr}(A_i X) \leq 0, i = 1, \dots, m, \\ \text{Tr}(B_\ell X) = 0, \ell = 1, \dots, k, \\ X \succeq 0, X_{n+1, n+1} = 1 \end{array} \right\} \quad (\text{Rel})$$

is a relaxation of (Ini).

♠ Another way to get the same relaxation is given by
Weak Lagrange Duality: Consider an optimization program

$$\text{Opt} = \min_x \left\{ f_0(x) : \begin{array}{l} f_i(x) \leq 0, \ i = 1, \dots, m; \\ h_\ell(x) = 0, \ \ell = 1, \dots, k. \end{array} \right\} \quad (\text{Ini})$$

Let

$$L(x; \lambda, \mu) = f_0(x) + \sum_{i=1}^m \lambda_i f_i(x) + \sum_{\ell=1}^k \mu_\ell h_\ell(x) \quad [\lambda_i \geq 0]$$

be the Lagrange function of (Ini). We clearly have

$$\lambda \geq 0, x \text{ feasible for (Ini)} \Rightarrow L(x; \lambda, \mu) \leq f_0(x)$$

and therefore

$$\lambda \geq 0 \Rightarrow F(\lambda, \mu) \equiv \inf_{x \in \mathbf{R}^n} L(x; \lambda, \mu) \leq \text{Opt}.$$

It follows that

$$\text{Opt}_{\text{Lag}} \equiv \sup_{\lambda \geq 0, \mu} F(\lambda, \mu) \leq \text{Opt}.$$

$$\begin{aligned}
(\text{Ini}) : \quad \text{Opt} &= \min_x \left\{ f_0(x) : \begin{array}{l} f_i(x) \leq 0, \quad i = 1, \dots, m; \\ h_\ell(x) = 0, \quad \ell = 1, \dots, k. \end{array} \right\} \\
\Rightarrow \quad L(x; \lambda, \mu) &= f_0(x) + \sum_i \lambda_i f_i(x) + \sum_\ell \mu_\ell h_\ell(x) \\
\Rightarrow \quad F(\lambda, \mu) &= \inf_{x \in \mathbf{R}^n} L(x; \lambda, \mu) \\
\Rightarrow \quad \boxed{\text{Opt}_{\text{Lag}} \equiv \sup_{\lambda \geq 0, \mu} F(\lambda, \mu) \leq \text{Opt}}
\end{aligned}$$

- **Shor's bounding scheme:** Assume that all functions $f_0, \dots, f_m, h_0, \dots, h_k$ are quadratic:

$$f_i(x) = x^T Q_i x + 2b_i^T x + c_i, \quad h_\ell = x^T R_\ell x + 2d_\ell^T x + e_\ell$$

and let us apply the Weak Duality:

$$\begin{aligned}
L(x; \lambda, \mu) &= f_0(x) + \sum_i \lambda_i f_i(x) + \sum_\ell \mu_\ell h_\ell(x) \\
&= x^T [Q(\lambda, \mu)] x + 2[q(\lambda, \mu)]^T x + r(\lambda, \mu) \\
Q(\lambda, \mu) &= Q_0 + \sum_{i \geq 1} \lambda_i Q_i + \sum_\ell \mu_\ell R_\ell, \\
q(\lambda, \mu) &= b_0 + \sum_{i \geq 1} \lambda_i b_i + \sum_\ell \mu_\ell d_\ell, \\
r(\lambda, \mu) &= c_0 + \sum_{i \geq 1} \lambda_i c_i + \sum_\ell \mu_\ell e_\ell
\end{aligned}$$

What is $\inf_x L(x; \lambda, \mu)$?

$$L(x; \lambda, \mu) = f_0(x) + \sum_i \lambda_i f_i(x) + \sum_\ell \mu_\ell h_\ell(x) = x^T [Q(\lambda, \mu)]x + 2[q(\lambda, \mu)]^T x + r(\lambda, \mu)$$

$$Q(\lambda, \mu) = Q_0 + \sum_{i \geq 1} \lambda_i Q_i + \sum_\ell \mu_\ell R_\ell, q(\lambda, \mu) = b_0 + \sum_{i \geq 1} \lambda_i b_i + \sum_\ell \mu_\ell d_\ell, r(\lambda, \mu) = c_0 + \sum_{i \geq 1} \lambda_i c_i + \sum_\ell \mu_\ell e_\ell$$

$$\text{Opt}_{\text{Lag}} = \max_{\lambda \geq 0, \mu} \{F(\lambda, \mu) := \inf_x \{L(x; \lambda, \mu)\}\}$$

Lemma: A quadratic form $x^T Q x + 2q^T x + r$ is $\geq s$ for all x iff $y^T Q y + 2tq^T y + rt^2 \geq st^2$ for all y, t (plug $x = y/t$), that is, iff

$$\left[\begin{array}{c|c} Q & q \\ \hline q^T & r - s \end{array} \right] \succeq 0.$$

By Lemma,

$$\inf_x L(x; \lambda, \mu) = \sup \left\{ s : \left[\begin{array}{c|c} Q(\lambda, \mu) & q(\lambda, \mu) \\ \hline q^T(\lambda, \mu) & r(\lambda, \mu) - s \end{array} \right] \succeq 0 \right\}$$

whence

$$\text{Opt}_{\text{Lag}} = \max_{\lambda, \mu, s} \left\{ s : \left[\begin{array}{c|c} Q(\lambda, \mu) & q(\lambda, \mu) \\ \hline q^T(\lambda, \mu) & r(\lambda, \mu) - s \end{array} \right] \succeq 0, \lambda \geq 0 \right\} \quad (\text{Lag})$$

and this optimal value is a lower bound for

$$\text{Opt} = \min_x \left\{ f_0(x) : \begin{array}{l} f_i(x) \leq 0, \quad i = 1, \dots, m; \\ h_\ell(x) = 0, \quad \ell = 1, \dots, k. \end{array} \right\}$$

$$[f_i(x) = x^T Q_i x + 2b_i^T x + c_i, \quad h_\ell = x^T R_\ell x + 2d_\ell^T x + e_\ell]$$

$$\begin{aligned} \text{Opt} = \min_x \left\{ f_0(x) : \begin{array}{l} f_i(x) \leq 0, \quad i = 1, \dots, m; \\ h_\ell(x) = 0, \quad \ell = 1, \dots, k. \end{array} \right\} \\ [f_i(x) = x^T Q_i x + 2b_i^T x + c_i, \quad h_\ell = x^T R_\ell x + 2d_\ell^T x + e_\ell] \end{aligned} \quad (\text{Ini})$$

The Semidefinite Relaxation and Shor's Bounding yield, respectively, the lower bounds

$$\begin{aligned} \text{Opt}_{\text{Rel}} = \min_X \left\{ \begin{array}{l} \text{Tr}(A_i X) \leq 0, \quad i = 1, \dots, m \\ \text{Tr}(B_\ell X) = 0, \quad \ell = 1, \dots, k \\ X \succeq 0, \quad X_{n+1, n+1} = 1 \end{array} \right\} \\ \left[A_i = \left[\begin{array}{c|c} Q_i & b_i^T \\ \hline b_i & c_i \end{array} \right], \quad i = 1, \dots, m, \quad B_\ell = \left[\begin{array}{c|c} R_\ell & d_\ell^T \\ \hline d_\ell & e_\ell \end{array} \right], \quad \ell = 1, \dots, k \right] \end{aligned} \quad (\text{Rel})$$

and

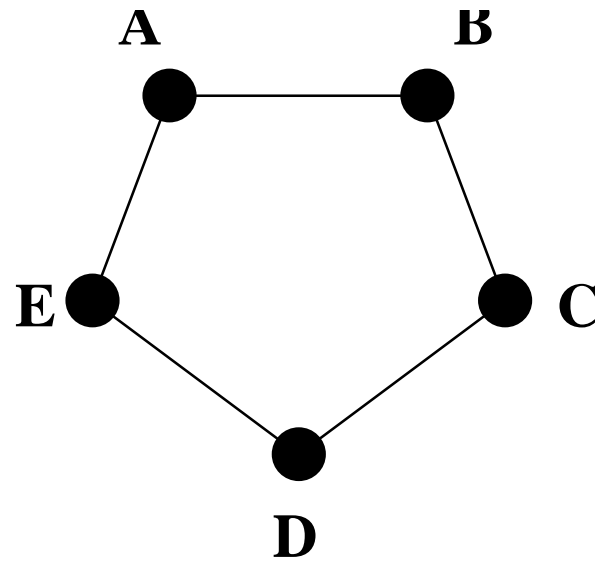
$$\begin{aligned} \text{Opt}_{\text{Lag}} = \max_{\lambda, \mu, s} \left\{ s : \left[\begin{array}{c|c} Q(\lambda, \mu) & q(\lambda, \mu) \\ \hline q^T(\lambda, \mu) & r(\lambda, \mu) - s \end{array} \right] \succeq 0, \quad \lambda \geq 0 \right\}, \\ \left[\begin{array}{lcl} Q(\lambda, \mu) & = & Q_0 + \sum_{i \geq 1} \lambda_i Q_i + \sum_{\ell} \mu_\ell R_\ell, \\ q(\lambda, \mu) & = & b_0 + \sum_{i \geq 1} \lambda_i b_i + \sum_{\ell} \mu_\ell d_\ell, \\ r(\lambda, \mu) & = & c_0 + \sum_{i \geq 1} \lambda_i c_i + \sum_{\ell} \mu_\ell e_\ell \end{array} \right] \end{aligned} \quad (\text{Lag})$$

on Opt.

- It is immediately seen that (Rel) is (equivalent to) the dual of (Lag), so that both bounds are the same (provided that one of the relaxations is essentially strictly feasible)!

Example: Lovasz ϑ -function

- A **graph** is a finite set of *nodes* linked by *arcs*. A subset S of the nodal set is called *independent*, if no pair of nodes from S are linked by an arc. The *stability number* $\alpha(\Gamma)$ of a graph Γ is the maximum cardinality of independent sets of nodes. E.g., the stability number of graph C_5



Graph C_5

is 2.

- To compute $\alpha(\Gamma)$ is an NP-complete combinatorial problem.

♠ **Shannon capacity** $\Theta(\Gamma)$ of a graph Γ is defined as follows. Imagine that the nodes are letters of an alphabet. We can send these letters through a communication channel. When passing through the channel, a letter may be corrupted by noise; as a result, two distinct letters on input to the channel may become the same on the output. We link every pair of letters with this property by an arc, thus getting a graph.

♥ Assume we are sending k -letter words, one letter per unit time, and want to avoid “misunderstandings” – the addressee should be capable to recognize which word was sent, without risk that “no!” will be read as “yes”.

To avoid misunderstandings, we should restrict the “dictionary” of k -letter words we actually use to be “independent” in the sense that no two distinct words from the dictionary, when sent through the channel, can produce the same output. If we agree with addressee what is the independent dictionary we use, no misunderstandings will occur.

♠ In order to fully utilize the capacity of the channel, it makes sense to use a maximum cardinality independent dictionary of k -letter words, let this cardinality be $f(k)$. It is clear that $f^{1/k}(k)$ is above bounded (e.g., by the number of letters) and that

$$f(k + l) \geq f(k)f(l)$$

(think about $(k + l)$ -letter words with the “ k -letter prefix” from the independent dictionary of cardinality $f(k)$, and the “ l -letter suffix” from the independent dictionary of cardinality $f(l)$). From these properties it follows that

$$\sup_{k \geq 1} f^{1/k}(k) = \lim_{k \rightarrow \infty} f^{1/k}(k) =: \sigma(\Gamma);$$

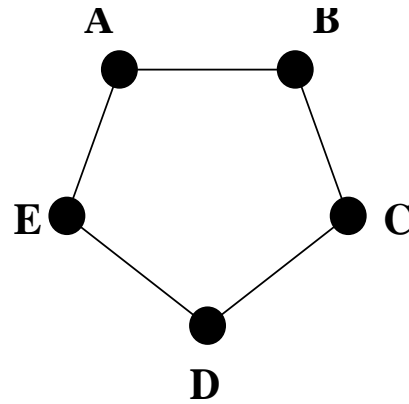
$\sigma(\Gamma)$ is called *Shannon capacity* of graph Γ .

● Since the maximum cardinality of independent single-letter dictionaries is the stability number of the graph, we have

$$\alpha(\Gamma) = f(1) \leq \sigma(\Gamma).$$

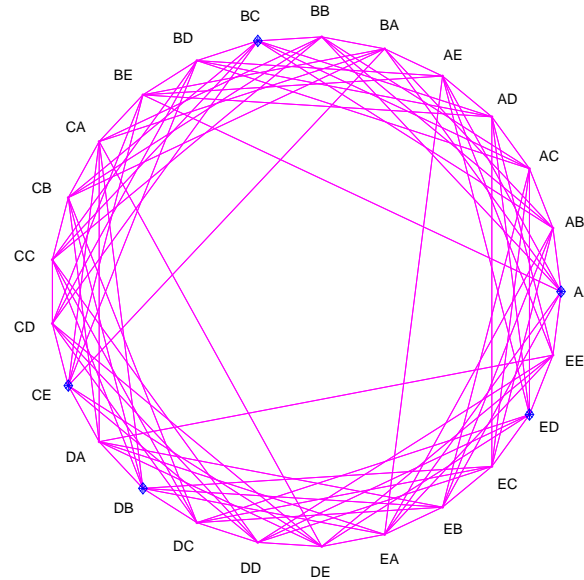
$$\alpha(\Gamma) \leq \sigma(\Gamma). \quad (*)$$

- Inequality (*) may be strict. E.g., $\alpha(C_5) = 2$:



Graph C_5

At the same time, for C_5 there exists independent dictionaries with 5 two-letter words, e.g., $\{AA, BC, CE, DB, ED\}$



Graph $C_5 \times C_5$

Thus,

$$\sigma(C_5) \geq \sqrt{f(2)} = \sqrt{5}.$$

The question whether this inequality is equality remained open for about 20 years!

- In early 70's, L. Lovasz found a computable upper bound $\vartheta(\Gamma)$ for $\alpha(\Gamma)$ and proved that

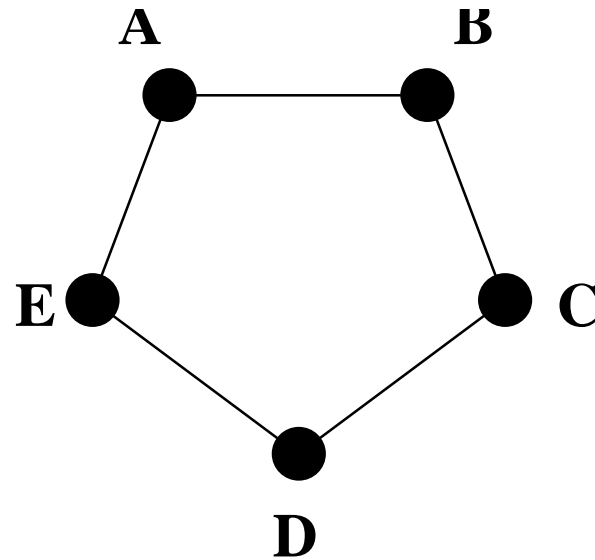
$$\alpha(\Gamma) \leq \sigma(\Gamma) \leq \vartheta(\Gamma)$$

(In particular, $\sqrt{5} \leq \sigma(C_5) \leq \vartheta(C_5) = \sqrt{5}$, whence $\sigma(C_5) = \sqrt{5}$).

- *By definition*, $\vartheta(\Gamma)$ is the optimal value in the following semidefinite program:

$$\min_{X \in \mathcal{L}} \lambda_{\max}(X) \equiv \min_{X \in \mathcal{L}, \mu} \{ \mu : \mu I \succeq X \} \quad (\text{Lov})$$

where \mathcal{L} is the set of all symmetric $n \times n$ matrices X (n is the number of nodes in the graph) such that $X_{ij} = 1$ when the nodes i, j are not adjacent.



Graph C_5

Example: For graph C_5 , the set \mathcal{L} is comprised of all matrices of the form

$$\begin{bmatrix} 1 & x_{AB} & 1 & 1 & x_{EA} \\ x_{AB} & 1 & x_{BC} & 1 & 1 \\ 1 & x_{BC} & 1 & x_{CD} & 1 \\ 1 & 1 & x_{CD} & 1 & x_{DE} \\ x_{EA} & 1 & 1 & x_{DE} & 1 \end{bmatrix}.$$

- The Lovasz upper bound on $\alpha(\Gamma)$ can be obtained from Shor's Bounding scheme.

Let the nodes of Γ be $1, \dots, n$.

- Observe that $\alpha(\Gamma)$ is the optimal value in the Boolean quadratic program:

$$\begin{array}{ll}
 (a) & \max_x \sum_{i=1}^n x_i \\
 (b) & 2x_i x_j = 0 \quad \forall \text{ adjacent } i, j \\
 (c) & x_i^2 - x_i = 0 \quad \Leftrightarrow \quad x_i \in \{0, 1\}
 \end{array} \tag{Stab}$$

- (c) associates with x the set of nodes $\{i : x_i = 1\}$;
- (b) says that the set $\{i : x_i = 1\}$ is independent;
- (a) counts the cardinality of $\{i : x_i = 1\}$.
- Applying Shor's scheme, we come to the "bounding program"

$$\min_{\mu, \nu, Y} \left\{ \mu : \left[\begin{array}{c|c} Y + \text{Diag}\{\nu\} & -\frac{1}{2}[\nu + \underline{1}] \\ \hline -\frac{1}{2}[\nu + \underline{1}]^T & \mu \end{array} \right] \succeq 0 \right\}, \quad \underline{1} = \begin{bmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{bmatrix} \tag{Lag}$$

$[\text{Opt(Lag)} \geq \alpha(\Gamma)]$

$$\min_{\mu, \nu, Y} \left\{ \mu : \begin{bmatrix} Y + \text{Diag}\{\nu\} & -\frac{1}{2}[\nu + \underline{1}] \\ -\frac{1}{2}[\nu + \underline{1}]^T & \mu \end{bmatrix} \succeq 0 \right\}, \quad \underline{1} = \begin{bmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{bmatrix} \quad (\text{Lag})$$

$$[\text{Opt}(\text{Lag}) \geq \alpha(\Gamma)]$$

- Applying Lemma on Schur Complement, we convert (Lag) to

$$\min_{\mu \geq 0, \nu, Y} \left\{ \mu : \begin{array}{l} \mu(Y + \text{Diag}\{\nu\}) \succeq \frac{1}{4}(\nu + \underline{1})(\nu + \underline{1})^T \\ Y_{ij} = 0 \quad \forall \text{ non-adjacent } i, j \end{array} \right\}$$

- Specifying ν -variables as ones, we can only increase the optimal value. The resulting problem is

$$\text{SDP} = \min_{\mu, Y} \left\{ \mu : \begin{array}{l} \mu I \succeq \overbrace{-\mu Y + \underline{1} \cdot \underline{1}^T}^X \\ Y_{ij} = 0 \quad \forall \text{ non-adjacent } i, j \end{array} \right\}$$

$$[\text{SDP} \geq \alpha(\Gamma)]$$

- When Y runs through the set of symmetric matrices such that $Y_{ij} = 0$ for non-adjacent i, j , X runs through the entire set of symmetric matrices with $X_{ij} = 1$ for non-adjacent i, j , so that

$$\text{SDP} = \min_{\mu, X} \left\{ \mu : \begin{array}{l} \mu I \succeq X \\ X_{ij} = 1 \quad \forall \text{ non-adjacent } i, j \end{array} \right\}$$

♠ How close is $\vartheta(\Gamma)$ to $\alpha(\Gamma)$?

- There exists an important class of *perfect graphs* for which $\vartheta(\Gamma) = \alpha(\Gamma)$
- However, for general-type graphs it may happen that

$$\vartheta(\Gamma) \gg \alpha(\Gamma).$$

Lovasz have proved that if Γ is an n -node graph and $\hat{\Gamma}$ is its complement (two distinct nodes are linked by arc in $\hat{\Gamma}$ iff they are not linked by arc in Γ), then

$$\vartheta(\Gamma)\vartheta(\hat{\Gamma}) \geq n \Rightarrow \max [\vartheta(\Gamma), \vartheta(\hat{\Gamma})] \geq \sqrt{n}.$$

On the other hand, for a random n -node graph Γ (probability for a pair $i < j$ to be linked by an arc is $\frac{1}{2}$) it holds

$$\max [\alpha(\Gamma), \alpha(\hat{\Gamma})] \leq O(\ln n)$$

with probability approaching 1 as $n \rightarrow \infty$.

Thus, for “typical” random graphs

$$\frac{\vartheta(\Gamma)}{\alpha(\Gamma)} \geq O\left(\frac{\sqrt{n}}{\ln n}\right).$$

B. Theorem of Goemans and Williamson. There exist hard combinatorial problems where bounds coming from semidefinite relaxations coincide with the actual optimal value within *absolute* constant factor. The most famous example is given by the MAXCUT problem which is as follows:

*Given a graph Γ with arcs assigned nonnegative weights a_{ij} ,
find a *cut* of maximal weight*

[A cut in a graph is partitioning (S, S') of the set of nodes into two non-overlapping subsets. The weight of a cut is the sum of weights of all arcs linking a node from S with a node from S'].

♠ MAXCUT is an NP-complete combinatorial problem which can be posed as quadratic program with variables ± 1 :

- We lose nothing by assuming that graph is complete (set $a_{ij} = 0$ for pairs i, j of nodes which in fact are not adjacent). Thus, assume that a_{ij} form a symmetric $n \times n$ matrix A with nonnegative entries and zero diagonal.

- A cut (S, S') can be represented by vector $x \in \mathbf{R}^n$ with $x_i = -1$ for $i \in S$ and $x_i = 1$ for $i \in S'$. With this representation, the weight of the cut is

$$\frac{1}{4} \sum_{i,j} a_{ij} (1 - x_i x_j) \quad (*)$$

- Thus, MAXCUT is the program

$$OPT = \max_x \left\{ \frac{1}{4} \sum_{i,j} a_{ij} (1 - x_i x_j) : x_i^2 = 1 \ \forall i \right\}. \quad (\text{MAXCUT})$$

- Applying the Semidefinite Relaxation scheme, we get an SDP relaxation of MAXCUT as follows:

$$SDP = \max_X \left\{ \frac{1}{4} \sum_{i,j} a_{ij} (1 - X_{ij}) : X \succeq 0, X_{ii} = 1, i \leq n \right\}. \quad (\text{SDP})$$

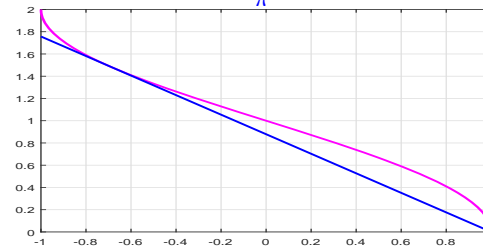
$$\begin{aligned}
 OPT &= \max_x \left\{ \frac{1}{4} \sum_{i,j} a_{ij} (1 - x_i x_j) : x_i^2 = 1, i \leq n \right\} & (\text{MAXCUT}) \\
 SDP &= \max_X \left\{ \frac{1}{4} \sum_{i,j} a_{ij} (1 - X_{ij}) : X \succeq 0, X_{ii} = 1, i \leq n \right\} & (\text{SDP})
 \end{aligned}$$

Theorem [Goemans & Williamson, 1995]

$$OPT \leq SDP \leq \alpha \cdot OPT, \quad \alpha = 1.138\dots$$

Proof. The left inequality is evident. Let X^* be optimal for (SDP), let $\xi \sim \mathcal{N}(0, X^*)$ and let $\zeta = \text{sign}[\xi]$. Then

$$\begin{aligned}
 [OPT \geq] \mathbb{E} \left\{ \frac{1}{4} \sum_{i,j} a_{ij} (1 - \zeta_i \zeta_j) \right\} &= \frac{1}{4} \sum_{i,j} a_{ij} (1 - \frac{2}{\pi} \text{asin}(X_{ij}^*)) \quad [\text{computation}] \\
 &\geq \frac{1}{4} \alpha^{-1} \sum_{i,j} a_{ij} (1 - X_{ij}^*) \\
 &\quad [\text{due to } a_{ij} \geq 0 \text{ and } (1 - \frac{2}{\pi} \text{asin}(t)) \geq \alpha^{-1} (1 - t), -1 \leq t \leq 1]
 \end{aligned}$$



$$= \alpha^{-1} \cdot SDP.$$

Thus, $SDP \leq \alpha \cdot OPT$. □

Note: We can rewrite equivalently the MAXCUT quadratic problem

$$OPT = \max_x \left\{ \frac{1}{4} \sum_{i,j} a_{ij} (1 - x_i x_j) : x_i^2 = 1 \forall i \right\}.$$

as

$$\begin{aligned} OPT &= \max_x \left\{ \frac{1}{4} \sum_{i,j} a_{ij} (x_i^2 - x_i x_j) : x_i^2 = 1 \forall i \right\} \\ &= \max_x \left\{ x^T Q x : x_i^2 = 1 \forall i \right\}, \quad (\text{Cmb}) \\ Q_{ij} &= \frac{1}{4} \begin{cases} \sum_p a_{ip}, & i = j \\ -a_{ij} & i \neq j \end{cases} \end{aligned}$$

The SDP relaxation of (Cmb) is equivalent to the original SDP relaxation of MAXCUT and reads

$$SDP = \max_X \{ \text{Tr}(QX) : X \succeq 0, X_{ii} = 1, i \leq n \} \quad (\text{SDP})$$

Note: Q stemming from MAXCUT is diagonal-dominated matrix and as such is positive semidefinite. It, however, is a very special positive semidefinite matrix: row sums are zero, and off-diagonal entries are non-positive.

$$\begin{aligned}
OPT &= \max_x \{x^T Q x : x_i^2 = 1, i \leq n\} && \text{(Cmb)} \\
SDP &= \max_X \{\text{Tr}(QX) : X \succeq 0, X_{ii} = 1, i \leq n\} && \text{(SDP)}
\end{aligned}$$

Question: What happens when passing from (Cmb) *with the only restriction $Q \succeq 0$ on Q* to the semidefinite relaxation (SDP) of (Cmb)? How tight is the relaxation now?

C. Nesterov's $\frac{\pi}{2}$ Theorem. *When $Q \succeq 0$, one has*

$$OPT \leq SDP \leq \frac{\pi}{2} \cdot OPT.$$

$$SDP = \max_X \{ \text{Tr}(QX) : X \succeq 0, X_{ii} = 1, i \leq n \} \quad (\text{SDP})$$

$$OPT = \max_x \{ \text{Tr}(Qxx^T) : x_i^2 = 1, i \leq n \} \quad (\text{QP})$$

$$\text{Claim: } OPT \leq \frac{\pi}{2} SDP$$

Proof. Let X^* be an optimal solution to (SDP), let $\xi \sim \mathcal{N}(0, X^*)$ and let $\zeta = \text{sign}[\xi]$. Then

$$[OPT \geq] \mathbf{E} \{ \zeta^T Q \zeta \} = \text{Tr}(Q \underbrace{\frac{2}{\pi} [\text{asin}(X_{ij}^*)]_{i,j}}_{\text{asin}[X^*]}) \quad (1)$$

Lemma: Let $X \succeq 0$ and $|X_{ij}| \leq 1$. Then $\text{asin}[X] \succeq X$.

Proof: Denoting $[X]^k = [X_{ij}^k]_{i,j}$ and taking into account that $X \succeq 0 \Rightarrow [X]^k \succeq 0, k = 1, 2, \dots$, one has

$$\text{asin}[X] - X = \sum_{k=1}^{\infty} \frac{1 \times 3 \times 5 \times \dots \times (2k-1)}{2^k k! (2k+1)} [X]^{2k+1} \succeq 0 \quad \square$$

By Lemma and since $Q \succeq 0$, the right hand side in (1) is $\geq \frac{2}{\pi} \text{Tr}(QX^*) = \frac{2}{\pi} SDP$, whence $SDP \leq \frac{\pi}{2} OPT$. \square

- We have used the following

Fact: *If $X = [x_{ij}]_{i,j \leq n}$, $Y = [y_{ij}]_{i,j \leq n}$ are positive semidefinite matrices of the same order, then the entrywise product of X and Y – the matrix*

$$X \bullet Y = [x_{ij}y_{ij}]_{i,j \leq n}$$

is positive semidefinite as well.

Indeed, symmetric matrix Q is $\succeq 0$ iff $Q = F^T F$ for some rectangular matrix F , or, which is the same, iff Q is a Gram matrix:

$$Q_{ij} = f_i^T f_j$$

for some $f_i \in \mathbf{R}^N$ (treat f_i as the columns of F). And entrywise product of Gram matrices again is a Gram matrix:

$$x_{ij} = f_i^T f_j, y_{ij} = g_i^T g_j \Rightarrow x_{ij}y_{ij} = \text{Vec}^T(f_i g_i^T) \text{Vec}(f_j g_j^T) \quad \square$$

What has happened?

♠ In nearly all known to us examples “tightness” of semidefinite relaxation is derived from considerations as follows:

- We start with quadratic quadratically constrained problem

$$\text{Opt} = \min_{x \in \mathbf{R}^n} \left\{ x^T Q_0 x + 2b_0^T x + c_0 : \begin{array}{l} f_i(x) = x^T Q_i x + 2b_i^T x + c_i \leq 0, i = 1, \dots, m \\ h_\ell(x) = x^T R_\ell x + 2d_\ell^T x + e_\ell = 0, \ell = 1, \dots, k \end{array} \right\}$$

and look for its *randomized* solutions x which satisfy the constraints *at average* and minimize under these restrictions the *expected value* of the objective.

- The expected value of a quadratic function $f(x) = x^T Q x + 2b^T x + c$ of random vector x is fully specified by the *moment matrix*

$$X = \mathbf{E}_x \left\{ \underbrace{\begin{bmatrix} xx^T & x \\ x^T & 1 \end{bmatrix}}_{[x;1][x;1]^T} \right\},$$

of x . Indeed, $\mathbf{E}_x \{f(x)\} = \text{Tr} \left(\begin{bmatrix} Q & b \\ b^T & c \end{bmatrix} X \right)$.

- It is easily seen that the only restrictions on a matrix $X \in \mathbf{S}^{n+1}$ to be the moment matrix of a random vector $x \in \mathbf{R}^n$ are $X \succeq 0$ & $X_{n+1,n+1} = 1$

$$\text{Opt} = \min_x \left\{ x^T Q_0 x + 2b_0^T x + c_0 : \begin{array}{l} f_i(x) = x^T Q_i x + 2b_i^T x + c_i \leq 0, i = 1, \dots, m \\ h_\ell(x) = x^T R_\ell x + 2d_\ell^T x + e_\ell = 0, \ell = 1, \dots, k \end{array} \right\}$$

$$\mathbb{E}_x\{x^T Q x + 2b^T x + c\} = \text{Tr} \left(\left[\begin{array}{c|c} Q & b \\ \hline b^T & c \end{array} \right] X \right), \quad X = \mathbb{E}_x\{[x; 1][x; 1]^T\}$$

\Rightarrow The "randomized" version of the problem of interest reads

$$\min_X \left\{ \begin{array}{l} \text{Tr}(A_i X) \leq 0, 1 \leq i \leq m \\ \text{Tr}(B_\ell X) = 0, \ell \leq k \\ X \succeq 0, X_{n+1, n+1} = 1 \end{array} \right\}$$

$$\left[A_i = \left[\begin{array}{c|c} Q_i & b_i \\ \hline b_i^T & c_i \end{array} \right], B_\ell = \left[\begin{array}{c|c} R_\ell & d_\ell \\ \hline d_\ell^T & e_\ell \end{array} \right] \right]$$

which is exactly the semidefinite relaxation of the problem of interest.

The advantage of this interpretation is that it allows to pass from optimal solution X_* to the relaxation to a sample x^1, x^2, \dots, x^N of realizations of the associated random "solution" to the problem of interest. In good cases, we can "correct" x^s to become feasible for the problem of interest and can look how much the correction costs us in terms of the objective.

For example: In MAXCUT and in Nesterov's $\frac{\pi}{2}$ Theorem we sample x^s from $\mathcal{N}(0, X_*)$ and correct x^s by passing from x^i to $\text{sign}[x^s]$.

♣ The $\frac{\pi}{2}$ Theorem admits important corollaries:

Corollary 1 [Nesterov '97] *Let $T \subset \mathbf{R}_+^n$ be a nonempty SDr compact set, and let Q be an $n \times n$ symmetric matrix. Then the quantities*

$$\begin{aligned} m_*(Q) &= \min_x \left\{ x^T Q x : (x_1^2, \dots, x_n^2)^T \in T \right\}, \\ m^*(Q) &= \max_x \left\{ x^T Q x : (x_1^2, \dots, x_n^2)^T \in T \right\} \end{aligned}$$

admit efficiently computable, via SDP, bounds

$$\begin{aligned} s_*(Q) &\equiv \min_X \left\{ \text{Tr}(QX) : X \succeq 0, (X_{11}, \dots, X_{nn})^T \in T \right\}, \\ s^*(Q) &\equiv \max_X \left\{ \text{Tr}(QX) : X \succeq 0, (X_{11}, \dots, X_{nn})^T \in T \right\} \end{aligned}$$

such that

$$s_*(Q) \leq m_*(Q) \leq m^*(Q) \leq s^*(Q)$$

and

$$m^*(Q) - m_*(Q) \leq s^*(Q) - s_*(Q) \leq \frac{\pi}{4 - \pi} (m^*(Q) - m_*(Q))$$

Thus, one can bound from above the variation $m^*(Q) - m_*(Q)$ by the efficiently computable quantity $s^*(Q) - s_*(Q)$, and this bound is tight within the absolute constant factor $\frac{\pi}{4 - \pi}$.

Corollary 2 [Nesterov '97] *Let $p \in [2, \infty]$, $r \in [1, 2]$, and let A be an $m \times n$ matrix. Consider the problem of computing the operator norm $\|A\|_{p,r}$ of the linear mapping $x \mapsto Ax$, considered as the mapping from the space \mathbf{R}^n equipped with the norm $\|\cdot\|_p$ to the space \mathbf{R}^m equipped with the norm $\|\cdot\|_r$:*

$$\|A\|_{p,r} = \max \{ \|Ax\|_r : \|x\|_p \leq 1 \};$$

(it is NP-hard to compute this norm, except for the case of $p = r = 2$). The “computationally intractable” quantity $\|A\|_{p,r}$ admits an efficiently computable upper bound

$$\omega_{p,r}(A) = \min_{\lambda \in \mathbf{R}^m, \mu \in \mathbf{R}^n} \left\{ \frac{1}{2} \left[\|\mu\|_{\frac{p}{p-2}} + \|\lambda\|_{\frac{r}{2-r}} \right] : \left[\begin{array}{c|c} \text{Diag}\{\mu\} & A^T \\ \hline A & \text{Diag}\{\lambda\} \end{array} \right] \succeq 0 \right\}.$$

This bound is exact for a nonnegative matrix A , and for an arbitrary A the bound is tight within the factor $\frac{\pi}{2\sqrt{3}-2\pi/3} = 2.293\dots$:

$$\|A\|_{p,r} \leq \omega_{p,r}(A) \leq \frac{\pi}{2\sqrt{3}-2\pi/3} \|A\|_{p,r}.$$

Moreover, if $p \in [1, \infty]$ and $r \in [1, 2]$ are rational, the bound $\omega_{p,r}(A)$ is an SDr function of A .

D. Semidefinite Relaxation on Ellitopes

♠ **A basic ellitope** is a set $\mathcal{X} \subset \mathbf{R}^n$ represented as

$$\mathcal{X} = \{x : \exists t \in \mathcal{T} : x^T S_k x \leq t_k, 1 \leq k \leq K\}$$

- $S_k \succeq 0, k \leq K, \sum_k S_k \succ 0$
- \mathcal{T} : convex compact set in \mathbf{R}_+^K containing a positive vector and monotone: $0 \leq t' \leq t \in \mathcal{T} \Rightarrow t' \in \mathcal{T}$

♠ **An ellitope** \mathcal{Y} is a set represented as a linear image of basic ellitope:

$$\mathcal{Y} = \{y : \exists(t \in \mathcal{T}, x) : y = Px, x^T S_k x \leq t_k, k \leq K\}.$$

Examples: A. Bounded intersection \mathcal{X} of K centered at the origin ellipsoids/elliptic cylinders $\{x : x^T S_k x \leq 1\} [S_k \succeq 0]$ is a basic ellitope:

$$\mathcal{X} = \{x : \exists t \in \mathcal{T} := [0, 1]^K : x^T S_k x \leq t_k, k \leq K\}$$

B. $\|\cdot\|_p$ -ball in \mathbf{R}^n with $p \in [2, \infty]$ is a basic ellitope:

$$\{x \in \mathbf{R}^n : \|x\|_p \leq 1\} = \{x : \exists t \in \mathcal{T} = \{t \in \mathbf{R}_+^n, \|t\|_{p/2} \leq 1\} : \underbrace{x_k^2}_{x^T S_k x} \leq t_k, k \leq K\}.$$

- ♣ **Fact:** Ellitopes admit fully algorithmic "calculus:" this family is closed with respect to basic operations preserving convexity and symmetry w.r.t. the origin, like taking
- finite intersections,
 - linear images,
 - inverse images under linear embeddings,
 - direct products,
 - arithmetic sums.
- What is missing, is *taking convex hulls of finite unions*.

♣ **Fact:** When maximizing a quadratic form $y^T C y$ over an ellitope

$$\mathcal{Y} = P\mathcal{X}, \mathcal{X} = \{x : \exists t \in \mathcal{T} : x^T S_k x \leq t_k, k \leq K\}$$

semidefinite relaxation works reasonably well.

This is how it works:

- Passing from the quadratic form $y^T C y$ to the “lifted” form $x^T \overbrace{[P^T C P]}^D x$, we reduce the situation to maximizing quadratic form $x^T D x$ over the basic ellitope \mathcal{X} .

- For $\lambda \in \mathbf{R}^K$, let $\phi_{\mathcal{T}}(\lambda) = \max_{t \in \mathcal{T}} t^T \lambda$ be the support function of \mathcal{T} . When $\lambda \geq 0$ is such that $D \preceq \sum_k \lambda_k S_k$, and $x \in \mathcal{X}$, there exists $t \in \mathcal{T}$ such that $x^T S_k x \leq t_k, k \leq K$,

$$\Rightarrow x^T D x \leq x^T [\sum_k \lambda_k S_k] x \leq \sum_k \lambda_k t_k \leq \phi_{\mathcal{T}}(\lambda)$$

$$\Rightarrow \max_{x \in \mathcal{X}} x^T D x \leq \text{Opt} := \min \{ \phi_{\mathcal{T}}(\lambda) : \lambda \geq 0, D \preceq \sum_k \lambda_k S_k \}$$

Theorem One has

$$\max_{x \in \mathcal{X}} x^T D x \leq \text{Opt} \leq 3 \ln(\sqrt{3}K) \max_{x \in \mathcal{X}} x^T D x$$

$$\text{Fact: } \begin{aligned} \text{Opt} &:= \min \left\{ \phi_{\mathcal{T}}(\lambda) : \lambda \geq 0, D \preceq \sum_k \lambda_k S_k \right\} & [\phi_{\mathcal{T}}(\lambda) = \max_{t \in \mathcal{T}} \lambda^T t] \\ \max_{x \in \mathcal{X}} x^T D x &\leq \text{Opt} \leq 3 \ln(\sqrt{3}K) \max_{x \in \mathcal{X}} x^T D x \end{aligned}$$

Note:

- When \mathcal{T} is SDR with essentially strictly feasible SDR, the support function $\phi_{\mathcal{T}}(\cdot)$ of \mathcal{T} is SDR by Calculus of CQRs/SDRs
 \Rightarrow *computing Opt reduces to SDP*
- $O(\ln(K))$ (not $O(1)$!) tightness factor correctly expresses the *actual* quality of SDP relaxation when maximizing quadratic forms over ellitopes.

$$\begin{aligned}
\text{Opt} &:= \min \left\{ \phi_{\mathcal{T}}(\lambda) : \lambda \geq 0, D \preceq \sum_k \lambda_k S_k \right\} & (*) \\
\geq \text{Opt}_* &:= \max_{z,t} \left\{ z^T D z : t \in \mathcal{T}, z^T S_k z \leq t_k, k \leq K \right\} \\
&\quad \phi_{\mathcal{T}}(\lambda) \max_{t \in \mathcal{T}} \lambda^T t
\end{aligned}$$

Claim: $\text{Opt} \leq 3 \ln(\sqrt{3}K) \text{Opt}_*$

Sketch of the Proof

Step 1. Rewriting (*) in conic form. Let $\mathbf{T} = \text{cl}\{[t; \tau] : \tau > 0, t/\tau \in \mathcal{T}\}$ be the closed conic hull of \mathcal{T} . Since \mathcal{T} is a convex compact set with a nonempty interior, \mathbf{T} is a regular cone, and

$$\mathcal{T} = \{t : [t; 1] \in \mathbf{T}\}.$$

It is immediately seen (check!) that

$$\mathbf{T}_* = \{[g; s] : s \geq \phi_{\mathcal{T}}(-g)\}$$

$\Rightarrow (*)$ is the conic problem

$$\text{Opt} = \min_{\lambda, \tau} \left\{ \tau : \lambda \geq 0, [-\lambda; \tau] \in \mathbf{T}_*, D \preceq \sum_k \lambda_k S_k \right\}$$

As is easily seen, the problem is strictly feasible and bounded, so that the dual problem is solvable with optimal value Opt :

$$\text{Opt} = \max_{X,t} \left\{ \text{Tr}(DX) : X \succeq 0, t \in \mathcal{T}, \text{Tr}(XS_k) \leq t_k, k \leq K \right\}.$$

Step 2: "derandomization. Let X_* , t^* be an optimal solution to (*). Let

$$\tilde{D} = X_*^{1/2} D X_*^{1/2} = U \text{Diag}\{\mu\} U^T \quad [U \text{ is orthogonal}]$$

$$\tilde{S}_k = U^T X_*^{1/2} S_k X_*^{1/2} U$$

$$\Rightarrow 0 \preceq \tilde{S}_k, \text{Tr}(\tilde{S}_k) = \text{Tr}(X_*^{1/2} S_k X_*^{1/2}) = \text{Tr}(S_k X_*) \leq t_k^*.$$

Let ζ be Rademacher random vector (independent entries taking values ± 1 with probability $1/2$), and let

$$\xi = X_*^{1/2} U \zeta.$$

Note:

$$\mathbf{E}\{\xi \xi^T\} = \mathbf{E}\{X_*^{1/2} U \zeta \zeta^T U^T X_*^{1/2}\} = X_* \quad (a)$$

$$\begin{aligned} \xi^T D \xi &= \zeta^T U^T X_*^{1/2} D X_*^{1/2} U \zeta = \zeta^T U^T \tilde{D} U \zeta \\ &= \zeta^T \text{Diag}\{\mu\} \zeta = \sum_i \mu_i = \text{Tr}(\tilde{D}) = \text{Tr}(D X_*) = \text{Opt} \end{aligned} \quad (b)$$

$$\xi^T S_k \xi = \zeta^T U^T X_*^{1/2} S_k X_*^{1/2} U \zeta = \zeta^T \tilde{S}_k \zeta \quad (c)$$

ζ : Rademacher random vector, $\xi = X_*^{1/2}U\zeta$, and

$$\mathbf{E}\{\xi\xi^T\} = \mathbf{E}\{X_*^{1/2}U\zeta\zeta^TU^TX_*^{1/2}\} = X_* \quad (a)$$

$$\begin{aligned} \xi^TD\xi &= \zeta^TU^TX_*^{1/2}DX_*^{1/2}U\zeta = \zeta^TU^T\tilde{D}U\zeta \\ &= \zeta^T\text{Diag}\{\mu\}\zeta = \sum_i \mu_i = \text{Tr}(\tilde{D}) = \text{Tr}(DX_*) = \text{Opt} \end{aligned} \quad (b)$$

$$\xi^TS_k\xi = \zeta^TU^TX_*^{1/2}S_kX_*^{1/2}U\zeta = \zeta^T\tilde{S}_k\zeta \quad (c)$$

$$\text{Tr}(\tilde{S}_k) \leq t_k^* \ \& \ \tilde{S}_k \succeq 0 \quad (d)$$

Observe that

A: for k with $t_k^* = 0$ we have $\tilde{S}_k = 0$ by (d) $\Rightarrow \xi^TS_k\xi \equiv 0$ by (c) \Rightarrow

$$\text{Prob}\{\xi^TS_k\xi > 3\gamma t_k^*\} \leq \sqrt{3}\exp\{-\gamma\} \ \forall \gamma \geq 0$$

B: for k with $t_k^* > 0$ we have $\text{Tr}(\tilde{S}_k/t_k^*) \leq 1$, whence

$$\mathbf{E}\left\{\exp\left\{\frac{\xi^TS_k\xi}{3t_k^*}\right\}\right\} \underbrace{=}_{(c)} \mathbf{E}\left\{\exp\left\{\frac{\zeta^T\tilde{S}_k\zeta}{3t_k^*}\right\}\right\} \underbrace{\leq}_{(!)} \sqrt{3} \Rightarrow$$

$$\text{Prob}\{\xi^TS_k\xi > 3t_k^*\gamma\} \leq \sqrt{3}\exp\{-\gamma\} \ \forall \gamma \geq 0$$

where (!) is due to

Lemma: Let Q be positive semidefinite $N \times N$ matrix with trace ≤ 1 and ζ be N -dimensional Rademacher random vector. Then $\mathbf{E}\{\exp\{\zeta^TQ\zeta/3\}\} \leq \sqrt{3}$.

applied to $Q = \tilde{S}_k/t_k^*$ with $\text{Tr}(Q) \leq 1$ by (d).

We have built random vector ξ with discrete distribution such that

$$\xi^T D \xi \equiv \text{Opt} \quad (e)$$

$$\text{Prob} \{ \xi^T S_k \xi > 3t_k^* \gamma \} \leq \sqrt{3} \exp\{-\gamma\} \forall k \forall \gamma > 0 \quad (f)$$

By (f) $\text{Prob}\{\exists k : \xi^T S_k \xi > 3 \ln(\sqrt{3}K) t_k^*\} < 1$

\Rightarrow exists realization $\bar{\xi}$ of ξ such that $\bar{\xi}^T S_k \bar{\xi} \leq 3 \ln(\sqrt{3}K) t_k^*, k \leq K$, while $\bar{\xi}^T D \bar{\xi} = \text{Opt}$ by (e)

\Rightarrow setting $z = \bar{\xi} / \sqrt{3 \ln(\sqrt{3}K)}$, we get

$$\underbrace{z^T S_k z \leq t_k^*, k \leq K}_{\Rightarrow z^T D z \leq \text{Opt}_*} \quad \& \quad z^T D z = \text{Opt} / [3 \ln(\sqrt{3}K)]$$

$\Rightarrow \text{Opt}_* \geq \text{Opt} / [3 \ln(\sqrt{3}K)], \text{ Q.E.D.}$

Lemma: Let Q be positive semidefinite $N \times N$ matrix with $\text{trace} \leq 1$ and ζ be N -dimensional Rademacher random vector. Then

$$\mathbf{E} \left\{ \exp \left\{ \zeta^T Q \zeta / 3 \right\} \right\} \leq \sqrt{3}.$$

Proof of Lemma: Let Q obey the premise of Lemma, and let $Q = \sum_i \sigma_i f_i f_i^T$ be the eigenvalue decomposition of Q , so that $f_i^T f_i = 1$, $\sigma_i \geq 0$, and $\sum_i \sigma_i \leq 1$. The function

$$f(\sigma_1, \dots, \sigma_N) = \mathbf{E} \left\{ e^{\frac{1}{3} \sum_i \sigma_i \zeta^T f_i f_i^T \zeta} \right\}$$

is convex on the simplex $\{\sigma \geq 0, \sum_i \sigma_i \leq 1\}$ and thus attains its maximum over the simplex at a vertex

\Rightarrow for some f with $f^T f = 1$ it holds

$$\mathbf{E} \left\{ e^{\frac{1}{3} \zeta^T Q \zeta} \right\} \leq \mathbf{E} \left\{ e^{\frac{1}{3} (f^T \zeta)^2} \right\}.$$

Let $\xi \sim \mathcal{N}(0, 1)$ be independent of ζ . We have

$$\begin{aligned} \mathbf{E}_\zeta \left\{ \exp \left\{ \frac{1}{3} (f^T \zeta)^2 \right\} \right\} &= \mathbf{E}_\zeta \left\{ \mathbf{E}_\xi \left\{ \exp \left\{ [\sqrt{2/3} f^T \zeta] \xi \right\} \right\} \right\} \\ &= \mathbf{E}_\xi \left\{ \mathbf{E}_\zeta \left\{ \exp \left\{ [\sqrt{2/3} f^T \zeta] \xi \right\} \right\} \right\} = \mathbf{E}_\xi \left\{ \prod_{j=1}^N \mathbf{E}_\zeta \left\{ \exp \left\{ \sqrt{2/3} \xi f_j \zeta_j \right\} \right\} \right\} \\ &= \mathbf{E}_\xi \left\{ \prod_{j=1}^N \cosh(\sqrt{2/3} \xi f_j) \right\} \leq \mathbf{E}_\xi \left\{ \prod_{j=1}^N \exp \left\{ \xi^2 f_j^2 / 3 \right\} \right\} \\ &= \mathbf{E}_\xi \left\{ \exp \left\{ \xi^2 / 3 \right\} \right\} = \sqrt{3} \end{aligned}$$

□

Application: Near-Optimal Linear Estimation

♣ Consider the following basic statistical problem: *Given noisy observation*

$$\omega = Ax + \xi \quad [A : m \times n; \xi \sim \mathcal{N}(0, I_m)]$$

of *unknown* signal x known to belong to a given “signal set” \mathcal{X} , recover the linear image $Bx \in \mathbb{R}^\nu$ of x .

♠ The performance of a candidate estimate $\hat{x}(\cdot)$ is quantified by *risk*

$$\text{Risk}[\hat{x}|\mathcal{X}] = \sup_{x \in \mathcal{X}} \mathbf{E}_\xi \{ \|\hat{x}(Ax + \xi) - Bx\| \} .$$

[$\|\cdot\|$: a given norm on \mathbb{R}^ν]

♠ Assuming \mathcal{X} symmetric w.r.t. the origin, the simplest estimates are *linear* ones: $\hat{x}(\omega) = \hat{x}_H(\omega) := H^T \omega$. In this case $Bx - \hat{x}_H(\omega) = Bx - H^T(Ax + \xi) = [B - H^T A]x - H^T \xi$
 \Rightarrow *The risk of a linear estimate can be tightly, within factor 2, upper-bounded as*

$$\text{Risk}[\hat{x}|\mathcal{X}] \leq \overline{\text{Risk}}[\hat{x}|\mathcal{X}] := \underbrace{\max_{x \in \mathcal{X}} \|[B - H^T A]x\|}_{\text{“bias”}} + \underbrace{\mathbf{E}\{\|H^T \xi\|\}}_{\text{stochastic term}} .$$

\Rightarrow *The minimum (within factor 2) risk linear estimate is given by an optimal solution to the convex optimization problem*

$$\text{Opt}_* = \min_H \{ \Phi(H) + \Psi(H) \}, \quad \Phi(H) := \max_{x \in \mathcal{X}} \|[B - H^T A]x\|, \quad \Psi(H) = \mathbf{E}_{\xi \sim \mathcal{N}(0, I)} \{\|H^T \xi\|\}$$

♠ **Difficulty:** Φ and Ψ , while convex, are, in general, difficult to compute. The only generic cases when computing Φ is easy are those when \mathcal{X} is an ellipsoid, or the convex hull of a finite set of moderate cardinality.

♠ **Proposed remedy:** *Replace Φ and Ψ with their efficiently computable upper bounds.*

♣ From now on, we make the following assumptions

A. the signal set \mathcal{X} is a basic ellitope:

$$\mathcal{X} = \{x : \exists t \in \mathcal{T} : x^T S_k x \leq t_k, k \leq K\}$$

B. the unit ball \mathcal{B}_* of the norm $\|\cdot\|_*$ conjugate to $\|\cdot\|$ is an ellitope:

$$\mathcal{B}_* = \{u : \|u\|_* := \max_{\|v\| \leq 1} v^T u \leq 1\} = \{u : \exists z \in \mathcal{Z} : u = Mz\}, \mathcal{Z} = \{z : \exists r \in \mathcal{R} : z^T R_\ell z \leq r_\ell, \ell \leq L\}$$

$$\Leftrightarrow \|v\| = \max_{\|u\|_* \leq 1} u^T v$$

as is the case, e.g., when $\|\cdot\| = \|\cdot\|_p$ with $1 \leq p \leq 2$.

• $\mathcal{T}, S_k, \mathcal{R}, R_\ell$: as required by the definition of an ellitope.

♣ **Upper-bounding Φ :**

$$\begin{aligned} \Phi(H) &= \max_{x \in \mathcal{X}} \|[B - H^T A]x\| = \max_{u \in \mathcal{B}_*, x \in \mathcal{X}} u^T [B - H^T A]x = \max_{z \in \mathcal{Z}, x \in \mathcal{X}} z^T [M^T [B - H^T A]]x \\ &= \max_{[z; x] \in \mathcal{Z} \times \mathcal{X}} [z; x]^T \left[\frac{\frac{1}{2}[B^T - A^T H]M}{\left| \frac{1}{2}M^T [B - H^T A] \right|} \right] [z; x] \end{aligned}$$

By Semidefinite Relaxation as applied on the ellitope $\mathcal{Z} \times \mathcal{X}$

$$\Phi(H) \leq \overline{\Phi}(H) := \min_{\lambda, \mu} \left\{ \phi_{\mathcal{T}}(\lambda) + \phi_{\mathcal{R}}(\mu) : \begin{array}{l} \lambda \geq 0, \mu \geq 0 \\ \left[\frac{\sum_{\ell} \mu_{\ell} R_{\ell}}{\frac{1}{2}[B^T - A^T H]M} \mid \frac{\frac{1}{2}M^T [B - H^T A]}{\sum_k \lambda_k S_k} \right] \succeq 0 \end{array} \right\}$$

• $\phi_U(w) = \max_{u \in U} w^T u$: the support function of set U

• **Note:** The upper bound $\overline{\Phi}$ on Φ is tight within the factor $3 \ln(\sqrt{3}(K + L))$.

A. the signal set \mathcal{X} is a basic ellitope:

$$\mathcal{X} = \{x : \exists t \in \mathcal{T} : x^T S_k x \leq t_k, k \leq K\}$$

B. the unit ball \mathcal{B}_* of the norm $\|\cdot\|_*$ conjugate to $\|\cdot\|$ is an ellitope:

$$\mathcal{B}_* = \{u : \|u\|_* := \max_{\|v\| \leq 1} v^T u \leq 1\} = \{u : \exists z \in \mathcal{Z} : u = Mz\}, \mathcal{Z} = \{z : \exists r \in \mathcal{R} : z^T R_\ell z \leq r_\ell, \ell \leq L\}$$

$$\Leftrightarrow \|v\| = \max_{\|u\|_* \leq 1} u^T v$$

♣ **Upper-bounding** $\Psi(H) := \mathbf{E}\{\|H^T \xi\|\}$. Observe that whenever nonnegative vector θ , symmetric matrix Θ , and H satisfy the matrix inequality

$$\left[\begin{array}{c|c} \sum_\ell \theta_\ell R_\ell & \frac{1}{2} M^T H^T \\ \hline \frac{1}{2} H M & \Theta \end{array} \right] \succeq 0,$$

we have

$$\mathbf{E}\{\|H^T \xi\|\} \leq \phi_{\mathcal{R}}(\theta) + \text{Tr}(\Theta). \quad (\#)$$

Indeed, under the claim's premise one has

$$\forall [z; \xi] : [Mz]^T H^T \xi \leq z^T [\sum_\ell \theta_\ell R_\ell] z + \xi^T \Theta \xi.$$

When $z \in \mathcal{Z}$, one has $z^T [\sum_\ell \theta_\ell R_\ell] z \leq \phi_{\mathcal{R}}(\theta)$ (check it!) \Rightarrow

$$\|H^T \xi\| = \max_{z \in \mathcal{Z}} [Mz]^T H^T \xi \leq \phi_{\mathcal{R}}(\theta) + \xi^T \Theta \xi.$$

Taking expectation over ξ , we arrive at (#).

As a result of our observation, we see that

♣ One has

$$\Psi(\eta) \leq \overline{\Psi}(H) := \min_{\theta, \Theta} \left\{ \phi_{\mathcal{R}}(\theta) + \text{Tr}(\Theta) : \theta \geq 0, \left[\begin{array}{c|c} \sum_\ell \theta_\ell R_\ell & \frac{1}{2} M^T H^T \\ \hline \frac{1}{2} H M & \Theta \end{array} \right] \succeq 0 \right\}.$$

• It turns out that $\overline{\Psi}$ on Ψ is tight within the factor $O(1)\sqrt{\ln(L+1)}$.

♣ **Bottom line:** Consider the convex optimization problem

$$\text{Opt} = \min_{H, \lambda, \mu, \theta, \Theta} \left\{ \phi_{\mathcal{T}}(\lambda) + \phi_{\mathcal{R}}(\mu) + \phi_{\mathcal{R}}(\theta) + \text{Tr}(\Theta) : \begin{array}{l} \lambda \geq 0, \mu \geq 0, \theta \geq 0 \\ \left[\begin{array}{c|c} \sum_{\ell} \mu_{\ell} R_{\ell} & \frac{1}{2} M^T [B - H^T A] \\ \hline \frac{1}{2} [B^T - A^T H] M & \sum_k \lambda_k S_k \end{array} \right] \succeq 0 \\ \left[\begin{array}{c|c} \sum_{\ell} \theta_{\ell} R_{\ell} & \frac{1}{2} M^T H^T \\ \hline \frac{1}{2} H M & \Theta \end{array} \right] \succeq 0 \end{array} \right\}$$

(which is nothing but the problem of minimizing $\overline{\Phi}(H) + \overline{\Psi}(H)$ over H). This problem is efficiently solvable, and the linear estimate \hat{x}_{H_*} yielded by the H -component H_* of optimal solution satisfies the relation

$$\text{Risk}[\hat{x}_{H_*} | \mathcal{X}] \leq \overline{\text{Risk}}[\hat{x}_{H_*} | \mathcal{X}] \leq \text{Opt}.$$

- **Note:** From remarks on tightness of the upper bounds $\overline{\Phi}$, $\overline{\Psi}$ it follows that the linear estimate from Bottom line is optimal, within the “moderate” factor $O(1) \ln(K + L)$, in terms of its risk *among all linear estimates*.
- Surprisingly, it turns out that *the estimate in question is optimal, within the factor $O(1) \sqrt{\ln(K + 1) \ln(L + 1)}$, among all estimates, linear and nonlinear alike.*

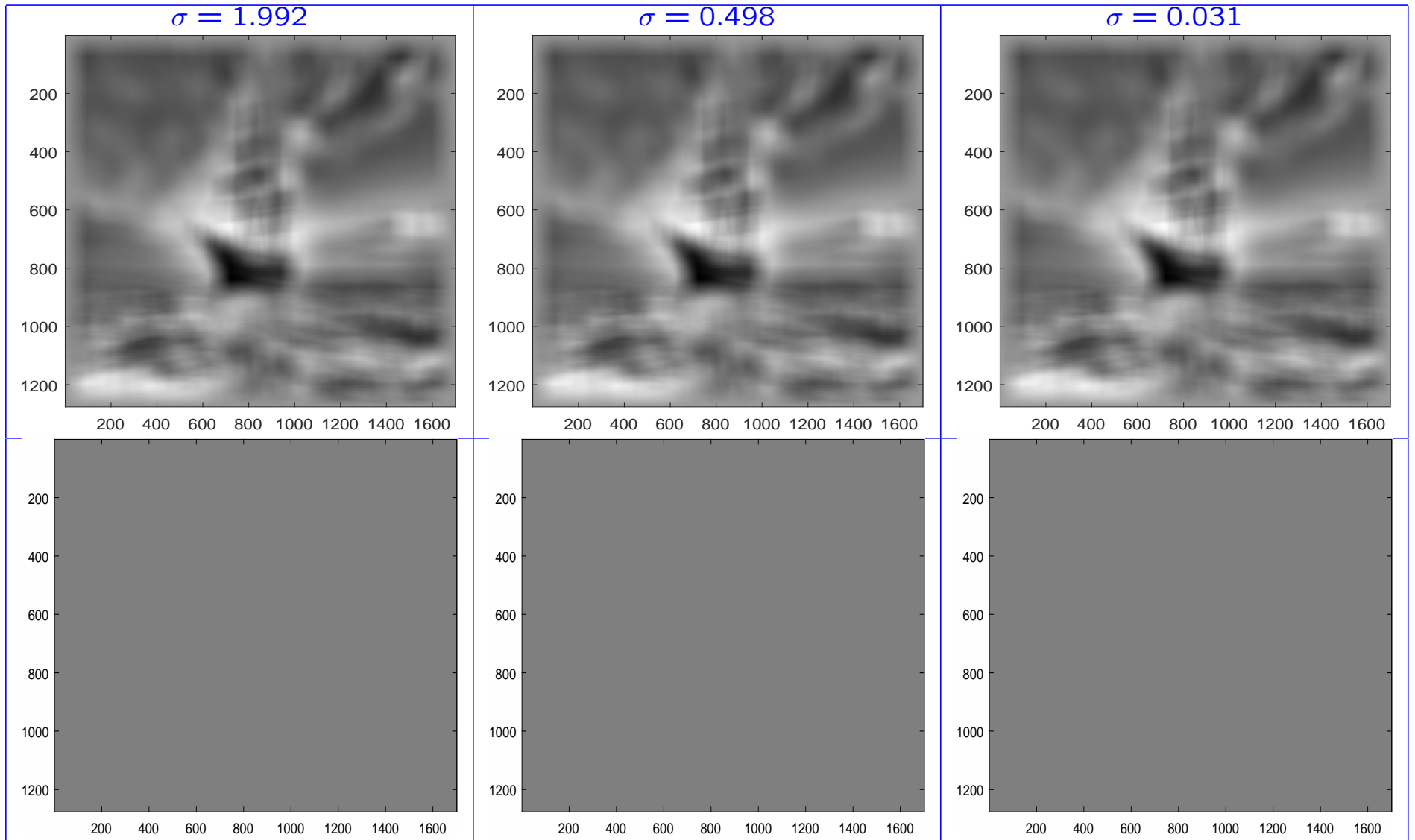
How it works

Situation: We want to recover image $x \in \mathcal{X}$ from its blurred noisy observation y :

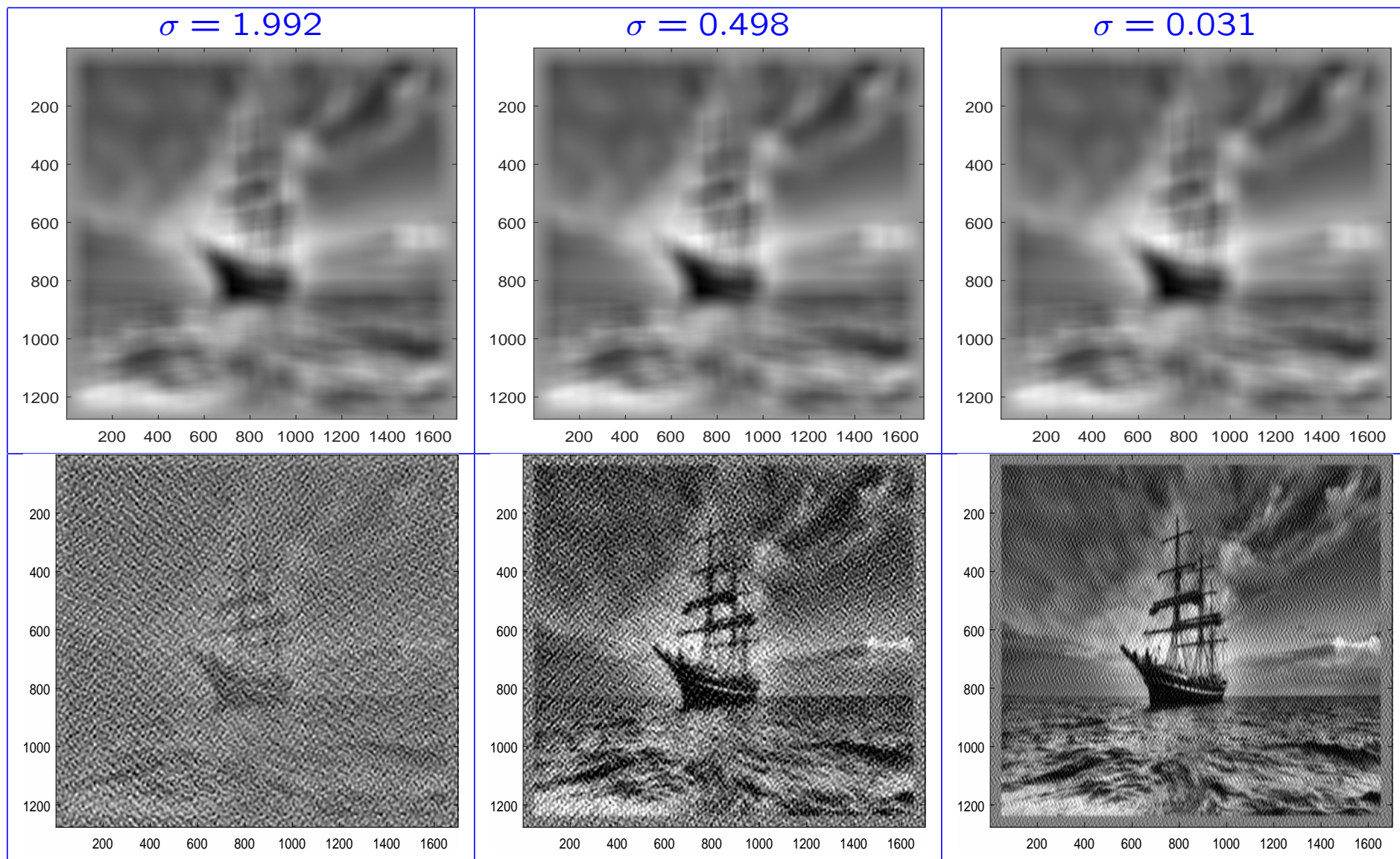
$$y = \kappa \star x + \sigma \xi$$

- $x \in \mathbf{R}^{m \times n}$: true image
- blur $x \mapsto \kappa \star x$: 2D convolution of x with given *blurring kernel* κ
- observation noise ξ : 2D White Gaussian with unit pixel-wise variance

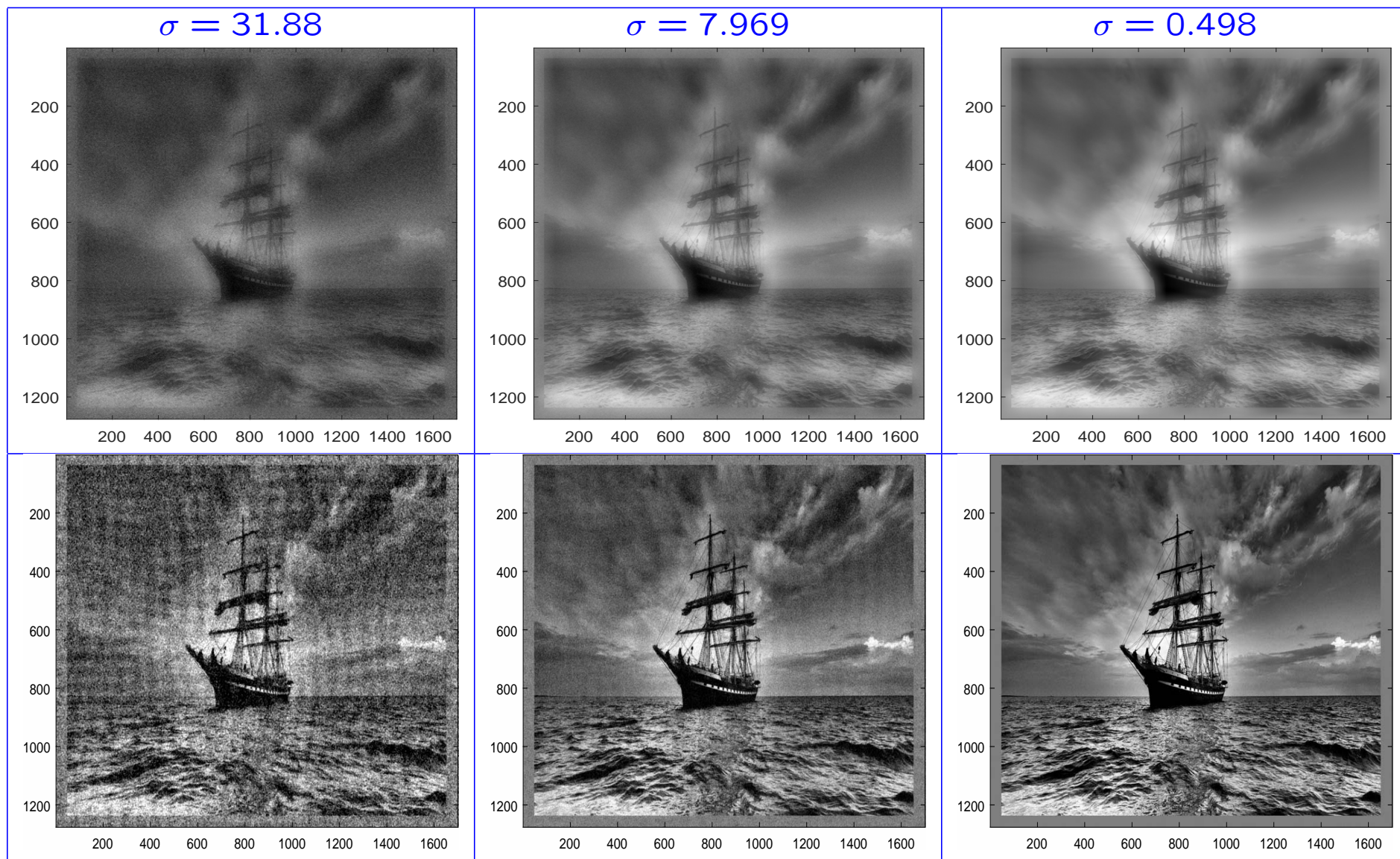
Blurred noisy observations (top) and recoveries (bottom) of 1200×1600 image, ill-posed case
[with \mathcal{X} given by trivial bound on signal's energy]



Blurred noisy observations (top) and recoveries (bottom) of 1200×1600 image, ill-posed case
[with \mathcal{X} given by Energy and rudimentary form of Total Variation constraints]



Blurred noisy observations (top) and recoveries (bottom) of 1200×1600 image, well-posed case
[with \mathcal{X} given by trivial bound on signal's energy]



E. The Matrix Cube Theorem. Consider the following problem:

MATRCUBE: Given symmetric $m \times m$ matrices $B_0 \succeq 0, B_1, \dots, B_L$, solve the optimization problem

$$\rho^* = \max \left\{ \rho : \mathcal{A}[\rho] \equiv \left\{ B_0 + \sum_{\ell=1}^L u_\ell B_\ell : \|u\|_\infty \leq \rho \right\} \subset \mathbf{S}_+^m \right\}$$

i.e., find the largest ρ such that the “matrix box” $\mathcal{A}[\rho]$ is contained in the semidefinite cone.

This problem is easy when all “edge matrices” $B_\ell, \ell \geq 1$, are of rank 1, and can be NP-hard already when the “edge matrices” are of rank 2.

Matrix Cube Theorem [Ben-Tal & Nemirovski, '00] *Given $\rho \geq 0$, consider the system of LMI's*

$$\begin{aligned} X^\ell &\succeq \pm B_\ell, \ell = 1, \dots, L, \\ \rho \sum_{\ell=1}^L X^\ell &\preceq B_0 \end{aligned} \quad (S[\rho])$$

in matrix variables X^1, \dots, X^L .

(i) If $(S[\rho])$ is solvable, then $\mathcal{A}[\rho]$ is contained in S_+^m

*(ii) If $(S[\rho])$ is unsolvable, then $\mathcal{A}[\vartheta(\mu)\rho]$ is **not** contained in S_+^m . Here*

$$\mu = \max_{1 \leq \ell \leq L} \text{Rank}(B_\ell)$$

(note $\ell \geq 1$ in the max!) and $\vartheta(\mu)$ is a universal function such that

$$\vartheta(1) = 1, \quad \vartheta(2) = \frac{\pi}{2}, \quad \vartheta(k) \leq \frac{\pi\sqrt{k}}{2}.$$

In particular, the efficiently computable quantity

$$\hat{\rho} = \max \{ \rho : (S[\rho]) \text{ is solvable} \}$$

is a lower bound on ρ^ , and this bound is tight within the factor $\vartheta(\mu)$:*

$$\hat{\rho} \leq \rho^* \leq \vartheta(\mu)\hat{\rho}.$$

Application: Lyapunov Stability Analysis revisited

♠ Recall that Lyapunov Stability Certificates, if any, for uncertain dynamical system

$$\dot{x} = A(t)x, \quad [A(t) \in \mathcal{U}]$$

are exactly the solutions X to the semi-infinite system of LMIs

$$X \succeq I, \quad A^T X + X A \preceq -I \quad \forall (A \in \mathcal{U}) \quad (\text{L}[\mathcal{U}])$$

Consider the case of “interval uncertainty”:

$$\mathcal{U} = \mathcal{U}_\rho \equiv \left\{ A : |A_{ij} - A_{ij}^*| \leq \rho D_{ij}, \quad i, j = 1, \dots, n \right\},$$

where A^* is the (stable) “nominal matrix”, ρ is the level of perturbations, and $D_{ij} \geq 0$ are “perturbation scales”.

How to compute the Lyapunov Stability Radius

$$LSR[A^*, D] = \sup \{ \rho : (\text{L}[\mathcal{U}_\rho]) \text{ is solvable} \} \quad ?$$

- The interval uncertainty is a polytopic one, so that the semi-infinite system of LMIs ($L[\mathcal{U}_\rho]$) is equivalent to the finite system of LMIs

$$X \succeq I, \quad A_j^T X + X A_j \preceq -I \quad \forall j = 1, \dots, J, \quad (*)$$

where A_1, \dots, A_J are the vertices of the matrix box \mathcal{U}_ρ . However, J can blow up exponentially with the size n of the underlying dynamical system, so that $(*)$ is not computationally tractable, except for the case when “nearly all” entries in $A(t)$ are certain.

- In fact, the problem of computing LSR for a general-type interval uncertainty is NP-hard.

- Observe that

$$\begin{aligned}
LSR[A^*, D] &= \sup \left\{ \rho : \exists X \succeq I : A^T X + X A \preceq -I \forall (A : |A_{ij} - A_{ij}^*| \leq \rho D_{ij}) \right\} \\
&= \sup \left\{ \rho : \exists X \succeq I : \underbrace{[-I - (A^*)^T X - X A^*]}_{B_0[X]} + \sum_{i,j} u_{ij} \underbrace{D_{ij} [e_j e_i^T X + X e_i e_j^T]}_{B_{ij}[X]} \succeq 0 \right. \\
&\quad \left. \forall (u : \|u\|_\infty \leq \rho) \right\} \\
&= \sup_{X \succeq I} \rho(X), \\
\rho(X) &= \sup \left\{ \rho : B_0[X] + \sum_{i,j} u_{ij} B_{ij}[X] \succeq 0 \forall (u : \|u\|_\infty \leq \rho) \right\}
\end{aligned}$$

$\rho(X)$ is the optimal value in a MATRCUBE problem with rank 2 edge matrices $B_{ij}[X]$. Applying the Matrix Cube Theorem, we conclude that *The efficiently computable quantity*

$$\widehat{LSR}[A^*, D] = \sup_{\rho, X, \{X^{ij}\}} \left\{ \rho : \begin{array}{l} X \succeq I \\ X^{ij} \succeq \pm B_{ij}[X], 1 \leq i, j \leq n \\ \rho \sum_{i,j} X^{ij} \preceq B_0[X] \end{array} \right\}$$

is a lower bound, tight within the factor $\frac{\pi}{2}$, on the Lyapunov Stability Radius $LSR[A^, D]$.*

♣ Similarly to Lyapunov Stability Analysis, the Matrix Cube Theorem allows to build tight, within an absolute constant factor, tractable approximations of numerous Control-originating semi-infinite LMIs affected by interval uncertainty.

Matrix Cube Theorem – Sketch of the Proof

Matrix Cube Theorem: Given $\rho \geq 0$, consider the system of LMI's

$$\begin{aligned} X^\ell &\succeq \pm B_\ell, \ell = 1, \dots, L, \\ \rho \sum_{\ell=1}^L X^\ell &\preceq B_0 \end{aligned} \tag{S[\rho]}$$

in matrix variables X^1, \dots, X^L .

(i) If $(S[\rho])$ is solvable, then the “matrix box”

$$\mathcal{A}[\rho] \equiv \left\{ B_0 + \rho \sum_{\ell} u_{\ell} B_{\ell} : \|u\|_{\infty} \leq 1 \right\}$$

is contained in \mathbf{S}_+^m

(ii) If $(S[\rho])$ is unsolvable, then the matrix box $\mathcal{A}[\vartheta(\mu)\rho]$ is **not** contained in \mathbf{S}_+^m . Here

$$\mu = \max_{1 \leq \ell \leq L} \text{Rank}(B_{\ell})$$

(note $\ell \geq 1$ in the max!) and $\vartheta(\mu)$ is a universal function such that

$$\vartheta(1) = 1, \quad \vartheta(2) = \frac{\pi}{2}, \quad \vartheta(k) \leq \frac{\pi\sqrt{k}}{2}.$$

(i) is evident: whenever X^1, \dots, X^L is a solution to $(S[\rho])$, we have

$$\|u\|_{\infty} \leq 1 \Rightarrow u_{\ell} B_{\ell} \succeq -X^{\ell} \forall \ell \Rightarrow B_0 + \rho \sum_{\ell} u_{\ell} B_{\ell} \succeq B_0 - \rho \sum_{\ell} X_{\ell} \succeq 0.$$

(ii): Assume that $(S[\rho])$ is **not** solvable, and let us prove that $\mathcal{A}[\vartheta(\mu)\rho]$ is not contained in the positive semidefinite cone, provided that $\vartheta(\mu)$ is chosen properly. There is nothing to prove when $B_0 \not\succeq 0$. Thus, let $B_0 \succeq 0$.

♣ **Step 1.** We have assumed that the system

$$\begin{array}{rcl} X^\ell & \succeq & \pm B_\ell, \ell = 1, \dots, L, \\ \rho \sum_{\ell=1}^L X^\ell & \preceq & B_0 \end{array} \quad (S[\rho])$$

has no solutions. Consider the semidefinite program

$$\text{Opt} = \min_{X^\ell, t} \left\{ t : \begin{array}{rcl} X^\ell & \succeq & \pm B_\ell, \ell = 1, \dots, L, \\ \rho \sum_{\ell=1}^L X^\ell & \preceq & B_0 + tI \end{array} \right\} \quad (P)$$

The problem clearly is feasible and has compact level sets, and is therefore solvable. Since $(S[\rho])$ has no solutions, the optimal value in (P) is positive. Since the problem clearly is strictly feasible, the dual problem is solvable with positive optimal value.

$$\text{Opt} = \min_{X^\ell, t} \left\{ t : \begin{array}{l} X^\ell \succeq \pm B_\ell, \ell = 1, \dots, L, \\ tI - \rho \sum_{\ell=1}^L X^\ell \succeq -B_0 \end{array} \right\} \quad (P)$$

♣ **Step 2.** Let us build the dual. Let

- $U_\ell \succeq 0$ be the “aggregation weights” for the constraints $X^\ell \succeq B_\ell$,
- $V_\ell \succeq 0$ be the aggregation weights for the constraints $X^\ell \succeq -B_\ell$,
- $W \succeq 0$ be the aggregation weight for the last LMI in (P).

♣ Aggregating the LMIs in (P) with the above weights, we get the inequality

$$\sum_{\ell} \text{Tr}([U_\ell + V_\ell - \rho W] X^\ell) + t \text{Tr}(W) \geq \sum_{\ell} \text{Tr}([U_\ell - V_\ell] B_\ell) - \text{Tr}(W B_0)$$

Restricting the weights to be such that the left hand side in this inequality, as a function of X^ℓ and t , is identically equal to the objective in (P):

$$U_\ell + V_\ell = \rho W, \ell = 1, \dots, L; \quad \text{Tr}(W) = 1 \quad (*)$$

we obtain the lower bound $\sum_{\ell} \text{Tr}([U_\ell - V_\ell] B_\ell) - \text{Tr}(W B_0)$ on Opt. The dual problem is to maximize this bound:

$$\max_{U_\ell, V_\ell, W} \left\{ \sum_{\ell} \text{Tr}([U_\ell - V_\ell] B_\ell) - \text{Tr}(W B_0) : \begin{array}{l} U_\ell + V_\ell = W, \ell = 1, \dots, L \\ \text{Tr}(W) = 1, U_\ell, V_\ell, W \succeq 0 \end{array} \right\} \quad (D)$$

and we know that the optimal value in the dual is positive.

$$0 < \max_{U_\ell, V_\ell, W} \left\{ \sum_{\ell} \text{Tr}([U_\ell - V_\ell]B_\ell) - \text{Tr}(WB_0) : \begin{array}{l} U_\ell + V_\ell = W, \ell = 1, \dots, L \\ \text{Tr}(W) = 1 \\ U_\ell, V_\ell, W \succeq 0 \end{array} \right\} \quad (D)$$

♣ In (D), we can carry out maximization in U_ℓ, V_ℓ analytically. Indeed, this maximization requires solving the problem of the form

$$m(B, Z) \equiv \max_{U, V} \{ \text{Tr}([U - V]B) : U \succeq 0, V \succeq 0, U + V = Z \}, \quad (A)$$

with given $Z \succeq 0$. Assume for a moment that $Z \succ 0$, and let us pass in (A) to new variables

$$P = Z^{-1/2}UZ^{-1/2}, \quad Q = Z^{-1/2}VZ^{-1/2}.$$

We have

$$\begin{aligned} U \succeq 0 &\Leftrightarrow P \succeq 0, \quad V \succeq 0 \Leftrightarrow Q \succeq 0, \quad U + V = Z \Leftrightarrow P + Q = I \\ \text{Tr}([U - V]B) &= \text{Tr}(Z^{1/2}[P - Q]Z^{1/2}B) = \text{Tr}([P - Q] \underbrace{(Z^{1/2}BZ^{1/2})}_C) \\ &\Rightarrow m(B, Z) = \max_P \{ \text{Tr}([2P - I]C) : 0 \preceq P \preceq I \} \end{aligned}$$

⇒ representing $C = U \text{Diag}\{\lambda(C)\} U^T$ with orthogonal U ,

$$\begin{aligned}
 m(B, Z) &= \max_P \{ \text{Tr}([2P - I]C) : 0 \preceq P \preceq I \} \\
 &= \max_P \{ \text{Tr}(U^T[2P - I]U \text{Diag}\{\lambda(C)\}) : 0 \preceq P \preceq I \} \\
 &= \max_P \left\{ \text{Tr}([2 \underbrace{U^T P U}_R - I] \text{Diag}\{\lambda(C)\}) : 0 \preceq P \preceq I \right\} \\
 &= \max_P \{ \text{Tr}(U^T[2P - I]U \text{Diag}\{\lambda(C)\}) : 0 \preceq P \preceq I \} \\
 &= \max_R \{ \text{Tr}([2R - I] \text{Diag}\{\lambda(C)\}) : 0 \preceq R \preceq I \} \\
 &= \max_R \{ \sum_i \lambda_i(C) (2R_{ii} - 1) : 0 \preceq R \preceq I \} \\
 &= \sum_i |\lambda_i(C)|.
 \end{aligned}$$

By continuity arguments, the resulting equality (proved when $Z \succ 0$) holds true for $Z \succeq 0$ as well.

$$0 < \max_{U_\ell, V_\ell, W} \left\{ \sum_\ell \text{Tr}([U_\ell - V_\ell] B_\ell) - \text{Tr}(W B_0) : \begin{array}{l} U_\ell + V_\ell = W, \ell = 1, \dots, L \\ \text{Tr}(W) = U_\ell, V_\ell, W \succeq 0 \end{array} \right\} \quad (D)$$

$$\max_{U, V} \left\{ \text{Tr}([U - V] B) : \begin{array}{l} U, V \succeq 0 \\ U + V = Z \end{array} \right\} = \|\lambda(Z^{1/2} B Z^{1/2})\|_1$$

♣ After optimization in U_ℓ and V_ℓ , (D) becomes

$$0 < \max_{W \succeq 0} \left\{ \sum_\ell \rho \|\lambda(W^{1/2} B_\ell W^{1/2})\|_1 - \text{Tr}(W B_0) : \text{Tr}(W) = 1 \right\},$$

so that

$$\rho \sum_{\ell=1}^L \|\lambda(W^{1/2} B_\ell W^{1/2})\|_1 > \text{Tr}(W^{1/2} B_0 W^{1/2})$$

for appropriately chosen $W \succeq 0$.

Situation: Assuming that $(S[\rho])$ has no solutions, there exists $W \succeq 0$ such that

$$\rho \sum_{\ell=1}^L \|\lambda(W^{1/2} B_\ell W^{1/2})\|_1 > \text{Tr}(W^{1/2} B_0 W^{1/2}). \quad (*)$$

Step 3: Probabilistic interpretation of (*). Let ξ be the standard (zero mean, unit covariance matrix) Gaussian random vector in \mathbf{R}^m , and A be a symmetric $m \times m$ matrix of rank k . What is the expectation of the *modulus* of the quadratic form $\xi^T A \xi$? Representing $A = U \text{Diag}\{\lambda\} U^T$ with orthogonal U and setting $\eta = U^T \xi$, observe that the distribution of η is exactly the same as the one of ξ ; thus, our question becomes what is the expectation of

$$\zeta = \left| \sum_{i=1}^k \lambda_i \eta_i^2 \right|$$

where $\eta_i \sim \mathcal{N}(0, 1)$ are independent of each other. Common sense says that the expectation of ζ is *at least* $O(1) \|\lambda\|_2 \geq O(1) k^{-1/2} \|\lambda\|_1$. Specifically, setting

$$\vartheta(k) = \frac{1}{\min \left\{ \int \left| \sum_{i=1}^k \lambda_i \eta_i^2 \right| (2\pi)^{-k/2} e^{-\frac{\eta_1^2 + \dots + \eta_k^2}{2}} d\eta_1 \dots d\eta_k : \|\lambda\|_1 = 1 \right\}}$$

one can easily verify that

$$\vartheta(1) = 1, \vartheta(2) = \frac{\pi}{2}, \vartheta(k) \leq \frac{\pi \sqrt{k}}{2},$$

while by definition of $\vartheta(\cdot)$ one has

$$\vartheta(\text{Rank}(A)) \mathbf{E} \{ |\xi^T A \xi| \} \geq \|\lambda(A)\|_1$$

for every symmetric matrix A .

Situation: Assuming that $(S[\rho])$ has no solutions, there exists $W \succeq 0$ such that

$$\rho \sum_{\ell=1}^L \|\lambda(W^{1/2} B_\ell W^{1/2})\|_1 > \text{Tr}(W^{1/2} B_0 W^{1/2}). \quad (*)$$

Besides this, we have seen that with properly chosen function $\vartheta(\cdot)$ such that

$$\vartheta(1) = 1, \vartheta(2) = \frac{\pi}{2}, \vartheta(k) \leq \frac{\pi\sqrt{k}}{2},$$

for standard Gaussian vector ξ and every symmetric matrix A one has

$$\vartheta(\text{Rank}(A)) \mathbf{E} \{ |\xi^T A \xi| \} \geq \|\lambda(A)\|_1 \quad (**)$$

- Let $\xi \sim \mathcal{N}(0, I_m)$ and let $\mu = \max_{\ell \geq 1} \text{Rank}(B_\ell)$. We have

$$\begin{aligned} \mathbf{E} \left\{ \rho \sum_{\ell=1}^k \vartheta(\mu) |\xi^T W^{1/2} B_\ell W^{1/2} \xi| \right\} &\geq \rho \sum_{\ell=1}^L \|\lambda(W^{1/2} B_\ell W^{1/2})\|_1 \text{ [by (**)]} \\ &> \text{Tr}(W^{1/2} B_0 W^{1/2}) \text{ [by (*)]} = \mathbf{E} \{ \xi^T W^{1/2} B_0 W^{1/2} \xi \} \text{ [evident]} \end{aligned}$$

Thus,

$$\mathbf{E} \left\{ \xi^T W^{1/2} B_0 W^{1/2} \xi - \xi \rho \vartheta(\mu) \sum_{\ell=1}^k |\xi^T W^{1/2} B_\ell W^{1/2} \xi| \right\} < 0.$$

It follows that there exists $\bar{\eta} = W^{1/2} \bar{\xi}$ such that $\bar{\eta}^T B_0 \bar{\eta} - \rho \vartheta(\mu) \sum_{\ell=1}^k |\bar{\eta}^T B_\ell \bar{\eta}| < 0$. Setting $u_\ell = -\rho \vartheta(\mu) \text{sign}(\bar{\eta}^T B_\ell \bar{\eta})$, we get $\|u\|_\infty = \rho \vartheta(\mu)$ and $\bar{\eta}^T \underbrace{\left[B_0 + \sum_{\ell} u_\ell B_\ell \right]}_{\in \mathcal{A}[\vartheta(\mu)\rho]} \bar{\eta} < 0$, i.e.,

$$\mathcal{A}[\vartheta(\mu)\rho] \not\subset \mathbf{S}_+^m.$$

□

F. Approximate S-Lemma. Consider quadratic maximization over a single-parametric family of similar to each other ellitopes:

$$\text{Opt}_*[\rho] = \max_{z \in \mathcal{Z}[\rho]} [z^T Q z + 2q^T z]$$

$$\left[\begin{array}{l} \mathcal{Z}[\rho] = \{z \in \mathbf{R}^n : \exists t \in \mathcal{T} : z^T S_k z \leq \rho t_k, k \leq K\} \\ S_k \succeq 0, \sum_k S_k \succ 0, \mathcal{T} \subset \mathbf{R}_+^K : \text{ tractable monotone convex compact set, } \text{int } \mathcal{T} \neq \emptyset \end{array} \right] [\rho > 0]$$

along with efficiently computable quantities

$$\text{Opt}[\rho] = \min_{\lambda, \mu} \left\{ \rho \phi_{\mathcal{T}}(\lambda) + \mu : \left\{ \begin{array}{l} \lambda, \mu \geq 0 \\ \left[\frac{Q}{q^T} \middle| q \right] \preceq \left[\frac{\sum_k \lambda_k S_k}{\mu} \right] \end{array} \right\} \right.$$

$$\left. \left[\phi_{\mathcal{T}}(\lambda) = \max_{t \in \mathcal{T}} \lambda^T t : \text{ support function of } \mathcal{T} \right] \right\}$$

Then

$$\text{Opt}_*[\rho] \leq \text{Opt}[\rho] \leq \text{Opt}_*[\kappa \rho], \quad \kappa = 3 \ln(6K).$$

- **Note:** In the homogeneous case $q = 0$, this result basically reproduces what we know about the quality of semidefinite relaxation when maximizing a homogeneous quadratic form over an ellitope. Moreover, $\text{Opt}[\rho]$ is *exactly* the semidefinite relaxation bound

$$\text{Opt}_*[\rho] = \max_{x \in \mathcal{Z}[\rho]} [x^T Q x + 2q^T x] = \max_{[x; \textcolor{red}{r}] \in \mathcal{Z}[\rho] \times \{\textcolor{red}{r}^2 \leq 1\}} [x^T Q x + 2\textcolor{red}{r} q^T x]$$

on the maximum of *homogeneous* quadratic form of $[x; r]$ over the ellitope $\mathcal{Z}[\rho] \times \{r^2 \leq 1\}$.

Note: The novelty is *not* in how the relaxation bound is built, but in *how we quantify its tightness*:

— in our previous results, the tightness was the ratio of the bound $\text{Opt}[\rho]$ on the “quantity of interest” $\text{Opt}_*[\rho]$ to the actual value of this quantity.
 — in constast, in Approximate \mathcal{S} -Lemma tightness is quantified in terms of the “size” ρ of the ellitope over which we are maximizing — by which factor κ should we increase ρ in order to get $\text{Opt}[\rho] \leq \text{Opt}_*[\textcolor{red}{\kappa}\rho]$.

- In the homogeneous case $q = 0$, $\text{Opt}_*[\rho]$ is proportional to ρ , both ways to quantify tightness are the same; in the inhomogeneous case, they are different.

Application: Affinely Adjustable Robust Counterparts of Uncertain Linear Programs with ellitopic uncertainty

♠ When speaking about AARC's (Affinely Adjustable Robust Counterparts) of uncertain LP's

$$\{\min\{c^T x : Ax \leq b\} \mid [A, b] \in \mathcal{U}\},$$

we have seen that *in the case of fixed recourse*, i.e., when all coefficients at adjustable variables are certain, *AARC becomes a problem with semi-infinite scalar linear constraints affinely affected by the uncertainty and therefore is reducible to Conic Quadratic Programming, provided the uncertainty set is CQr with essentially strictly feasible CQR.*

- Needless to say, we can replace here CQr (CQR) with SDr (SDR).

♠ When there is no fixed recourse, the AARC becomes a problem with semiinfinite scalar constraints *quadratically* affected by ucertainty

⇒ AARC can become intractable even for a simple uncertainty set \mathcal{U} .

Partial remedy: Pass from intractable AARC to its safe tractable approximation. *With parametric ellitopic uncertainty*, a good approximation is given by Approximate S-Lemma.

Uncertain LP to be solved in Affine Decision Rules: $\{\min\{c^T x : Ax \leq b\} \mid [A, b] \in \mathcal{U}\}$

♠ Assume that the uncertainty is *elliptic* and is parameterized by “uncertainty level” ρ , specifically, $[A, b]$ is affinely parameterized by “perturbation” ζ running through ellitope from a single-parametric family:

$$\mathcal{U} = \mathcal{U}[\rho] := \left\{ [A, b] = [A_*, b_*] + \sum_i \zeta_i [A^i, b^i] : \zeta \in \mathcal{Z}[\rho] \right\}$$

$$\mathcal{Z}[\rho] = \{ \zeta : \exists t \in \mathcal{T} : \zeta^T S_k \zeta \leq \rho t_k, k \leq K \}$$

with S_k, \mathcal{T} as required by the definition of an ellitope.

With this parametric uncertainty, the AARC of our uncertain LP becomes parametric semiinfinite problem of the form

$$\text{Opt}_*[\rho] = \min_y \left\{ c^T y : [\zeta; 1]^T Q_i[y] [\zeta; 1] \leq 0 \forall \zeta \in \mathcal{Z}[\rho], i \leq I \right\} \quad \text{AARC}[\rho]$$

- y : design variables (coefficients of affine decision rules)
- $Q_i[y]$: *affine* in y symmetric matrices.

$$\text{Opt}_*[\rho] = \min_y \{c^T y : [\zeta; 1]^T Q_i[y][\zeta; 1] \leq 0 \forall \zeta \in \mathcal{Z}[\rho], i \leq I\} \quad \text{AARC}[\rho]$$

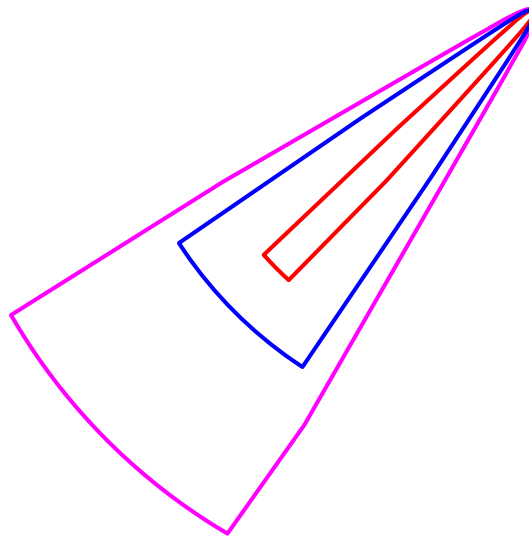
Fact: Utilizing Approximate \mathcal{S} -Lemma as explained in Section 3.5.3.1 of Lecture Notes, one can build an *efficiently solvable* parametric convex problem APPR $[\rho]$ in “variables of interest” y and “analysis variables” ω and with the same objective $c^T y$ as in AARC $[\rho]$ such that

- *The y -component of every feasible solution (y, ω) to APPR $[\rho]$ is feasible for AARC $[\rho]$.*

As a result, we can find efficiently an approximate solution to the problem of interest AARC $[\rho]$ with the value of the objective upper-bounded by the optimal value Opt $[\rho]$ of the approximating problem APPR $[\rho]$

- *One has $\text{Opt}[\rho] \leq \text{Opt}_*[\kappa\rho]$, $\kappa = 3 \ln(6K)$*

In words: We can build a computationally tractable *safe approximation* $\text{APPR}[\rho]$ of (by itself, *NP*-hard in general) problem of interest $\text{AARC}[\rho]$, safety meaning that the projection $\mathcal{A}[\rho]$ of the feasible set of the approximation on the space of variables of interest y is *inside* the true feasible set $\mathcal{R}[\rho]$ of $\text{AARC}[\rho]$. This approximation is reasonably tight: $\mathcal{A}[\rho]$ contains $\mathcal{R}[\kappa\rho]$ with “moderate” – logarithmic in K – factor κ .



Magenta curve: boundary of the true robust feasible set $\mathcal{R}[\rho]$ at uncertainty level ρ
Red curve: boundary of the true robust feasible set $\mathcal{R}[\kappa\rho]$ at uncertainty level $\kappa\rho$
Blue curve: boundary of safe tractable approximation $\mathcal{A}[\rho]$ of $\mathcal{R}[\rho]$

Application: Robust Conic Quadratic Programming with ellitopic uncertainty. Consider a conic quadratic inequality

$$\|Ax + b\|_2 \leq \tau \quad (\text{CQI})$$

in variables x, τ and assume that the data $[A, b]$ of this c.q.i. is not known exactly and run through a given uncertainty set \mathcal{U} . How to process the Robust Counterpart

$$\|Ax + b\|_2 \leq \tau \quad \forall [A, b] \in \mathcal{U} \quad (\text{RC})$$

of (CQI)?

♣ Assume that the uncertainty is *ellitopic* and is parameterized by “uncertainty level” ρ , specifically, $[A, b]$ is affinely parameterized by “perturbation” ζ running through ellitope from a single-parametric family:

$$\mathcal{U} = \mathcal{U}[\rho] := \left\{ [A, b] = [A_*, b_*] + \sum_i \zeta_i [A^i, b^i] : \zeta \in \mathcal{Z}[\rho] \right\}$$

$$\mathcal{Z}[\rho] = \{ \zeta : \exists t \in \mathcal{T} : \zeta^T S_k \zeta \leq \rho t_k, k \leq K \}$$

with S_k, \mathcal{T} as required by definition of an ellitope.

With this parametric uncertainty, (RC) also becomes parametric:

$$\|Ax + b\|_2 \leq \tau \quad \forall [A, b] \in \mathcal{U}[\rho] \quad \text{RC}[\rho]$$

$$\mathcal{U} = \mathcal{U}[\rho] := \{[A, b] = [A_*, b_*] + \sum_i \zeta_i [a^i, b^i] : \zeta \in \mathcal{Z}[\rho]\}$$

$$\mathcal{Z}[\rho] = \{\zeta : \exists t \in \mathcal{T} : \zeta^T S_k \zeta \leq \rho t_k, k \leq K\}$$

$$\|Ax + b\|_2 \leq \tau \quad \forall [A, b] \in \mathcal{U}[\rho] \quad \text{RC}[\rho]$$

Fact: Utilizing Approximate \mathcal{S} -Lemma as explained in Section 3.5.3.2 of Lecture Notes, one can build an explicit computationally tractable system $\mathcal{S}[\rho]$ of parameterized by ρ convex constraints on “variables of interest” x, τ and “analysis variables” ω such that

- if x, τ can be extended by appropriately chosen value of ω to yield a feasible solution to $\mathcal{S}[\rho]$, then x, τ is feasible for $\text{RC}[\rho]$
- if x, τ **cannot** be extended to a feasible solution to $\mathcal{S}[\rho]$, then x, τ is **not** feasible for $\text{RC}[\kappa\rho]$, with $\kappa = 3 \ln(6K)$

In words: We can build a computationally tractable *safe approximation* $\mathcal{S}[\rho]$ of (by itself, NP -hard in general) Robust Counterpart $\text{RC}[\rho]$, safety meaning that the projection $\mathcal{A}[\rho]$ of the feasible set of the approximation on the space of “variables of interest” x, τ is *inside* the true feasible set $\mathcal{R}[\rho]$ of $\text{RC}[\rho]$. This approximation is reasonably tight: $\mathcal{A}[\rho]$ contains $\mathcal{R}[\kappa\rho]$ with “moderate” – logarithmic in K – factor κ .

How it Works: Antenna Synthesis revisited

♠ When motivating Robust LP, we considered Antenna Design problem, where one was interested to approximate the desired *target diagram* (periodic function of altitude angle) by a linear combination of diagrams of given antenna elements (concentric planar circles), and the uncertainty came from implementation errors.

- When quantifying the discrepancy between the target and the synthesized diagrams by $\|\cdot\|_\infty$ -distance on a finite grid, the problem of interest becomes uncertain LP problem, and its Robust Counterpart is easy to process.
- When quantifying the discrepancy between the target diagram D_* and the synthesized diagrams D by $\|\cdot\|_2$ -distance on a finite grid, the problem of interest becomes an uncertain conic quadratic inequality

$$\{\|Ax + b\|_2 \leq \tau \mid [A, b] \in \mathcal{U}[\epsilon]\}, \quad \mathcal{U}[\epsilon] = \{[A, b] = [A_*, b_*] + \sum_k \zeta_k [A^k, 0] : |\zeta_k| \leq \epsilon \forall k\}$$

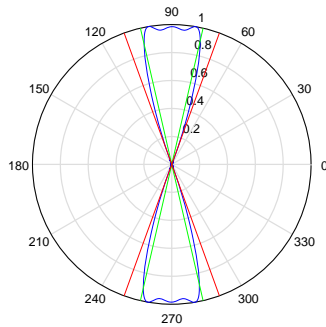
- A^i : obtained from A_* by zeroing out all columns except for k -th one

Note: Up to reparameterization $\epsilon \mapsto \rho = \epsilon^2$, we are in the case of single-parametric ellitopic uncertainty:

$$\mathcal{U}[\epsilon] = \{\zeta : \exists t \in [0, 1]^K : \zeta_k^2 \leq \epsilon^2 t_k, k \leq K\}$$

and can apply our machinery to safely approximate the Robust Counterpart of our uncertain problem.

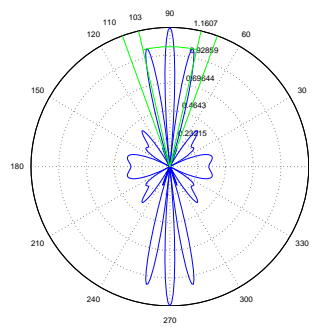
- The *Nominal design* – the one corresponding to $\epsilon = 0$ – is completely unstable w.r.t. small implementation errors



Dream

no errors

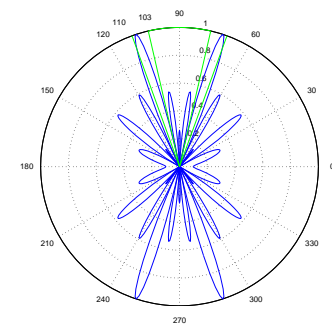
$$\|D_* - D\|_2 = 0.014$$



Reality

sampled D , 0.1% errors

$$\|D_* - D\|_2 \in [0.17, 0.89]$$



Reality

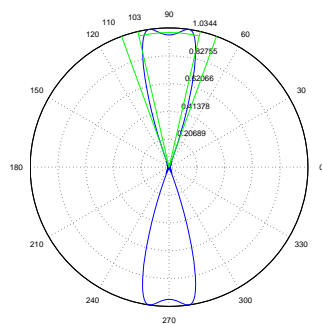
sampled D , 2% errors

$$\|D_* - D\|_2 \in [2.9, 19.6]$$

Nominal design: dream and reality. Range $\|D_* - D\|$ obtained from 100-element sample.

Note: D_* is 1 in acute angles with green sides and is 0 in obtuse angles with magenta sides

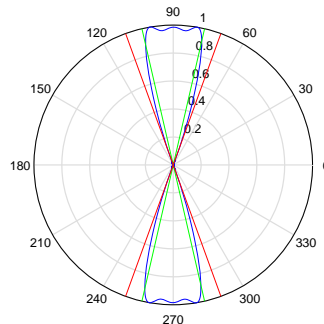
♠ Safe tractable approximation of the Robust Counterpart of our uncertain conic quadratic inequality yields incomparably more meaningful *Robust design*:



Dream

no errors

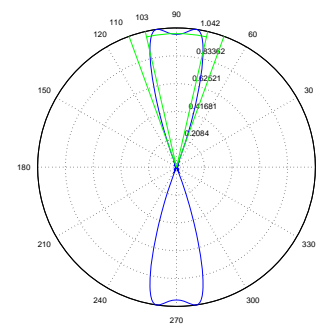
$$\|D_* - D\|_2 = 0.025$$



Reality

sampled D , 0.1% errors

$$\|D_* - D\|_2 \approx 0.025$$



Reality

sampled D , 2% errors

$$\|D_* - D\|_2 \approx 0.025$$

Robust design: dream and reality, data over 100-element sample.

Proof of \mathcal{S} -Lemma

\mathcal{S} -Lemma: Let A, B be symmetric $m \times m$ matrices such that $\bar{x}^T A \bar{x} > 0$ for certain \bar{x} . Then the implication

$$\forall x : x^T A x \geq 0 \Rightarrow x^T B x \geq 0 \quad (*)$$

holds true *iff*

$$\exists \lambda \geq 0 : B \succeq \lambda A \quad (**)$$

- $(**) \Rightarrow (*)$: evident.
- $(*) \Rightarrow (**)$: Consider the following “relaxation” of $(*)$:

$$\forall (X \succeq 0) : \quad \text{Tr}(XA) \geq 0 \Rightarrow \text{Tr}(XB) \geq 0 \quad (R)$$

Step 1: Claim: Under the premise of \mathcal{S} -Lemma, (R) is equivalent to $(**)$.

Indeed, under the premise of \mathcal{S} -Lemma, the semidefinite program

$$\min_X \{ \text{Tr}(BX) : X \succeq 0, \text{Tr}(AX) \geq 0 \}$$

is strictly feasible, and (R) just says that the optimal value in this problem (which is either 0, or $-\infty$) is 0. Applying Conic Duality Theorem, this is the case *iff* the dual problem

$$\max_{\lambda, S} \{ 0 : B = \lambda A + S, S \succeq 0, \lambda \geq 0 \}$$

is feasible, i.e., *iff* $(**)$ takes place.

- Thus, to complete the proof of \mathcal{S} -Lemma, it suffices to verify that

$$(*) \Rightarrow (R).$$

$$\forall x : x^T A x \geq 0 \Rightarrow x^T B x \geq 0 \quad (*)$$

$$\forall (X \succeq 0) : \quad \text{Tr}(XA) \geq 0 \Rightarrow \text{Tr}(XB) \geq 0 \quad (R)$$

Goal: to prove that $(*) \Rightarrow (R)$.

Proof: Assume that $(*)$ takes place and that $X \succeq 0$ is such that $\text{Tr}(AX) \geq 0$; we should prove that then $\text{Tr}(BX) \geq 0$ as well.

Let us set

$$\bar{A} \equiv X^{1/2} A X^{1/2} = U \text{Diag}\{\lambda\} U^T, \quad \eta = X^{1/2} U \xi,$$

where ξ is a random vector with independent coordinates taking values ± 1 with probabilities $1/2$. We have

$$\eta^T A \eta = \xi^T U^T X^{1/2} A X^{1/2} U \xi = \xi^T \text{Diag}\{\lambda\} \xi = \text{Tr}(\text{Diag}\{\lambda\}) = \text{Tr}(X^{1/2} A X^{1/2}) = \text{Tr}(AX) \geq 0$$

$$\Downarrow \quad (*)$$

$$\eta^T B \eta \geq 0$$

$$\Downarrow$$

$$0 \leq \mathbf{E} \{ \eta^T B \eta \} = \mathbf{E} \{ \xi^T U^T X^{1/2} B X^{1/2} U \xi \} = \text{Tr}(U^T X^{1/2} B X^{1/2} U) = \text{Tr}(X^{1/2} B X^{1/2}) = \text{Tr}(BX)$$

Q.E.D.

Extremal Ellipsoids

♣ *An ellipsoid* in \mathbf{R}^n is, by definition, the image of the unit Euclidean ball

$$B_n = \{u \in \mathbf{R}^n : u^T u \leq 1\}$$

under an affine mapping $u \mapsto Au + a$:

$$E = \{x = Au + a : u^T u \leq 1\}. \quad (*)$$

Note:

- An ellipsoid is a convex compact set symmetric w.r.t. a . Consequently, *The center a of an ellipsoid E is uniquely defined by the set E .*
- An ellipsoid E is “full-dimensional”, that is, possesses a nonempty interior, *iff* A in $(*)$ is nonsingular.
- Matrix A in $(*)$ is *not* uniquely defined by E ; replacing in $(*)$ A with AU , where U is orthogonal, we preserve the right hand side set. In particular, *Among the matrices A participating in representations of a given ellipsoid E , there exists a positive semidefinite one, which is uniquely defined by the set E .*

$$E = \{x = Au + a : u^T u \leq 1\}. \quad (*)$$

♣ **Bottom line:** If a set $E \subset \mathbb{R}^n$ is an ellipsoid, that is, admits a representation $(*)$, then E admits a representation $(*)$ with $A \succeq 0$. In this *image representation* of E , both $A \succeq 0$ and a are uniquely defined by the set E .

● An ellipsoid with image representation given by matrix $A \succeq 0$ and vector a will be denoted $\mathcal{E}(A, a)$:

$$\mathcal{E}(A, a) = \{Au + a : u^T u \leq 1\} \subset \mathbb{R}^n \quad [A \in \mathbf{S}_+^n, a \in \mathbb{R}^n]$$

Inequality Representation of Full-Dimensional Ellipsoid and Elliptic Cylinders

♣ Consider a quadratic form

$$f(x) = x^T P x - 2p^T x \quad (f)$$

on \mathbf{R}^n . This form is below bounded if and only if the following two conditions hold:

- The form is convex: $P \succeq 0$
- The Fermat equation

$$\nabla f(x) = 0 \Leftrightarrow P x = p \quad (F)$$

has a solution x_* .

In particular, *if* $f(\cdot)$ is below bounded, *then* there exists a representation

$$f(x) = x^T B^2 x - 2b^T B x, \quad (*)$$

where $B \succeq 0$ and $b \in \text{Im } B$. Indeed, in the case of 1), 2) one can set $B = P^{1/2}$, $b = P^{1/2} x_*$.

Vise versa, *if* $f(\cdot)$ can be represented in the form $(*)$ with $B \succeq 0$ and $b \in \text{Im } B$, then 1), 2) hold true, so that below boundedness of f is *equivalent* to the possibility to represent f by $(*)$ with $B \succeq 0$, $b \in \text{Im } B$.

♣ A below bounded quadratic form $f(x)$ can be represented as

$$\begin{aligned} f(x) &= x^T B^2 x - 2b^T Bx \\ [B \succeq 0, b \in \text{Im} B] \end{aligned} \quad (*)$$

Note that *Form (*) attains its minimum, which is equal to $-b^T b$.* Indeed, relation $b \in \text{Im} B$ means that $b = Bx_*$ for certain x_* . Then

$$\nabla f(x_*) = 2B^2 x_* - 2Bb = 2B^2 x_* - 2B^2 x_* = 0$$

that is, x_* is a critical point and thus – a minimizer of the *convex* function f . We have

$$f(x_*) = \underbrace{(Bx_*)^T}_b (Bx_*) - 2b^T Bx_* = -b^T Bx_* = -b^T b.$$

♣ Let f be a below bounded quadratic form on \mathbf{R}^n , and let f_* be its minimum value. The “nontrivial” levels sets of f , that is, level sets of the form

$$C = \{x : f(x) \leq f_* + r^2\} \quad [r > 0] \quad (C)$$

are called “elliptic cylinders”.

$$C = \{x : f(x) \leq f_* + r^2\} \quad [r > 0] \quad (C)$$

♠ In representation (*), an elliptic cylinder is

$$C = \{x : \|Bx - b\|_2^2 \leq r^2\}$$

When $\theta > 0$, the data (B, b, r) and $(\theta B, \theta b, \theta r)$ define the same cylinder, so that by normalization we may assume that $r = 1$. The representation

$$C = \{x : \|b - Bx\|_2^2 \leq 1\} \quad [B \succeq 0, b \in \text{Im} B]$$

is called *inequality representation* of elliptic cylinder. The data B, b of this representation are uniquely defined by the set C .

$$C = \{x : \|b - Bx\|_2^2 \leq 1\} \quad [B \succeq 0, b \in \text{Im}B]$$

- C is bounded iff $B \succ 0$, and iff C is a full-dimensional ellipsoid. Indeed,
- We clearly have $C = C + \text{Ker}B$. Thus, if C is bounded, then $\text{Ker}B = \{0\}$, that is, $B \succ 0$. Vice versa, if $B \succ 0$, then C clearly is bounded.
- We have

$$\begin{aligned} B \succ 0 \Rightarrow \{x : \|\underbrace{Bx - b}_u\|_2^2 \leq 1\} &= \{x = B^{-1}u + B^{-1}b : u^T u \leq 1\} \\ A \succ 0 \Rightarrow \{x = Au + a : u^T u \leq 1\} &= \{x : \|\underbrace{A^{-1}x - A^{-1}a}_u\|_2^2 \leq 1\} \end{aligned}$$

- When $B \succeq 0$ is degenerate, the elliptic cylinder C can be represented as the sum of the set

$$C_0 = \{x \in \text{Im}B : \|b - Bx\|_2^2 \leq 1\}$$

(which is a full-dimensional ellipsoid in the subspace $\text{Im}B = (\text{Ker}B)^\perp$) and the linear subspace $\text{Ker}B$.

Bottom line: We have defined

- Ellipsoids in \mathbf{R}^n – sets representable as

$$E = \mathcal{E}(A, a) \equiv \{x = Au + a : u^T u \leq 1\}, \quad (E)$$

where $A \succeq 0$. The data A, a of this *image representation* of E are uniquely defined by the set E itself.

Ellipsoid E is full-dimensional (that is, $\text{int } E \neq \emptyset$) if and only if $A \succ 0$, otherwise the ellipsoid is “flat” – it is contained in the plane $a + \text{Im}A$, which is a proper affine subspace of \mathbf{R}^n .

- Elliptic cylinders in \mathbf{R}^n – sets representable as

$$C = \mathcal{C}(B, b) \equiv \{x : \|Bx - b\|_2^2 \leq 1\} \quad (C)$$

where $B \succeq 0$ and $b \in \text{Im}B$. The data B, b of this *inequality representation* of C are uniquely defined by the set C itself.

Elliptic cylinder C is bounded if and only if $B \succ 0$, and in this case C is just a full-dimensional ellipsoid, otherwise C is the sum of the kernel of B and a full-dimensional ellipsoid in the image space of B .

- Full-dimensional ellipsoids E admit both image and inequality representations:

$$A \succ 0 \Rightarrow E \equiv \{x = Au + a : u^T u \leq 1\} = \{x : \|Bx - b\|^2 \leq 1\}$$

with the parameters of the representations linked by the relations

$$\begin{aligned} B = A^{-1} &\Leftrightarrow A = B^{-1} \\ b = A^{-1}a &\Leftrightarrow a = B^{-1}b \end{aligned}$$

Volume of an Ellipsoid

♣ Under affine transformation

$$x \mapsto Ax + a : \mathbf{R}^n \rightarrow \mathbf{R}^n,$$

n -dimensional volumes of sets are multiplied by $|\text{Det}(A)|$:

$$\text{Vol}(\{y = Ax + a : x \in U\}) = |\text{Det}(A)|\text{Vol}(U).$$

In particular, *The volume of ellipsoid $\mathcal{E}(A, a)$ is $|\text{Det}(A)|$ times the volume of the unit Euclidean ball in \mathbf{R}^n .*

♣ In what follows, it is convenient to choose, as the unit of volume in \mathbf{R}^n , the volume of the unit Euclidean ball rather than the volume of the unit cube. With this convention, *The volume of ellipsoid $\mathcal{E}(A, a)$ is $|\text{Det}(A)|$, and the volume of full-dimensional ellipsoid $\mathcal{C}(B, b)$ is*

$$\frac{1}{|\text{Det}(B)|}.$$

Half-Axes of an Ellipsoid

♣ Let $E = \mathcal{E}(A, a)$, let e_i be the orthonormal eigenbasis of A , and λ_i be the corresponding eigenvalues. Let $\xi_i(x)$ be the coordinates of x in the basis e_1, \dots, e_n . The fact that $x = Au + a$ is equivalent to the relations

$$\xi_i(x) - \xi_i(a) = \lambda_i \xi_i(u),$$

so that the fact that $x \in E$ is equivalent to

$$\sum_i \frac{(\xi_i(x) - \xi_i(a))^2}{\lambda_i^2} \leq 1 \qquad \left[\frac{t^2}{0^2} = \begin{cases} 0, & t = 0 \\ +\infty, & t \neq 0 \end{cases} \right]$$

Geometrically: λ_i are the *half-axes* $\chi_i(E)$ of E , and e_i are the directions of the principal axes of E .

♣ For a full-dimensional ellipsoid $E = \mathcal{E}(A, a)$, all half-axes $\chi_i(E) \equiv \lambda_i(A)$ are positive. In terms of the inequality representation $E = \mathcal{C}(B, b)$ of the ellipsoid, the half-axes are

$$\chi_i(E) = \lambda_i^{-1}(B).$$

♣ In the case of degenerate B , elliptic cylinder $C = \mathcal{C}(B, b)$ is the sum of an ellipsoid C_0 in the subspace $\text{Im}B$ and the linear subspace $\text{Ker}B$ which is orthogonal to C_0 . It makes sense to define the first $\text{Rank}(B)$ half-axes of C as $\chi_i(C) = \lambda_i^{-1}(B)$, where $\lambda_i(B)$, $i = 1, \dots, \text{Rank}(B)$, are the nonzero eigenvalues of B , and the remaining $n - \text{Rank}(B)$ half-axes of C as $+\infty$.

♣ *The* basic problems on extremal ellipsoids are as follows:

Outer Approximation: (O): *Given a bounded nonempty set $X \subset \mathbf{R}^n$, find the “smallest” ellipsoid containing X .*

Inner Approximation: (I): *Given a nonempty set $X \subset \mathbf{R}^n$, find the “largest” ellipsoid contained in X .*

♣ In these problems, the “size” of an ellipsoid is an appropriate symmetric function of the half-axes, e.g.

- $\chi_1 \chi_2 \dots \chi_n$ (the volume),
- $\max_i \chi_i$ (the radius of the smallest circumscribed ball),
- $\min_i \chi_i$ (the radius of the largest inscribed ball),
- $\sum_i \chi_i^\alpha$,
- ...

- ♣ Extremal ellipsoids have numerous applications, including
 - “optimal” methods of Nonsmooth Convex Optimization,
 - identification and estimation in Control
 - accurate integration of ordinary differential equations,
 - ...

Example 1: Inscribed Ellipsoid Method. Theoretically optimal, in certain precise sense, method for solving to high accuracy a general non-smooth Convex Programming program

$$\min_X f(x)$$

(X is a convex polytope given by linear inequalities, f is convex and continuous on X) is the *Inscribed Ellipsoid Method*. At every step of this method, one should solve an auxiliary problem of the form *Find the largest volume ellipsoid contained in a polytope given by a list of linear inequalities*.

Example 2: Estimation in Dynamical System. Consider a Discrete Time Linear Dynamical System:

$$\begin{aligned} z(t+1) &= Az(t) \\ y(t) &= Cz(t) + \xi_t \end{aligned}$$

where

- $z(t)$ is the state at time t ,
- $y(t)$ is the observation at time t ,
- ξ_t is norm-bounded observation error: $\|\xi_t\|_2 \leq \rho$,
- A and C are known matrices.

Example: $z(t)$ is the position $x(t)$ and the velocity $v(t)$ of a plane flying at (unknown) constant velocity, and $y(t)$ are the observations of the position of the plane coming from a radar:

$$\begin{aligned} \begin{bmatrix} x(t+1) \\ v(t+1) \end{bmatrix} &= \begin{bmatrix} I_3 & I_3 \\ \hline & I_3 \end{bmatrix} \begin{bmatrix} x(t) \\ v(t) \end{bmatrix} \\ y(t) &= x(t) + \xi_t \end{aligned}$$

$$\begin{aligned} z(t+1) &= Az(t) \\ y(t) &= Cz(t) + \xi_t \end{aligned}$$

Since the dynamics is known, all we need to identify the motion is the initial state $z(0)$. Some information on $z(0)$ is contained in observations $y(t)$: given $y(t)$, we know that $z(0)$ belongs to the elliptic cylinder

$$C_t = \{z : \|CA^t z - y(t)\|_2^2 \leq \rho^2\},$$

and all we know at time T is that $z(0)$ belongs to the set

$$C^T = \bigcap_{t=0}^T C_t.$$

We may now want to build an estimate of $z(0)$ as the center of the smallest ball containing the set C^T , which is the Outer Ellipsoidal Approximation problem where you are interested to minimize the maximal half-axis of a circumscribed ellipsoid.

Example 3: Approximating reachable sets. Consider a controlled Discrete Time Linear Dynamical System:

$$z(t+1) = A_t z(t) + B_t u(t) + f_t, \quad z(0) = z_0 \quad (1)$$

• $z(t)$: states; • $u(t)$: controls; • f_t : known inputs; • A_t, B_t : known matrices.

Assume that the control $u(t)$ is bounded:

$$\|u(t)\|_2 \leq \rho_t. \quad (2)$$

The *reachable set* Z^T of system (1) – (2) at time T is the set of all possible states z of the system at time T :

$$Z^T = \{z : \exists \{u(t), \|u(t)\|_2 \leq \rho_t\}_{t=0}^{T-1} : z(T) = z\}.$$

$$Z^T = \{z : \exists \{u(t), \|u(t)\|_2 \leq \rho_t\}_{t=0}^{T-1} : z(T) = z\}.$$

Note:

- Z^T is “computationally tractable”; e.g., to optimize a linear form $c^T z$ over Z^T is the same as to solve the conic quadratic problem

$$\min_{\substack{u(0), \dots, u(T-1) \\ z(1), \dots, z(T)}} \left\{ c^T z(T) : \begin{array}{l} z(t+1) = A_t z(t) + B_t u(t) + f_t, \ 0 \leq t < T \\ \|u(t)\|_2 \leq \rho_t, \ 0 \leq t < T, \ z(0) = z_0 \end{array} \right\}$$

- Z^T is the arithmetic sum of T ellipsoids:

$$z(T) = z_0(T) + \sum_{\tau=0}^{T-1} \underbrace{A_T A_{T-1} \dots A_{\tau+1} B_{\tau}}_{B_{T,\tau}} u(\tau),$$

where $z_0(\cdot)$ is the trajectory of (1) corresponding to $u(\cdot) \equiv 0$. \Rightarrow

$$Z^T = z_0(T) + \sum_{\tau=0}^{T-1} B_{T,\tau} \{u : u^T u \leq \rho_t^2\}.$$

The reachable set Z^T , while computationally tractable, becomes more and more complicated as T grows. *In many applications it makes sense to look for simple – ellipsoidal – inner and outer approximations of Z^T .*

Tractability of Outer Ellipsoidal Approximation

♣ **Observation O.1:** *Let $X \subset \mathbb{R}^n$ be a nonempty compact set. Then the set \mathcal{X} of parameters B, b of *inequality* representations of elliptic cylinders containing X is convex.*

To prove that \mathcal{X} is convex, let $\lambda \in [0, 1]$, $(B, b), (C, c) \in \mathcal{X}$, so that $B \succ 0$, $C \succ 0$ and

$$\forall x \in X : \begin{cases} \|Bx - b\|_2 \leq 1 & [b \in \text{Im}B] \\ \|Cx - c\|_2 \leq 1 & [c \in \text{Im}C] \end{cases} \quad (*)$$

we should prove that $(D, d) = \lambda(B, b) + (1 - \lambda)(C, c) \in \mathcal{X}$. There is nothing to prove when $\lambda = 0$ or $\lambda = 1$, thus let $0 < \lambda < 1$. From (*) and Triangle inequality we get

$$\forall x \in X : \|Dx - d\|_2 \leq \lambda \|Bx - b\|_2 + (1 - \lambda) \|Cx - c\|_2 \leq 1;$$

thus, all we need is to verify that $d \in \text{Im}D$.

Situation:

$$\lambda \in (0, 1) \text{ \& } (D, d) = \lambda(B, b) + (1 - \lambda)(C, c)$$

Claim: $d \in \text{Im} D$.

Mini-Lemma: Let $A_i \succeq 0$ and $\lambda_i > 0$, $i = 1, \dots, K$, and let $A = \sum_i \lambda_i A_i$.
Then

$$\text{Ker} A = \bigcap_i \text{Ker} A_i \quad (a); \quad \text{Im} A = \text{Im} A_1 + \dots + \text{Im} A_K \quad (b)$$

Proof: For $C \succeq 0$, one has $\text{Ker} C = \{x : x^T C x = 0\}$. Since $\lambda_i > 0$ and $A_i \succeq 0$, it follows that $x^T A x = 0$ **iff** $x^T A_i x = 0$ for all i , which gives (a). (b) is equivalent to (a) by elementary Linear Algebra. \square

Since $0 < \lambda < 1$, both $B \succeq 0$ and $C \succeq 0$ enter the expression $D = \lambda B + (1 - \lambda)C$ with positive weights. By MiniLemma, it follows that $\text{Im} D = \text{Im} B + \text{Im} C$, whence $d = \lambda b + (1 - \lambda)c \in \text{Im} D$ due to $b \in \text{Im} B$, $c \in \text{Im} C$. \square

♣ **Observation O.2:** “Typical sizes” of full-dimensional ellipsoids E are *convex* (and thus easy-to-minimize) functions of the parameters B, b of the *inequality* representation of E . This is so, e.g., for the sizes

- $\text{Vol}(E) = \prod_i \chi_i(E)$ (volume) or $\text{Vol}^{1/n}(E)$ (average linear size)
- $\max_i \chi_i(E)$ (minimal radius of circumscribed balls),
- $\sum_i \chi_i^p(E)$, $p > 0$,

where $\chi_i(E)$ are the half-axes of E .

Indeed, the half-axes of E are the eigenvalues of the “parameter” $A = B^{-1}$ of the image representation of E , that is, $\chi_i(E) = \lambda_i^{-1}(B)$. Therefore

$$\begin{aligned} (a) \quad \text{Vol}(E) &= \lambda_1^{-1}(B) \dots \lambda_n^{-1}(B), \quad \text{Vol}^{1/n}(E) = \lambda_1^{-1/n}(B) \dots \lambda_n^{-1/n}(B) \\ (b) \quad \max_i \chi_i(E) &= \max_i \lambda_i^{-1}(B) \\ (c) \quad \sum_i \chi_i^p(E) &= \sum_i \lambda_i^{-p}(B) \end{aligned}$$

are convex symmetric functions of the eigenvalues of $B \succ 0$ and thus are convex functions of $B \succ 0$.

Note: From Calculus of SDr Functions/Sets it follows that the sizes (a), (b) are SDr functions of B ; the same is true for size (c) provided that $p > 0$ is rational.

♣ **Summary of observations:** *With the **inequality** representation of ellipsoids, typical problems of outer ellipsoidal approximation become problems of minimizing convex SDr functions over convex feasible sets.*

⇒ If the feasible set of a problem of outer ellipsoidal approximation is “computationally tractable” (in particular, is SDr), the problem itself is computationally tractable (in particular, is an SDP).

Note: *“If the feasible set ... is computationally tractable” is a big “IF” indeed!*

Tractability of Inner Ellipsoidal Approximation

♣ **Observation I.1:** Let $X \subset \mathbb{R}^n$ be a nonempty *convex* set. Then the set \mathcal{X} of parameters A, a of *image* representations of ellipsoids contained in X is convex.

To prove that \mathcal{X} is convex, let $\lambda \in [0, 1]$, $(A', a'), (A'', a'') \in \mathcal{X}$, so that $A \succeq 0$, $A' \succeq 0$ and

$$\forall(u : u^T u \leq 1) : \begin{cases} a' + A'u \in X \\ a'' + A''u \in X \end{cases} \quad (*)$$

we should prove that $\lambda(A', a') + (1 - \lambda)(A'', a'') \in \mathcal{X}$, that is,

$$\begin{aligned} \forall(u : u^T u \leq 1) : & [\lambda a' + (1 - \lambda)a''] + [\lambda A' + (1 - \lambda)A'']u \\ & \equiv \lambda[a' + A'u] + (1 - \lambda)[a'' + A''u] \in X. \end{aligned}$$

But this is an immediate corollary of $(*)$ and the convexity of X .

♣ **Observation 1.2:** “Typical sizes” of an ellipsoid E are *concave* (and thus easy-to-maximize) functions of the parameters A, a of the *image* representation of E . This is the case, e.g., for the sizes

- $\text{Vol}^{1/n}(E) = \prod_i \chi_i^{1/n}(E)$,
- $\min_i \chi_i(E)$ (“minimal width” of E , the radius of the largest Euclidean ball contained in E)
- $(\sum_i \chi_i^p(E))^{1/p}$, $0 < p \leq 1$,
where $\chi_i(E)$ are the half-axes of E .

Indeed, the half-axes of E are the eigenvalues of the “parameter” A of the image representation of $E \Rightarrow$

$$\begin{aligned} (a) \quad \text{Vol}^{1/n}(E) &= (\lambda_1(A) \dots \lambda_n(A))^{1/n} \\ (b) \quad \min_i \chi_i(E) &= \min_i \lambda_i(A) \\ (c) \quad (\sum_i \chi_i^p(E))^{1/p} &= (\sum_i \lambda_i^p(A))^{1/p} \end{aligned}$$

are concave symmetric functions of the eigenvalues of $A \succeq 0$ and thus are concave functions of $A \succeq 0$.

Note: From Calculus of SDr Functions/Sets it follows that *minus* sizes (a), (b), and *minus* size (c) with rational p — this is what we should minimize in order to maximize the actual sizes — are SDr functions of A .

♣ **Summary of observations:** *With the **image** representation of ellipsoids, typical problems of inner ellipsoidal approximation become problems of minimizing convex SDr functions over convex feasible sets.*

⇒ If the feasible set of a typical problem of inner ellipsoidal approximation is “computationally tractable” (in particular, is SDr), the problem itself is computationally tractable (in particular, is an SDP).

Note: “If the feasible set ... is computationally tractable” is a big “IF” indeed!

♣ We have seen that the typical problems of inner and outer ellipsoidal approximation are problems of minimizing explicit convex (usually even SDr) functions over convex feasible sets. As we shall see in the mean time, problems of this type are “computationally tractable” if the feasible sets are so.

♣ A sufficient condition for “computational tractability” of a convex set \mathcal{X} is the possibility to solve efficiently the *Analysis problem* “Given x , check whether $x \in \mathcal{X}$. ”

In our context, the Analysis problem is

- in *Outer ellipsoidal approximation of a set X* – problem
(AO) Given an ellipsoid E , check whether $E \supset X$.
- in *Inner ellipsoidal approximation of a set X* – problem
(AI) Given an ellipsoid E , check whether $E \subset X$.

Whether these analysis problems are/are not tractable, it depends on the structure of X .

(AO) *Given an ellipsoid E , check whether $E \supset X$.*

- (AO) is easy when X is a polytope given as a convex hull of a finite set $\{x^1, \dots, x^M\}$. Indeed, $\text{Conv}\{x^1, \dots, x^M\} \subset E$ iff $x^i \in E$ for all i , and it is easy to check whether or not a point belongs to E .
- (AO) can be NP-hard when X is a polytope given by a list of linear inequalities. Indeed, to check whether the unit cube $\{x : \|x\|_\infty \leq 1\}$ belongs to the centered at the origin ellipsoid $\{x : x^T Q x \leq r^2\}$, where $Q \succ 0$, is the same as to verify whether

$$\max_x \{x^T Q x : \|x\|_\infty \leq 1\} \leq r^2,$$

and the latter problem is, essentially, the NP-hard problem of maximizing positive definite homogeneous quadratic form over the unit cube.

(AI) *Given an ellipsoid E , check whether $E \subset X$.*

- (AI) is easy when X is a polytope P given by a list of linear inequalities $a_i^T x \leq b_i$, $1 \leq i \leq M$. Indeed, to check whether an ellipsoid E is contained in P is the same as to check whether $\max_{x \in E} a_i^T x \leq b_i$ for all i , and it is easy to maximize a linear form over an ellipsoid.
- (AI) can be NP-hard when X is a polytope given as $\text{Conv}\{x^1, \dots, x^M\}$.

Basic fact [Boyd et al.] *Let $E = \mathcal{E}(A, a)$ and $C = \mathcal{C}(B, b)$ be ellipsoid and elliptic cylinder given, respectively, by image and inequality representations. Then*

$$E \equiv \mathcal{E}(A, a) \subset C \equiv \mathcal{C}(B, b) \quad (*)$$

$$\Leftrightarrow \exists \lambda : \left[\begin{array}{c|c|c} 1 - \lambda & & a^T B - b^T \\ \hline & \lambda I & AB \\ \hline Ba - b & BA & I \end{array} \right] \succeq 0 \quad (**)$$

Note: For E fixed, $(**)$ is an LMI in variable λ and in the parameters B, b of C . For C fixed, $(**)$ is an LMI in variable λ and in the parameters A, a of E .

Thus, both the facts that

- an ellipsoid is contained in a fixed elliptic cylinder
 - an elliptic cylinder contains a fixed ellipsoid
- are semidefinite representable!

$$E \equiv \mathcal{E}(A, a) \subset C \equiv \mathcal{C}(B, b)?? \Leftrightarrow ?? \exists \lambda : \left[\begin{array}{c|c|c} 1 - \lambda & & a^T B - b^T \\ \hline & \lambda I & AB \\ \hline Ba - b & BA & I \end{array} \right] \succeq 0$$

Proof of equivalence:

$$\{Au + a : u^T u \leq 1\} \subset \{x : \|Bx - b\|_2^2 \leq 1\} \Leftrightarrow \forall (u : u^T u \leq 1) : \|BAu + \underbrace{Ba - b}_c\|_2^2 \leq 1$$

$$\underbrace{\Leftrightarrow}_{[u=t^{-1}v]} \forall (v, t : v^T v \leq t^2, t \neq 0) : \|t^{-1}BAv + c\|_2^2 \leq 1 \Leftrightarrow \forall (v, t : v^T v \leq t^2, t \neq 0) : \|BAv + tc\|_2^2 \leq t^2$$

$$\Leftrightarrow$$

$$\forall (v, t : t^2 - v^T v \geq 0) : t^2 - \|BAv + tc\|_2^2 \geq 0 \quad \underbrace{\Leftrightarrow}_{\text{S-Lemma}} \quad \exists \lambda \geq 0 : t^2 - \|BAv + tc\|_2^2 - \lambda [t^2 - v^T v] \geq 0 \quad \forall (v, t)$$

$$\Leftrightarrow \exists \lambda \geq 0 : \left[\begin{array}{c|c} 1 - \lambda & \\ \hline & \lambda I \end{array} \right] - \left[\begin{array}{c} c^T \\ AB \end{array} \right] \left[\begin{array}{c} c^T \\ AB \end{array} \right]^T \succeq 0$$

$\underbrace{\Leftrightarrow}_{\text{Schur Complement Lemma}}$

$$\exists \lambda \geq 0 : \left[\begin{array}{c|c|c} 1 - \lambda & & a^T B - b^T \\ \hline & \lambda I & AB \\ \hline Ba - b & BA & I \end{array} \right] \succeq 0 \Leftrightarrow \boxed{\exists \lambda : \left[\begin{array}{c|c|c} 1 - \lambda & & a^T B - b^T \\ \hline & \lambda I & AB \\ \hline Ba - b & BA & I \end{array} \right] \succeq 0}$$

♣ Conclusions, problem (O):

♠ Let X be a union of finitely many ellipsoids. The problem of finding the smallest ellipsoid E containing X can be posed as an explicit semidefinite program, provided that the size to be minimized is

- either the volume $\text{Vol}(E)$ (or the average linear size $\text{Vol}^{1/n}(E)$) of E ,
- or the maximal half-axis $\max_i \chi_i(E)$ of E ,
- or $\sum_i \chi_i^p(E)$ with rational $p > 0$.

“Good” design variables in the problem are the parameters B, b of the *inequality* representation of E .

In particular, the problem of finding the smallest ellipsoid containing a polytope *given as a convex hull of a finite set of points* can be posed as an explicit semidefinite program

♣ Conclusions, problem (I):

♠ Let X be an intersection of finitely many elliptical cylinders. The problem of finding the largest ellipsoid E contained in X can be posed as an explicit semidefinite program, provided that the size to be maximized is

- either average linear size $\text{Vol}^{1/n}(E)$ of E ,
- or the minimal half-axis $\min_i \chi_i(E)$ of E ,
- or $\left(\sum_i \chi_i^p(E)\right)^{1/p}$ with rational p , $0 < p \leq 1$.

“Good” design variables in the problem are the parameters A, a of the *image* representation of E .

In particular, the problem of finding the largest ellipsoid contained in a polytope *given by a finite list of linear inequalities* can be posed as explicit semidefinite program

♣ **Important Difficult Open problem:** Outer Ellipsoidal approximation of *intersection*

$$\hat{E} = \bigcap_{i=1}^m E_i$$

of ellipsoids (or elliptic cylinders).

♣ **Source of difficulty:** Given two ellipsoids, we understand how to check efficiently that one of them is contained in the other one, but we do *not* know how to check efficiently that a given ellipsoid contains the *intersection* of a collection of ellipsoids.

● The latter problem reduces to describing strongly convex quadratic inequalities

$$x^T A x + 2b^T x + c \leq 0 \quad [A \succ 0]$$

which are consequences of *systems*

$$x^T A_i x + 2b_i^T x + c_i \leq 0, \quad 1 \leq i \leq m \quad [A_i \succ 0 \forall i]$$

of strongly convex quadratic inequalities.

This problem is NP-hard, and the SDP Relaxation, based on replacing the set of *all* consequences with the set of *all linear* consequences, fails to work properly!

$$\hat{E} = \bigcap_{i=1}^m E_i, \quad E_i: \text{ ellipsoids}$$

♣ There are several interesting “ad hoc” approximations of the *smallest in volume* Outer Ellipsoidal approximation of \hat{E} . In all schemes, one builds efficiently two similar to each other concentric ellipsoids E, \overline{E} which “bracket” \hat{E} :

$$E \subset \hat{E} \subset \overline{E},$$

and guarantees certain bounds on the similarity ratio θ of the “brackets”.

- One scheme allows to ensure $\theta \leq n$ and is based on the following nice fact:

Fritz John Theorem: *For every convex compact set $X \subset \mathbb{R}^n$ with a nonempty interior, there exists a unique smallest volume ellipsoid E_{out} containing X , same as there exists a unique largest volume ellipsoid E_{in} contained in X .*

When shrinking E_{out} to its center with the coefficient n , one gets an ellipsoid which is contained in X , and when enlarging E_{in} by factor n (keeping the center fixed), one gets an ellipsoid which contains X .

When X has a symmetry center, the shrinkage/enlargement by factor n can be replaced with shrinkage/enlargement by factor \sqrt{n} .

We would like to build E_{out} , but we do not know how to do it efficiently. However, we *do* know how to build efficiently E_{in} . Building E_{in} and enlarging it by factor n , we, by Fritz John Theorem, get an ellipsoid containing \hat{E} , the ratio of the linear sizes of the resulting “brackets” being n .

- Another scheme allows to ensure $\theta \leq m + 2\sqrt{m}$ (non-optimality in volume by factor $\leq (m + 2\sqrt{m})^n$). Without essential loss of generality, we can assume that

$$E_i = \{x : \|B_i x - b_i\|_2^2 \leq 1\}$$

\hat{E} is bounded, and $\text{int } \hat{E} \neq \emptyset$. We form the *analytical barrier* for \hat{E} – the explicit convex function

$$F(x) = -\sum_i \ln(1 - \|B_i x - b_i\|_2^2)$$

with the domain $\text{int } \hat{E}$, solve the convex optimization problem

$$x_* = \underset{x \in \text{int } \hat{E}}{\text{argmin}} F(x)$$

(this can be done efficiently) and set

$$\begin{aligned} \underline{E} &= \{x : (x - x_*)^T \nabla^2 F(x_*) (x - x_*) \leq 1\}, \\ \overline{E} &= \{x : (x - x_*)^T \nabla^2 F(x_*) (x - x_*) \leq (m + 2\sqrt{m})^2\} \end{aligned}$$

♣ In Outer Ellipsoidal approximation of intersection of ellipsoids, SDP Relaxation “recovers its power” when all the ellipsoids in the intersection have a common center (w.l.o.g., 0). In fact, it works well when the set to be approximated is an ellitope:

$$\hat{E} = \{x \in \mathbf{R}^n : \exists(t \in \mathcal{T}, y) : x = Py, y^T S_i y \leq t_i, i \leq m\}$$

where $S_i \succeq 0$, $\sum_i S_i \succ 0$, and $\mathcal{T} \subset \mathbf{R}_+^m$ is a convex compact set intersecting $\text{int} \mathbf{R}_+^m$ and *monotone*: whenever $0 \leq t' \leq t$ with $t \in \mathcal{T}$, one has $t' \in \mathcal{T}$.

Note: When $P = I$ and $\mathcal{T} = [0, 1]^m$, $\hat{E} = \{x : x^T S_i x \leq 1, i \leq m\}$ is the intersection of ellipsoids/elliptic cylinders.

♠ Let \hat{E} be ellitope. Observe that the optimal circumscribed ellipsoid is centered at the origin.

Indeed, if

$$C_+ \equiv \{x : \|Bx - b\|_2^2 \leq 1\} \supset \hat{E},$$

then, due to symmetry of \hat{E} , we have

$$C_- \equiv \{x : \|Bx + b\|_2^2 \leq 1\} \supset \hat{E}$$

as well, whence, due to the convexity of the set $\{(P, p) : \mathcal{C}(P, p) \supset \hat{E}\}$, we have

$$C \equiv \{x : \|Bx\|_2^2 \leq 1\} \supset \hat{E},$$

and C has the same size as C_+ and C_- .

$$\hat{E} = \{x \in \mathbf{R}^n : \exists(t \in \mathcal{T}, y) : x = Py, y^T S_i y \leq t_i, i \leq m\} \quad [S_i \succeq 0, \sum_i S_i \succ 0]$$

Fact: the minimum size circumscribed ellipsoid is centred at the origin

\Rightarrow the Outer Ellipsoidal approximation problem for \hat{E} is to minimize a desired size $\text{Size}(E_B)$ of ellipsoid $E_B = \{x : x^T B^2 x \leq 1\}$ over $B \succeq 0$ satisfying the constraint $x^T B^2 x \leq 1 \forall x \in \hat{E}$, that is, the constraint

$$1 \geq \text{Opt}(B) := \max_{x \in \hat{E}} x^T B^2 x = \max_{y, t} \left\{ y^T P^T B^2 P y : y^T S_i y \leq t_i, i \leq m, t \in \mathcal{T} \right\}$$

♣ For the sake of definiteness, we restrict ourselves with the sizes $\text{Size}(E_B)$ as follows:

- $\text{Vol}^{1/n}(E_B)$ – average linear size of E_B
- maximum $\max_i \chi_i(E_B)$ of the half-axes $\chi_i(E_B)$ of E_B
- $\sum_i \chi_i^p(E_B)$ with $p > 0$

♣ By what we know on Semidefinite Relaxation on ellitopes,

$$\begin{aligned} \text{Opt}(B) \leq \text{Opt}_+(B) &:= \min \left\{ \phi_{\mathcal{T}}(\lambda) : P^T B^2 P \preceq \sum_i \lambda_i S_i, \lambda \geq 0 \right\} \\ &\left[\phi_{\mathcal{T}}(\lambda) = \max_{t \in \mathcal{T}} \lambda^T t : \text{ support function of } \mathcal{T} \right] \end{aligned}$$

and $\text{Opt}_+(B) \leq 3 \ln(\sqrt{3}m) \text{Opt}(B)$.

Conclusion: The difficult problem

$$\text{SizeOpt} = \min_B \left\{ \text{Size}(E_B) : B \succeq 0, \mathbf{1} \geq \text{Opt}(B) := \max_{y,t} \{y^T P^T B^2 P : y^T S_i y \leq t_i, i \leq m\} \right\}$$

of Outer ellipsoidal approximation of ellitope \hat{E} can be approximated by the efficiently solvable problem

$$\begin{aligned} \text{Size}_+ &= \min_B \left\{ \text{Size}(E_B) : B \succeq 0, \mathbf{1} \geq \text{Opt}_+(B) := \min_{\lambda} \left\{ \phi_{\mathcal{T}}(\lambda) : \lambda \geq 0, P^T B^2 P \preceq \sum_i \lambda_i S_i \right\} \right\} \\ \Leftrightarrow \text{Size}_+ &= \min_{B,\lambda} \left\{ \text{Size}(E_B) : B \succeq 0, P^T B^2 P \preceq \sum_i \lambda_i S_i, \lambda \geq 0, \phi_{\mathcal{T}}(\lambda) \leq 1 \right\} \\ \Leftrightarrow \text{Size}_+ &= \min_{B,\lambda} \left\{ \text{Size}(E_B) : B \succeq 0, \left[\frac{\sum_i \lambda_i S_i}{BP} \middle| \frac{P^T B}{I} \right] \succeq 0, \lambda \geq 0, \phi_{\mathcal{T}}(\lambda) \leq 1 \right\} \quad (*) \end{aligned}$$

• The approximation is safe: whenever B, λ is feasible for $(*)$, the ellipsoid $E_B = \{x : x^T B^2 x \leq 1\}$ contains $\hat{E} \Rightarrow$ the B -component B_* of optimal solution to $(*)$ yields ellipsoid $E_{B_*} \supset \hat{E}$ of size Size_+

• The approximation is reasonably tight: setting $\vartheta = \sqrt{3 \ln(\sqrt{3}m)}$, it holds $\text{Size}_+ \leq \vartheta^{\kappa} \text{SizeOpt}$, where

- $\kappa = 1$ when $\text{Size}(E_B)$ is the maximum of half-axes $\chi_i(E_B)$ of the ellipsoid E_B or its average linear size $\text{Vol}^{1/n}(E_B)$,
- $\kappa = p$ when $\text{Size}(E_B)$ is $\sum_i \chi_i^p(E_B)$, $p > 0$.

Note: When $\phi_{\mathcal{T}}$ and $\text{Size}(\cdot)$ are SDr, $(*)$ reduces to SDP.

Ellipsoidal Approximation and Polarity

Preliminaries on Polarity. Let $X \subset \mathbf{R}^n$ be a closed convex set containing the origin. The *polar* of X is the set

$$X_* = \{y \in \mathbf{R}^n : y^T x \leq 1 \forall x \in X\}.$$

Fact 0: *The polar X_* of a closed convex set X containing the origin is a closed convex set containing the origin, and twice taken, the polar recover the original set:*

$$(X_*)_* = X.$$

- *Taking polars reverses inclusion: if X and Y are two closed convex sets containing the origin, then $X \subset Y$ iff $Y_* \subset X_*$.*
- *Taking polar preserves symmetry w.r.t. the origin.*

Fact I: *The polar X_* of a closed convex set X containing the origin is bounded if and only if $0 \in \text{int } X$.*

Fact II: [finite-dimensional Hahn-Banach Theorem] *Let $X \subset \mathbb{R}^n$ be a closed convex set with $0 \in \text{int } X$ and L be a linear subspace in \mathbb{R}^n . Then the polar of $X \cap L$ taken with respect to L – the set*

$$X_{*,L} = \{y \in L : y^T x \leq 1 \forall x \in X \cap L\}$$

is the orthogonal projection of the polar X_ of X onto L :*

$$X_{*,L} = \{y \in L : \exists z \in L^\perp : y + z \in X_*\}$$

More traditional formulation: *A linear form **on** L does not exceed 1 on $X \cap L$ iff it can be extended from L onto \mathbb{R}^n to a linear form which does not exceed 1 on X .*

Examples of polars

- Polar of polytope $X = \{x \in \mathbb{R}^n : a_i^T x \leq 1, i \leq m\}$ is the polytope $X_* = \text{Conv}\{a_1, \dots, a_m\}$.

Indeed, by LP Duality

$$\max_x \{y^T x : -1 \leq a_i^T x \leq 1, i \leq m\} = \min_{\lambda, \mu} \{\sum_i \lambda_i + \sum_i \mu_i : \sum_i \lambda_i a_i - \sum_i \mu_i a_i = y, \lambda \geq 0, \mu \geq 0\}$$

- Polar of full-dimensional ellipsoid $X = \{x : x^T A x \leq 1\}$, $A \succeq 0$, centered at the origin, is the ellipsoid $X_* = \{y : y^T A^{-1} y \leq 1\}$

Indeed,

$$\begin{aligned} \max_x \{y^T x : x^T A x \leq 1\} &= \max_{u=A^{1/2}x} \{y^T A^{-1/2} u : u^T u \leq 1\} = \|A^{-1/2} y\|_2 \\ \Rightarrow \{y : \max_x \{y^T x : x^T A x \leq 1\} \leq 1\} &= \{y : \|A^{-1/2} y\|_2 \leq 1\} = \{y : y^T A^{-1} y \leq 1\} \end{aligned}$$

- Polar of "flat" centered at the origin ellipsoid $X = \{[u; v] \in \mathbb{R}^p \times \mathbb{R}^q : u^T A u \leq 1, v = 0\}$, where A is positive definite $p \times p$ matrix, is the elliptic cylinder $X_* = \{[w; z] \in \mathbb{R}^p \times \mathbb{R}^q : w^T A^{-1} w \leq 1\}$

- Polar of ellitope $X = \{x : \exists(t \in \mathcal{T}, y) : x = P y, y^T S_i^{1/2} y \leq t_i, i \leq m\}$ is

$$X_* = \left\{ \xi : \exists(p, s, r, \{z_i, i \leq m\}) : \begin{aligned} &\sum_i S_i^{1/2} z_i = P^T \xi, \|z_i\|_2 \leq s_i, i \leq m \\ &s_i^2 \leq p_i r_i, i \leq m, p \geq 0, r \geq 0 \\ &\sum_i r_i + \phi_{\mathcal{T}}(p) \leq 1 \end{aligned} \right\}$$

♠ **Situation:** We want to approximate by ellipsoids *symmetric w.r.t. the origin convex body* $X \subset \mathbf{R}^n$ ("convex body" meaning a closed and bounded convex set with a nonempty interior).

Note: By the above facts, the polar of centered at the origin convex body itself is centered at the origin convex body.

Note: *When X is centered at the origin convex body, for all standard sizes of ellipsoids, when finding the largest (the smallest) ellipsoid contained in (resp., containing) $X \cap L$, we lose nothing when restricting ourselves with ellipsoids centered at the origin.*

Note: When $A \succ 0$, volumes of the ellipsoids $\mathcal{A} = \{x : x^T A x \leq 1\}$ and $\mathcal{A}_* = \{y : y^T A^{-1} y \leq 1\}$, same as half-axes of these ellipsoids, are reciprocals of each other.

Conclusion: When X is a symmetric w.r.t. the origin convex body, basic problems (O)/(I) associated with X are equivalent to “swapped” problems (I)/(O) associated with the polar X_* of X , namely

- finding the largest radius Euclidean ball contained in X is equivalent to finding the smallest radius Euclidean ball containing X_* ;
- finding the largest volume ellipsoid contained in X is equivalent to finding the smallest volume ellipsoid containing X_* .

Example A: As we know, when X is a symmetric w.r.t. the origin convex body given as $X = \{x : -1 \leq a_i^T x \leq 1, i \leq m\}$, the problem of finding the *largest* radius ball (or the *largest* volume ellipsoid) *contained in* X is efficiently solvable via Semidefinite Programming

\Leftrightarrow When Y is a symmetric w.r.t. the origin convex body given as $X = \text{Conv}\{\pm a_1, \pm a_2, \dots, \pm a_m\}$, the problem of finding the *smallest* radius ball (or the *smallest* volume ellipsoid) *containing* X is efficiently solvable via Semidefinite Programming

Example B: As we know, when X is symmetric w.r.t. the origin convex body given as $X = \text{Conv}\{\pm a_1, \pm a_2, \dots, \pm a_m\}$, the problem of finding the *largest* radius ball (or the *largest* volume ellipsoid) *contained in* X can be difficult.

\Leftrightarrow When X is symmetric w.r.t. the origin convex body given as $X = \{x : -1 \leq a_i^T x \leq 1, i \leq m\}$, the problem of finding the *smallest* radius ball (or the *smallest* volume ellipsoid) *containing* X can be difficult.

♠ Assume that $X = \{x : -1 \leq a_i^T x \leq 1, i \leq m\} \subset \mathbf{R}^n$ is bounded. The problem of Inner ellipsoidal approximation of X is easy, but the problem of Outer ellipsoidal approximation can be difficult.

But: X is an ellitope: $X = \{x : x^T [a_i a_i^T] x \leq 1, i \leq m\}$, so that *semidefinite relaxation* provides a *tight efficiently verifiable sufficient condition* for the ellipsoid $\{x^T Q x \leq 1\}$ with $Q \succ 0$ to contain X , namely, the condition

$$\exists \lambda \geq 0 : Q \preceq \sum_i \lambda_i a_i a_i^T \text{ \& } \sum_i \lambda_i \leq 1.$$

Specifically, when the condition is satisfied, the ellipsoid $\{x : x^T Q x \leq 1\}$ does contain X , and when it is violated, the shrinkage $\{x : x^T Q x \leq \theta^{-2}\}$ of the ellipsoid with “moderate” θ : $\theta = 3 \ln(\sqrt{3}m)$ does *not* contain X

\Rightarrow An explicit efficiently solvable semidefinite program

$$\text{Opt} = \max_{\rho, \lambda} \left\{ \rho : \rho I_n \preceq \sum_i \lambda_i a_i a_i^T : \lambda \geq 0, \sum_i \lambda_i \leq 1 \right\} \quad (*)$$

is a safe tractable approximation of the problem of finding the minimum of radii of balls containing X : whenever ρ, λ is a feasible solution to $(*)$, the centered at the origin ball of radius $1/\sqrt{\rho}$ contains X . This approximation is tight: $1/\sqrt{\text{Opt}}$ is within the factor $\sqrt{3 \ln(\sqrt{3}m)}$ of the minimum of radii of balls containing X .

Recall that similar result holds true for every ellitope X .

Note: “Safe efficiently solvable tight approximation” can be built for the problem of finding the smallest volume ellipsoid containing X . This approximation reads

$$\text{Opt} = \max_{\rho, Q, \lambda} \left\{ \rho : \rho \leq [\text{Det}(Q)]^{1/n}, 0 \preceq Q \preceq \sum_i \lambda_i a_i a_i^T : \lambda \geq 0, \sum_i \lambda_i \leq 1 \right\}.$$

The ellipsoid $\{x : x^T Q x \leq 1\}$ yielded by the Q -component of a feasible solution to the problem contains X , its volume is $\leq \rho^{-n/2}$, and $\text{Opt}^{-n/2}$ is within factor $[3 \ln(\sqrt{3}m)]^{n/2}$ of the smallest of volumes of ellipsoids containing X .

By the equivalence we have established, *when $X = \text{Conv}\{\pm a_1, \dots, \pm a_m\} \subset \mathbb{R}^n$ has a nonempty interior, the problem of finding the largest radius ball (or the largest volume ellipsoid) contained in X admits safe efficiently solvable semidefinite approximation with the same as above tightness guarantees.*

♠ Along with the problems of Outer/Inner ellipsoidal approximation of centered at the origin convex body X , an important problem is approximating the *cross-sections of X with linear subspace L of dimension $p < n$* by “flat” (p -dimensional) ellipsoid.

The complexity of these problems depends on how X is given.

♠ **When** $X = \{x \in \mathbf{R}^n : -1 \leq a_i^T x \leq 1, i \leq m\}$, $X \cap L$ admits similar representation, and we can apply the techniques we already have; in this case,
— problem of finding the largest radius ball (or the largest p -dimensional volume ellipsoid) contained in $X \cap L$ is efficiently solvable;
— problem of finding the smallest radius ball (or the smallest p -dimensional volume) containing $X \cap L$ admits safe and tight semidefinite approximation.

♠ **When** $X = \text{Conv}\{\pm a_1, \dots, \pm a_m\}$,

— the problem of finding the smallest radius ball (or the smallest p -dimensional volume ellipsoid) containing $X \cap L$ is *not* known to have tight safe tractable approximation.

What *does* have such an approximation, is the problem of finding the smallest radius ball (or the smallest p -dimensional volume ellipsoid) containing the *orthogonal projection* of X onto L (which for $L = \mathbf{R}^n$ is the same as finding largest ball/ellipsoid contained in X)

— the problem of finding the largest radius ball (or the largest p -dimensional volume ellipsoid) contained in $X \cap L$ still admits tight safe tractable approximation.

Indeed, assuming w.l.o.g. that L is the linear span of the first p basic orths, centered at the origin ellipsoid $E = \{u \in L : u^T Q u \leq 1\}$ (Q is positive definite $p \times p$ matrix) is contained in $X \cap L$ if and only if

$$E_* = \{[w; z] \in \mathbf{R}^p \times \mathbf{R}^{n-p} : w^T Q^{-1} w \leq 1\}$$

contains $X_* = \{y : -1 \leq a_i^T y \leq 1, i \leq m\}$, which in turn happens if and only if the quadratic form

$$[w; z]^T \left[\begin{array}{c|c} Q^{-1} & \\ \hline & \end{array} \right] [w; z]$$

does not exceed 1 everywhere on $X_* = \{y : -1 \leq a_i^T y \leq 1, i \leq m\}$. The latter restriction on Q , same as in the case of $L = \mathbf{R}^n$, can be safely and tightly approximated via Semidefinite Relaxation.

Inner and Outer Ellipsoidal Approximations of Sums of Ellipsoids

Problems of interest: *Given m full-dimensional ellipsoids W_1, \dots, W_m in \mathbf{R}^n , find the best in the volume inner (problem (I)) and outer (problem (O)) ellipsoidal approximations of the arithmetic sum*

$$W = \{x = w_1 + w_2 + \dots + w_m : w_i \in W_i, i = 1, \dots, m\}$$

of the ellipsoids W_1, \dots, W_m .

♠ **Note:** When shifting one of the sets A, B, \dots, Z by a vector a , the arithmetic sum $A + B + \dots + Z$ of the sets is shifted by the same vector a .
 \Rightarrow *We may assume w.l.o.g. that all the ellipsoids W_i are centered at the origin:*

$$W_i = \{x \in \mathbf{R}^n : x^T Z_i x \leq 1\} \quad [Z_i \succ 0].$$

In this case the solutions to (I) and (O) also can be sought among the ellipsoids centered at the origin.

Outer Ellipsoidal Approximation of Sum of Ellipsoids

Observation: *Ellipsoid*

$$E = \{x : x^T Z x \leq 1\} \quad [Z \succ 0]$$

contains the arithmetic sum of ellipsoids

$$W_i = \{x : x^T Z_i x \leq 1\}, \quad i = 1, \dots, m$$

iff

$$\max_{u=[u^1; \dots; u^m]} \left\{ \underbrace{(u^1 + \dots + u^m)^T Z (u^1 + \dots + u^m)}_{u^T \mathcal{M}[Z] u} : \underbrace{(u^i)^T Z_i u^i}_{u^T \mathcal{M}_i u} \leq 1, \quad i = 1, \dots, m \right\} \leq 1$$

$$\left[\mathcal{M}[Z] = \begin{bmatrix} Z & Z & \dots \\ Z & \ddots & \dots \\ \vdots & \vdots & Z \end{bmatrix}, \mathcal{M}_1 = \begin{bmatrix} Z_1 & & \\ & & \\ & & \end{bmatrix}, \dots, \mathcal{M}_m = \begin{bmatrix} & & \\ & & \\ & & Z_m \end{bmatrix} \right]$$

♠ Applying Semidefinite Relaxation, we arrive at the following *conservative approximation* of (O):

$$\min_{Z, \mu} \left\{ \text{Det}^{-1/2}(Z) : \begin{array}{l} Z \succ 0, \mu \geq 0, \sum_i \mu_i \leq 1 \\ \mathcal{M}[Z] \preceq \sum_i \mu_i \mathcal{M}_i \end{array} \right\} \quad (*)$$

♠ Matrices \mathcal{M}_i are positive semidefinite and commute with each other. Applying (corollary of) Nesterov's $\frac{\pi}{2}$ Theorem, it is easily seen that the optimal solution to (*) yields an optimal, up to factor $(\frac{\pi}{2})^{n/2}$, solution to (O).

Inner Ellipsoidal Approximation of Sum of Ellipsoids

♣ **Observation:** *An ellipsoid*

$$E = \{x = Au : u^T u \leq 1\}$$

is contained in the sum of ellipsoids

$$W_i = \{x = A_i u : u^T u \leq 1\}, i = 1, \dots, m$$

iff for every vector ξ one has

$$\|A^T \xi\|_2 \leq \sum_i \|A_i^T \xi\|_2. \quad (*)$$

Proof. Let P, Q be closed nonempty convex sets. From Separation Theorem it immediately follows that

$$P \subset Q \Leftrightarrow \max_{x \in Q} \xi^T x \geq \max_{x \in P} \xi^T x \quad \forall \xi.$$

With $P = E$, we have

$$\max_{x \in P} \xi^T x = \max_u \{\xi^T Au : u^T u \leq 1\} = \|A^T \xi\|_2.$$

With $Q = W_1 + \dots + W_m$, we have

$$\max_{x \in Q} \xi^T x = \max_{u^1, \dots, u^m} \{\xi^T [A_1 u^1 + \dots + A_m u^m] : \|u^i\|_2 \leq 1 \quad \forall i\} = \sum_i \|A_i^T \xi\|_2.$$

Thus, $E \subset W_1 + \dots + W_m$ if and only if $(*)$ takes place.

$$\|A^T \xi\|_2 \leq \sum_i \|A_i^T \xi\|_2. \quad (*)$$

Observation I: *Given matrices A_i , the simplest way to generate matrix A satisfying $(*)$ is to set*

$$A = \sum_i A_i X_i, \quad \|X_i\| \leq 1 \quad (**)$$

Observation II: *Let $A = S + C$ with symmetric positive definite S and skew-symmetric C . Then*

$$\text{Det}(A) = |\text{Det}(A)| \geq \text{Det}(S)$$

Indeed, by “scaling”

$$A = S + C \mapsto \hat{A} = S^{-1/2} A S^{-1/2} = I + \underbrace{S^{-1/2} C S^{-1/2}}_{\hat{C}}$$

we reduce the general case to the one where $S = I$. Here the statement is evident: since the eigenvalues of skew-symmetric real matrix C are pairs of conjugate purely imaginary complex numbers $\pm i\nu_\ell$, we have

$$\begin{aligned} \text{Det}(A) &= \text{Det}(I + C) = \prod_\ell [(1 - i\nu_\ell)(1 + i\nu_\ell)] \\ &= \prod_\ell [1 + \nu_\ell^2] \geq 1 = \text{Det}(I). \end{aligned}$$

♣ We arrive at the following conservative approximation of (I):

$$\max_{\{X_i\}} \left\{ \text{Det}^{1/n} \left(\frac{1}{2} \sum_i [X_i^T A_i + A_i X_i] \right) : \overbrace{\left[\begin{array}{c|c} I & X_i \\ \hline X_i & I \end{array} \right]}^S \succeq 0 \forall i \atop \equiv \|X_i\| \leq 1} \right\} \quad (P)$$

where $A_i \succeq 0$ are the matrices from the image representations of the ellipsoids W_i .

Every feasible solution $\{X_i\}$ of (P) produces ellipsoid

$$E = \{x = Au : u^T u \leq 1\}, \quad A = \sum_i A_i X_i$$

which is contained in $W_1 + \dots + W_m$, and the volume of this ellipsoid is at least

$$\text{Det} \left(\frac{1}{2} \sum_i [X_i^T A_i + A_i X_i] \right).$$

Problems (O) and (I) in the Co-Axial Case

♣ **Observation:** Problems (O) and (I) (same as all problems of “optimal in volume” ellipsoidal approximation) admit certain symmetry. Specifically, let

$$y = Qx$$

be a nondegenerate linear transformation of \mathbf{R}^n . Such a transformation multiplies the volumes of all sets by the same factor $|\text{Det}(Q)|$; consequently, problems (I)/(O) involving ellipsoids

$$W_i = \{x : x^T Z_i x \leq 1\} \quad [Z_i \succ 0]$$

can be reduced to similar problems involving the images

$$\widehat{W}_i = \{y : (Q^{-1}y)^T Z_i (Q^{-1}y) \leq 1\} = \{y : y^T \underbrace{[Q^{-T} Z_i Q^{-1}]}_{\widehat{Z}_i} y \leq 1\}$$

of ellipsoids W_i under this transformation.

♠ Let us call ellipsoids W_i *co-axial*, if, with a proper choice of Q , the matrices \hat{Z}_i commute with each other.

◇ Co-axiality is equivalent to the existence of a basis (not necessarily orthogonal) where all quadratic forms $x^T Z_i x$ become diagonal:

$$x^T Z_i x = \sum_j \nu_j^i \xi_j^2(x)$$

$[\xi_j(x) : \text{coordinates of } x \text{ in the basis}]$

◇ Linear Algebra says that every two (full-dimensional) ellipsoids W_1, W_2 are co-axial.

Indeed, if $W_i = \{x : x^T Z_i x \leq 0\}$ and $Z_i \succ 0$, $i = 1, 2$, then, setting $Q = Z_1^{-1/2}$, we arrive at commuting matrices

$$\hat{Z}_1 = Z_1^{-1/2} Z_1 Z_1^{-1/2} = I, \quad \hat{Z}_2 = Z_1^{-1/2} Z_2 Z_1^{-1/2}.$$

♠ We have seen that in the co-axial case problems (I) and (O) can be reduced to similar problems for the sum of ellipsoids given by *diagonal* matrices:

$$W_i = \{x : \sum_j \nu_j^i x_j^2 \leq 1\} \quad [\nu_j^i > 0]$$

It turns out that *in the case of ellipsoids W_i given by diagonal matrices, the tractable approximations of (O) and (I) we have presented yield exactly optimal solutions to the respective problems.*

This is a corollary of simple and powerful *Symmetry Principle*.

Symmetry Principle: Consider a *convex* and solvable optimization problem

$$\min_{x \in X} f(x) \quad (P)$$

and assume that it admits a finite group G of symmetries, that is,

- G is a finite subset of the group \mathcal{L}_n of nonsingular $n \times n$ matrices,
- G is a sub-group of \mathcal{L}_n : $U \in G \Rightarrow U^{-1} \in G$, $U, V \in G \Rightarrow UV \in G$ and
- every $U \in G$ is a symmetry of (P):

$$U(X) := \{Ux : x \in X\} = X, \quad f(Ux) = f(x) \quad \forall x \in X.$$

Then (P) admits a “ G -symmetric” optimal solution x_* :

$$Ux_* = x_* \quad \forall U \in G.$$

Proof. Let \bar{x} be an optimal solution to (P). Since (P) is G -symmetric, every point of the form

$$U\bar{x}, \quad U \in G$$

is an optimal solution to (P) along with \bar{x} . Since (P) is convex, it follows that the point

$$x_* = \frac{1}{\text{Card}(G)} \sum_{U \in G} U\bar{x} \quad (*)$$

also is an optimal solution to (P); this solution is clearly G -symmetric.

Remark: Assuming X closed, the statement remains valid when G is a compact, rather than finite, group of symmetries of (P) . The proof remains essentially the same, with averaging $(*)$ replaced by integration over the invariant probabilistic measure on G .

From Symmetry Principle to Co-Axial (O)/(I). Let ellipsoids W_i be given by diagonal matrices:

$$W_i = \{x : \sum_j \nu_j^i x_j^2 \leq 1\} \quad [\nu_j^i > 0]$$

Consider problem (O):

$$\min_{B,b} \left\{ \text{Det}^{-1}(B) \equiv \prod_j \lambda_j^{-1}(B) : \mathcal{C}(B,b) \supset \underbrace{W_1 + \dots + W_m}_W, B \succ 0 \right\} \quad (\text{O})$$

The problem is convex and solvable (the latter – by Fritz John Theorem). Let \mathcal{J} be a transformation of \mathbf{R}^n of the form

$$x \mapsto (\epsilon_1 x_1, \epsilon_2 x_2, \dots, \epsilon_n x_n), \quad \epsilon_j = \pm 1.$$

Since W_i are given by diagonal matrices, this transformation keeps W invariant and therefore maps an ellipsoid $\mathcal{C}(B,b)$ containing W into another ellipsoid also containing W ; this “other ellipsoid” is $\mathcal{C}(JBJ, Jb)$. Thus, *the feasible set of convex and solvable problem (O) is invariant under the transformations*

$$\mathcal{J} : (B,b) \mapsto (JBJ, Jb) \equiv (J^T B J, Jb)$$

generated by 2^n “reflections” J . The transformations \mathcal{J} clearly form a finite sub-group of the group of orthogonal rotations of the Euclidean space $\mathbf{S}^n \times \mathbf{R}^n$ where the feasible set of (O) lives, and that these transformations preserve the objective in (O). Applying Symmetry Principle, we conclude that (O) admits an optimal solution (B_*, b_*) which remains invariant under all transformations of the form

$$(B,b) \mapsto (J^T B J, Jb), \quad J = \text{Diag}\{\epsilon_1, \dots, \epsilon_n\}, \quad \epsilon_i = \pm 1,$$

which clearly is possible iff $b_* = 0$ and B_* is diagonal.

$$W_i = \{x : \sum_j \nu_j^i x_j^2 \leq 1\} \quad [\nu_j^i > 0]$$

$$\min_{B,b} \left\{ \frac{1}{\text{Det}(B)} \equiv \prod_j \lambda_j^{-1}(B) : \mathcal{C}(B,b) \supset W_1 + \dots + W_m, B \succ 0 \right\} \quad (\text{O})$$

We have seen that when solving (O), we lose nothing by assuming that $b = 0$ and B is diagonal, so that (O) is equivalent to the problem

$$\min_{\beta} \left\{ \prod_j \beta_j^{-1} : \beta > 0, \sum_j \beta_j x_j^2 \leq 1 \quad \forall (x = x^1 + \dots + x^m : \sum_j \nu_j^i \underbrace{(x_j^i)^2}_{y_j^i}) \right\} \quad (\text{O}')$$

$$\Leftrightarrow \min_{\beta > 0} \left\{ \prod_j \beta_j^{-1} : \sum_j \beta_j \left(\sum_j \sqrt{y_j^i} \right)^2 \leq 1 \quad \forall \left(\{y_j^i \geq 0\} : \begin{array}{l} \sum_j \nu_j^i y_j^i \leq 1, \\ 1 \leq i \leq m \end{array} \right) \right\}$$

We claim that

(!) *A vector $\beta > 0$ is feasible for (O') if and only if there exists $\mu \geq 0$ such that $\mathcal{M}[\text{Diag}\{\beta\}] \preceq \sum_i \mu_i \mathcal{M}_i$.*

(!) says that *the matrices $\text{Diag}\{\beta\}$ associated with feasible solutions to (O') are feasible solutions to the tractable approximation of (O) we have built.*

\Rightarrow *Optimal solution to our approximation of (O) is optimal solution of (O) as well.*

$$\min_{\beta > 0} \left\{ \prod_j \beta_j^{-1} : \sum_j \beta_j \left(\sum_j \sqrt{y_j^i} \right)^2 \leq 1 \quad \forall (\{y_j^i \geq 0\} : \sum_j \nu_j^i y_j^i \leq 1, 1 \leq i \leq m) \right\} \quad (\mathbf{O}')$$

Claim: (!) A vector $\beta > 0$ is feasible for (\mathbf{O}') if and only if there exists $\mu \geq 0$ such that $\sum_i \mu_i \leq 1$ and

$$\mathcal{M}[\text{Diag}\{\beta\}] \preceq \sum_i \mu_i \mathcal{M}_i. \quad \left[\mathcal{M}_i = \begin{bmatrix} & & & & \\ & & & & \\ & & & & \\ & & \text{Diag}\{\nu^i\} & & \\ & & & & \end{bmatrix} \right]$$

Proof of (!): The only nontrivial part of (!) is the claim that (!!) *if $\beta > 0$ is feasible for (\mathbf{O}') , then there exists $\mu \geq 0$ such that...*

By Semidefinite Duality, the property “exists $\mu \geq 0$ such that...” is exactly equivalent to the validity of the implication

$$Y \in \mathbf{S}_+^{mn}, \text{Tr}(\mathcal{M}_i Y) \leq 1, 1 \leq i \leq m \Rightarrow \text{Tr}(\mathcal{M}[\text{Diag}\{\beta\}]Y) \leq 1 \quad (1)$$

so that to prove (!!) is the same as to prove that

(!!!) *If β is feasible for (\mathbf{O}') , then (1) takes place.*

To prove (!!!), let β be feasible for (\mathbf{O}') , and let Y satisfy the premise in (1). Let us split Y into m^2 blocks Y^{ik} of the size $n \times n$ each.

Situation: β is feasible for

$$\min_{\beta > 0} \left\{ \prod_j \beta_j^{-1} : \sum_j \beta_j \left(\sum_j \sqrt{y_j^i} \right)^2 \leq 1 \quad \forall (\{y_j^i \geq 0\} : \sum_j \nu_j^i y_j^i \leq 1, 1 \leq i \leq m) \right\} \quad (\text{O}')$$

$Y = [Y^{k\ell} \in \mathbf{R}^{n \times n}]_{k,\ell \leq m}$ satisfies the premise in

$$Y \in \mathbf{S}_+^{mn}, \text{Tr}(\mathcal{M}_i Y) \leq 1, 1 \leq i \leq m \Rightarrow \text{Tr}(\mathcal{M}[\text{Diag}\{\beta\}]Y) \leq 1 \quad (1)$$

Goal: to justify the validity of the conclusion in (1).

Taking into account that $Y \succeq 0$, we have $|Y_{jj}^{ik}| \leq \sqrt{Y_{jj}^{ii} Y_{jj}^{kk}}$, whence

$$\text{Tr}(\mathcal{M}[\text{Diag}\{\beta\}]Y) = \sum_{i,k=1}^m \sum_{j=1}^n \beta_j Y_{jj}^{ik} \leq \sum_{i,k=1}^m \sum_{j=1}^n \beta_j \sqrt{Y_{jj}^{ii} Y_{jj}^{kk}} = \sum_{j=1}^n \beta_j \left(\sum_{i=1}^m \sqrt{Y_{jj}^{ii}} \right)^2$$

Since Y satisfies the premise in (1), we have

$$\text{Tr}(\mathcal{M}_i Y) \equiv \sum_j \nu_j^i Y_{jj}^{ii} \leq 1,$$

whence, since β is feasible for (O'),

$$\text{Tr}(\mathcal{M}[\text{Diag}\{\beta\}]Y) = \sum_{j=1}^n \beta_j \left(\sum_{i=1}^m \sqrt{Y_{jj}^{ii}} \right)^2 \leq 1,$$

as required in the conclusion of (1). □

♣ Let ellipsoids W_i be given by diagonal matrices:

$$W_i = \{x : \sum_j \nu_j^i x_j^2 \leq 1\} \quad [\nu_j^i > 0]$$

$$\Rightarrow W_i = \{x = \underbrace{\text{Diag}\{\theta_j^i\}}_{A_i} u : u^T u \leq 1\} \quad [\theta_j^i = (\nu_j^i)^{-1/2}]$$

Problem (I). *In the case of diagonal matrices $A_i \succeq 0$, our approximation scheme recovers **exactly optimal** ellipsoid contained in $W_1 + \dots + W_m$. Moreover, this ellipsoid is just*

$$W = \{x = \underbrace{[A_1 + \dots + A_m]}_A u : u^T u = 1\}. \quad (!)$$

Indeed, ellipsoid (!) is given by our approximation scheme:

$$A = \frac{1}{2} \sum_i [X_i^T A_i + A_i X_i] \succeq 0 \quad [X_i = I, \|X_i\| \leq 1]$$

thus, the ellipsoid is contained in $W_1 + \dots + W_m$.

On the other hand, it is clear that the set $W_1 + \dots + W_m$ is contained in the box

$$\{x : |x_j| \leq \theta_j^1 + \theta_j^2 + \dots + \theta_j^m, j = 1, \dots, n\},$$

so that the largest volume ellipsoid contained in this box (which is exactly W !) can be only larger than the largest volume ellipsoid contained in $W_1 + \dots + W_m$.

♣ **Application: On-line approximation of reachable sets.**

$$z(t+1) = A_t z(t) + B_t u(t) + f_t, \quad z(0) = z_0 \quad (1)$$

♣ The set Z^T of all states $z(T)$ of (1) reachable with norm-bounded control:

$$\|u(t)\|_2 \leq \rho_t, \quad t = 0, 1, \dots, T-1$$

is the sum of T ellipsoids and thus can be approximated from inside and from outside by ellipsoids via our techniques. We can further “trade quality for simplicity” and look at *on-line approximations*, where, given ellipsoidal approximations of Z^t :

$$E_t \subset Z^t \subset E^t$$

and observing that

$$Z^{t+1} = A_t Z^t + \{B_t u + f_t : u^T u \leq \rho_t^2\},$$

we conclude that

$$A_t E_t + \{B_t u + f_t : u^T u \leq \rho_t^2\} \subset Z^{t+1} \subset A_t E^t + \{B_t u + f_t : u^T u \leq \rho_t^2\}$$

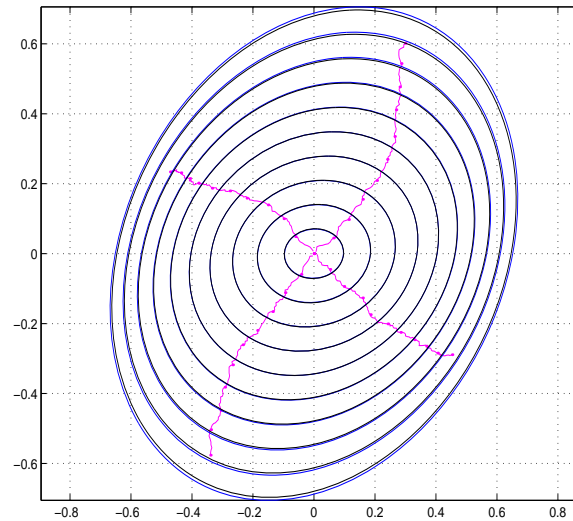
Thus, setting

$$\begin{aligned} E_{t+1} &= \text{largest volume ellipsoid} \subset A_t E_t + \{B_t u + f_t : u^T u \leq \rho_t^2\} \\ E^{t+1} &= \text{smallest volume ellipsoid} \supset A_t E^t + \{B_t u + f_t : u^T u \leq \rho_t^2\} \end{aligned}$$

we get (non-optimal!) “greedy” inner and outer ellipsoidal approximations of Z^{t+1} by solving recursively simple problems of approximating sums of *just two* ellipsoids (co-axial case!).

$$\frac{d}{dt} \begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix} = \underbrace{\begin{bmatrix} -0.8147 & -0.4163 \\ 0.8167 & -0.1853 \end{bmatrix}}_P \begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix} + \begin{bmatrix} u_1(t) \\ 0.7071u_2(t) \end{bmatrix}, \begin{bmatrix} x_1(0) \\ x_2(0) \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}, \|u(t)\|_2 \leq 1$$

$$\Rightarrow z(k+1) = \underbrace{\exp\{P\Delta t\}}_A z(k) + \underbrace{\left[\int_0^{\Delta t} \exp\{As\} \begin{bmatrix} 1 & 0 \\ 0 & 0.7071 \end{bmatrix} ds \right]}_B u(k), z(0) = \begin{bmatrix} 0 \\ 0 \end{bmatrix}, [\Delta t = 0.01]$$



Outer and inner on-line approximation of Z^t , $t = 10\ell$, $\ell = 1, \dots, 10$, and 4 sample trajectories

♣ A continuous time Linear Dynamical System

$$\begin{aligned} \dot{z} &= A(t)z + B(t)u(t) + f(t), & t \geq 0 \\ z(0) &\in E_0 \equiv \{z : z^T G_{\text{ini}} z \leq 1\} & [G_{\text{ini}} \succ 0] \end{aligned} \quad (*)$$

with norm-bounded control:

$$\|u(t)\|_2 \leq 1 \quad \forall t,$$

can be viewed as a limit of discrete time systems with norm-bounded control. The above discrete time greedy on-line policies for building ellipsoidal approximations yield continuous-time counterparts as follows:

We associate with (*) ordinary differential equations for *matrix-valued* functions G_t and W_t :

$$\begin{aligned} \frac{d}{dt} G_t &= -A^T(t)G_t - G_t A(t) - \left(\frac{n}{\text{Tr}(G_t B(t) B^T(t))} \right)^{1/2} G_t B(t) B^T(t) G_t - \left(\frac{\text{Tr}(G_t B(t) B^T(t))}{n} \right)^{1/2} G_t, \quad t \geq 0, \\ G_0 &= G_{\text{ini}}; \\ \frac{d}{dt} W_t &= -A^T(t)W_t - W_t A(t) - 2W_t^{1/2} (W_t^{1/2} B(t) B^T(t) W_t^{1/2})^{1/2} W_t^{1/2}, \quad t \geq 0, \\ W_0 &= G_{\text{ini}}. \end{aligned}$$

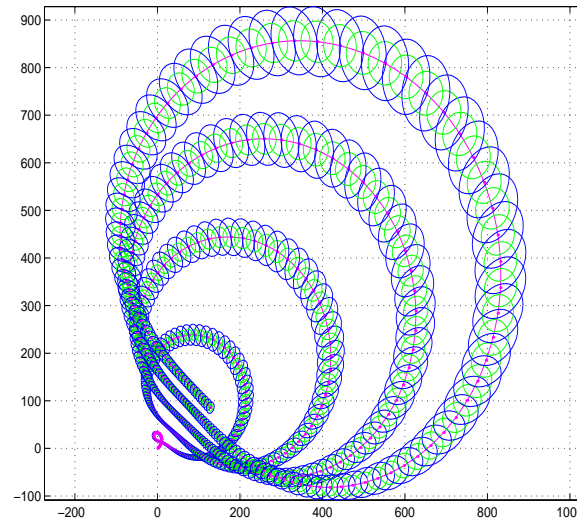
Let also z_t be the “central trajectory”:

$$\frac{d}{dt} z_t = A(t)z_t + f(t), \quad z_0 = 0.$$

Then $G_t \succ 0$, $W_t \succ 0$ for all $t \geq 0$, and for all t one has

$$\{z : (z - z_t)^T W_t (z - z_t) \leq 1\} \subset Z^t \subset \{z : (z - z_t)^T G_t (z - z_t) \leq 1\}$$

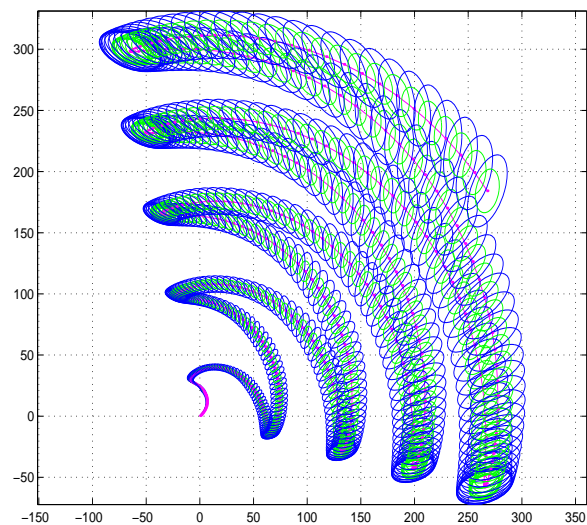
where Z^t is the set of all possible states of (*) at time t .



"Spiral"

$$\frac{d}{dt} \begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix} = \begin{bmatrix} \cos(t) & -\sin(t) \\ \sin(t) & \cos(t) \end{bmatrix} \begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix} + u(t) \begin{bmatrix} \cos(t) \\ \sin(t) \end{bmatrix} + \begin{bmatrix} 10 \\ 10 \end{bmatrix}$$

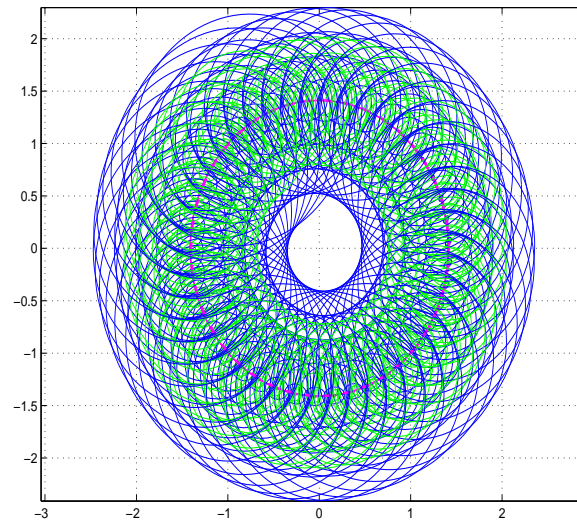
$x(0) = 0, \quad |u(\cdot)| \leq 1, \quad 0 \leq t \leq 30$



"Snake"

$$\frac{d}{dt} \begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix} = \begin{bmatrix} 0 & -\sin(t) \\ \sin(t) & 0 \end{bmatrix} \begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix} + u(t) \begin{bmatrix} \cos(t) \\ \sin(t) \end{bmatrix} + \begin{bmatrix} 10 \\ 10 \end{bmatrix}$$

$x(0) = 0, \quad |u(\cdot)| \leq 1, \quad 0 \leq t \leq 30$



“Pendulum”

$$\frac{d}{dt} \begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix} = \left[\begin{array}{c|c} 0 & 1 \\ -1 & 0 \end{array} \right] \begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix} + u(t) \begin{bmatrix} 0 \\ 0.05 \end{bmatrix}$$

$$\begin{bmatrix} \frac{d^2}{dt^2} x_1(t) = -x_1(t) + 0.05u(t) \\ x_2(t) = \frac{d}{dt} x_1(t) \end{bmatrix}$$

$$x_1(0) = 0, x_2(0) = 1, \quad |u(\cdot)| \leq 1, \quad 0 \leq t \leq 30$$

IV. COMPUTATIONAL TRACTABILITY OF CONVEX PROGRAMMING

A Mathematical Programming problem is

$$\min_x \left\{ p_0(x) : x \in X(p) \subset \mathbf{R}^{n(p)} \right\} \quad (p)$$

- $n(p)$ is the *design dimension* of problem (p) ;
- $X(p) \subset \mathbf{R}^n$ is the *feasible domain* of the problem;
- $p_0(x) : \mathbf{R}^n \rightarrow \mathbf{R}$ is the *objective* of (p) .

E.g., a conic program

$$\min_x \left\{ c^T x : Ax - b \in \mathbf{K} \right\}, \quad (\text{CP})$$

is a Mathematical Programming program given by

$$X(p) = \{x : Ax - b \in \mathbf{K}\}, \quad p_0(x) = c^T x.$$

Definition: A Mathematical Programming program

$$\min_x \left\{ p_0(x) : x \in X(p) \subset \mathbf{R}^{n(p)} \right\} \quad (p)$$

is called *convex*, if

- The domain $X(p)$ of the program is a convex set;
- The objective $p_0(x)$ is convex and real valued on the entire $\mathbf{R}^{n(p)}$.
- E.g., a conic program

$$\min_x \left\{ p_0(x) \equiv c^T x : x \in X(p) \equiv \{x : Ax - b \in \mathbf{K}\} \right\}$$

is convex.

Claim: (!) *Convex optimization programs are “computationally tractable”: there exist solution methods which “efficiently solve” every convex optimization program satisfying “very mild” computability and boundedness restrictions.*

(!!) *In contrast to this, no efficient universal solution methods for non-convex Mathematical Programming programs are known, and there are strong reasons to expect that no methods of this type exist.*

- To make (!) a rigorous statement, one should specify the notions of
- *solution method*
- *efficiency*

- Intuitively, a (numerical) solution method is a computer code; when solving a particular instance of optimization problem, computer loaded with this code inputs the data of the instance, executes the code on these data and outputs the result – a real array representing the solution, or the message “no solution exists”.

The efficiency of such a solution method on a particular problem’s instance can be measured by the *running time* of the code as applied to the data of the instance – the # of elementary operations performed by the computer when executing the code; the less is the running time, the higher is the efficiency.

When formalizing these intuitive considerations, we should specify a number of elements:

- *Model of computations*: What our computer can do, in particular, what are its “elementary operations”?
- *Encoding of program instances*: What are the problems we intend to solve and what are the “data of particular instances?”
- *Quality of solution*: Solution of what kind we expect to get? An exactly optimal or an approximate one? Even for simple convex programs, it would be unrealistic to expect that the data can be converted into an *exactly optimal* solution in *finitely many* elementary operations!

Real Arithmetics Complexity Model

Model of computations: idealized computer capable to store arbitrary many *reals* and to perform *exactly* the following standard operations with reals:

- four arithmetic operations
- comparisons
- computing elementary functions like \log , \exp , $\sqrt{}$, \sin ,...

(idealization comes from the assumption that reals can be stored and processed exactly!)

Generic optimization problem: a family of Mathematical Programming problems of a given “analytical structure”, like Linear, Conic Quadratic and Semidefinite Programming.

Formally: a generic optimization problem \mathcal{P} is a family of “instances” – optimization programs

$$\min_x \left\{ p_0(x) : x \in X(p) \subset \mathbf{R}^{n(p)} \right\} \quad (p)$$

where every instance $(p) \in \mathcal{P}$ is specified by a finite-dimensional *data vector* $\text{Data}(p)$.

The maximum of the design dimension $n(p)$ of an instance and the dimension of the data vector is called the *size* of the instance:

$$\text{Size}(p) = \max[n(p), \dim \text{Data}(p)].$$

Examples:

- **Linear Programming \mathcal{LP} :** collection of all possible LP programs

$$\min_x \{c^T x : Ax \geq b\} \quad [A : m \times n],$$

the data vector of an instance being

$$[n; m; c; \text{Vec}(A); b]$$

where for $A \in \mathbb{M}^{m,n}$

$$\text{Vec}(A) = [A_{11}; A_{21}; \dots; A_{m1}; A_{12}; \dots; A_{m2}; \dots; A_{1n}; \dots; A_{mn}].$$

- **Conic Quadratic Programming \mathcal{CQP} :** collection of all possible conic quadratic programs

$$\min_x \left\{ c^T x : \|D_i x - d_i\|_2 \leq e_i^T x - c_i, \ i = 1, \dots, k \right\}$$
$$[D_i : m_i \times n]$$

the data vector of an instance being

$$\left[n; k; m_1; \dots; m_k; c; \text{Vec} \left(\begin{bmatrix} D_1 & d_1 \\ e_1^T & c_1 \end{bmatrix} \right); \dots; \text{Vec} \left(\begin{bmatrix} D_k & d_k \\ e_k^T & c_k \end{bmatrix} \right) \right]$$

- **Semidefinite programming** SDP : collection of all possible semidefinite programs

$$\min_x \left\{ c^T x : \sum_{i=1}^n x_i A_i - B \succeq 0 \right\} \quad [A_i \in \mathbf{S}^m]$$

the data vector of an instance being

$$[n; m; c; \text{Vec}(A_1); \dots; \text{Vec}(A_n); \text{Vec}(B)] .$$

Accuracy of approximate solutions: Let \mathcal{P} be a generic *convex* optimization problem, meaning that all instances of \mathcal{P} are convex programs. We assume that \mathcal{P} is equipped with *infeasibility measure*

$$\text{Infeas}_{\mathcal{P}}(x, p)$$

– a real-valued function of $(p) \in \mathcal{P}$ and $x \in \mathbf{R}^{n(p)}$ which is nonnegative everywhere, is zero when $x \in X(p)$, and is *convex* in x .

Note: The infeasibility measure is a part of the description of \mathcal{P} .

• Given an infeasibility measure, we can define the notion of an *ϵ -solution* to an instance

$$(p) : \quad \min_x \{p_0(x) : x \in X(p) \subset \mathbf{R}^{n(p)}\}$$

of \mathcal{P} as a point $x \in \mathbf{R}^{n(p)}$ which is both ϵ -feasible and ϵ -optimal:

$$\text{Infeas}_{\mathcal{P}}(x, p) \leq \epsilon \ \& \ p_0(x) - \text{Opt}(p) \leq \epsilon,$$

where

$$\text{Opt}(p) \equiv \begin{cases} \inf_{x \in X(p)} p_0(x), & X(p) \neq \emptyset \\ +\infty, & \text{otherwise} \end{cases}$$

is the optimal value of (p) .

Example: Natural infeasibility measures for \mathcal{LP} , \mathcal{CQP} , \mathcal{SDP} are given by the following construction: An instance of the generic problem \mathcal{P} in question is a conic problem of the form

$$\min_x \left\{ c_{(p)}^T x : A_{(p)}x - b_{(p)} \in \mathbf{K}_{(p)} \right\} \quad (p)$$

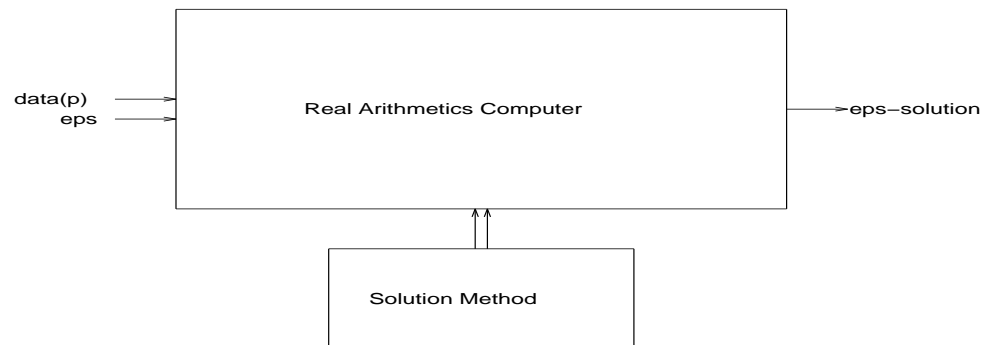
The infeasibility measure is

$$\text{Infeas}_{\mathcal{P}}(x, p) = \min_t \left\{ t \geq 0 : A_{(p)}x - b_{(p)} + te[\mathbf{K}_{(p)}] \in \mathbf{K}_{(p)} \right\},$$

where $e[\mathbf{K}] \in \text{int } \mathbf{K}$ is the “central point” of cone \mathbf{K} , specifically,

- 1 when \mathbf{K} is the nonnegative ray \mathbf{R}_+ ,
- the vector $[0; \dots; 0; 1] \in \mathbf{R}^m$, when \mathbf{K} is the Lorentz cone \mathbf{L}^m ;
- the unit matrix I_m , when $\mathbf{K} = \mathbf{S}_+^m$ is a semidefinite cone,
- the direct sum of the central points of the direct factors, when \mathbf{K} is a direct product of the just listed standard cones

- Let \mathcal{P} be a generic optimization problem. A *solution method* \mathcal{M} for \mathcal{P} is a code for the Real Arithmetics computer such that when loaded by \mathcal{M} and getting on input the data vector $\text{Data}(p)$ of an instance $(p) \in \mathcal{P}$ and $\epsilon > 0$, the computer in finitely many operations returns
 - either an $n(p)$ -dimensional vector $\text{Res}_{\mathcal{M}}(p, \epsilon)$ which is an ϵ -solution to (p) ,
 - or a correct message “ (p) is infeasible”,
 - or a correct message “ (p) is below unbounded”.



- The *complexity* of a solution method \mathcal{M} on input $((p), \epsilon)$ is

$$\text{Compl}_{\mathcal{M}}(p, \epsilon) = \# \text{ of real arithmetic operations carried out on input } (\text{Data}(p), \epsilon)$$

- The *complexity* of a solution method \mathcal{M} on input $((p), \epsilon)$ is

$$\text{Compl}_{\mathcal{M}}(p, \epsilon) = \begin{array}{l} \# \text{ of real arithmetic operations} \\ \text{carried out on input } (\text{Data}(p), \epsilon) \end{array}$$

- A solution method is called *polynomial time* (“theoretically efficient”) on \mathcal{P} , if its complexity is bounded by a polynomial of the size of (p) and the “number of accuracy digits”:

$$\exists \text{ polynomial } \pi : \forall (p) \in \mathcal{P} \forall \epsilon > 0 : \text{Compl}_{\mathcal{M}}(p, \epsilon) \leq \pi(\text{Size}(p), \text{Digits}(p, \epsilon))$$

$$\text{Digits}(p, \epsilon) = \ln \left(\frac{\text{Size}(p) + \|\text{Data}(p)\|_1 + \epsilon^2}{\epsilon} \right)$$

$$\left[\text{Size}(p) = \max[n(p), \dim \text{Data}(p)], \|u\|_1 = \sum_{i=1}^{\dim u} |u_i| \right]$$

- A generic optimization problem \mathcal{P} is called *polynomially solvable* (“computationally tractable”), if it admits a polynomial time solution method.

- A polynomial time method:

$$\exists \text{ polynomial } \pi : \forall (p) \in \mathcal{P} \ \forall \epsilon > 0 : \text{Compl}_{\mathcal{M}}(p, \epsilon) \leq \pi(\text{Size}(p), \text{Digits}(p, \epsilon))$$

$$\text{Digits}(p, \epsilon) = \ln \left(\frac{\text{Size}(p) + \|\text{Data}(p)\|_1 + \epsilon^2}{\epsilon} \right)$$

$$\left[\text{Size}(p) = \max[n(p), \dim \text{Data}(p)], \|u\|_1 = \sum_{i=1}^{\dim u} |u_i| \right]$$

- For a polynomial time method, increasing by absolute constant factor (say, by 10) computer's performance, we can increase by (another) absolute constant factor the size of instances which can be processed in a fixed time and the number of accuracy digits to which the instances are processed in this time. In contrast to this,

- for a solution method with exponential in $\text{Size}(\cdot)$ complexity like

$$\text{Compl}_{\mathcal{M}}(p, \epsilon) \approx f(\epsilon) \exp\{\text{Size}(p)\}$$

10-fold progress in computer power allows to increase the sizes of problems solvable to a fixed accuracy in a fixed time only by *additive* absolute constant ≈ 2 .

- for a solution method with sublinear in $1/\epsilon$ complexity like

$$\text{Compl}_{\mathcal{M}}(p, \epsilon) \approx f(\text{Size}(p)) \frac{1}{\epsilon}$$

10-fold progress in computer power allows to increase the # of accuracy digits available in a fixed time only by *additive* absolute constant ≈ 1 .

- The complexity bound of a typical polynomial time method is just linear in the # of accuracy digits:

$$\text{Compl}_{\mathcal{M}}(p, \epsilon) \leq O(1) \text{Size}^{\alpha}(p) \text{Digits}(p, \epsilon).$$

For such a method, polynomially means that the “arithmetic cost” of an extra accuracy digit is independent of the position of the digit (is it the 1-st or the 10,000-th) and is polynomial in the dimension of the data vector.

Polynomial Solvability of Convex Programming

- We are about to prove that under “mild assumptions” a generic *convex* optimization problem \mathcal{P} is polynomially solvable.

The assumptions are

- Polynomial computability
- Polynomial growth
- Polynomial boundedness of feasible sets.

1. Polynomial computability

- We say that a generic convex optimization problem

$$\mathcal{P} = \left\{ (p) : \min_x \left\{ p_0(x) \mid x \in X(p) \in \mathbf{R}^{n(p)} \right\} \right\}$$

is *polynomially computable*, if

1.1. There exists a code \mathcal{C}_{obj} for the Real Arithmetics computer which, given on input the data vector $\text{Data}(p)$ of an instance $(p) \in \mathcal{P}$ and a vector $x \in \mathbf{R}^{n(p)}$, reports on output the value $p_0(x)$ and a subgradient $p'_0(x)$ of the objective of (p) at x , and the $\# T_{\text{obj}}(x, p)$ of operations in course of this computation is bounded by a polynomial of $\text{Size}(p)$:

$$\forall \left((p) \in \mathcal{P}, x \in \mathbf{R}^{n(p)} \right) : \quad T_{\text{obj}}(x, p) \leq \chi \text{Size}^\chi(p) \\ [\text{Size}(p) = \max[n(p), \dim \text{Data}(p)]]$$

From now on, χ stands for positive constants “characteristic for \mathcal{P} ” and independent of particular choice of $(p) \in \mathcal{P}$, $\epsilon > 0$, etc.

1.2. There exists a code $\mathcal{C}_{\text{cons}}$ for the Real Arithmetics computer which, given on input the data vector $\text{Data}(p)$ of an instance $(p) \in \mathcal{P}$, a vector $x \in \mathbf{R}^{n(p)}$ and $\epsilon > 0$, reports on output whether $\text{Infeas}_{\mathcal{P}}(x, p) \leq \epsilon$, and if it is **not** the case, returns vector e which separates x and the set $\{y : \text{Infeas}_{\mathcal{P}}(y, p) \leq \epsilon\}$:

$$\text{Infeas}_{\mathcal{P}}(y, p) \leq \epsilon \Rightarrow e^T x > e^T y,$$

and the $\# T_{\text{cons}}(x, \epsilon, p)$ of operations in course of this computation is bounded by a polynomial of $\text{Size}(p)$ and $\text{Digits}(p, \epsilon)$:

$$\forall \left(\begin{array}{c} (p) \in \mathcal{P} \\ x \in \mathbf{R}^{n(p)} \\ \epsilon > 0 \end{array} \right) : \quad T_{\text{cons}}(x, \epsilon, p) \leq \chi (\text{Size}(p) + \text{Digits}(p, \epsilon))^{\chi}.$$

2. Polynomial growth

- We say that a generic convex optimization problem

$$\mathcal{P} = \left\{ (p) : \min_x \left\{ p_0(x) : x \in X(p) \in \mathbf{R}^{n(p)} \right\} \right\}$$

is of *polynomial growth*, if the objectives and the infeasibility measures, as functions of x , grow polynomially with $\|x\|_1$, the degree of the polynomial being a power of $\text{Size}(p)$:

$$\forall \left((p) \in \mathcal{P}, x \in \mathbf{R}^{n(p)} \right) : \\ |p_0(x)| + \text{Infeas}_{\mathcal{P}}(x, p) \leq (\chi [\text{Size}(p) + \|x\|_1 + \|\text{Data}(p)\|_1]) (\chi^{\text{Size}(p)}).$$

3. Polynomial boundedness of feasible sets

- We say that a generic convex optimization problems \mathcal{P} has polynomially bounded feasible sets, if the feasible set $X(p)$ of every instance $(p) \in \mathcal{P}$ is bounded and is contained in the centered at the origin Euclidean ball of “not too large” radius:

$$\forall (p) \in \mathcal{P} : X(p) \subset \left\{ x \in \mathbf{R}^{n(p)} : \|x\|_2 \leq (\chi [\text{Size}(p) + \|\text{Data}(p)\|_1])^{\chi \text{Size}^x(p)} \right\}.$$

♣ It is easily seen that the generic convex programs \mathcal{LP} , \mathcal{CQP} , \mathcal{SDP} (same as basically all other generic convex programs) satisfy the assumptions of polynomial computability and polynomial growth.

At the same time, \mathcal{LP} , \mathcal{CQP} , \mathcal{SDP} (and most of other generic convex programs) “as they are” do *not* satisfy the assumption of polynomial boundedness. *We can enforce polynomial boundedness of feasible sets by rejecting to deal with instances where an upper bound on the norm of a feasible solution is not stated explicitly.* To this end we pass from a generic problem \mathcal{P} to the problem \mathcal{P}_b with instances $(p^+) = ((p), R)$:

$$\begin{aligned} (p) : \quad & \min_x \{p_0(x) : x \in X(p)\} \\ \Rightarrow (p^+) : \quad & \min_x \{p_0(x) : x \in X_R(p) = \{x \in X(p) : \|x\|_\infty \leq R\}\} \\ & \left[\text{Data}(p^+) = (\text{Data}(p), R) \right] \end{aligned}$$

Note that $\mathcal{LP}_b \subset \mathcal{LP}$, $\mathcal{CQP}_b \subset \mathcal{CQP}$, $\mathcal{SDP}_b \subset \mathcal{SDP}$, and the generic convex programs \mathcal{LP}_b , \mathcal{CQP}_b , \mathcal{SDP}_b satisfy the assumption of polynomial boundedness of feasible sets (same as the assumptions of polynomial computability and polynomial growth).

Theorem [Polynomial Solvability of Convex Programming] *Let \mathcal{P} be a generic convex optimization problem which is*

- (a) polynomially computable*
- (b) of polynomial growth*
- (c) with polynomially bounded feasible sets.*

Then \mathcal{P} is polynomially solvable.

Key Component: Ellipsoid Algorithm

♣ Consider an optimization program

$$f_* = \min_X f(x) \quad (\text{P})$$

- $X \subset \mathbf{R}^n$ is a closed and bounded convex set with a nonempty interior;
- f is a continuous convex function on \mathbf{R}^n .

♠ Assume that our “environment” when solving (P) is as follows:

A. We have access to a *Separation Oracle* $\text{Sep}(X)$ for X – a routine which, given on input a point $x \in \mathbf{R}^n$, reports whether $x \in X$, and in the case of $x \notin X$, returns a *separator* – a vector $e \neq 0$ such that

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

B. We have access to a *First Order Oracle* which, given on input a point $x \in X$, returns the value $f(x)$ and a *subgradient* $f'(x)$ of f at x :

$$\forall y : f(y) \geq f(x) + (y - x)^T f'(x).$$

Note: When f is differentiable, one can set $f'(x) = \nabla f(x)$.

C. We are given positive reals R, r, V such that for some (unknown) c one has

$$\{x : \|x - c\|_2 \leq r\} \subset X \subset \{x : \|x\|_2 \leq R\}$$

and

$$\max_{x \in X} f(x) - \min_{x \in X} f(x) \leq V.$$

♠ **Example:** Consider an optimization program

$$\min_x \left\{ f(x) \equiv \max_{1 \leq \ell \leq L} [p_\ell + q_\ell^T x] : x \in X = \{x : a_i^T x \leq b_i, 1 \leq i \leq m\} \right\}$$

W.l.o.g. we assume that $a_i \neq 0$ for all i .

♠ A Separation Oracle can be as follows: given x , the oracle checks whether $a_i^T x \leq b_i$ for all i . If it is the case, the oracle reports that $x \in X$, otherwise it finds $i = i_x$ such that $a_{i_x}^T x > b_{i_x}$, reports that $x \notin X$ and returns a_{i_x} as a separator. This indeed is a separator:

$$y \in X \Rightarrow a_{i_x}^T y \leq b_{i_x} < a_{i_x}^T x$$

♠ A First Order Oracle can be as follows: given x , the oracle computes the quantities $p_\ell + q_\ell^T x$ for $\ell = 1, \dots, L$ and identifies the largest of these quantities, which is exactly $f(x)$, along with the corresponding index ℓ , let it be ℓ_x : $f(x) = p_{\ell_x} + q_{\ell_x}^T x$. The oracle returns the computed $f(x)$ and, as a subgradient $f'(x)$, the vector q_{ℓ_x} . This indeed is a subgradient:

$$f(y) \geq p_{\ell_x} + q_{\ell_x}^T y = [p_{\ell_x} + q_{\ell_x}^T x] + (y - x)^T q_{\ell_x} = f(x) + (y - x)^T f'(x).$$

$$f_* = \min_X f(x) \quad (\text{P})$$

- $X \subset \mathbf{R}^n$ is a closed and bounded convex set with a nonempty interior;
- f is a continuous convex function on \mathbf{R}^n .
- We have access to a *Separation Oracle* which, given on input a point $x \in \mathbf{R}^n$, reports whether $x \in X$, and in the case of $x \notin X$, returns a separator $e \neq 0$:

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

- We have access to a *First Order Oracle* which, given on input a point $x \in X$, returns the value $f(x)$ and a subgradient $f'(x)$ of f :

$$\forall y : f(y) \geq f(x) + (y - x)^T f'(x).$$

- We are given positive reals R, r, V such that for some (unknown) c one has

$$\{x : \|x - c\|_2 \leq r\} \subset X \subset \{x : \|x\|_2 \leq R\}$$

and

$$\max_{x \in X} f(x) - \min_{x \in X} f(x) \leq V.$$

- ♠ *How to build a good solution method for (P)?*

To get an idea, let us start with univariate case.

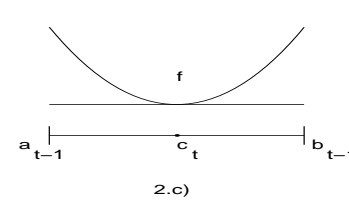
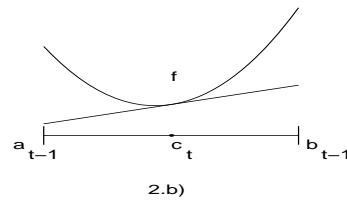
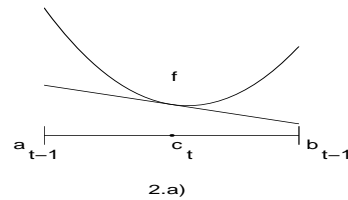
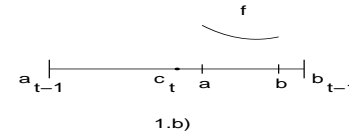
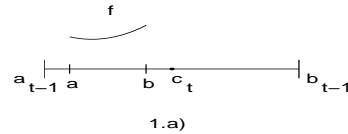
Univariate Case: Bisection

♣ When solving a problem $\min_x \{f(x) : x \in X = [a, b] \subset [-R, R]\}$, by bisection, we recursively update *localizers* – segments $\Delta_t = [a_{t-1}, b_{t-1}]$ containing the optimal set X_{opt} .

● **Initialization:** Set $\Delta_1 = [-R, R] \supset X_{\text{opt}}$

$$\min_x \{f(x) : x \in X = [a, b] \subset [-R, R]\},$$

- **Step t :** Given $\Delta_t = [a_{t-1}, b_{t-1}] \supset X_{\text{opt}}$ let c_t be the midpoint of Δ_t . Calling Separation and First Order oracles at c_t , we replace Δ_t by *twice smaller* localizer Δ_{t+1} .



1)	Sep _X says that $c_t \notin X$ and reports, via separator e , on which side of c_t X is. 1.a): $\Delta_{t+1} = [a_{t-1}, c_t]$; 1.b): $\Delta_{t+1} = [c_t, b_{t-1}]$
2)	Sep _X says that $c_t \in X$, and \mathcal{O}_f reports, via $\text{sign}[f'(c_t)]$, on which side of c_t X_{opt} is. 2.a): $\Delta_{t+1} = [c_t, b_{t-1}]$; 2.b): $\Delta_{t+1} = [a_{t-1}, c_t]$; 2.c): $c_t \in X_{\text{opt}}$

♠ Since the localizers rapidly shrink and X is of positive length, eventually some of search points will become feasible, and the nonoptimality of the best found so far feasible search point will rapidly converge to 0 as process goes on.

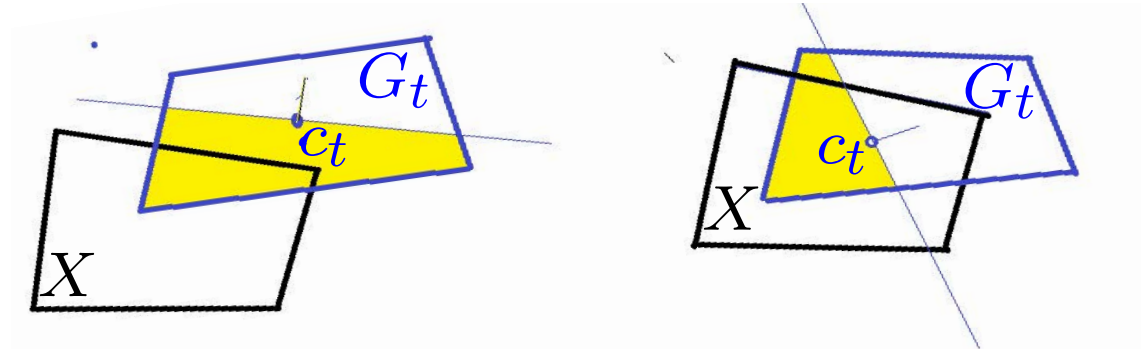
$$\text{Opt}(P) = \min_{x \in X \subset \mathbf{R}^n} f(x) \quad (P)$$

♠ Bisection admits multidimensional extension, called *Generic Cutting Plane Algorithm*, where one builds a sequence of “shrinking” *localizers* G_t – closed and bounded convex domains containing the optimal set X_{opt} of (P) .

Generic Cutting Plane Algorithm is as follows:

♠ **Initialization** Select as G_1 a closed and bounded convex set containing X and thus being a localizer.

$$\text{Opt}(P) = \min_{x \in X \subset \mathbf{R}^n} f(x) \quad (P)$$



Left: $c_t \notin X$ (case A); right: $c_t \in X$ (case B). Yellow polygon: \hat{G}_t .

♠ **Step** $t = 1, 2, \dots$: Given current localizer G_t ,

- Select current **search point** $c_t \in G_t$ and call Separation and First Order oracles to form a **cut** – to find $e_t \neq 0$ s.t. $X_{\text{opt}} \subset \hat{G}_t := \{x \in G_t : e_t^T x \leq e_t^T c_t\}$.

To this end

— call Sep_X , c_t being the input. If Sep_X says that $c_t \notin X$ and returns a separator, take it as e_t (case A on the picture).

Note: $c_t \notin X \Rightarrow$ all points from $G_t \setminus \hat{G}_t$ are infeasible

— if $c_t \in X$, call \mathcal{O}_f to compute $f(c_t)$, $f'(c_t)$. If $f'(c_t) = 0$, terminate, otherwise set $e_t = f'(c_t)$ (case B on the picture).

Note: When $f'(c_t) = 0$, c_t is optimal for (P) , otherwise $f(x) > f(c_t)$ at all feasible $x \in G_t \setminus \hat{G}_t$

- By the two “Note” above, \hat{G}_t is a localizer along with G_t . Select a closed and bounded convex set $G_{t+1} \supset \hat{G}_t$ (it also will be a localizer) and pass to step $t + 1$.

$$\text{Opt}(P) = \min_{x \in X \subset \mathbf{R}^n} f(x) \quad (P)$$

♣ **Summary:** Given current localizer G_t , selecting a point $c_t \in G_t$ and calling the Separation and the First Order oracles, we can

♠ in the *productive case* $c_t \in X$, find e_t such that

$$e_t^T(x - c_t) > 0 \Rightarrow f(x) > f(c_t)$$

♠ in the *non-productive case* $c_t \notin X$, find e_t such that

$$e_t^T(x - c_t) > 0 \Rightarrow x \notin X$$

\Rightarrow the set $\hat{G}_t = \{x \in G_t : e_t^T(x - c_t) \leq 0\}$ is a localizer

♣ We can select as the next localizer G_{t+1} any set containing \hat{G}_t .

♠ We define approximate solution x^t built in course of $t = 1, 2, \dots$ steps as the best – with the smallest value of f – of the *feasible* search points c_1, \dots, c_t built so far.

If in course of the first t steps no feasible search points were built, x^t is undefined.

$$\text{Opt}(P) = \min_{x \in X \subset \mathbf{R}^n} f(x) \quad (P)$$

♣ Analysing Cutting Plane algorithm

- Let $\text{Vol}(G)$ be the n -dimensional volume of a closed and bounded convex set $G \subset \mathbf{R}^n$.

Note: For convenience, we use, as the unit of volume, the volume of n -dimensional unit ball $\{x \in \mathbf{R}^n : \|x\|_2 \leq 1\}$, and not the volume of n -dimensional unit box.

- Let us call the quantity $\rho(G) = [\text{Vol}(G)]^{1/n}$ the *radius* of G . $\rho(G)$ is the radius of n -dimensional ball with the same volume as G , and this quantity can be thought of as the average linear size of G .

Theorem. *Let convex problem (P) satisfying our standing assumptions be solved by Generic Cutting Plane Algorithm generating localizers G_1, G_2, \dots and ensuring that $\rho(G_t) \rightarrow 0$ as $t \rightarrow \infty$. Let \bar{t} be the first step where $\rho(G_{\bar{t}+1}) < \rho(X)$. Starting with this step, approximate solution x^t is well defined and obeys the “error bound”*

$$f(x^t) - \text{Opt}(P) \leq \min_{\tau \leq t} \left[\frac{\rho(G_{\tau+1})}{\rho(X)} \right] \left[\max_X f - \min_X f \right]$$

$$\text{Opt}(P) = \min_{x \in X \subset \mathbf{R}^n} f(x) \quad (P)$$

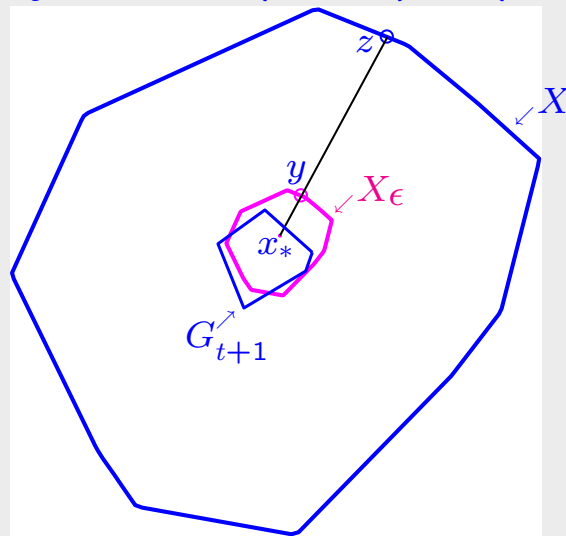
Explanation: Since $\text{int } X \neq \emptyset$, $\rho(X)$ is positive, and since X is closed and bounded, (P) is solvable. Let x_* be an optimal solution to (P).

- Let us fix $\epsilon \in (0, 1)$ and set $X_\epsilon = x_* + \epsilon(X - x_*)$.

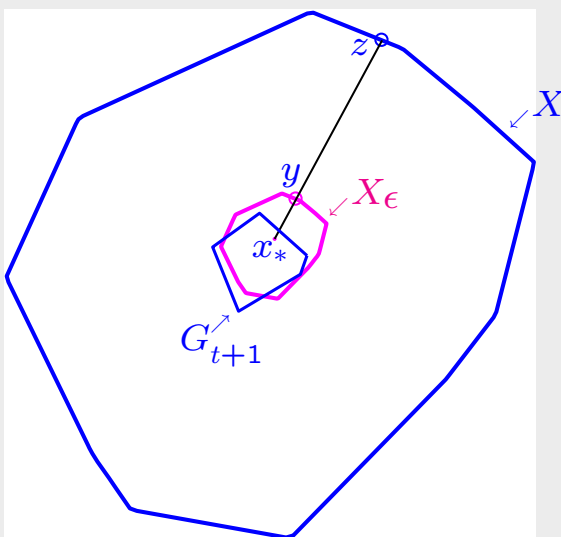
X_ϵ is obtained X by similarity transformation which keeps x_* intact and “shrinks” X towards x_* by factor ϵ . This transformation multiplies volumes by $\epsilon^n \Rightarrow \rho(X_\epsilon) = \epsilon \rho(X)$.

- Let t be such that $\rho(G_{t+1}) < \epsilon \rho(X) = \rho(X_\epsilon)$. Then $\text{Vol}(G_{t+1}) < \text{Vol}(X_\epsilon) \Rightarrow$ the set $X_\epsilon \setminus G_{t+1}$ is nonempty \Rightarrow for some $z \in X$, the point

$$y = x_* + \epsilon(z - x_*) = (1 - \epsilon)x_* + \epsilon z$$



does **not** belong to G_{t+1} .



- G_1 contains X and thus y , and G_{t+1} does not contain y , implying that *for some* $\tau \leq t$, it holds

$$e_\tau^T y > e_\tau^T c_\tau \quad (!)$$

- We definitely have $c_\tau \in X$ – otherwise e_τ separates c_τ and $X \ni y$, and (!) witnesses otherwise.

$$\Rightarrow c_\tau \in X \Rightarrow e_\tau = f'(c_\tau) \Rightarrow f(c_\tau) + e_\tau^T (y - c_\tau) \leq f(y)$$

\Rightarrow [by (!)]

$$f(c_\tau) \leq f(y) = f((1 - \epsilon)x_* + \epsilon z) \leq (1 - \epsilon)f(x_*) + \epsilon f(z)$$

$$\Rightarrow f(c_\tau) - f(x_*) \leq \epsilon[f(z) - f(x_*)] \leq \epsilon \left[\max_X f - \min_X f \right].$$

Bottom line: If $0 < \epsilon < 1$ and $\rho(G_{t+1}) < \epsilon \rho(X)$, then x^t is well defined (since $\tau \leq t$ and c_τ is feasible) and $f(x^t) - \text{Opt}(P) \leq \epsilon \left[\max_X f - \min_X f \right]$.

$$\text{Opt}(P) = \min_{x \in X \subset \mathbf{R}^n} f(x) \quad (P)$$

“Starting with the first step \bar{t} where $\rho(G_{\bar{t}+1}) < \rho(X)$, x^t is well defined, and

$$f(x^t) - \text{Opt} \leq \underbrace{\min_{\tau \leq t} \left[\frac{\rho(G_{\tau+1})}{\rho(X)} \right]}_{\epsilon_t} \underbrace{\left[\max_X f - \min_X f \right]}_V$$

♣ We are done. Let $t \geq \bar{t}$, so that $\epsilon_t < 1$, and let $\epsilon \in (\epsilon_t, 1)$. Then for some $t' \leq t$ we have

$$\rho(G_{t'+1}) < \epsilon \rho(X)$$

\Rightarrow [by bottom line] $x^{t'}$ is well defined and

$$f(x^{t'}) - \text{Opt}(P) \leq \epsilon V$$

\Rightarrow [since $f(x^t) \leq f(x^{t'})$ due to $t \geq t'$] x^t is well defined and $f(x^t) - \text{Opt}(P) \leq \epsilon V$

\Rightarrow [passing to limit as $\epsilon \rightarrow \epsilon_t + 0$] x^t is well defined and $f(x^t) - \text{Opt}(P) \leq \epsilon_t V$

□

$$\text{Opt}(P) = \min_{x \in X \subset \mathbf{R}^n} f(x) \quad (P)$$

♠ **Corollary:** Let (P) be solved by cutting Plane Algorithm which ensures, for some $\vartheta \in (0, 1)$, that $\rho(G_{t+1}) \leq \vartheta \rho(G_t)$. Then, for every desired accuracy $\epsilon > 0$, finding feasible ϵ -optimal solution x_ϵ to (P) (i.e., a feasible solution x_ϵ satisfying $f(x_\epsilon) - \text{Opt} \leq \epsilon$) takes at most

$$N = \frac{1}{\ln(1/\vartheta)} \ln \left(\mathcal{R} \left[1 + \frac{V}{\epsilon} \right] \right) + 1$$

steps of the algorithm. Here

$$\mathcal{R} = \frac{\rho(G_1)}{\rho(X)}$$

says how well, in terms of volume, the initial localizer G_1 approximates X , and

$$V = \max_X f - \min_X f$$

is the variation of f on X .

Note: \mathcal{R} and V/ϵ are under log, implying that high accuracy and poor approximation of X by G_1 cost “nearly nothing.”

What matters, is the factor *at the log* which is the larger the closer $\vartheta < 1$ is to 1.

“Academic” Implementation: Centers of Gravity

♠ In high dimensions, to ensure progress in volumes of subsequent localizers in a Cutting Plane algorithm is not an easy task: we do *not* know how the cut through c_t will pass, and thus should select c_t in G_t in such a way that *whatever be the cut*, it cuts off the current localizer G_t a “meaningful” part of its volume.

♠ The most natural choice of c_t in G_t is the *center of gravity*:

$$c_t = \left[\int_{G_t} x dx \right] / \left[\int_{G_t} 1 dx \right],$$

the expectation of the random vector uniformly distributed on G_t .

Good news: The Center of Gravity policy with $G_{t+1} = \hat{G}_t$ results in

$$\vartheta = \left(1 - \left[\frac{n}{n+1} \right]^n \right)^{1/n} \leq [0.632...]^{1/n} \quad (*)$$

This results in the complexity bound (# of steps needed to build ϵ -solution)

$$N = 2.2n \ln \left(\mathcal{R} \left[1 + \frac{V}{\epsilon} \right] \right) + 1$$

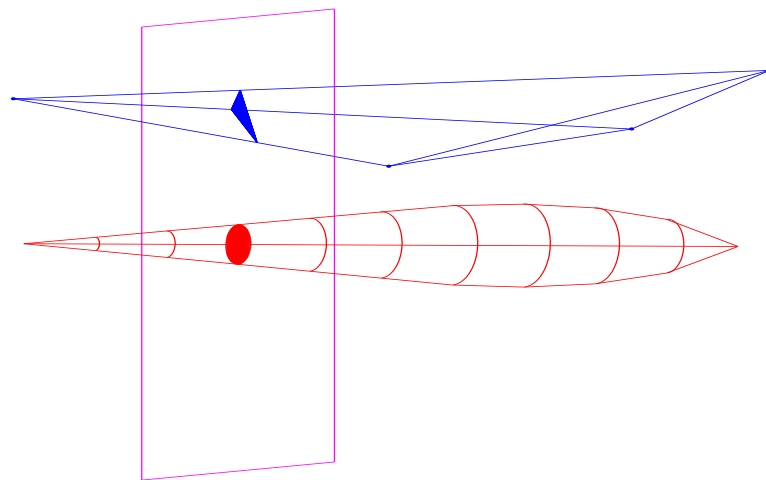
Note: It can be proved that *within absolute constant factor*, like 4, *this is the best complexity bound achievable by whatever algorithm for convex minimization which can “learn” the objective via First Order oracle only.*

♣ Reason for (*): Brunn-Minkowski Symmeterization Principle:

Let Y be a convex compact set in \mathbb{R}^n , e be a unit direction and Z be “equi-cross-sectional” to X body symmetric w.r.t. e , so that

- Z is rotationally symmetric w.r.t. the axis e
- for every hyperplane $H = \{x : e^T x = \text{const}\}$, one has

$$\text{Vol}_{n-1}(X \cap H) = \text{Vol}_{n-1}(Z \cap H)$$



Then Z is a **convex** compact set.

Equivalently: Let U, V be convex compact nonempty sets in \mathbb{R}^n . Then

$$\text{Vol}^{1/n}(U + V) \geq \text{Vol}^{1/n}(U) + \text{Vol}^{1/n}(V).$$

In fact, convexity of U, V is redundant!

Disastrously bad news: Centers of Gravity are *not* implementable, unless the dimension n of the problem is like 2 or 3.

Reason: We have no control on the shape of localizers. When started with a polytope G_1 given by M linear inequalities (e.g., a box), G_t for $t \gg n$ can be a more or less arbitrary polytope given by $M + t - 1$ linear inequalities. *Computing center of gravity of a general-type high-dimensional polytope is a computationally intractable task – it requires astronomically many computations already in the dimensions like 5 – 10.*

Remedy: *Maintain the shape of G_t simple and convenient for computing centers of gravity, sacrificing, if necessary, the value of ϑ .*

The most natural implementation of this remedy is enforcing G_t to be *ellipsoids*. As a result,

- c_t becomes computable in $O(n^2)$ operations (nice!)
- $\vartheta = [0.632\dots]^{1/n} \approx \exp\{-0.367/n\}$ increases to $\vartheta \approx \exp\{-0.5/n^2\}$, spoiling the complexity bound

$$N = 2.2n \ln \left(\mathcal{R} \left[1 + \frac{V}{\epsilon} \right] \right) + 1$$

to

$$N = 4n^2 \ln \left(\mathcal{R} \left[1 + \frac{V}{\epsilon} \right] \right) + 1$$

(unpleasant, but survivable...)

Practical Implementation - Ellipsoid Method

♠ *Ellipsoid in \mathbf{R}^n* is the image of the unit n -dimensional ball under one-to-one affine mapping:

$$E = E(B, c) = \{x = Bu + c : u^T u \leq 1\}$$

where B is $n \times n$ nonsingular matrix, and $c \in \mathbf{R}^n$.

- c is the center of ellipsoid $E = E(B, c)$: when $c + h \in E$, $c - h \in E$ as well
- When multiplying by $n \times n$ matrix B , n -dimensional volumes are multiplied by $|\text{Det}(B)|$

$$\Rightarrow \text{Vol}(E(B, c)) = |\text{Det}(B)|, \quad \rho(E(B, c)) = |\text{Det}(B)|^{1/n}.$$

$$E = E(B, c) = \{x = Bu + c : u^T u \leq 1\}$$

Simple fact: Let $E(B, c)$ be ellipsoid in \mathbb{R}^n and $e \in \mathbb{R}^n$ be a nonzero vector. The “half-ellipsoid”

$$\hat{E} = \{x \in E(B, c) : e^T x \leq e^T c\}$$

is covered by the ellipsoid $E^+ = E(B^+, c^+)$ given by

$$c^+ = c - \frac{1}{n+1} Bp, \quad p = B^T e / \sqrt{e^T B B^T e}$$

$$B^+ = \frac{n}{\sqrt{n^2-1}} B + \left(\frac{n}{n+1} - \frac{n}{\sqrt{n^2-1}} \right) (Bp)p^T,$$

- $E(B^+, c^+)$ is the ellipsoid of the smallest volume containing the half-ellipsoid \hat{E} , and the volume of $E(B^+, c^+)$ is *strictly smaller* than the one of $E(B, c)$:

$$\vartheta := \frac{\rho(E(B^+, c^+))}{\rho(E(B, c))} \leq \exp\left\{-\frac{1}{2n^2}\right\}.$$

- Given B, c, e , computing B^+, c^+ costs $O(n^2)$ arithmetic operations.

$$\text{Opt}(P) = \min_{x \in X \subset \mathbf{R}^n} f(x) \quad (P)$$

♣ **Ellipsoid method** is the Cutting Plane Algorithm where

- all localizers G_t are ellipsoids:

$$G_t = E(B_t, c_t),$$

- the search point at step t is c_t , and
- G_{t+1} is the smallest volume ellipsoid containing the half-ellipsoid

$$\hat{G}_t = \{x \in G_t : e_t^T x \leq e_t^T c_t\}$$

Computationally, at every step of the algorithm we once call the Separation oracle Sep_X , (at most) once call the First Order oracle \mathcal{O}_f and spend $O(n^2)$ operations to update (B_t, c_t) into (B_{t+1}, c_{t+1}) by explicit formulas.

♠ **Complexity bound** of the Ellipsoid algorithm is

$$N = 4n^2 \ln \left(\mathcal{R} \left[1 + \frac{V}{\epsilon} \right] \right) + 1$$

$$\mathcal{R} = \frac{\rho(G_1)}{\rho(X)} \leq \frac{R}{r}, \quad V = \max_{x \in X} f(x) - \min_{x \in X} f(x)$$

Pay attention:

- \mathcal{R}, V, ϵ are under log \Rightarrow large magnitudes in data entries and high accuracy are not issues
- the factor at the log depends only on the **structural** parameter of the problem (its design dimension n) and is independent of the remaining data.

What is Inside Simple Fact

♠ Messy formulas describing the updating

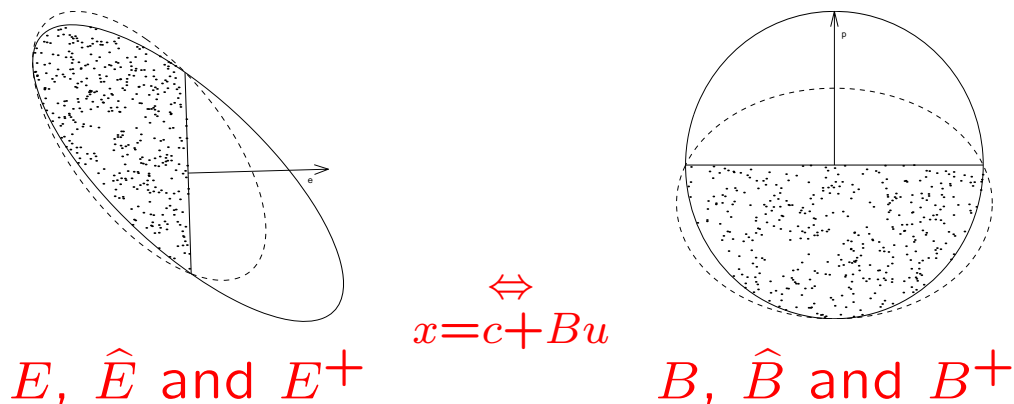
$$(B_t, c_t) \rightarrow (B_{t+1}, c_{t+1})$$

in fact are easy to get.

- Ellipsoid E is the image of the unit ball B under affine transformation.

Affine transformation preserves ratio of volumes

\Rightarrow Finding the smallest volume ellipsoid containing a given half-ellipsoid \hat{E} reduces to finding the smallest volume ellipsoid B^+ containing half-ball \hat{B} :



- The “ball” problem is highly symmetric, and solving it reduces to a simple exercise in elementary Calculus.

Why Ellipsoids?

(?) When enforcing the localizers to be of “simple and stable” shape, why we make them ellipsoids (i.e., affine images of the unit Euclidean ball), and not something else, say parallelotopes (affine images of the unit box)?

Answer: In a “simple stable shape” version of Cutting Plane Scheme all localizers are affine images of some fixed n -dimensional *solid* \mathbf{C} (closed and bounded convex set in \mathbf{R}^n with a nonempty interior). To allow for reducing step by step volumes of localizers, \mathbf{C} cannot be arbitrary. What we need is the following property of \mathbf{C} :

One can fix a point \mathbf{c} in \mathbf{C} in such a way that *whatever be a cut*

$$\hat{\mathbf{C}} = \{x \in \mathbf{C} : e^T x \leq e^T \mathbf{c}\} \quad [e \neq 0]$$

this cut can be covered by the affine image of \mathbf{C} with the volume less than the one of \mathbf{C} :

$$\exists B, b : \hat{\mathbf{C}} \subset BC + b \ \& \ |\text{Det}(B)| < 1 \quad (!)$$

Note: The Ellipsoid method corresponds to unit Euclidean ball in the role of \mathbf{C} and to $\mathbf{c} = 0$, which allows to satisfy (!) with $|\text{Det}(B)| \leq \exp\{-\frac{1}{2n}\}$, finally yielding $\vartheta \leq \exp\{-\frac{1}{2n^2}\}$.

- Solids C with the above property are “rare commodity.” For example, n -dimensional box does *not* possess it.
 - Another “good” solid is n -dimensional simplex (this is not that easy to see!). Here (!) can be satisfied with $|\text{Det}(B)| \leq \exp\{-O(1/n^2)\}$, finally yielding $\vartheta = (1 - O(1/n^3))$.
- \Rightarrow From the complexity viewpoint, “simplex” Cutting Plane algorithm is worse than the Ellipsoid method.
- The same is true for handful of other known so far (and quite exotic) “good solids.”

Ellipsoid Method: pro's & con's

♣ **Academically speaking,** *Ellipsoid method is an indispensable tool underlying basically all results on efficient solvability of generic convex problems, most notably, the famous theorem of L. Khachiyan (1978) on polynomial time solvability of Linear Programming with rational data in Rational Arithmetic Complexity model.*

♠ *What matters from theoretical perspective, is “universality” of the algorithm (nearly no assumptions on the problem except for convexity) and complexity bound of the form “structural parameter outside of log, all else, including required accuracy, under the log.”*

♠ Another theoretical (and to some extent, also practical) advantage of the Ellipsoid algorithm is that *as far as the representation of the feasible set X is concerned, all we need is a Separation oracle, and not the list of constraints describing X .* The number of these constraints can be astronomically large, making impossible to check feasibility by looking at the constraints one by one; however, in many important situations the constraints are “well organized,” allowing to implement Separation oracle efficiently.

♠ Theoretically, the only (and minor!) drawbacks of the algorithm is the necessity for the feasible set X to be bounded, with known “upper bound,” and to possess nonempty interior.

As of now, there is not way to cure the first drawback without sacrificing universality. The second “drawback” is artifact: given nonempty set

$$X = \{x : g_i(x) \leq 0, 1 \leq i \leq m\},$$

we can extend it to

$$X^\epsilon = \{x : g_i(x) \leq \epsilon, 1 \leq i \leq m\},$$

thus making the interior nonempty, and minimize the objective within accuracy ϵ on this larger set, seeking for ϵ -optimal **ϵ -feasible** solution instead of ϵ -optimal and *exactly feasible* one.

This is quite natural: to find a feasible solution is, in general, not easier than to find an optimal one. Thus, *either ask for exactly feasible and exactly optimal solution* (which beyond LO is unrealistic), or allow for controlled violation in *both* feasibility and optimality!

♠ **From practical perspective**, theoretical drawbacks of the Ellipsoid method become irrelevant: for all practical purposes, bounds on the magnitude of variables like 10^{100} are the same as no bounds at all, and infeasibility like 10^{-10} is the same as feasibility. And since the bounds on the variables and the infeasibility are under log in the complexity estimate, 10^{100} and 10^{-10} are not a disaster.

♠ **Practical limitations** (rather severe!) of Ellipsoid algorithm stem from method's sensitivity to problem's design dimension n . Theoretically, with ϵ, V, \mathcal{R} fixed, the number of steps grows with n as n^2 , and the effort per step is *at least* $O(n^2)$ a.o.

⇒ *Theoretically, computational effort grows with n at least as $O(n^4)$,*

⇒ *n like 1000 and more is beyond the “practical grasp” of the algorithm.*

Note: *Nearly all modern applications of Convex Optimization deal with n in the range of tens and hundreds of thousands!*

♠ By itself, growth of *theoretical* complexity with n as n^4 is not a big deal: for Simplex method, this growth is exponential rather than polynomial, and nobody dies – in reality, Simplex does *not* work according to its disastrous theoretical complexity bound.

Ellipsoid algorithm, unfortunately, works more or less according to its complexity bound.

⇒ *Practical scope of Ellipsoid algorithm is restricted to convex problems with few tens of variables.*

However: Low-dimensional convex problems from time to time do arise in applications. More importantly, these problems arise “on a permanent basis” as auxiliary problems within some modern algorithms aimed at solving *extremely large-scale* convex problems.

⇒ *The scope of practical applications of Ellipsoid algorithm is nonempty, and within this scope, the algorithm, due to its ability to produce high-accuracy solutions (and surprising stability to rounding errors) can be considered as the method of choice.*

How It Works

$$\text{Opt} = \min_x f(x), X = \{x \in \mathbf{R}^n : a_i^T x - b_i \leq 0, 1 \leq i \leq m\}$$

♠ Real-life problem with $n = 10$ variables and $m = 81,963,927$ “well-organized” linear constraints:

CPU, sec	t	$f(x^t)$	$f(x^t) - \text{Opt} \leq$	$\rho(G_t)/\rho(G_1)$
0.01	1	0.000000	6.7e4	1.0e0
0.53	63	0.000000	6.7e3	4.2e-1
0.60	176	0.000000	6.7e2	8.9e-2
0.61	280	0.000000	6.6e1	1.5e-2
0.63	436	0.000000	6.6e0	2.5e-3
1.17	895	-1.615642	6.3e-1	4.2e-5
1.45	1250	-1.983631	6.1e-2	4.7e-6
1.68	1628	-2.020759	5.9e-3	4.5e-7
1.88	1992	-2.024579	5.9e-4	4.5e-8
2.08	2364	-2.024957	5.9e-5	4.5e-9
2.42	2755	-2.024996	5.7e-6	4.1e-10
2.66	3033	-2.024999	9.4e-7	7.6e-11

♠ Similar problem with $n = 30$ variables and
 $m = 1,462,753,730$ “well-organized” linear constraints:

CPU, sec	t	$f(x^t)$	$f(x^t) - \text{Opt} \leq$	$\rho(G_t)/\rho(G_1)$
0.02	1	0.000000	5.9e5	1.0e0
1.56	649	0.000000	5.9e4	5.0e-1
1.95	2258	0.000000	5.9e3	8.1e-2
2.23	4130	0.000000	5.9e2	8.5e-3
5.28	7080	-19.044887	5.9e1	8.6e-4
10.13	10100	-46.339639	5.7e0	1.1e-4
15.42	13308	-49.683777	5.6e-1	1.1e-5
19.65	16627	-50.034527	5.5e-2	1.0e-6
25.12	19817	-50.071008	5.4e-3	1.1e-7
31.03	23040	-50.074601	5.4e-4	1.1e-8
37.84	26434	-50.074959	5.4e-5	1.0e-9
45.61	29447	-50.074996	5.3e-6	1.2e-10
52.35	31983	-50.074999	1.0e-6	2.0e-11

From Ellipsoid Method to Polynomial Solvability of Convex Programming

♣ Consider a generic Convex Programming problem \mathcal{P} which is *polynomially computable*, of *polynomial growth* and with *polynomially bounded feasible sets*.
In order to solve an instance

$$\min_{x \in X(p)} p_0(x) \quad (p)$$

within accuracy ϵ , we act as follows:

- We rewrite (p) as

$$\min_{x \in X} p_0(x), \quad X = \{x : \|x\|_2 \leq R, \text{Infeas}_{\mathcal{P}}(x, p) \leq \epsilon\} \quad (*)$$

where R is the a priori bound on the size of $X(p)$ given by assumption on *polynomial boundedness of feasible sets*. Note that $X(p) \subset X$;

- The *polynomial computability* assumption allows to equip (*) with First Order and Separation oracles
- Assuming (p) feasible, *polynomial growth* assumption allows to bound from above $\text{Var}_R(p_0)$ and to bound from below the radius $r > 0$ of a ball contained in the feasible set of (*)

♣ We now are in a position to solve (*) by the Ellipsoid method. The complexity bound for the method combines with the bounds on the effort to mimic the First Order and the Separation oracles to yield a *polynomial-time* bound on the complexity of finding ϵ -solution to (p).

Complexity bounds for LP , CQP , SDP

♣ The theorem on polynomial time solvability of Convex Programming is “constructive” – we can explicitly point out the underlying polynomial time solution algorithm (e.g., the Ellipsoid method). However, from the practical viewpoint this is a kind of “existence theorem” – the resulting complexity bounds, although polynomial, are “too large” for practical large-scale computations.

The intrinsic drawback of the Ellipsoid method (and all other “universal” polynomial time methods in Convex Programming) is that the method utilizes just the convex structure of instances and is unable to facilitate our a priori knowledge of the particular analytic structure of these instances.

- In late 80’s, a new family of polynomial time methods for “well-structured” generic convex programs was found – the *Interior Point* methods which indeed are able to facilitate our knowledge of the analytic structure of instances.
- LP , CQP and SDP are especially well-suited for processing by the IP methods, and these methods yield the best known so far theoretical complexity bounds for the indicated generic problems.

- ♣ As far as practical computations are concerned *and high-accuracy solutions are sought*, the IP methods
 - *in the case of Linear Programming, are competitive* (to say the least) *with the Simplex method*
 - *in the case of Conic Quadratic and Semidefinite Programming, are the best known so far numerical techniques.*

V. INTERIOR POINT ALGORITHMS FOR *LP/CQP/SDP*

Preliminaries: The Newton method and the Interior Penalty Scheme

♠ The classical Newton method for unconstrained minimization of a smooth convex function $f : \mathbf{R}^n \rightarrow \mathbf{R} \cup \{+\infty\}$ with an open domain is the linearization scheme for solving the Fermat equation

$$\nabla f(x) = 0. \quad (*)$$

Given current iterate x_t , we linearize $(*)$ at x_t :

$$\nabla f(x) \approx \nabla f(x_t) + \nabla^2 f(x_t)(x - x_t);$$

the next iterate is the solution to the linearized Fermat equation:

$$\nabla f(x_t) + \nabla^2 f(x_t)(x - x_t) = 0$$

\Rightarrow

$$x_{t+1} = x_t - [\nabla^2 f(x_t)]^{-1} \nabla f(x_t) \quad (\text{Nwt})$$

- Assuming that x_* is a nondegenerate minimum of f :

$$\nabla f(x_*) = 0, \quad \nabla^2 f(x_*) \succ 0,$$

the Newton method converges to x_* quadratically, *provided that it is started close enough to x_** :

$$\exists(r > 0, C < \infty) : \|x_t - x_*\|_2 \leq r \Rightarrow \|x_{t+1} - x_*\|_2 \leq C\|x_t - x_*\|_2^2 \leq \frac{1}{2}\|x_t - x_*\|_2.$$

- In order to ensure *global* convergence of the method, one incorporates linesearch, thus coming to the *damped Newton scheme*

$$x_{t+1} = x_t - \gamma_t [\nabla^2 f(x_t)]^{-1} \nabla f(x_t).$$

♠ A Convex Programming program

$$\min_x \{c^T x : x \in X \subset \mathbf{R}^n\} \quad (\text{C})$$

with closed and bounded feasible domain X (int $X \neq \emptyset$) can be represented as a “limiting case” of convex unconstrained problems.

Indeed, introducing an *interior penalty* $F(\cdot) : \text{int } X \rightarrow \mathbf{R}$ such that

- F is smooth and $\nabla^2 F(x) \succ 0$ for $x \in \text{int } X$,
- $F(x_i) \rightarrow \infty$ along every sequence $\{x_i \in \text{int } X\}$ converging to a point $x \in \partial X$,

one can approximate (C) by a “penalized” problem

$$\min_x \left\{ f_t(x) \equiv c^T x + \frac{1}{t} F(x) \right\} \quad (\text{C}_t).$$

- For every $t > 0$, f_t is a smooth convex function with the domain int X , and f_t attains its minimum on the domain at a unique point $x_*(t)$;
- As $t \rightarrow \infty$, the path $x_*(t)$ converges to the solution set of (C).
- In order to solve (C), one can trace the path $x_*(t)$, iterating the updating

$$(a) \quad t_i \mapsto t_{i+1} > t_i$$

$$(b) \quad x_i \mapsto x_{i+1} \text{ “close enough” to } x_*(t_{i+1})$$

Usually, (b) is obtained by minimizing $f_{t_{i+1}}(\cdot)$ with the (damped) Newton method started at x_i .

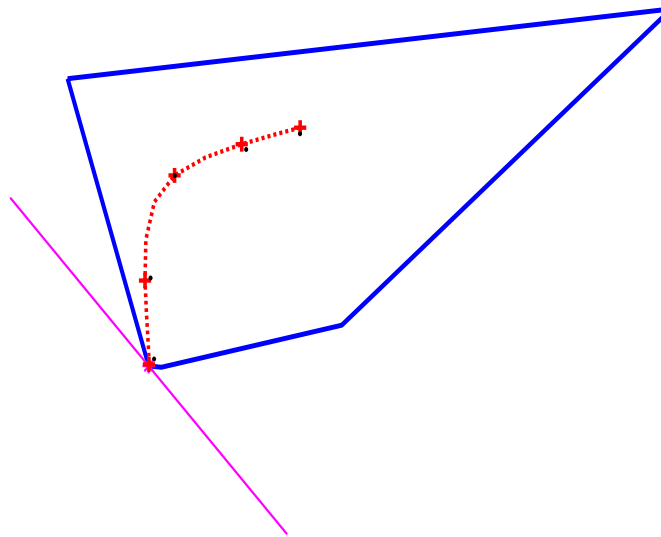
$$\min_x \{c^T x : x \in X\}; \quad F : \text{int } X \rightarrow \mathbf{R}$$

$$\Downarrow$$

$$f_t(x) = c^T x + \frac{1}{t} F(x) \quad x_*(t) = \underset{x}{\operatorname{argmin}} f_t(x)$$

$$\Downarrow$$

$$(a) \quad t_i \mapsto t_{i+1} > t_i \quad (b) \quad x_i \mapsto x_{i+1} - \gamma_i [\nabla^2 f_{t_{i+1}}(x_i)]^{-1} \nabla f_{t_{i+1}}(x_i)$$



- blue polygon: X
- magenta: $\{u : c^T u = \min_X c^T x\}$
- red dots and crosses: path $x_*(t)$ and “targets” $x_*(t_i)$
- black points: iterates x_i

♠ Traditional theory of Newton method predicted slowing the process down as penalty t grows—by this theory, the larger t , the more difficult is to minimize f_t by the Newton method.

However: In 1985-94, it was discovered that

- With an appropriate choice of the interior penalty F , there is no slowing down, and Interior Penalty Scheme admits a *polynomial time* implementation;
- LP, CQP and SDP are especially well-suited for the resulting IP (Interior Point) methods.

IP methods for LP–CQP–SDP: building blocks

♣ We are interested in a generic conic problem

$$\min_x \left\{ c^T x : \mathcal{A}x - B \in \mathbf{K} \right\} \quad (\text{CP})$$

where \mathbf{K} is a *canonical cone* – a direct product of several Semidefinite and Lorentz cones:

$$\mathbf{K} = \mathbf{S}_+^{k_1} \times \dots \times \mathbf{S}_+^{k_p} \times \mathbf{L}^{k_{p+1}} \times \dots \times \mathbf{L}^{k_m} \subset E = \mathbf{S}^{k_1} \times \dots \times \mathbf{S}^{k_p} \times \mathbf{R}^{k_{p+1}} \times \dots \times \mathbf{R}^{k_m}. \quad (\text{Cone})$$

♠ We equip the Semidefinite and the Lorentz cones with *canonical barriers*:

- The canonical barrier for \mathbf{S}_+^k is

$$S_k(X) = -\ln \text{Det}(X) : \text{int } \mathbf{S}_+^k \rightarrow \mathbf{R};$$

the *parameter* of this barrier is $\theta(S_k) = k$.

$$\mathbf{K} = \mathbf{S}_+^{k_1} \times \dots \times \mathbf{S}_+^{k_p} \times \mathbf{L}^{k_{p+1}} \times \dots \times \mathbf{L}^{k_m} \subset E = \mathbf{S}^{k_1} \times \dots \times \mathbf{S}^{k_p} \times \mathbf{R}^{k_{p+1}} \times \dots \times \mathbf{R}^{k_m}$$

$$S_k(X) = -\ln \text{Det}(X) : \text{int } \mathbf{S}_+^k \rightarrow \mathbf{R}, \quad \theta(S_k) = k$$

- The canonical barrier for \mathbf{L}^k is

$$L_k(x) = -\ln(x_k^2 - x_1^2 - \dots - x_{k-1}^2) = -\ln(x^T J_k x),$$

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

the *parameter* of this barrier is $\theta(L_k) = 2$.

- The canonical barrier K for \mathbf{K} is the direct sum of the canonical barriers of the factors:

$$K(X) = S_{k_1}(X_1) + \dots + S_{k_p}(X_p) + L_{k_{p+1}}(X_{p+1}) + \dots + L_{k_m}(X_m),$$

$$X_i \in \begin{cases} \text{int } \mathbf{S}_+^{k_i}, & i \leq p \\ \text{int } \mathbf{L}^{k_i}, & p < i \leq m \end{cases};$$

the *parameter* of this barrier is the sum of parameters of the components:

$$\theta(K) = \theta(S_{k_1}) + \dots + \theta(S_{k_p}) + \theta(L_{k_{p+1}}) + \dots + \theta(L_{k_m}) = \sum_{i=1}^p k_i + 2(m - p).$$

$$\begin{aligned}
\mathbf{K} &= \mathbf{S}_+^{k_1} \times \dots \times \mathbf{S}_+^{k_p} \times \mathbf{L}^{k_{p+1}} \times \dots \times \mathbf{L}^{k_m} \subset E = \mathbf{S}^{k_1} \times \dots \times \mathbf{S}^{k_p} \times \mathbf{R}^{k_{p+1}} \times \dots \times \mathbf{R}^{k_m} \\
K(X) &= -\sum_{i=1}^p \ln \text{Det}(X_i) - \sum_{i=p+1}^m \ln(X_i^T J_i X_i), \quad J_k = \begin{bmatrix} -1 & & & \\ & \ddots & & \\ & & -1 & \\ & & & 1 \end{bmatrix}; \\
\theta(K) &= \sum_{i=1}^p k_i + 2(m-p).
\end{aligned}$$

Elementary properties of canonical barriers:

- [barrier property] $K(\cdot)$ is C^∞ strongly convex function: $\nabla^2 K(\cdot) \succ 0$ on $\text{int } \mathbf{K}$, and

$$X^i \in \text{int } \mathbf{K}, \lim_{i \rightarrow \infty} X^i = X \in \partial \mathbf{K} \Rightarrow K(X^i) \rightarrow \infty, i \rightarrow \infty;$$

- [logarithmic homogeneity]

$$\begin{aligned}
X \in \text{int } \mathbf{K}, t > 0 &\Rightarrow K(tX) = K(X) - \theta(K) \ln t \\
&\Rightarrow \nabla K(tX) = t^{-1} \nabla K(X); \quad \langle \nabla K(X), X \rangle_E = -\theta(K)
\end{aligned}$$

- [self-duality] The mapping $X \mapsto -\nabla K(X)$ is a one-to-one mapping of $\text{int } \mathbf{K}$ onto $\text{int } \mathbf{K}$, and this mapping is self-inverse:

$$X \in \text{int } \mathbf{K}, S = -\nabla K(X) \Leftrightarrow S \in \text{int } \mathbf{K}, X = -\nabla K(S).$$

Central Path

♠ Consider a primal-dual pair of conic problems associated with a canonical cone \mathbf{K} :

$$\begin{array}{l}
 \boxed{\begin{array}{ll}
 \min_x \{c^T x : \mathcal{A}x - B \in \mathbf{K}\} & \text{(CP)} \\
 \max_S \{\langle B, S \rangle_E : \mathcal{A}^*S = c, S \in \mathbf{K}\} & \text{(CD)}
 \end{array}} \quad [\mathcal{A}^* : \langle X, \mathcal{A}x \rangle_E \equiv x^T \mathcal{A}^*X] \\
 \Leftrightarrow \boxed{\begin{array}{ll}
 \min_X \{\langle C, X \rangle_E : X \in (\mathcal{L} - B) \cap \mathbf{K}\} & \text{(P)} \\
 \max_S \{\langle B, S \rangle_E : S \in (\mathcal{L}^\perp + C) \cap \mathbf{K}\} & \text{(D)}
 \end{array}} \quad [\mathcal{L} = \text{Im} \mathcal{A}, C : \mathcal{A}^*C = c]
 \end{array}$$

Note: We assume from now on that $\text{Ker} \mathcal{A} = \{0\}$, implying that reformulation $[(\text{CP}), (\text{CD})] \mapsto [(\text{P}), (\text{D})]$ is possible: the required C does exist.

♣ *In the sequel, we assume that problems (P), (D) are strictly feasible: the primal feasible plane $\mathcal{L} - B$ and the dual feasible plane $\mathcal{L}^\perp + C$ intersect the interior of \mathbf{K} .*

$$\begin{array}{l}
\boxed{\begin{array}{ll} \min_x \{c^T x : \mathcal{A}x - B \in \mathbf{K}\} & \text{(CP)} \\ \max_S \{\langle B, S \rangle_E : \mathcal{A}^* S = c, S \in \mathbf{K}\} & \text{(CD)} \end{array}} \quad [\mathcal{A}^* : \langle X, \mathcal{A}x \rangle_E \equiv x^T \mathcal{A}^* X] \\
\Leftrightarrow \boxed{\begin{array}{ll} \min_X \{\langle C, X \rangle_E : X \in (\mathcal{L} - B) \cap \mathbf{K}\} & \text{(P)} \\ \max_S \{\langle B, S \rangle_E : S \in (\mathcal{L}^\perp + C) \cap \mathbf{K}\} & \text{(D)} \end{array}} \quad [\mathcal{L} = \text{Im} \mathcal{A}, C : \mathcal{A}^* C = c]
\end{array}$$

- The canonical barrier of \mathbf{K} induces the barrier $F(x) = K(\mathcal{A}x - B)$ for the feasible set of (CP), and thus defines the path

$$x_*(t) = \underset{x}{\operatorname{argmin}} \left[c^T x + \frac{1}{t} F(x) \right]$$

which turns out to be well-defined for all $t > 0$.

- The image $X_*(t) = \mathcal{A}x_*(t) - B$ of the path $x_*(t)$ is the path of minimizers of $\langle C, X \rangle + \frac{1}{t} K(X)$ over strictly primal feasible X 's and is fully characterized by the following two properties:

$$\begin{array}{ll}
\diamond & X_*(t) \text{ is strictly primal feasible} \\
\heartsuit & -t^{-1} \nabla K(X_*(t)) \text{ is strictly dual feasible}
\end{array}$$

$$x_*(t) = \operatorname{argmin}_x \left[c^T x + \frac{1}{t} F(x) \right] \Rightarrow X_*(t) = \mathcal{A}x_*(t) - B$$

Claim: $X_*(t)$ is fully characterized by the following two properties:

- | | |
|---|---|
| ◇ | $X_*(t)$ is strictly primal feasible |
| ♡ | $-t^{-1}\nabla K(X_*(t))$ is strictly dual feasible |

Indeed, $X_*(t)$ is the minimizer of the function $\langle C, X \rangle_E + t^{-1}K(X)$ over the set of strictly feasible solutions to (P)

$$\Rightarrow C + t^{-1}\nabla K(X_*(t)) \in \mathcal{L}^\perp \Leftrightarrow [-t^{-1}\nabla K(X_*(t))] \in \mathcal{L}^\perp + C;$$

besides this, $-t^{-1}\nabla K(X_*(t)) \in \operatorname{int} \mathbf{K}$.

$$\min_X \{ \langle C, X \rangle_E : X \in (\mathcal{L} - B) \cap \mathbf{K} \} \quad (\text{P}) \quad \max_S \{ \langle B, S \rangle_E : S \in (\mathcal{L}^\perp + C) \cap \mathbf{K} \} \quad (\text{D})$$

\Rightarrow Primal central path $X_*(t)$: $\begin{cases} (a) & X_*(t) \text{ is strictly primal feasible} \\ (b) & -t^{-1} \nabla K(X_*(t)) \text{ is strictly dual feasible} \end{cases}$

- Due to primal-dual symmetry, the dual problem (D) defines the *dual central path* $S_*(t)$ comprised of minimizers of $-\langle B, S \rangle_E + \frac{1}{t} K(S)$ over strictly dual feasible S 's and fully characterized by the following two properties:

$$\begin{aligned} (c) & \quad S_*(t) \text{ is strictly dual feasible} \\ (d) & \quad -t^{-1} \nabla K(S_*(t)) \text{ is strictly primal feasible} \end{aligned}$$

♣ The paths are closely related:

$$X_*(t) = -t^{-1} \nabla K(S_*(t)); \quad S_*(t) = -t^{-1} \nabla K(X_*(t)).$$

Indeed, setting $S = -t^{-1} \nabla K(X_*(t))$, we see that S is strictly dual feasible by (b), while

$$\begin{aligned} -t^{-1} \nabla K(S) &= -t^{-1} \nabla K(-t^{-1} \nabla K(X_*(t))) = -\nabla K(-\nabla K(X_*(t))) \text{ [by logarithmic homogeneity of } K] \\ &= X_*(t) \text{ [since the mapping } X \mapsto -\nabla K(X) \text{ is self-inverse]} \end{aligned}$$

i.e., $-t^{-1} \nabla K(S)$ is strictly primal feasible. Thus, S satisfies (c), (d), whence

$$S := -t^{-1} \nabla K(X_*(t)) = S_*(t).$$

$$\Rightarrow \boxed{\begin{array}{ll} \min_x \{c^T x : Ax - B \in \mathbf{K}\} & \text{(CP)} \quad \max_S \{\langle B, S \rangle_E : \mathcal{A}^* S = c, S \in \mathbf{K}\} & \text{(CD)} \\ \min_X \{\langle C, X \rangle_E : X \in (\mathcal{L} - B) \cap \mathbf{K}\} & \text{(P)} \quad \max_S \{\langle B, S \rangle_E : S \in (\mathcal{L}^\perp + C) \cap \mathbf{K}\} & \text{(D)} \end{array}}$$

$$\Rightarrow \text{Primal-Dual Central Path } (X_*(t), S_*(t)) : \begin{cases} X_*(t) \text{ is strictly primal feasible} \\ S_*(t) \text{ is strictly dual feasible} \\ X_*(t) = -t^{-1} \nabla K(S_*(t)) \Leftrightarrow S_*(t) = -t^{-1} \nabla K(X_*(t)). \end{cases}$$

♣ *The Duality Gap on the primal-dual central path equals to $\frac{\theta(K)}{t}$. Thus, $X_*(t)$ is $\frac{\theta(K)}{t}$ -primal optimal, and $S_*(t)$ is $\frac{\theta(K)}{t}$ -dual optimal:*

$$\begin{aligned} \text{DualityGap}(X_*(t), S_*(t)) &\equiv [\langle C, X_*(t) \rangle_E - \text{Opt(P)}] + [\text{Opt(D)} - \langle B, S_*(t) \rangle_E] \\ &= \langle S_*(t), X_*(t) \rangle_E = t^{-1} \langle -\nabla K(X_*(t)), X_*(t) \rangle_E \\ &= t^{-1} \theta(K). \end{aligned}$$

♠ *Consequently, our “ideal goal” could be to move along the primal-dual central path, thus staying strictly primal-dual feasible and approaching the primal and the dual optimal sets.*

However: We do not know how to stay on a “curved” path, although can move close to the path.

In a neighbourhood of the central path

$$\boxed{\min_X \{ \langle C, X \rangle_E : X \in (\mathcal{L} - B) \cap \mathbf{K} \} \quad (\text{P}) \quad \max_S \{ \langle B, S \rangle_E : S \in (\mathcal{L}^\perp + C) \cap \mathbf{K} \} \quad (\text{D})}$$

$$\Rightarrow \text{Primal-Dual Central Path } (X_*(t), S_*(t)) : \begin{cases} X_*(t) \text{ is strictly primal feasible} \\ S_*(t) \text{ is strictly dual feasible} \\ X_*(t) = -t^{-1} \nabla K(S_*(t)) \Leftrightarrow S_*(t) = -t^{-1} \nabla K(X_*(t)). \end{cases}$$

♠ Given a triple (t, X, S) , where $t > 0$, X is strictly primal feasible, and S is strictly dual feasible, a good for our purposes measure of closeness of (X, S) to $(X_*(t), S_*(t))$ turns out to be

$$\begin{aligned} \text{dist}(t, X, S) &= \sqrt{\langle [\nabla^2 K(X)]^{-1} [tS + \nabla K(X)], tS + \nabla K(X) \rangle_E} \\ &= \sqrt{\langle [\nabla^2 K(S)]^{-1} [tX + \nabla K(S)], tX + \nabla K(S) \rangle_E}. \end{aligned}$$

• Let \mathcal{N}_r be “ r -neighbourhood” of the primal-dual central path comprised of triples (t, X, S) with $t > 0$ and primal-dual strictly feasible X, S satisfying $\text{dist}(t, X, S) \leq r$. The duality gap in \mathcal{N}_1 is nearly the same as on the central path:

$$(t, X, S) \in \mathcal{N}_1 \Rightarrow \text{DualityGap}(X, S) \leq \frac{2\theta(K)}{t}.$$

♠ Consequently, our “realistic goal” could be to trace the primal-dual central path as $t \rightarrow \infty$, staying in (or periodically entering) the neighbourhood \mathcal{N}_1 of the primal-dual central path.

How to trace the central path?

♠ The central path is given by

Strict primal feasibility: (a) $X \in \mathcal{L} - B$ [$\mathcal{L} = \text{Im}\mathcal{A}$] (b) $X \in \text{int } K$	Strict dual feasibility: (c) $S \in \mathcal{L}^\perp + C$ (d) $S \in \text{int } K$
Augmented complementary slackness: (e) $\underbrace{S + t^{-1} \nabla K(X)}_{G_t(X,S)=0} = 0$	

♠ The most natural way to trace the path is as follows:

Given a current triple t_i, X_i, S_i with strictly primal-dual feasible X_i, S_i , we

- increase the penalty parameter t : $t_i \mapsto t_{i+1} > t_i$;
- linearize at t_{i+1}, X_i, S_i the system of nonlinear equations (e), thus coming to the system of *linear* equations for the (approximate) “corrections” $\Delta X \approx X_*(t_{i+1}) - X_i$, $\Delta S \approx S_*(t_{i+1}) - S_i$:

$$\Delta X \in \mathcal{L}, \Delta S \in \mathcal{L}^\perp, G_{t_{i+1}}(X_i, S_i) + \frac{\partial G_{t_{i+1}}(X_i, S_i)}{\partial X} \Delta X + \frac{\partial G_{t_{i+1}}(X_i, S_i)}{\partial S} \Delta S = 0 \quad (\text{N})$$

- solve (N), thus getting the corrections (“search directions”) $\Delta X_i, \Delta S_i$, and update X_i, S_i according to

$$X_{i+1} = X_i + \alpha_i \Delta X_i, \quad S_{i+1} = S_i + \beta_i \Delta S_i.$$

♠ **Note:** The **A**ugmented **C**omplementary **S**lackness (**ACS**) equation can be written in many equivalent forms:

$$S + t^{-1}\nabla K(X) = 0, \quad X + t^{-1}\nabla K(S) = 0, \dots$$

Different equivalent formulations of ACS equation result in *different* linearizations and thus in *different* path-following schemes.

$$\begin{array}{ll}
\min_x \{c^T x : Ax - B \in \mathbf{K}\} & \text{(CP)} \\
\Leftrightarrow \min_X \{ \langle C, X \rangle_E : X \in [\mathcal{L} - B] \cap \mathbf{K} \} & \text{(P)} \\
[\mathcal{A}^* C = c, \mathcal{L} = \text{Im} \mathcal{A}] &
\end{array}$$

Example: Primal path-following method. Let us use the ACS equation “as it is”:

$$S + t^{-1} \nabla K(X) = 0.$$

Then the system for corrections becomes

$$\begin{aligned}
\Delta X &= \mathcal{A} \Delta x \quad [\Leftrightarrow \Delta X \in \mathcal{L} = \text{Im} \mathcal{A}] \\
\mathcal{A}^* \Delta S &= 0 \quad [\Leftrightarrow \Delta S \in \mathcal{L}^\perp] \\
\Delta S + t_{i+1}^{-1} [\nabla^2 K(X_i)] \Delta X &= -S_i - t_{i+1}^{-1} \nabla K(X_i),
\end{aligned}$$

which is equivalent to

$$\begin{aligned}
\Delta X &= \mathcal{A} \Delta x \\
\Delta S &= -t_{i+1}^{-1} [\nabla^2 K(X_i)] \Delta X - S_i - t_{i+1}^{-1} \nabla K(X_i), \\
t_{i+1}^{-1} \mathcal{A}^* [\nabla^2 K(X_i)] \mathcal{A} \Delta x &= -\underbrace{\mathcal{A}^* S_i}_c - t_{i+1}^{-1} \mathcal{A}^* \nabla K(X_i).
\end{aligned}$$

\Leftrightarrow	$\min_x \{c^T x : \mathcal{A}x - B \in \mathbf{K}\} \quad (\text{CP})$ $\min_X \{ \langle C, X \rangle_E : X \in [\mathcal{L} - B] \cap \mathbf{K} \} \quad (\text{P})$ $[\mathcal{A}^*C = c, \mathcal{L} = \text{Im}\mathcal{A}]$
	$\Delta X = \mathcal{A}\Delta x$ $\Delta S = -t_{i+1}^{-1}[\nabla^2 K(X_i)]\Delta X - S_i - t_{i+1}^{-1}\nabla K(X_i),$ $t_{i+1}^{-1}\mathcal{A}^*[\nabla^2 K(X_i)]\mathcal{A}\Delta x = -\underbrace{\mathcal{A}^*S_i}_c - t_{i+1}^{-1}\mathcal{A}^*\nabla K(X_i).$

Setting

$$F(x) = K(\mathcal{A}x - B),$$

the method becomes

$t_i \mapsto t_{i+1} > t_i,$ $x_{i+1} = x_i - [\nabla^2 F(x_i)]^{-1}[t_{i+1}c + \nabla F(x_i)],$ $X_{i+1} = \mathcal{A}x_{i+1} - B,$ $S_{i+1} = \dots$
--

which is *exactly* the classical Interior Penalty Scheme for tracing the path

$$x_*(t) = \operatorname{argmin}_x [c^T x + t^{-1}F(x)] = \operatorname{argmin}_x [tc^T x + F(x)].$$

$$\begin{array}{lcl}
& \min_x \{c^T x : \mathcal{A}x - B \in \mathbf{K}\} & \text{(CP)} \\
\Rightarrow & \begin{array}{l} t_i \mapsto t_{i+1} > t_i, \\ x_{i+1} = x_i - [\nabla^2 F(x_i)]^{-1} [t_{i+1}c + \nabla F(x_i)], \\ F(x) = K(\mathcal{A}x - B); \\ \hline X_{i+1} = \mathcal{A}x_{i+1} - B, \\ S_{i+1} = \dots \end{array} & \text{(PF)}
\end{array}$$

Theorem. Let the starting point (t_0, X_0, S_0) in the Primal Path-Following method belong to the neighbourhood $\mathcal{N}_{0.1}$ of the central path, i.e.,

- $t_0 > 0$, X_0 is strictly primal feasible, S_0 is strictly dual feasible;

- $\sqrt{\langle [\nabla^2 K(X_0)]^{-1} [t_0 S_0 + \nabla K(X_0)], t_0 S_0 + \nabla K(X_0) \rangle_E} \leq 0.1$.

With the penalty updating rule

$$t_{i+1} = \left(1 + \frac{0.1}{\sqrt{\theta(K)}} \right) t_i,$$

the Primal Path-Following method is well-defined and keeps all iterates in $\mathcal{N}_{0.1}$. In particular, it takes no more than

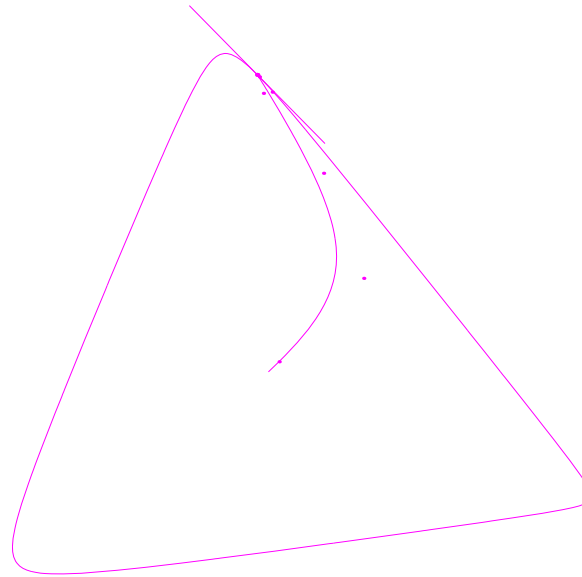
$$O(1) \sqrt{\theta(K)} \ln \left(2 + \frac{\theta(K)}{t_0 \epsilon} \right)$$

steps of (PF) to get a feasible ϵ -solution of (CP).

- ♠ Theorem implies the best known so far polynomial time complexity bounds for LP, CQP and SDP.
- ♠ Writing the Augmented Complementarity Slackness equation in the “symmetric” form

$$X + t^{-1} \nabla K(S) = 0,$$

one arrives at the *Dual* Path-Following method with exactly the same theoretical properties as the Primal method.



2D feasible set of a toy SDP ($\mathbf{K} = \mathbf{S}_+^3$).
 “Continuous curve” is the primal central path
 Dots are iterates x_i of the Primal Path-Following method.

Itr#	Objective	DualityGap	Itr#	Objective	DualityGap
1	-0.100000	2.96	7	-1.359870	8.4e-4
2	-0.906963	0.51	8	-1.360259	2.1e-4
3	-1.212689	0.19	9	-1.360374	5.3e-5
4	-1.301082	6.9e-2	10	-1.360397	1.4e-5
5	-1.349584	2.1e-2	11	-1.360404	3.8e-6
6	-1.356463	4.7e-3	12	-1.360406	9.5e-7

Semidefinite Case

♠ In spite of being “theoretically perfect”, Primal and Dual Path-Following methods in practice are inferior as compared with the methods based on “less straightforward” forms of the ACS equation. Let us look at these “more advanced” methods *in the SDP case*:

$$\mathbf{K} = \mathbf{S}_+^k \subset E = \mathbf{S}^k, \quad K(X) = -\ln \text{Det}(X).$$

In this case,

- $\nabla K(X) = -X^{-1}$, $[\nabla^2 K(X)]H = X^{-1}HX^{-1}$:

$$\left. \frac{d}{dt} \right|_{t=0} K(X + tH) = \text{Tr}(-X^{-1}H), \quad \left. \frac{d^2}{dt^2} \right|_{t=0} K(X + tH) = \text{Tr}(HX^{-1}HX^{-1}).$$

- The ACS equation reads

$$S = t^{-1}X^{-1} \Leftrightarrow SX = t^{-1}I. \quad (*)$$

♠ An important class of equivalent representations of (*) is as follows: given a “scaling matrix” $Q \succ 0$, one can rewrite (*) in two equivalent forms:

$$Q^{-1}SXQ = t^{-1}I, \quad QXSQ^{-1} = t^{-1}I,$$

whence also

$$QXSQ^{-1} + Q^{-1}SXQ = 2t^{-1}I; \quad (**)$$

in fact, (*) and (**) regarded as nonlinear equations with *positive definite unknowns* X, S are equivalent to each other.

$$QXSQ^{-1} + Q^{-1}SXQ = 2t^{-1}I \quad (**)$$

Explanation: Let $Q \in \mathbf{S}^k$ be nonsingular. The *Q-scaling*

$$X \mapsto QXQ$$

is a one-to-one linear mapping of \mathbf{S}^k onto itself, the inverse being the mapping

$$X \mapsto Q^{-1}XQ^{-1}.$$

Q-scaling is a symmetry of the positive semidefinite cone – it maps the cone onto itself.

⇒ Given a primal-dual pair of semidefinite programs

$$\text{Opt}(\mathcal{P}) = \min_X \{ \text{Tr}(CX) : X \in [\mathcal{L} - B] \cap \mathbf{S}_+^k \} \quad (\mathcal{P})$$

$$\text{Opt}(\mathcal{D}) = \max_S \{ \text{Tr}(BS) : S \in [\mathcal{L}^\perp + C] \cap \mathbf{S}_+^k \} \quad (\mathcal{D})$$

and a nonsingular matrix $Q \in \mathbf{S}^k$, one can pass in (\mathcal{P}) from variable X to variable $\hat{X} = QXQ$, and in (\mathcal{D}) from variable S to variable $\tilde{S} = Q^{-1}SQ^{-1}$. The resulting problems are

$$\begin{aligned} \text{Opt}(\mathcal{P}) = \min_{\hat{X}} \{ \text{Tr}(\tilde{C}\hat{X}) : \hat{X} \in [\hat{\mathcal{L}} - \hat{B}] \cap \mathbf{S}_+^k \} \quad (\hat{\mathcal{P}}) \quad & \text{Opt}(\mathcal{D}) = \max_{\tilde{S}} \{ \text{Tr}(\hat{B}\tilde{S}) : \tilde{S} \in [\tilde{\mathcal{L}}^\perp + \tilde{C}] \cap \mathbf{S}_+^k \} \quad (\tilde{\mathcal{D}}) \\ & \left[\hat{B} = QBQ, \hat{\mathcal{L}} = \{QXQ : X \in \mathcal{L}\}, \tilde{C} = Q^{-1}CQ^{-1}, \tilde{\mathcal{L}}^\perp = \{Q^{-1}SQ^{-1} : S \in \mathcal{L}^\perp\} \right] \end{aligned}$$

♠ $\hat{\mathcal{P}}$ and $\tilde{\mathcal{D}}$ are dual to each other, the primal-dual central path of this pair is the image of the primal-dual path of $(\mathcal{P}), (\mathcal{D})$ under the *primal-dual Q-scaling*

$$(X, S) \mapsto (\hat{X} = QXQ, \tilde{S} = Q^{-1}SQ^{-1})$$

Q preserves closeness to the path, etc.

♠ Writing down the ACS equation as

$$QXSQ^{-1} + Q^{-1}SXQ = 2t^{-1}I \quad (!)$$

we in fact

- pass from (\mathcal{P}) , (\mathcal{D}) to the equivalent primal-dual pair of problems $(\widehat{\mathcal{P}})$, $(\widetilde{\mathcal{D}})$
- write down the ACS equation for the latter pair in the simplest primal-dual symmetric form

$$\widehat{X}\widetilde{S} + \widetilde{S}\widehat{X} = 2t^{-1}I,$$

- “scale back” to the original primal-dual variables X, S , thus arriving at (!).

$$\boxed{QXSQ^{-1} + Q^{-1}SXQ = 2t^{-1}I} \quad (**)$$

- With the ACS equation written in the form of (**), one can use iteration-dependent scaling matrices Q_i . The system defining the search directions at i -th iteration becomes

$$\begin{aligned} & \Delta X \in \mathcal{L}, \quad \Delta S \in \mathcal{L}^\perp, \\ & Q_i[\Delta XS_i + X_i\Delta S]Q_i^{-1} + Q_i^{-1}[S_i\Delta X + \Delta SX_i]Q_i = 2t_{i+1}^{-1}I - Q_iX_iS_iQ_i^{-1} - Q_i^{-1}S_iX_iQ_i \end{aligned}$$

♠ Popular choices of the scaling matrices Q_i are:

- $Q_i = I$ [Alizadeh-Haeberly-Overton method]
- $Q_i = S_i^{1/2}$ [the XS -method]
- $Q_i = X_i^{-1/2}$ [the SX -method]
- $Q_i = \left(X_i^{-1/2} (X_i^{1/2} S_i X_i^{1/2})^{-1/2} X_i^{1/2} S_i \right)^{1/2}$ [Nesterov-Todd method]

Note: The XS -, the SX -, and the NT-method are based on *commutative scalings*, where the matrices

$$\widehat{X}_i = Q_i X_i Q_i, \quad \widetilde{S}_i = Q_i^{-1} S_i Q_i^{-1}$$

commute with each other. Specifically,

- in the XS -method, $\widetilde{S} = I$
- in the SX -method, $\widehat{X} = I$,
- in the NT-method, $\widetilde{S} = \widehat{X}$.

$$\begin{array}{ll} \min_X \left\{ \text{Tr}(CX) : X \in (\mathcal{L} - B) \cap \mathbf{S}_+^k \right\} & \text{(P)} \\ \max_S \left\{ \text{Tr}(BS) : S \in (\mathcal{L}^\perp + C) \cap \mathbf{S}_+^k \right\} & \text{(D)} \end{array}$$

♣ **Theorem.** Let a primal-dual pair (P), (D) of strictly feasible semidefinite programs be solved by a primal-dual path-following method *based on commutative scalings*, and let the penalty updating policy in the method be

$$t_{i+1} = \left(1 + \frac{0.1}{\sqrt{k}} \right) t_i. \quad (U)$$

Assume that the starting triple (t_0, X_0, S_0) is such that

- X_0 is strictly primal feasible, S_0 is strictly dual feasible, $t_0 = k^{-1} \text{Tr}(X_0 S_0)$;
- The triple (t_0, X_0, S_0) is close to the central path:

$$\begin{aligned} \text{dist}(t_0, X_0, S_0) &:= \sqrt{\langle [\nabla^2 K(X_0)]^{-1} [t_0 S_0 + \nabla K(X_0)], t_0 S_0 + \nabla K(X_0) \rangle_E} \\ &\equiv \sqrt{\text{Tr}([t_0 X_0^{1/2} S_0 X_0^{1/2} - I]^2)} \leq 0.1. \end{aligned}$$

Then the method is well-defined and keeps all iterates in $\mathcal{N}_{0.1}$. In particular, it takes no more than

$$O(1) \sqrt{k} \ln \left(2 + \frac{k}{t_0 \epsilon} \right)$$

steps of the method to build feasible ϵ -solutions of (P), (D).

- ♠ To improve the practical performance of primal-dual path-following methods, in actual computations
 - the penalty parameter is updated in a “more aggressive,” as compared to (U), *on line adjusted* fashion;
 - the primal-dual methods are allowed to travel in “much wider,” as compared to $\mathcal{N}_{0.1}$, neighbourhoods of the central path.
- ♠ The constructions and the complexity results we have presented are “incomplete” in the sense that they do not take into account the necessity to come close to the central path before starting path-tracing and do not take care of the case when the pair (P), (D) is not strictly feasible. All these “gaps” can be easily closed via the same path-following technique as applied to appropriate augmented versions of the original problem.

Complexity bounds for \mathcal{LP}_b

♣ A program from \mathcal{LP}_b :

$$(p) : \min_x \left\{ c^T x : Ax \geq b, \|x\|_\infty \leq R \right\} \quad [A \in \mathbf{M}^{m,n}]$$

can be solved within accuracy ϵ in

$$\mathcal{N}_{\mathcal{LP}} = O(1)\sqrt{m} \ln \left(\frac{\|\text{Data}(p)\|_1 + \epsilon^2}{\epsilon} \right)$$

iterations.

The computational effort per iteration is dominated by the necessity, given a positive definite diagonal matrix Δ and a vector r , to assemble the matrix of the system of linear equations

$$[A; I; -I]^T \Delta [A; I; -I] x = h$$

and to solve this system.

- In the case $m = O(n)$, the overall complexity of solving (p) within accuracy ϵ is cubic in n :

$$O(1)mn^2 \ln \left(\frac{\|\text{Data}(p)\|_1 + \epsilon^2}{\epsilon} \right)$$

Complexity bounds for \mathcal{CQP}_b

♣ A program from \mathcal{CQP}_b :

$$(p) : \quad \left\{ c^T x : \|D_i x - d_i\|_2 \leq e_i^T x - c_i, \ i = 1, \dots, k; \|x\|_2 \leq R \right\}$$

can be solved within accuracy ϵ in

$$\mathcal{N}_{\mathcal{CQP}} = O(1)\sqrt{k} \ln \left(\frac{\|\text{Data}(p)\|_1 + \epsilon^2}{\epsilon} \right)$$

iterations.

The computational effort per iteration is dominated by the necessity, given vectors δ_i , $i = 1, \dots, k$ and a vector r , to assemble the matrices

$$H_i = D_i^T (I - \delta_i \delta_i^T) D_i, \ i = 1, \dots, k$$

and to solve a $\dim x \times \dim x$ linear system

$$Hu = r$$

with positive definite matrix H “readily given” by H_1, \dots, H_k .

Complexity bounds for \mathcal{SDP}_b

♣ A program from \mathcal{SDP}_b :

$$(p) : \min_x \left\{ c^T x : \mathcal{A}(x) = \sum_{i=1}^n x_i A_i - B \succeq 0, \|x\|_2 \leq R \right\}$$

can be solved within accuracy ϵ in

$$\mathcal{N}_{\mathcal{SDP}} = O(1) \sqrt{\mu} \ln \left(\frac{\|\text{Data}(p)\|_1 + \epsilon^2}{\epsilon} \right)$$

iterations, where μ is the row size of matrices A_1, \dots, A_n .

The computational effort per iteration is dominated by the necessity, given a positive definite matrix X of the same size and block-diagonal structure as those of A_i and a vector r s

- to compute $n \times n$ symmetric matrix \hat{H} with entries

$$\hat{H}_{ij} = \text{Tr}(X^{-1} A_i X^{-1} A_j), \quad i, j = 1, \dots, n;$$

- to solve $n \times n$ linear system

$$Hu = r$$

with positive definite matrix H “readily given” by \hat{H} .

VI. FIRST ORDER METHODS

Simple methods for extremely large-scale problems

♣ The arithmetic complexity of a step in *all* known polynomial time methods for Convex Programming grows up *nonlinearly* with the design dimension n of the problem – at least as $O(n^2)$, if not as $O(n^3)$ (the only exception are extremely sparse real-world LPs with favourable sparsity patterns).

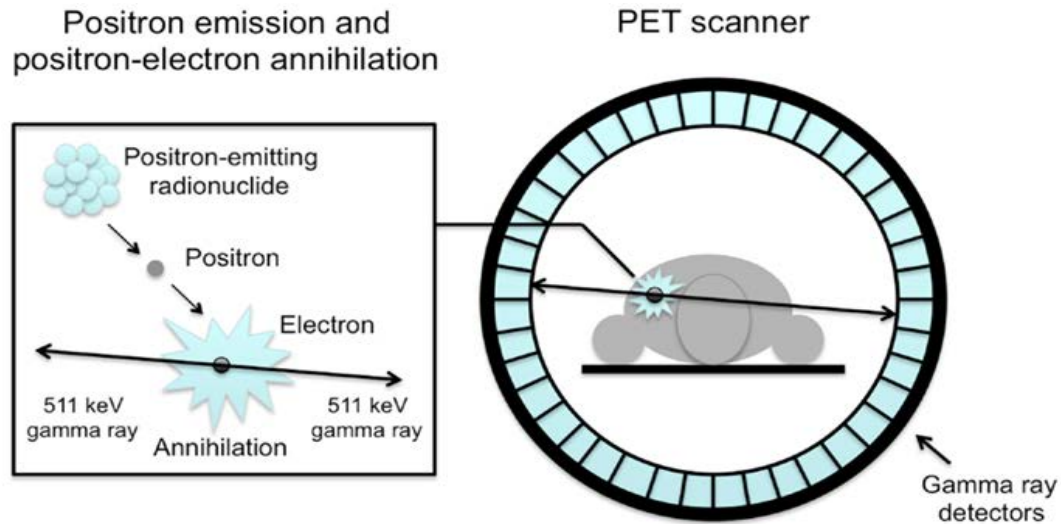
What to do when the design dimension is of order of tens and hundreds of thousands, and the problem is not a “very sparse LP”?

Nonlinear convex problems of huge design dimension do arise in numerous applications, e.g., in

- SDP relaxations of large combinatorial problems,
- Structural Design (especially for 3D structures),
- Signal Processing, High-dimensional Statistics, Machine Learning
- *3D Medical imaging problems*

Example of Medical Imaging problem: PET Image Reconstruction

♣ **PET** (Positron Emission Tomography) is a powerful, non-invasive, medical diagnostic imaging technique for measuring the metabolic activity of cells in the human body. It has been in clinical use since the early 1990s. PET imaging is unique in that it shows the *chemical functioning* of organs and tissues, while other imaging techniques - such as X-ray, computerized tomography (CT) and magnetic resonance imaging (MRI) - show *anatomic structures*.



♣ **Physics of PET.** A PET scan uses *radioactive tracer* – a biologically active fluid with a radio-active component capable of emitting positrons. When administered to a patient, the tracer distributes within the body and, with properly chosen biologically active “carrier”, concentrates in desired locations, e.g., in the areas of high metabolic activity where cancer tumors can be expected.

- The tracer disintegrates, emitting positrons.
- A positron immediately annihilates with a near-by electron, giving rise to two photons flying at the speed of light off the point of annihilation in nearly opposite directions. They are registered outside the patient by cylindrical PET scanner consisting of several rings of detectors.
- When two detectors “simultaneously” (within $\sim 10^{-8}$ sec time window) are hit by photons, this event is registered, indicating that somewhere on the line linking the detectors (**LOR – “Line of Response”**) a disintegration act took place.

- The measured data is the collection of numbers of LOR's counted by different pairs of detectors ("bins"), and the problem is to recover from these measurements the 3D density of the tracer.

♣ Mathematically, the PET Image Reconstruction problem, after appropriate discretization, becomes the problem of recovering a vector $\lambda \geq 0$ from a noisy observation y of the vector $P\lambda$:

$$\lambda \mapsto y = P\lambda + \text{noise} \quad ? \mapsto ? \quad \text{estimate of } \lambda.$$

Specifically,

- entries of λ are indexed by voxels – small cubes into which we partition the field of view; λ_j is the average density of the tracer in voxel j ;
- entries of y are indexed by bins (pairs of detectors); y_i is the number of LORs registered by bin i ;
- $P = [p_{ij}]$ is a given matrix; p_{ij} is the probability for a LOR originating in voxel j to be registered by bin i .

Statistical model of PET states that the entries y_i in y are realizations of independent Poisson random variables with the expectations $(P\lambda)_i$.

♡ In the PET Reconstruction problem, we are interested, given observations y , to find the Maximum Likelihood estimate λ_* of tracer's density:

$$\lambda_* = \underset{\lambda \geq 0}{\operatorname{argmin}} \left[\sum_{j=1}^n p_j \lambda_j - \sum_{i=1}^m y_i \ln \left(\sum_j p_{ij} \lambda_j \right) \right] \quad [p_j = \sum_i p_{ij}] \quad (\text{PET})$$

(PET) is a nicely structured constrained convex program; the only difficulty – a true one! – is in huge sizes of (PET): for problems of actual interest,

- the design dimension n varies from 300,000 to 3,000,000
- the number m of log-terms in the objective varies from 6,000,000 to 25,000,000

♣ As far as nonlinear programs are concerned, design dimension $n \sim 10^4 - 10^5 - 10^6$ makes it necessary to use “cheap” algorithms – those with nearly linear in n arithmetic cost of a step (otherwise you never will finish the very first iteration). This requirement rules out all “advanced” polynomial time optimization techniques and leaves us with, essentially, just two options:

I. Traditional tools of *smooth unconstrained* minimization: gradient descent, conjugate gradients, quasi-Newton methods, etc.

II. Simple subgradient-type techniques for solving *convex nonsmooth constrained* optimization problems:
subgradient descent, restricted memory bundle methods, etc.

- We are interested in extremely large-scale constrained convex problems, and thus intend to focus on cheap subgradient-type techniques. The question of primary importance here is:
(?) *What are the limits of performance of cheap optimization techniques?*
- When answering (?), we shall restrict ourselves with the *black-box-represented* convex programs. As a matter of fact, this is exactly the “working environment” for cheap optimization algorithms.

Black-box-represented convex programs and Information-based complexity

♣ Let us fix a family $\mathcal{P}(X)$ of convex programs

$$\min_x \{f(x) : x \in X\}; \quad (\text{CP})$$

where $X \subset \mathbf{R}^n$ is a given *instance-independent* convex compact set, and $f : \mathbf{R}^n \rightarrow \mathbf{R}$ is convex.

• Formally, $\mathcal{P}(X)$ is some family of convex objectives $f : X \rightarrow \mathbf{R}$.

$$\min_x \{f(x) : x \in X\}; \quad (\text{CP})$$

♣ A *black-box-oriented* solution method \mathcal{B} for $\mathcal{P}(X)$ is as follows:

- When starting to solve (CP), \mathcal{B} is given an accuracy $\epsilon > 0$, knows what is X , and knows that f belongs to a given family $\mathcal{P}(X)$. However, \mathcal{B} does *not* know in advance what is the particular f it deals with and must “learn” f to solve the problem.
- When solving the problem, \mathcal{B} has access to the First Order oracle for f . Given on input $x \in \mathbf{R}^n$, the oracle returns $f(x)$ and a subgradient $f'(x)$ of f at x . \mathcal{B} generates a sequence of *search points* x_1, x_2, \dots and calls the First Order oracle to get values and subgradients of f at these points. The rules for building x_t can be arbitrary, *except for the fact that they should be non-anticipative*: x_t can depend only on the information $f(x_1), f'(x_1), \dots, f(x_{t-1}), f'(x_{t-1})$ on f accumulated by \mathcal{B} at the first $t - 1$ steps.
- After a number $T = T_{\mathcal{B}}(f, \epsilon)$ of calls to the oracle, \mathcal{B} terminates and outputs a result $z_{\mathcal{B}}(f, \epsilon)$ which *should depend solely on the information on f accumulated by \mathcal{B} at the T search steps*, and *must be an ϵ -solution to (CP)*:

$$z_{\mathcal{B}}(f, \epsilon) \in X \ \& \ f(z_{\mathcal{B}}(f, \epsilon)) - \min_X f \leq \epsilon.$$

♣ The *complexity* of $\mathcal{P}(X)$ w.r.t. a solution method \mathcal{B} is

$$\text{Compl}_{\mathcal{B}}(\epsilon) = \max_{f \in \mathcal{P}(X)} T_{\mathcal{B}}(f, \epsilon)$$

which is the minimal number of steps sufficient for \mathcal{B} to solve within accuracy ϵ every instance of $\mathcal{P}(X)$.

♣ The *Information-based complexity* of a family $\mathcal{P}(X)$ of problems is

$$\text{Compl}(\epsilon) = \min_{\mathcal{B}} \text{Compl}_{\mathcal{B}}(\epsilon),$$

the minimum being taken over all solution methods. Relation

$$\text{Compl}(\epsilon) = N$$

means that

- there exists a solution method \mathcal{B} capable to solve within accuracy ϵ every instance of $\mathcal{P}(X)$ in no more than N calls to the First Order oracle;
- for every solution method \mathcal{B} , there exists an instance of $\mathcal{P}(X)$ such that \mathcal{B} solves the instance within the accuracy ϵ in at least N steps.

♣ The information-based complexity $\text{Compl}(\epsilon)$ of a family $\mathcal{P}(X)$ is a *lower bound* on “actual” computational effort, whatever it means, sufficient to find ϵ -solution to every instance of the family.

Main results on Information-based complexity of Convex Programming

♣ Let

$X \subset \mathbf{R}^n$ – a convex compact set, $\text{int } X \neq \emptyset$

$$\mathcal{P}(X) = \left\{ \left\{ \min_{x \in X} f(x) \right\} : f \text{ is convex on } \mathbf{R}^n \text{ and is normalized by } \max_X f - \min_X f \leq 1. \right\}$$

For the family $\mathcal{P}(X)$,

I. Complexity of finding high-accuracy solutions in fixed dimension is independent of the geometry of X . Specifically,

$$\begin{aligned} \forall(\epsilon \leq \epsilon(X)) : \quad & O(1)n \ln \left(2 + \frac{1}{\epsilon} \right) \leq \text{Compl}(\epsilon); \\ \forall(\epsilon > 0) : \quad & \text{Compl}(\epsilon) \leq O(1)n \ln \left(2 + \frac{1}{\epsilon} \right), \end{aligned}$$

where

$O(1)$ are appropriately chosen positive absolute constants,
 $\epsilon(X)$ depends on the geometry of X , but never is less than $\frac{1}{n^2}$.

$$X \subset \mathbf{R}^n - \text{a convex compact set, } \text{int } X \neq \emptyset$$

$$\mathcal{P}(X) = \left\{ \{\min_{x \in X} f(x)\} : f \text{ is convex on } \mathbf{R}^n \text{ and normalized by } \max_X f - \min_X f \leq 1. \right\}$$

II. Complexity of finding solutions of fixed accuracy in high dimensions does depend on the geometry of X . Here are 3 typical results:

Let $X = \{x \in \mathbf{R}^n : \|x\|_\infty \leq 1\}$. Then

$$\epsilon \leq \frac{1}{2} \Rightarrow O(1)n \ln\left(\frac{1}{\epsilon}\right) \leq \text{Compl}(\epsilon) \leq O(1)n \ln\left(\frac{1}{\epsilon}\right). \quad (\|\cdot\|_\infty\text{-Ball})$$

Let $X = \{x \in \mathbf{R}^n : \|x\|_2 \leq 1\}$. Then

$$n \geq \frac{1}{\epsilon^2} \Rightarrow \frac{O(1)}{\epsilon^2} \leq \text{Compl}(\epsilon) \leq \frac{O(1)}{\epsilon^2}. \quad (\|\cdot\|_2\text{-Ball})$$

Let $X = \{x \in \mathbf{R}^n : \|x\|_1 \leq 1\}$. Then

$$n \geq \frac{1}{\epsilon^2} \Rightarrow \frac{O(1)}{\epsilon^2} \leq \text{Compl}(\epsilon) \leq \frac{O(\ln n)}{\epsilon^2}. \quad (\|\cdot\|_1\text{-Ball})$$

$(O(1))$ in the lower bound can be replaced with $O(\ln n)$, provided that $n \gg \frac{1}{\epsilon^2}$.

$$\boxed{\text{Compl}(\epsilon) \geq O(1)n \ln(2 + 1/\epsilon) \quad \forall(\epsilon \leq \epsilon(X))} \quad (\text{I})$$

$$\boxed{X = \{x \in \mathbf{R}^n : \|x\|_2 \leq 1\} \Rightarrow \text{Compl}(\epsilon) \leq \frac{O(1)}{\epsilon^2} \quad \forall(\epsilon > 0) :} \quad (\text{II})$$

♣ Consequences for large-scale convex minimization:

Bad news: I says that we have no hope to guarantee high-accuracy solutions (like $\epsilon = 10^{-6}$) when solving large-scale problems with black-box-oriented methods: it would require at least $O(n)$ calls to the first order oracle with at least $O(n)$ a.o. per call, i.e., totally at least $O(n^2)$ a.o. (with known methods – even $O(n^4)$ a.o.), which is too much for large n ...

Good news: II says that there exist cases when medium accuracy solutions can be found in (nearly) dimension-independent number of oracle calls...

♣ **Good news:** There exist cases when medium accuracy solutions of convex programs

$$\min_{x \in X} f(x), \quad \max_X f - \min_X f \leq 1 \quad (*)$$

can be found in (nearly) dimension-independent number of oracle calls, e.g., the cases of

$$X = B_n^2 \equiv \{x \in \mathbf{R}^n : \|x\|_2 \leq 1\} \quad (\|\cdot\|_2\text{-Ball})$$

or

$$X = B_n^1 \equiv \{x \in \mathbf{R}^n : \|x\|_1 \leq 1\} \quad (\|\cdot\|_1\text{-Ball})$$

(but, unfortunately, *not* the case when X is a box).

$$\min_{x \in X} f(x), \quad \max_X f - \min_X f \leq 1 \quad (*)$$

♣ Problems of minimizing over a $\|\cdot\|_p$ -ball, $p = 1, 2$, are not that typical. Fortunately, the corresponding (nearly) dimension-independent complexity bounds remain valid when X in $(*)$ is a subset of a “good” set B_n^p , $p = 1, 2$, *and* the normalization condition on f in $(*)$ is strengthened to

$$|f(x) - f(y)| \leq \|x - y\|_p \quad \forall x, y \in X.$$

In particular, $O(\frac{\ln n}{\epsilon^2})$ oracle calls are sufficient to minimize, within accuracy ϵ , a convex function f over the *standard simplex*

$$\Delta_n = \{x \in \mathbf{R}^n : x \geq 0, \sum_i x_i = 1\},$$

provided that f is Lipschitz continuous, with constant 1, w.r.t. $\|\cdot\|_1$ (i.e., that the magnitudes of all first order partial derivatives of f are ≤ 1).

♣ **More good news:** The nearly dimension independent complexity bounds for minimization over ball and simplex are given by cheap minimization methods!

Where the lower complexity bounds come from? (cases of ball and box)

♣ Let $2 \leq p \leq \infty$ and $X = \{x : \|x\|_p \leq 1\}$. Consider the families of convex functions

$$\mathcal{F}_k = \{f(x) \equiv \max_{1 \leq i \leq k} [\epsilon_i x_i + \delta_i]\} \quad [k \leq n]$$

given by all 2^k collections $\epsilon_i = \pm 1$ and all collections $\{\delta_i\}_{i=1}^k$ with $0 \leq \delta_i \leq \frac{1}{2k^{1/p}}$.

Observe that when $f \in \mathcal{F}_k$, the variation of f on X does not exceed 2, and the $\|\cdot\|_\infty$ -Lipschitz constant of f does not exceed 1.

We claim that

(!) For every $k \leq n$, the $\frac{1}{4k^{1/p}}$ -complexity of the class of problems $\min_{x \in X} f(x)$ is at least $k - 1$

whence, of course,

(!!) For $0 < \epsilon < \frac{1}{4}$, the ϵ -complexity of the class of optimization problems $\min_X f(x)$ with Lipschitz continuous, with constant 1 w.r.t. $\|\cdot\|_\infty$, objectives f is at least $\min[n, \lfloor \frac{1}{4\epsilon} \rfloor^p] - 1$.

♠ We should prove that if \mathcal{B} is a method for solving problems

$$\min_{x \in X} f_{\epsilon, \delta}(x) = \max_{1 \leq i \leq k} [\epsilon_i x_i + \delta_i] \quad [X = \{x \in \mathbf{R}^n : \|x\|_p \leq 1\}]$$

which, as applied to every problem of this type, terminates after at most $k - 1$ steps, then the accuracy to which the method solves at least one problem from the family is worse than $\epsilon \equiv \frac{1}{2k^{1/p}}$.

We lose nothing when assuming that \mathcal{B} , as applied to every problem from the family, performs exactly k steps, and the approximate solution is the last – the k -th – search point.

♣ Let us associate with \mathcal{B} the following construction:

First step. Let

- x^1 be the first search point generated by \mathcal{B} (this point depends solely on \mathcal{B}),
- i_1 be the index of the largest in absolute value coordinate of x^1 ,
- $\epsilon_{i_1}^* = \pm 1$ be such that $\epsilon_{i_1}^* x_{i_1}^1 = |x_{i_1}^1|$
- $\delta_{i_1}^* = \frac{1}{2k^{1/p}}$

We set

$$\mathcal{F}^1 = \left\{ f(x) = \max_{1 \leq i \leq k} [\epsilon_i x_i + \delta_i] : \begin{array}{l} |\epsilon_i| = 1, \epsilon_{i_1} = \epsilon_{i_1}^*, \\ \delta_{i_1} = \delta_{i_1}^* > \max_{i \neq i_1} \delta_i \geq 0 \end{array} \right\}$$

Note: All functions from \mathcal{F}^1 coincide with each other in a neighbourhood of x^1 , so that the Oracle, being asked at x^1 about every one of the objectives from \mathcal{F}^1 , reports the same.

Step $\ell + 1$, $1 \leq \ell < k$. At the beginning of ℓ -th step, we have ℓ points x^1, \dots, x^ℓ and a set of objectives

$$\mathcal{F}^\ell = \left\{ f(x) = \max_{1 \leq i \leq k} [\epsilon_i x_i + \delta_i] : \begin{array}{l} |\epsilon_i| = 1, i = 1, \dots, k \\ \epsilon_{i_s} = \epsilon_{i_s}^*, s = 1, \dots, \ell \\ \delta_{i_s} = \delta_{i_s}^*, s = 1, \dots, \ell \\ \delta_{i_1}^* > \dots > \delta_{i_\ell}^* > \max_{i \notin \{i_1, \dots, i_\ell\}} \delta_i \geq 0 \end{array} \right\}$$

such that

(A_ℓ): x^1, \dots, x^ℓ are the first ℓ points of the trajectory of \mathcal{B} as applied to every objective $f \in \mathcal{F}^\ell$

(B_ℓ): for every $s \leq \ell$, $\max_{i \notin \{i_1, \dots, i_\ell\}} |x_i^s| \leq |x_{i_s}^s| = \epsilon_{i_s}^* x_{i_s}^s$

At step ℓ , we shrink \mathcal{F}^ℓ to $\mathcal{F}^{\ell+1}$ and extend $\{x^1, \dots, x^\ell\}$ to $\{x^1, \dots, x^{\ell+1}\}$ as follows:

- By (A_ℓ), x^1, \dots, x^ℓ are the first ℓ points of the trajectory of \mathcal{B} applied to every one of the objectives $f \in \mathcal{F}^\ell$, and by (B_ℓ) all these objectives are identically equal to each other in a neighbourhood of $\{x^1, \dots, x^\ell\} \Rightarrow (\ell + 1)$ -st point $x^{\ell+1}$ of the trajectory of \mathcal{B} as applied to every one of the objectives $f \in \mathcal{F}^\ell$ is the same.

- Consider the coordinates of $x^{\ell+1}$ with indexes different from i_1, \dots, i_ℓ , and let $i_{\ell+1}$ be the index of the largest in magnitude of these coordinates. We choose $\epsilon_{i_{\ell+1}}^* = \pm 1$ in such a way that $\epsilon_{i_{\ell+1}}^* x_{i_{\ell+1}}^{\ell+1} = |x_{i_{\ell+1}}^{\ell+1}|$ thus ensuring (B_{ℓ+1}), choose $\delta_{i_{\ell+1}}^* \in (0, \delta_{i_\ell}^*)$ and set

$$\mathcal{F}^{\ell+1} = \left\{ f(x) = \max_{1 \leq i \leq k} [\epsilon_i x_i + \delta_i] : \begin{array}{l} |\epsilon_i| = 1, i = 1, \dots, k \\ \epsilon_{i_s} = \epsilon_{i_s}^*, s = 1, \dots, \ell + 1 \\ \delta_{i_s} = \delta_{i_s}^*, s = 1, \dots, \ell + 1 \\ \delta_{i_1}^* > \dots > \delta_{i_{\ell+1}}^* > \max_{i \notin \{i_1, \dots, i_{\ell+1}\}} \delta_i \geq 0 \end{array} \right\}$$

thus ensuring (A_{ℓ+1}).

♣ After k steps of the construction, we end up with a single-function family

$$\mathcal{F}^k = \{f_k(x) = \max_{1 \leq s \leq k} [\epsilon_{i_s}^* x_{i_s} + \delta_{i_s}^*]\}$$

such that the trajectory x^1, \dots, x^k of \mathcal{B} as applied to $f_k(\cdot)$ satisfies

$$\epsilon_{i_s}^* x_{i_s}^s \geq 0, \quad s = 1, \dots, k,$$

whence, in particular, $f_k(x_k) > 0$. On the other hand,

$$\min_{x \in X} f_k(x) \leq -\frac{1}{k^{1/p}} + \max_i \delta_i^* = -\frac{1}{k^{1/p}} + \frac{1}{2k^{1/p}} = \epsilon_k \equiv -\frac{1}{2k^{1/p}}.$$

Thus, the result x_k of \mathcal{B} as applied to $f_k(\cdot)$ is *not* an ϵ_k -solution of $\min_X f_k$, as claimed.

Convention: From now on, speaking about optimization problem

$$\min_{x \in X} f(x), \quad (*)$$

we assume *by default* that

- X is nonempty closed and bounded convex subset of Euclidean space E (by default, $E = \mathbf{R}^n$)
- $f(x) : X \rightarrow \mathbf{R}$ is convex and Lipschitz continuous:

$$\forall (x, y \in X) : |f(x) - f(y)| \leq L\|x - y\| \quad [L < \infty]$$

Note: The property of f to be Lipschitz continuous is independent of the choice of norm $\|\cdot\|$ on E ; in contrast, the allowed values of the *Lipschitz constant* L do depend on $\|\cdot\|$. In the sequel,

$$L_{\|\cdot\|}(f) = \sup_{x \neq y, x, y \in X} \frac{|f(x) - f(y)|}{\|x - y\|}$$

stands for the best – the smallest – of the Lipschitz constants, taken w.r.t. $\|\cdot\|$, of a Lipschitz continuous function $f : X \rightarrow \mathbf{R}$.

$$\min_{x \in X} f(x), \quad (*)$$

♠ Recall that a *subgradient* $f'(x)$ of a convex function $f : X \rightarrow \mathbb{R}$ at a point $x \in X$ is the slope of a linear function which underestimates f everywhere on X and coincides with f at x :

$$f(y) \geq f(x) + \langle y - x, f'(x) \rangle \quad \forall y \in X.$$

For Lipschitz continuous convex f , a norm $\|\cdot\|$ on E , and every $x \in X$ there exists a subgradient $f'(x)$ of f at x satisfying the norm bound

$$\begin{aligned} \|f'(x)\|_* &\leq L_{\|\cdot\|}(f) \quad (!) \\ \left[\|z\|_* = \max_{u: \|u\| \leq 1} \langle z, u \rangle \right] \end{aligned}$$

When $x \in \text{int } X$, the above relation holds true for every subgradient of f at x .

Convention: In the sequel, when speaking about First Order oracles for Lipschitz continuous convex functions f , we always assume that the subgradients $f'(x)$ reported by the oracles satisfy (!).

The simplest of the cheapest – Subgradient Descent (N. Shor, 1967)

♣ The *Subgradient Descent* method (SD) for solving a convex program

$$\min_{x \in X} f(x) \quad (P)$$

- X – convex compact set in \mathbf{R}^n
 - f – Lipschitz continuous on X convex function
- is the recurrence

$$x_{t+1} = \Pi_X(x_t - \gamma_t f'(x_t)) \quad [x_1 \in X] \quad (\text{SD})$$

where

- $\gamma_t > 0$ are *stepsizes*
- $\Pi_X(x) = \operatorname{argmin}_{y \in X} \|x - y\|_2^2$ is the standard *projector* on X ,
- $f'(x)$ is a *subgradient* of f at x :

$$f(y) \geq f(x) + (y - x)^T f'(x) \quad \forall y \in X.$$

When, why and how SD converges?

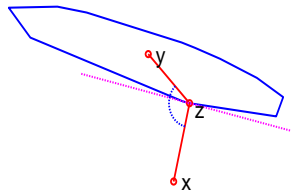
$$x_{t+1} = \Pi_X(x_t - \gamma_t f'(x_t)) \quad (\text{SD})$$

♣ We start with a simple geometric fact:

(!) Let $X \subset \mathbf{R}^n$ be a closed convex set, $x \in \mathbf{R}^n$, and $z = \Pi_X(x)$. Then the vector $e = x - z$ forms an obtuse angle with every vector of the form $y - z$, $y \in X$:

$$(x - z)^T(y - z) \leq 0 \quad \forall y \in X.$$

In particular, $y \in X \Rightarrow \|y - \Pi_X(x)\|_2^2 \leq \|y - x\|_2^2 - \|x - \Pi_X(x)\|_2^2$



In words: When projecting a point x onto a closed convex set X , the squared $\|\cdot\|_2$ distance to any point from X is decreased by at least the squared $\|\cdot\|_2$ -distance from the point x to its projection onto X .

Indeed, when $y \in X$ and $0 \leq t \leq 1$, one has

$$\phi(t) = \|\underbrace{[\Pi_X(x) + t(y - \Pi_X(x))]}_{y_t \in X} - x\|_2^2 \geq \|\Pi_X(x) - x\|_2^2 = \phi(0),$$

whence $0 \leq \phi'(0) = 2(\Pi_X(x) - x)^T(y - \Pi_X(x))$. Consequently,

$$\|y - x\|_2^2 = \|y - \Pi_X(x)\|_2^2 + \|\Pi_X(x) - x\|_2^2 + 2(y - \Pi_X(x))^T(\Pi_X(x) - x) \geq \|y - \Pi_X(x)\|_2^2 + \|\Pi_X(x) - x\|_2^2.$$

$$x_{t+1} = \Pi_X(x_t - \gamma_t f'(x_t)) \quad (\text{SD})$$

♠ By Simple Geometric Fact, for every $u \in X$ one has

$$\begin{aligned} \|x_{t+1} - u\|_2^2 &= \|\Pi_X(x_t - \gamma_t f'(x_t)) - u\|_2^2 \\ &\leq \|x_t - \gamma_t f'(x_t) - u\|_2^2 = \|x_t - u\|_2^2 - 2\gamma_t(x_t - u)^T f'(x_t) + \gamma_t^2 \|f'(x_t)\|_2^2 \end{aligned}$$

and we arrive at

Corollary: For every $u \in X$ one has

$$\gamma_t(x_t - u)^T f'(x_t) \leq \underbrace{\frac{1}{2}\|x_t - u\|_2^2}_{d_t} - \underbrace{\frac{1}{2}\|x_{t+1} - u\|_2^2}_{d_{t+1}} + \frac{1}{2}\gamma_t^2 \|f'(x_t)\|_2^2$$

Note: Since f is convex, one has $(x_t - u)^T f'(x_t) \geq f(x_t) - f(u)$, which combines with Corollary to yield

$$\boxed{\gamma_t[f(x_t) - f(u)] \leq \underbrace{\frac{1}{2}\|x_t - u\|_2^2}_{d_t} - \underbrace{\frac{1}{2}\|x_{t+1} - u\|_2^2}_{d_{t+1}} + \frac{1}{2}\gamma_t^2 \|f'(x_t)\|_2^2}$$

$f_* = \min_{x \in X} f(x)$	(1)
$x_{t+1} = \Pi_X(x_t - \gamma_t f'(x_t))$	(2)
$\gamma_t[f(x_t) - f(u)] \leq \underbrace{\frac{1}{2}\ x_t - u\ _2^2}_{d_t} - \underbrace{\frac{1}{2}\ x_{t+1} - u\ _2^2}_{d_{t+1}} + \frac{1}{2}\gamma_t^2\ f'(x_t)\ _2^2 \quad \forall u \in X$	(3)

Summing up inequalities (3) over $t = T_0, T_0 + 1, \dots, T$, we get

$$\sum_{t=T_0}^T \gamma_t(f(x_t) - f(u)) \leq \underbrace{d_{T_0} - d_{T+1}}_{\leq \Theta} + \sum_{t=T_0}^T \frac{1}{2}\gamma_t^2\|f'(x_t)\|_2^2$$

$$\left[\Theta = \max_{x,y \in X} \frac{1}{2}\|x - y\|_2^2 \right]$$

Setting $u = x_* \equiv \operatorname{argmin}_X f$, we arrive at the bound

$$\forall (T, T_0, T \geq T_0 \geq 1) : \epsilon_T \equiv \min_{t \leq T} f(x_t) - f_* \leq \frac{\Theta + \frac{1}{2} \sum_{t=T_0}^T \gamma_t^2 \|f'(x_t)\|_2^2}{\sum_{t=T_0}^T \gamma_t}$$

$$\forall(T, T_0, T \geq T_0 \geq 1) : \epsilon_T \equiv \min_{t \leq T} f(x_t) - f_* \leq \frac{\Theta + \frac{1}{2} \sum_{t=T_0}^T \gamma_t^2 \|f'(x_t)\|_2^2}{\sum_{t=T_0}^T \gamma_t}$$

♣ The resulting relation leads to various convergence results.

Example 1: “Divergent Series”. Let $\gamma_t \rightarrow 0$ as $t \rightarrow \infty$, while $\sum_t \gamma_t = \infty$. Then

$$\lim_{T \rightarrow \infty} \epsilon_T = 0.$$

Proof. Set $T_0 = 1$ and note that

$$\frac{\sum_{t=1}^T \gamma_t^2 \|f'(x_t)\|_2^2}{\sum_{t=1}^T \gamma_t} \leq L_{\|\cdot\|_2}^2(f) \frac{\sum_{t=1}^T \gamma_t^2}{\sum_{t=1}^T \gamma_t} \rightarrow 0, T \rightarrow \infty.$$

$$\begin{array}{c}
\boxed{f_* = \min_{x \in X} f(x)} \\
\Downarrow \\
\boxed{\forall (T, T_0, T \geq T_0 \geq 1) : \epsilon_T \equiv \min_{t \leq T} f(x_t) - f_* \leq \frac{\Theta + \frac{1}{2} \sum_{t=T_0}^T \gamma_t^2 \|f'(x_t)\|_2^2}{\sum_{t=T_0}^T \gamma_t}} \\
\quad \left[\Theta = \frac{1}{2} \max_{x, y \in X} \|x - y\|_2^2 \right]
\end{array}$$

Example 2: “Optimal stepsizes” :

$$\gamma_t = \frac{\sqrt{2\Theta}}{\|f'(x_t)\|_2 \sqrt{t}} \Rightarrow \boxed{\epsilon_T \equiv \min_{t \leq T} f(x_t) - f_* \leq O(1) \frac{L_{\|\cdot\|_2}(f) \sqrt{\Theta}}{\sqrt{T}}, T \geq 1}$$

Proof. Setting $T_0 = \lfloor T/2 \rfloor$, we get

$$\begin{aligned}
\epsilon_T &\leq \left[\Theta + \Theta \sum_{t=T_0}^T t^{-1} \right] \left[\sum_{t=T_0}^T \frac{\sqrt{2\Theta}}{\sqrt{t} \|f'(x_t)\|_2} \right]^{-1} \leq \left[\Theta + \Theta \sum_{t=T_0}^T t^{-1} \right] \left[\sum_{t=T_0}^T \frac{\sqrt{2\Theta}}{\sqrt{t} L_{\|\cdot\|_2}(f)} \right]^{-1} \\
&\leq L_{\|\cdot\|_2}(f) \sqrt{\Theta} \frac{1+O(1)}{O(1)\sqrt{T}} = O(1) \frac{L_{\|\cdot\|_2}(f) \sqrt{\Theta}}{\sqrt{T}}
\end{aligned}$$

[note that with $T_0 = \lfloor T/2 \rfloor$ we have $\sum_{t=T_0}^T t^{-1} = O(1)$ and $\sum_{t=T_0}^T \frac{1}{\sqrt{t}} = O(1)\sqrt{T}$].

$$\begin{aligned}
f_* &= \min_{x \in X} f(x) \\
\Rightarrow x_{t+1} &= \Pi_X(x_t - \gamma_t f'(x(t))), \quad \gamma_t = \frac{\max_{x,y \in X} \|x-y\|_2}{\sqrt{t} \|f'(x_t)\|_2} \\
&\quad \text{Var}_{\|\cdot\|_2, X}(f) \\
\Rightarrow \epsilon_T \equiv \min_{1 \leq t \leq T} f(x_t) - f_* &\leq O(1) \underbrace{L_{\|\cdot\|_2}(f) \max_{x,y \in X} \|x-y\|_2}_{\text{Var}_{\|\cdot\|_2, X}(f)} / \sqrt{T}
\end{aligned}$$

Good news: We have arrived at efficiency estimate which is *dimension-independent*, provided that the “ $\|\cdot\|_2$ -variation” of the objective on the feasible domain

$$\text{Var}_{\|\cdot\|_2, X}(f) = L_{\|\cdot\|_2}(f) \max_{x,y \in X} \|x-y\|_2$$

is fixed. Moreover, when X is a Euclidean ball in \mathbf{R}^n , this efficiency estimate “is as good as an efficiency estimate of a black-box-oriented method can be”, provided that the dimension is large:

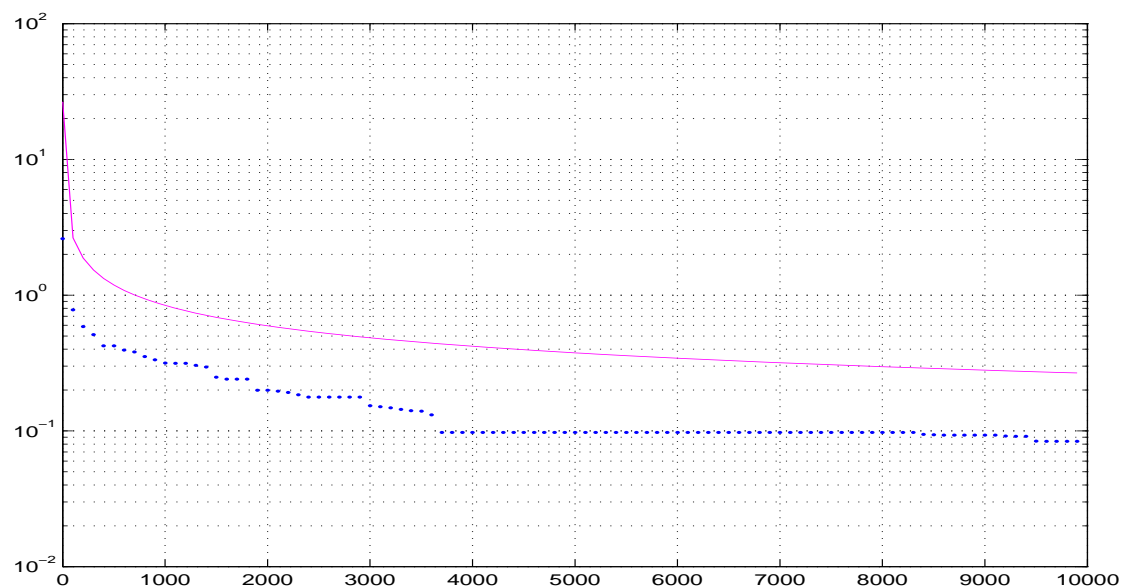
$$n \geq (\text{Var}_{\|\cdot\|_2, X}(f)/\epsilon)^2$$

$$\epsilon_T \equiv \min_{1 \leq t \leq T} f(x_t) - f_* \leq O(1) \text{Var}_{\|\cdot\|_2, X}(f) / \sqrt{T}$$

$$[\text{Var}_{\|\cdot\|_2, X}(f) = L_{\|\cdot\|_2}(f) \max_{x, y \in X} \|x - y\|_2]$$

Bad news: Our “dimension-independent” efficiency estimate

- is pretty slow
- is indeed dimension-independent only for problems with “Euclidean geometry” – those with moderate $\|\cdot\|_2$ -variation. As a matter of fact, in some (but not all!) important applications problems of this type are pretty rare.



SD as applied to $\min_{\|x\|_2 \leq 1} \|Ax - b\|_1$, $A : 50 \times 50$
 [red: efficiency estimate; blue: actual error]

$$x_{t+1} = \Pi_X(x_t - \gamma_t f'(x(t)))$$

♣ An evident drawback of SD is that all information on the objective accumulated so far is “summarized” in the current iterate, and this “summary” is very incomplete. With better usage of past information, one arrives at *bundle methods* which outperform SD significantly in practice, while preserving the most attractive theoretical property of SD – dimension-independent and optimal, in favourable circumstances, rate of convergence.

Bundle-Level method for solving $f_* = \min_{x \in X} f(x)$

- ♣ At the beginning of step t of BL, we have at our disposal
 - the first-order information $\{f(x_\tau), f'(x_\tau)\}_{1 \leq \tau < t}$ on f along the previous search points $x_\tau \in X$, $\tau < t$;
 - current iterate $x_t \in X$.

- ♣ At step t we
 - compute $f(x_t), f'(x_t)$; this information, along with the past first-order information on f , provides us with the current *model of the objective*

$$f_t(x) = \max_{\tau \leq t} [f(x_\tau) + (x - x_\tau)^T f'(x_\tau)]$$

- This model underestimates the objective and is exact at the points x_1, \dots, x_t ;
- define the *best found so far value* $f^t = \min_{\tau \leq t} f(x_\tau)$ of f
 - define the current *lower bound* f_t on f_* by solving the auxiliary problem

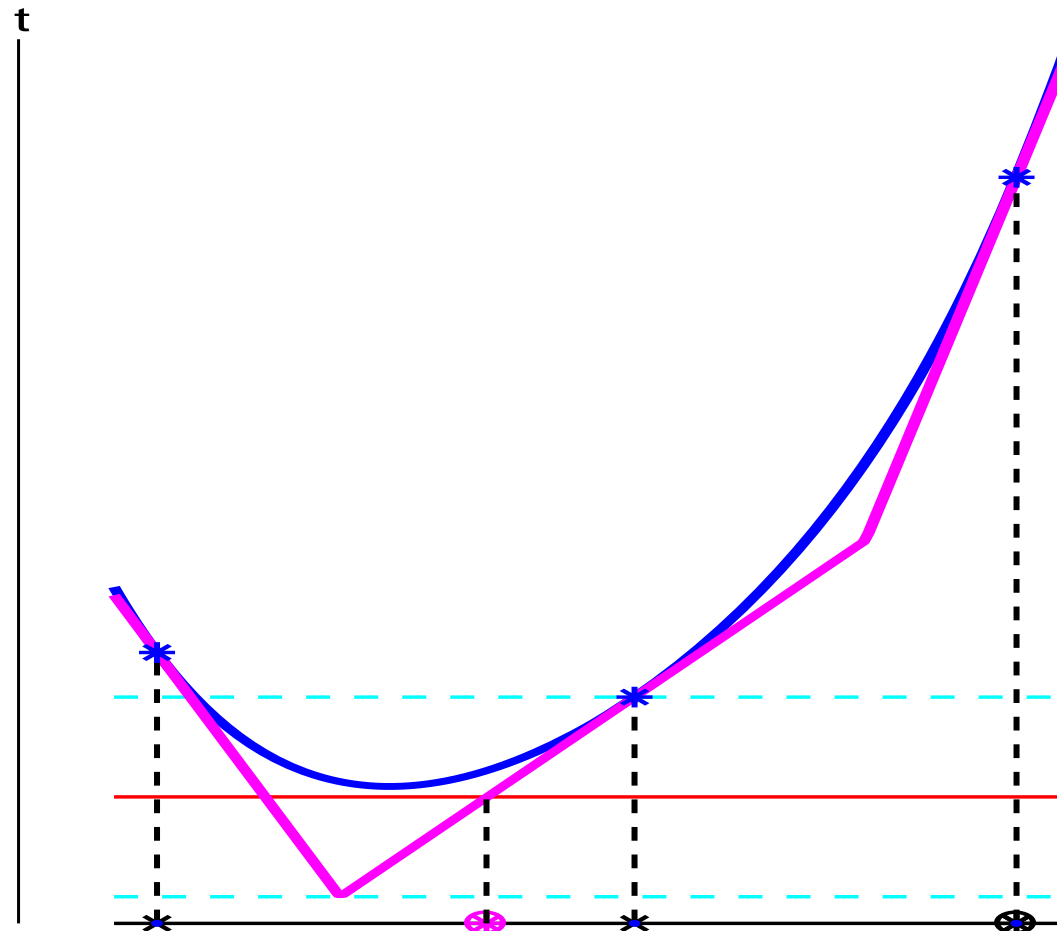
$$f_t = \min_{x \in X} f_t(x) \quad (\text{LP}_t)$$

Note: current *gap* $\Delta_t = f^t - f_t$ upper-bounds the inaccuracy of the best found so far solution;

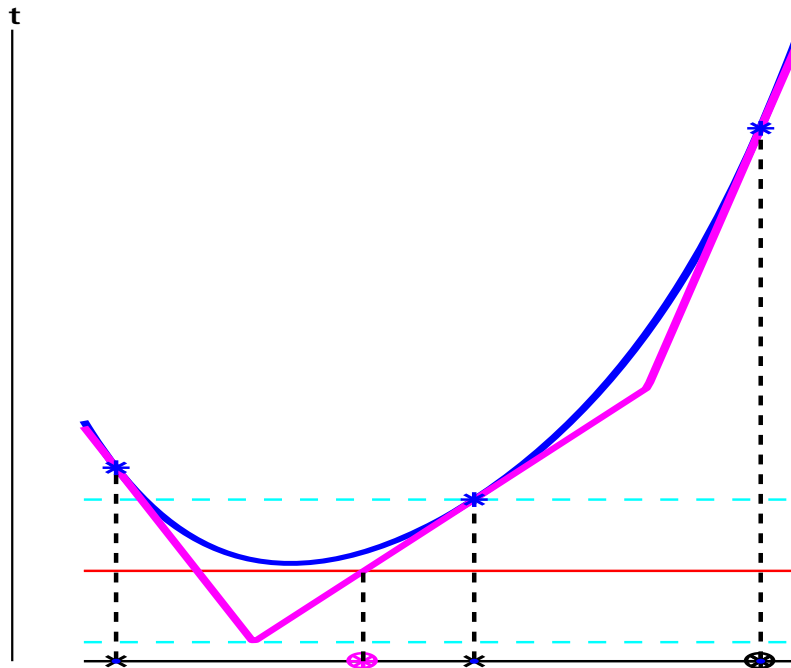
- compute the current *level* $\ell_t = f_t + \lambda \Delta_t$ ($\lambda \in (0, 1)$ is a parameter)
- build a new search point by solving the auxiliary problem

$$x_{t+1} = \underset{x}{\operatorname{argmin}} \{ \|x - x_t\|_2^2 : x \in X, f_t(x) \leq \ell_t \} \quad (\text{QP}_t)$$

and loop to step $t + 1$.



- blue: the objective f
- *: x_1, x_2, x_3
- magenta: current piecewise linear model $f_3(\cdot)$ of f
- cyan horizontal lines: $t = \min_{i \leq 3} f(x_i)$ and $t = \min_x f_3(x)$
- red horizontal line: $t = \ell_3$
- red circle: new iterate x_4



Note: It seems to be more intuitive to “fully trust” in model and take, as the next iterate, the minimizer of the model or, which is the same, to set the level ℓ_t equal to f_t rather than to

$$\ell_t = f_t + \lambda \Delta_t \quad \Delta_t = \min_{\tau \leq t} f(x_\tau) - f_t. \quad [\lambda \in (0, 1), \text{ usually } \lambda = 0.5]$$

Unfortunately, the resulting *Kelley method* has disastrously bad theoretical complexity (and from time to time exhibits disastrously bad actual performance).

How BL converges?

Claim: For every ϵ , $0 < \epsilon < \Delta_1$, the number N of steps before a gap $\leq \epsilon$ is obtained (i.e., before an ϵ -solution is found) does not exceed the bound

$$N(\epsilon) = \frac{\text{Var}_{\|\cdot\|_2, X}^2(f)}{\lambda(1-\lambda)^2(2-\lambda)\epsilon^2},$$

\Rightarrow Inaccuracy after $T = 1, 2, \dots$ steps is upper-bounded by

$$C(\lambda) \frac{\text{Var}_{\|\cdot\|_2, X}(f)}{\sqrt{T}}$$

— the same efficiency estimate as for SD with optimal stepsizes.

Why and how BL converges?

Preliminary observations:

♠ The models $f_t(x) = \max_{\tau \leq t} [f(x_\tau) + (x - x_\tau)^T f'(x_\tau)]$ grow with t and underestimate f , while the best found so far values of the objective decrease with t and overestimate f_* . Thus,

$$\begin{aligned} f_1 &\leq f_2 \leq f_3 \leq \dots \leq f_* \\ f^1 &\geq f^2 \geq f^3 \geq \dots \geq f_* \\ \Delta_1 &\geq \Delta_2 \geq \dots \geq 0 \end{aligned}$$

♠ Let us say that a group of subsequent iterations $J = \{s, s+1, \dots, r\}$ form a segment, if $\Delta_r \geq (1-\lambda)\Delta_s$. We claim that *If $J = \{s, s+1, \dots, r\}$ is a segment, then*

- (i) *All the sets $L_t = \{x \in X : f_t(x) \leq \ell_t\}$, $t \in J$, have a point in common, specifically, (any) minimizer u of $f_r(\cdot)$ over X ;*
- (ii) *For $t \in J$, one has $\|x_t - x_{t+1}\|_2 \geq \frac{(1-\lambda)\Delta_r}{L_{\|\cdot\|_2}(f)}$.*

We claim that if $J = \{s, s+1, \dots, r\}$ is a segment, then

- (i) All the sets $L_t = \{x \in X : f_t(x) \leq \ell_t\}$, $t \in J$, have a point in common, specifically, (any) minimizer u of $f_r(\cdot)$ over X ;
- (ii) For $t \in J$, one has $\|x_t - x_{t+1}\|_2 \geq \frac{(1-\lambda)\Delta_r}{L_{\|\cdot\|_2}(f)}$.

Indeed,

(i): for $t \in J$ we have

$$f_t(u) \leq f_r(u) = f_r = f^r - \Delta_r \leq f^t - \Delta_r \leq f^t - (1-\lambda)\Delta_s \leq f^t - (1-\lambda)\Delta_t = \ell_t.$$

(ii): We have $f_t(x_t) = f(x_t) \geq f^t$, and $f_t(x_{t+1}) \leq \ell_t = f^t - (1-\lambda)\Delta_t$. Thus, when passing from x_t to x_{t+1} , t -th model decreases by at least $(1-\lambda)\Delta_t \geq (1-\lambda)\Delta_r$. It remains to note that $f_t(\cdot)$ is Lipschitz continuous w.r.t. $\|\cdot\|_2$ with constant $L_{\|\cdot\|_2}(f)$.

$$(ii) \text{ For } t \in J, \text{ one has } \|x_t - x_{t+1}\|_2 \geq \frac{(1-\lambda)\Delta_r}{L_{\|\cdot\|_2}(f)}.$$

♣ **Main observation:** The cardinality of a segment $J = \{s, s+1, \dots, r\}$ of iterations can be bounded as follows:

$$\text{Card}(J) \leq \frac{\text{Var}_{\|\cdot\|_2, X}^2(f)}{(1-\lambda)^2 \Delta_r^2}.$$

Indeed, when $t \in J$, the sets $L_t = \{x \in X : f_t(x) \leq \ell_t\}$ have a point u in common, and x_{t+1} is the projection of x_t onto L_t . It follows that

$$\begin{aligned} & \|x_{t+1} - u\|_2^2 \leq \|x_t - u\|_2^2 - \|x_t - x_{t+1}\|_2^2 \quad \forall t \in J \\ \Rightarrow & \sum_{t \in J} \|x_t - x_{t+1}\|_2^2 \leq \|x_s - u\|_2^2 \leq \max_{x, y \in X} \|x - y\|_2^2 \\ \Rightarrow & \text{Card}(J) \leq \frac{\max_{x, y \in X} \|x - y\|_2^2}{\min_{t \in J} \|x_t - x_{t+1}\|_2^2} \\ \Rightarrow & \text{Card}(J) \leq \frac{L_{\|\cdot\|_2}^2(f) \max_{x, y \in X} \|x - y\|_2^2}{(1-\lambda)^2 \Delta_r^2} \quad [\text{by (ii)}] \end{aligned}$$

Corollary: For every ϵ , $0 < \epsilon < \Delta_1$, the number N of steps before a gap $\leq \epsilon$ is obtained (i.e., before an ϵ -solution is found) does not exceed the bound

$$N(\epsilon) = \frac{\text{Var}_{\|\cdot\|_2, X}^2(f)}{\lambda(1-\lambda)^2(2-\lambda)\epsilon^2}.$$

Proof of Corollary. Assume that N is such that $\Delta_N > \epsilon$, and let us bound N from above.

- Let us split the set of iterations $I = \{1, \dots, N\}$ into segments J_1, \dots, J_m as follows: •
 J_1 is the maximal segment which ends with iteration N :

$$J_1 = \{t : t \leq N, (1 - \lambda)\Delta_t \leq \Delta_N\}$$

- J_1 is certain group of subsequent iterations $\{s_1, s_1 + 1, \dots, N\}$. If J_1 differs from I : $s_1 > 1$, we define J_2 as the maximal segment which ends with iteration $s_1 - 1$:

$$J_2 = \{t : t \leq s_1 - 1, (1 - \lambda)\Delta_t \leq \Delta_{s_1-1}\} = \{s_2, s_2 + 1, \dots, s_1 - 1\}$$

- If $J_1 \cup J_2$ differs from I : $s_2 > 1$, we define J_3 as the maximal segment which ends with iteration $s_2 - 1$:

$$J_3 = \{t : t \leq s_2 - 1, (1 - \lambda)\Delta_t \leq \Delta_{s_2-1}\} = \{s_3, s_3 + 1, \dots, s_2 - 1\}$$

and so on.

- As a result, I will be partitioned “from the end to the beginning” into segments of iterations J_1, J_2, \dots, J_m . Let d_ℓ be the gap corresponding to the last iteration from J_ℓ . By maximality of segments J_ℓ , we have

$$d_1 \geq \Delta_N > \epsilon \& d_{\ell+1} > (1 - \lambda)^{-1} d_\ell, \ell = 1, 2, \dots, m - 1$$

whence

$$d_\ell > \epsilon(1 - \lambda)^{-(\ell-1)}.$$

We now have

$$\begin{aligned} N &= \sum_{\ell=1}^m \text{Card}(J_\ell) \leq \sum_{\ell=1}^m \frac{\text{Var}_{\|\cdot\|_2, X}^2(f)}{(1-\lambda)^2 d_\ell^2} \leq \frac{\text{Var}_{\|\cdot\|_2, X}^2(f)}{(1-\lambda)^2} \sum_{\ell=1}^m (1 - \lambda)^{2(\ell-1)} \epsilon^{-2} \\ &\leq \frac{\text{Var}_{\|\cdot\|_2, X}^2(f)}{(1-\lambda)^2 \epsilon^2} \sum_{\ell=1}^{\infty} (1 - \lambda)^{2(\ell-1)} = \frac{\text{Var}_{\|\cdot\|_2, X}^2(f)}{(1-\lambda)^2 [1 - (1-\lambda)^2] \epsilon^2} = N(\epsilon). \end{aligned}$$

♣ We have seen that Bundle-Level shares the dimension-independent (and optimal in the “favourable” large-scale case) theoretical complexity bound

For every $\epsilon > 0$, the number of steps before an ϵ -solution to convex program $\min_{x \in X} f(x)$ is found, does not exceed

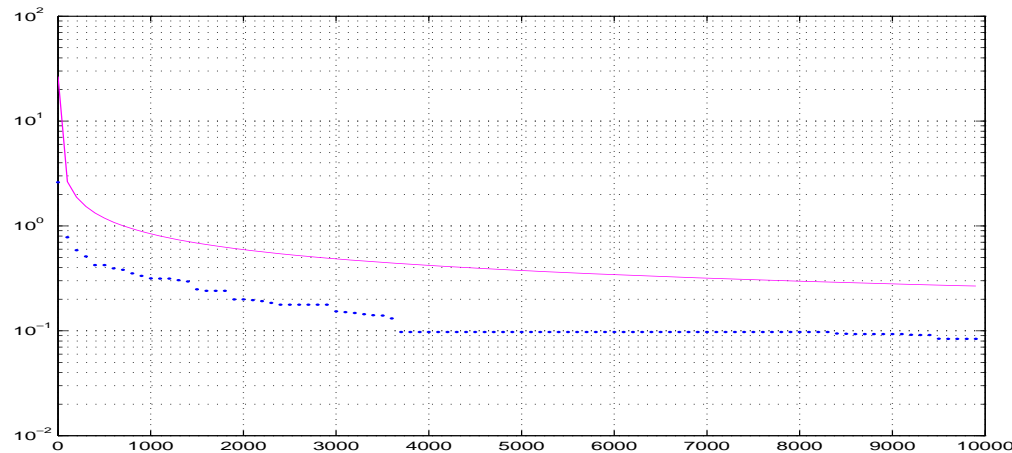
$$O(1) \left(\frac{\text{Var}_{\|\cdot\|_2, X}(f)}{\epsilon} \right)^2.$$

♣ There exists quite convincing *experimental* evidence that Bundle-Level obeys the optimal in fixed dimension “polynomial time” complexity bound:

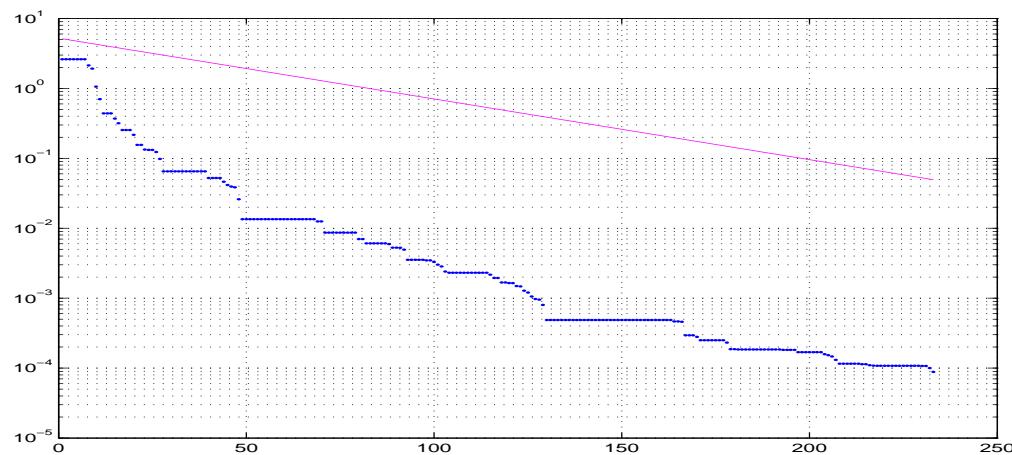
For every $\epsilon \in (0, \text{Var}_X(f) \equiv \max_X f - \min_X f)$, the number of steps before an ϵ -solution to convex program $\min_{x \in X} f(x)$ with $X \subset \mathbf{R}^n$ is found, does not exceed $n \ln \left(\frac{\text{Var}_X(f)}{\epsilon} \right) + 1$.

♠ **Experimental rule:** When solving convex program with n variables by BL, every n steps add new accuracy digit.

Illustration: $\min_{x: \|x\|_2 \leq 1} f(x) \equiv \|Ax - b\|_1$, $\dim x = 50$ ($f(0) = 2.61$, $f_* = 0$)



SD, accuracy vs. iteration count. blue: errors; red: efficiency estimate $3 \frac{\text{Var}_{\|\cdot\|_2, X}(f)}{\sqrt{t}}$; $\epsilon_{10000} = 0.084$



BL, accuracy vs. iteration count. blue: errors; red: efficiency estimate $e^{-t/n} \text{Var}_X(f)$; $\epsilon_{233} < 1.e - 4$

♣ In BL, the number of linear constraints in the auxiliary problems

$$\begin{aligned} f_t &= \min_{x \in X} f_t(x) && (\text{LP}_t) \\ x_{t+1} &= \operatorname{argmin}_x \{ \|x_t - x\|_2^2 : x \in X, f_t(x) \leq \ell_t \} && (\text{QP}_t) \end{aligned}$$

is equal to the size t of the current *bundle* – the collection of affine forms $g_\tau(x) = f(x_\tau) + (x - x_\tau)^T f'(x_\tau)$ participating in the model $f_t(\cdot)$. Thus, the complexity of an iteration in BL grows with the iteration number. In order to suppress this phenomenon, one needs a mechanism for *shrinking* the bundle (and thus – simplifying the models of f).

♠ The simplest way of shrinking the bundle is to initialize d as Δ_1 and to run plain BL until an iteration t with $\Delta_t \leq d/2$ is met. At such an iteration, we — shrink the current bundle, keeping in it the minimum number of the forms g_τ sufficient to ensure that

$$f_t \equiv \min_{x \in X} \max_{1 \leq \tau \leq t} g_\tau(x) = \min_{x \in X} \max_{\text{selected } \tau} g_\tau(x)$$

(this number is at most n),

— reset d as Δ_t ,

and proceed with plain BL until the gap is again reduced by factor 2, etc.

♣ Computational experience demonstrates that the outlined approach does not slow BL down, while keeping the size of the bundle below the level of about $2n$.

Truncated Proximal Level Method for $\min_{x \in X} f(x)$

- ♣ The *Truncated Proximal Level* method has the same efficiency estimate as SD and BL, but keeps the cardinality of bundle not exceeding a given level m (which can be as small as 1).
- ♣ Playing with m , one can trade “practical rate of convergence” for arithmetic complexity of an iteration, which is important when solving large-scale problems.

Truncated Proximal Level Method for $\min_{x \in X} f(x)$ – construction

- ♣ The *Truncated Proximal Level* method keeps the cardinality of bundle not exceeding a given level m .
- ♣ Execution of TLM is split into *phases*. Phase s is associated with
 - *prox-center* $c_s \in X$
 - s -th upper bound f^s on f_* , which is the best value of the objective observed before the phase begins
 - s -th lower bound f_s on f_* , which is the best lower bound on f_* observed before the phase begins
 - f^s and f_s define
 - ◊ s -th optimality gap $\Delta_s = f^s - f_s$
 - ◊ s -th level $\ell_s = f_s + \lambda \Delta_s$, where $\lambda \in (0, 1)$ is parameter of the method.
 - current model $\tilde{f}^s(\cdot) \leq f(\cdot)$ of $f(\cdot)$, which is the maximum of $\leq m$ affine forms.
- ♠ To initialize the first phase, we choose $c_1 \in X$, compute $f(c_1), f'(c_1)$ and set

$$\tilde{f}^1(x) = f(c_1) + (x - c_1)^T f'(c_1), \quad f^1 = f(c_1), \quad f_1 = \min_{x \in X} \tilde{f}^1(x).$$

- ♣ At the beginning of step $t = 1, 2, \dots$ of phase s , we have at our disposal
 - upper bound $f^{s,t-1} \leq f^s$ on f_* , which is the best found so far value of the objective,
 - lower bound $f_{s,t-1} \geq f_s$ on f_* ,
 - model $\tilde{f}^{s,t-1}(\cdot) \leq f(\cdot)$ of the objective which is the maximum of $\leq m$ affine forms
 - iterate $x_t \in X$ and set

$$H_{t-1} = \{x : \alpha_{t-1}^T x \geq \beta_{t-1}\}$$

such that

$$x \in X, f(x) \leq \ell_s \Rightarrow x \in H_{t-1} \quad (a_t)$$

$$x_t = \operatorname{argmin}_x \{\|x - c_s\|_2^2 : x \in X \cap H_{t-1}\} \quad (b_t)$$

- ♠ To initialize the first step of phase s , we set

$$f^{s,0} = f^s, f_{s,0} = f_s, \tilde{f}^{s,0}(\cdot) = \tilde{f}^s(\cdot), \alpha_0 = 0, \beta_0 = 0 [\Rightarrow H_0 = \mathbf{R}^n]$$

thus ensuring (a_1) , and set $x_1 = c_s$, thus ensuring (b_1) .

Step t phase s : Given

- bounds $f^{s,t-1} \geq f_*$, $f_{s,t-1} \leq f_*$, • model $\tilde{f}^{s,t-1}(\cdot) \leq f(\cdot)$,
- x_t and $H_{t-1} = \{x : \alpha_{t-1}^T x \geq \beta_{t-1}\}$ such that

$$x \in X, f(x) \leq \ell_s \Rightarrow x \in H_{t-1} \quad (a_t) \quad \& \quad x_t = \operatorname{argmin}_x \{\|x - c_s\|_2^2 : x \in X \cap H_{t-1}\} \quad (b_t)$$

1. we compute $f(x_t)$, $f'(x_t)$ and set $g_t(x) = f(x_t) + (x - x_t)^T f'(x_t)$;
2. we define $\tilde{f}^{s,t}(\cdot)$ as the maximum of $g_t(\cdot)$ and affine forms associated with $\tilde{f}^{s,t-1}$ (dropping, if necessary, one of the latter forms to make $\tilde{f}^{s,t}$ the maximum of at most m forms). If $f(x_t) \leq \ell_s + 0.5(f^s - \ell_s)$ (“significant progress in the upper bound”), we terminate phase s and set

$$f^{s+1} = f^{s,t}, \quad f_{s+1} = f_{s,t-1}, \quad \tilde{f}^{s+1}(\cdot) = \tilde{f}^{s,t}(\cdot),$$

otherwise we proceed as follows:

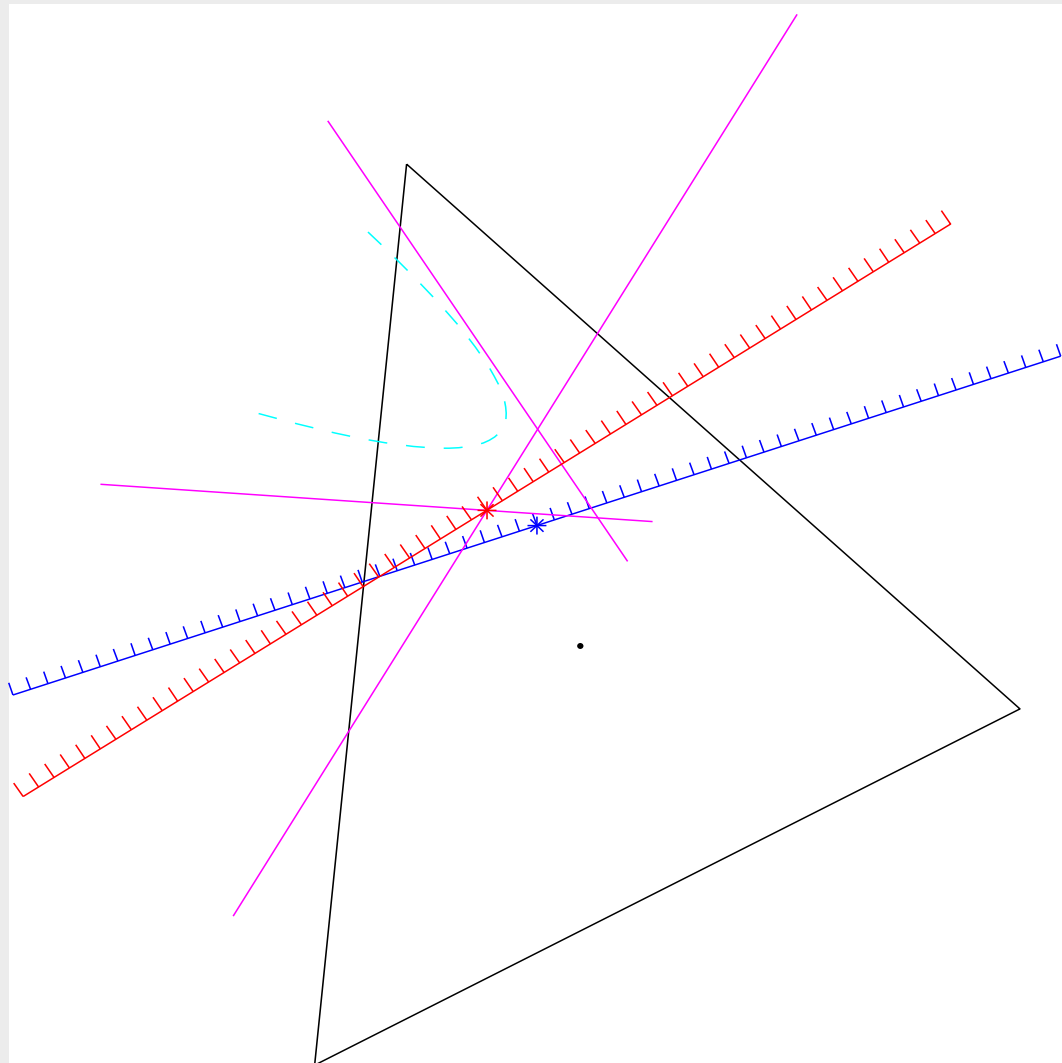
3. we compute $f_t = \min_x \{\tilde{f}^{s,t}(x) : x \in H_{t-1} \cap X\}$. Since $f(x) \geq \ell_s$ in $X \setminus H_{t-1}$, we have $f_* \geq \min[\ell_s, f_t]$, so that $f_{s,t} \equiv \max\{f_{s,t-1}, \min[\ell_s, f_t]\} \leq f_*$. If $f_{s,t} \geq \ell_s - 0.5(\ell_s - f_s)$ (“significant progress in the lower bound”), we terminate phase s and set

$$f^{s+1} = f^{s,t}, \quad f_{s+1} = f_{s,t}, \quad \tilde{f}^{s+1}(\cdot) = \tilde{f}^{s,t}(\cdot)$$

otherwise we set

$$\begin{aligned} x_{t+1} &= \operatorname{argmin}_x \left\{ \|x - c_s\|_2^2 : x \in X \cap H_{t-1}, \tilde{f}^{s,t}(x) \leq \ell_s \right\} \\ H_t &= \{x : (x_{t+1} - c_s)^T (x - x_{t+1}) \geq 0\} \end{aligned}$$

and loop to step $t + 1$ of phase s .



Step of TPL

$$x_{t+1} = \operatorname{argmin}_x \left\{ \|x - c_s\|_2^2 : x \in X \cap H_{t-1}, \tilde{f}^{s,t}(x) \leq \ell_s \right\} \quad (1)$$

$$H_t = \{x : (x_{t+1} - c_s)^T (x - x_{t+1}) \geq 0\} \quad (2)$$

Note: When passing to step $t + 1$, we have ensured the relations

$$x \in X, f(x) \leq \ell_s \Rightarrow x \in H_t \quad (a_{t+1})$$

$$x_{t+1} = \operatorname{argmin}_x \left\{ \|x - c_s\|_2^2 : x \in X \cap H_t, \tilde{f}^{s,t}(x) \leq \ell \right\} \quad (b_{t+1})$$

Indeed, x_{t+1} is the minimizer of $\omega_s(x) \equiv \frac{1}{2}\|x - c_s\|_2^2$ on the set

$$Y_t = X \cap H_{t-1} \cap \{x : \tilde{f}^{t,s}(x) \leq \ell_s\}$$

whence

$$\overbrace{[\omega'_s(x_{t+1})]^T}^{x_{t+1} - c_s} (x - x_{t+1}) \geq 0 \quad \forall x \in Y_t$$



$$Y_t \subset H_t = \{x : [\omega'_s(x_{t+1})]^T (x - x_{t+1}) \geq 0\} \quad (*)$$

Thus,

$$\begin{aligned} (x \in X, f(x) \leq \ell_s) &\underbrace{\Rightarrow}_{(a_t)} (x \in X \cap H_{t-1}, f(x) \leq \ell_s) \\ &\Rightarrow \underbrace{(x \in X \cap H_{t-1}, \tilde{f}^{s,t}(x) \leq \ell_s)}_{x \in Y_t} \underbrace{\Rightarrow}_{(*)} x \in H_t \end{aligned}$$

as required in (a_{t+1}) . (b_{t+1}) readily follows from the definition of H_t .

Convergence of TPL

♣ Preliminary observations:

- When passing from phase s to phase $s + 1$, the optimality gap is decreased at least by the factor

$$\theta(\lambda) = \frac{\min[1 + \lambda, 2 - \lambda]}{2}.$$

Indeed, phase s can be terminated at step t due to significant progress either in the upper bound on f_* : $f^{s+1} = f^{s,t} \leq \ell_s + \frac{1}{2}(f^s - \ell_s)$

$$\Rightarrow \Delta_{s+1} = f^{s+1} - f_{s+1} \leq \frac{1}{2}\ell_s + \frac{1}{2}f^s - f_s = \frac{1 + \lambda}{2}\Delta_s$$

or in the lower bound: $f_{s+1} = f_{s,t} \geq \ell_s - \frac{1}{2}(\ell_s - f_s)$

$$\Rightarrow \Delta_{s+1} = f^{s+1} - f_{s+1} \leq f^s - \frac{1}{2}f_s - \frac{1}{2}\ell_s = \frac{2 - \lambda}{2}\Delta_s$$

- Let x_t, x_{t+1} be two subsequent search points of phase s . Then

$$\|x_t - x_{t+1}\|_2 > \frac{(1 - \lambda)\Delta_s}{2L_{\|\cdot\|_2}(f)}.$$

Indeed, we have $f(x_t) = g_t(x_t) = \tilde{f}^{s,t}(x_t) \geq \ell_s + \frac{1}{2}(f^s - \ell_s)$, since otherwise phase s would be terminated at step t . At the same time, $g_t(x_{t+1}) \leq \tilde{f}^{s,t}(x_{t+1}) \leq \ell_s$. Thus, passing from x_t to x_{t+1} , we decrease Lipschitz continuous, with constant $L_{\|\cdot\|_2}(f)$ w.r.t. $\|\cdot\|_2$, function $g_t(\cdot)$ by at least $\frac{1}{2}(f^s - \ell_s) = \frac{1-\lambda}{2}\Delta_s$.

♣ **Main observation:** *Number of steps at phase s does not exceed*

$$N_s = \frac{4V_{\|\cdot\|_2, X}^2(f)}{(1-\lambda)^2\Delta_s^2} + 1. \quad (*)$$

Indeed, let the number of steps of the phase be $> N$. By construction, $x_{t+1} \in H_{t-1}$ and x_t is the minimizer of $\omega_s(x) = \frac{1}{2}\|x - c_s\|_2^2$ on H_{t-1} , whence

$$1 \leq t \leq N \Rightarrow \omega_s(x_{t+1}) = \omega_s(x_t) + \underbrace{(x_{t+1} - x_t)^T \omega'_s(x_t)}_{\geq 0} + \frac{1}{2}\|x_t - x_{t+1}\|_2^2 \geq \omega_s(x_t) + \frac{1}{2}\|x_t - x_{t+1}\|_2^2.$$

It follows that $\sum_{t=1}^N \underbrace{\frac{1}{2}\|x_t - x_{t+1}\|_2^2}_{\geq \frac{(1-\lambda)^2\Delta_s^2}{8L^2\|\cdot\|_2(f)}} \leq \frac{1}{2} \max_{x,y \in X} \|y - x\|_2^2$, whence $N \leq \frac{4V_{\|\cdot\|_2, X}^2(f)}{(1-\lambda)^2\Delta_s^2}$.

♣ Same as in the case of BL, $(*)$ combines with the relation $\Delta_{s+1} \leq \theta(\lambda)\Delta_s$ to yield the following

Corollary: *For every ϵ , $0 < \epsilon < \Delta_1$, the total number of TPL steps before a gap $\leq \epsilon$ is obtained (i.e., before an ϵ -solution is found) does not exceed the bound*

$$N(\epsilon) = c(\lambda) \frac{\text{Var}_{\|\cdot\|_2, X}^2(f)}{\epsilon^2}.$$

$$f_* = \min_{x \in X} f(x) \quad (*)$$

From Gradient to Mirror Descent

♣ Subgradient Descent method and its bundle versions are “intrinsically adjusted” to problems with Euclidean geometry; this is where the role of the $\|\cdot\|_2$ -variation of the objective

$$\text{Var}_{\|\cdot\|_2, X}(f) = L_{\|\cdot\|_2}(f) \max_{x, x' \in X} \|x - x'\|_2$$

in the efficiency estimate

$$\min_{t \leq T} f(x_t) - f_* \leq O(1) \frac{\text{Var}_{\|\cdot\|_2, X}(f)}{\sqrt{T}}$$

comes from.

♣ An extension of SD and its bundle versions onto problems with “nice non-Euclidean geometry” is offered by the *Mirror Descent* scheme.

Mirror Descent – Building Blocks

♣ Building block #1: Distance-Generating Function.

♠ A SD step

$$x \mapsto x_+ = \Pi_X(x - \gamma f'(x)) \quad (1)$$

can be viewed as follows: given an iterate $x \in X$, we

1) Compute $f'(x)$

2) Perform the *prox-step* $x \mapsto x_+ = \text{Prox}_x(\gamma f'(x))$

$$\text{Prox}_x(\xi) := \underset{u \in X}{\operatorname{argmin}} [\langle \xi, u \rangle + V_x(u)]$$

$\xi \mapsto \text{Prox}_x(\xi)$: prox-mapping with prox-center x

$$V_x(u) = \omega(u) - \omega(x) - \langle u - x, \nabla \omega(x) \rangle$$

where

$$\omega(u) = \frac{1}{2} \|u\|_2^2 \quad (2)$$

is a specific “distance-generating function.”

Indeed, with the above $\omega(\cdot)$, we have

$$V_x(u) := \frac{1}{2} u^T u - x^T(u - x) - \frac{1}{2} x^T x = \frac{1}{2} \|u - x\|_2^2$$

$$\Downarrow$$

$$\text{Prox}_x(\xi) = \underset{u \in X}{\operatorname{argmin}} \left[\xi^T u + \frac{1}{2} (u - x)^T (u - x) \right] = \underset{u \in X}{\operatorname{argmin}} \frac{1}{2} \|u - (x - \xi)\|_2^2 = \Pi_x(x - \xi)$$

$$\begin{aligned}\text{Prox}_x(\xi) &= \operatorname{argmin}_{u \in X} [\langle \xi, u \rangle + V_x(u)] \\ V_x(u) &= \omega(u) - \omega(x) - \langle \nabla \omega(x), u - x \rangle\end{aligned}$$

♠ The “Main Inequality”

$$x_+ = \Pi_X(x - \gamma f'(x)) \Rightarrow \forall u \in X : \gamma \langle f'(x), x - u \rangle \leq \frac{1}{2} \|x - u\|_2^2 - \frac{1}{2} \|x_+ - u\|_2^2 + \frac{1}{2} \gamma^2 \|f'(x)\|_2^2$$

underlying all our convergence and rate-of-convergence results is an immediate corollary of the following “Magic Inequality:”

(!) *With convex and continuously differentiable $\omega(\cdot) : X \rightarrow \mathbf{R}$ for all $x \in X$, $\xi \in \mathbf{R}^n$ one has:*

$$x_+ = \text{Prox}_x(\xi) \Rightarrow \forall u \in X : \langle \xi, x_+ - u \rangle \leq V_x(u) - V_{x_+}(u) - V_x(x_+)$$

where $V_x(u) = \omega(u) - [\omega(x) + \langle u - x, \nabla \omega(x) \rangle]$ is the generated by $\omega(\cdot)$ Bregman distance from u to x , $u, x \in X$.

as applied to $\omega(u) \equiv \frac{1}{2} u^T u$.

- **Justifying Magic Inequality:**

$$x_+ = \operatorname{argmin}_{u \in X} [\langle \xi, u \rangle + V_x(u)] \Rightarrow \forall u \in X : \langle \xi - \nabla \omega(x) + \nabla \omega(x_+), u - x_+ \rangle \geq 0$$

[optimality conditions]

$$\begin{aligned} \Leftrightarrow \forall u \in X : \langle \xi, x_+ - u \rangle &\leq \langle \nabla \omega(x_+) - \nabla \omega(x), u - x_+ \rangle \\ &= [\omega(u) - \omega(x) - \langle \nabla \omega(x), u - x \rangle] \\ &\quad - [\omega(u) - \omega(x_+) - \langle \nabla \omega(x_+), u - x_+ \rangle] \\ &\quad - [\omega(x_+) - \omega(x) - \langle \nabla \omega(x), x_+ - x \rangle] \\ &= V_x(u) - V_{x_+}(u) - V_x(x_+) \end{aligned}$$

- **Magic Inequality \Rightarrow Main Inequality:** As we know, with $\omega(u) = \frac{1}{2}\|u\|_2^2$ we have $\Pi_X(x - \xi) = \operatorname{Prox}_x(\xi)$. Thus,

$$\begin{aligned} x_+ &= \Pi_X(x - \gamma f'(x)) \Rightarrow x_+ = \operatorname{Prox}_x(\gamma f'(x)) \\ &\Rightarrow \forall u \in X : \langle \gamma f'(x), x_+ - u \rangle \leq V_x(u) - V_{x_+}(u) - V_x(x_+) \\ &\Rightarrow \forall u \in X : \langle \gamma f'(x), x - u \rangle \leq V_x(u) - V_{x_+}(u) + \underbrace{[\langle \gamma f'(x), x - x_+ \rangle - V_x(x_+)]}_{\delta} \end{aligned}$$

With our $\omega(\cdot)$, $V_x(x_+) = \frac{1}{2}\|x - x_+\|_2^2$, whence

$$\delta = \langle \gamma f'(x), x - x_+ \rangle - \frac{1}{2}\|x - x_+\|_2^2 \leq \frac{1}{2}\|\gamma f'(x)\|_2^2,$$

and we arrive at the Main Inequality.

Distance-Generating Functions

- ♣ Let $\|\cdot\|$ be a norm on \mathbf{R}^n . A function $\omega(\cdot) : X \rightarrow \mathbf{R}$ is called *Distance-Generating Function (DGF) for X compatible with $\|\cdot\|$* , if
- $\omega(\cdot) : X \rightarrow \mathbf{R}$ is convex and continuously differentiable
 - $\omega(\cdot)$ is strongly convex, modulus 1, w.r.t. $\|\cdot\|$, that is,

$$\forall x, y \in X : \langle \nabla \omega(x) - \nabla \omega(y), x - y \rangle \geq \|y - x\|^2$$

or, equivalently,

$$\forall (x \in X, u \in X) : V_x(u) := \omega(u) - \omega(x) - \langle u - x, \nabla \omega(x), \rangle \geq \frac{1}{2} \|u - x\|^2.$$

Note: For every convex compact set $X \subset \mathbf{R}^n$, the function $\omega(u) = \frac{1}{2} \|u\|_2^2$ restricted to X is a DGF compatible with $\|\cdot\| = \|\cdot\|_2$. For this DGF, $V_x(y) = \frac{1}{2} \|y - x\|_2^2$.

$$\forall (x \in X, u \in X) : V_x(u) := \omega(u) - \omega(x) - \langle \nabla \omega(x), u - x \rangle \geq \frac{1}{2} \|u - x\|^2.$$

Fact: Whenever $\omega(\cdot)$ is a DGF for X compatible with $\|\cdot\|$, for $x \in X$, $\xi \in \mathbf{R}^n$, the prox-mapping

$$x_+ = \text{Prox}_x(\xi) := \operatorname{argmin}_{u \in X} [\langle \xi, u \rangle + V_x(u)]$$

is well-defined, takes values in X , and ensures that

$$\forall (u \in X) : \langle \xi, x_+ - u \rangle \leq V_x(u) - V_{x_+}(u) - V_x(x_+), \quad (1)$$

whence also

$$\forall (u \in X) : \langle \xi, x - u \rangle \leq V_x(u) - V_{x_+}(u) + \frac{1}{2} \|\xi\|_*^2, \quad (2)$$

where $\|\cdot\|_*$ is the norm conjugate to $\|\cdot\|$:

$$\|\xi\|_* = \max_x \{ \langle \xi, x \rangle : \|x\| \leq 1 \}.$$

$$V_x(u) = \omega(u) - \omega(x) - \langle u - x, \nabla \omega(x) \rangle \geq \frac{1}{2} \|u - x\|^2$$

$$x_+ = \text{Prox}_x(\xi) := \operatorname{argmin}_{u \in X} [\langle \xi, u \rangle + V_x(u)]$$

Claims:

$$\forall (u \in X) : \langle \xi, x_+ - u \rangle \leq V_x(u) - V_{x_+}(u) - V_x(x_+) \quad (1)$$

$$\forall (u \in X) : \langle \xi, x - u \rangle \leq V_x(u) - V_{x_+}(u) + \frac{1}{2} \|\xi\|_*^2 \quad (2)$$

Indeed, as we have seen, (1) follows from optimality conditions as applied to the problem defining x_+ . To derive (2) from (1), we need to show that

$$\langle \xi, x - x_+ \rangle - V_x(x_+) \leq \frac{1}{2} \|\xi\|_*^2,$$

which is immediate due to

$$\langle \xi, x - x_+ \rangle \leq \|\xi\|_* \|x - x_+\| \quad \& \quad V_x(x_+) \geq \frac{1}{2} \|x - x_+\|^2.$$

♣ **Conclusion:** *Subgradient Descent step*

$$x \mapsto x_+ = \Pi_X(x - \gamma f'(x)) \quad (1)$$

is nothing but the prox-step

$$\begin{aligned} x \mapsto x_+ &= \operatorname{argmin}_{y \in X} [\langle \gamma f'(x), y \rangle + V_x(y)] \\ V_x(y) &= \omega(y) - [\omega(x) + \langle y - x, \nabla \omega(x) \rangle] \end{aligned} \quad (*)$$

associated with the specific distance-generating function

$$\omega(u) = \frac{1}{2} u^T u \quad (2)$$

$$X \ni x \mapsto x_+ = \operatorname{argmin}_{y \in X} [\langle \xi, y \rangle + V_x(y)] \quad (*)$$

$$\Rightarrow \forall (u \in X) : \langle \xi, x - u \rangle \leq V_x(u) - V_{x_+}(u) + \frac{1}{2} \|\xi\|_*^2 \quad (2)$$

$$\left[\begin{array}{l} V_x(u) = \omega(u) - [\langle u - x, \nabla \omega(x) \rangle + \omega(x)] \\ \omega(z) : X \rightarrow \mathbf{R} : \text{continuously differentiable \& } \langle \nabla \omega(x) - \nabla \omega(y), x - y \rangle \geq \|x - y\|^2 \end{array} \right]$$

♣ **Building block #2: the potential.** Convergence analysis of SD was based on the ensured by SD step inequality

$$\begin{aligned} \forall u \in X : \gamma \langle f'(x), x - u \rangle &\leq \underbrace{\frac{1}{2} \|x - u\|_2^2 - \frac{1}{2} \|x_+ - u\|_2^2}_{= [\frac{1}{2} x^T x - x^T u] - [\frac{1}{2} x_+^T x_+ - x_+^T u]} + \frac{1}{2} \|\gamma f'(x)\|_2^2 \\ &= V_x(u) - V_{x_+}(u) \end{aligned} \quad (3)$$

where V_x stems from $\omega(\cdot) = \frac{1}{2} \|\cdot\|_2^2$. This inequality states that when $\omega(\cdot) = \frac{1}{2} \|\cdot\|_2^2$, a SD iteration $x \mapsto x_+$ reduces the “potential” – the Bregman distance

$$V_x(u) = \omega(u) - [\omega(x) + \langle u - x, \nabla \omega(x) \rangle] = \frac{1}{2} (u - x)^T (u - x)$$

from $u \in X$ to the iterate by at least $\gamma \langle f'(x), x - u \rangle - O(\gamma^2)$.

♠ (2) says that *when $\omega(\cdot)$ is continuously differentiable and strongly convex, modulus 1 w.r.t. $\|\cdot\|$, on X :*

$$\langle \nabla \omega(u) - \nabla \omega(v), u - v \rangle \geq \|u - v\|^2 \quad \forall u, v \in X$$

prox-step $x \mapsto x_+ = \operatorname{argmin}_{y \in X} [\langle \gamma f'(x), y \rangle + V_x(y)]$ ensures inequality similar to (3):

$$\begin{aligned} \forall u \in X : \gamma \langle f'(x), x - u \rangle &\leq V_x(u) - V_{x_+}(u) + \frac{1}{2} \gamma^2 \|f'(x)\|_*^2 \\ &[\|\xi\|_* = \max_u \{ \langle \xi, u \rangle : \|u\| \leq 1 \}] \end{aligned} \quad (!)$$

Non-Euclidean SD – Mirror Descent

$$\min_{x \in X} f(x) \quad (P)$$

- X : convex compact set in Euclidean space E
- f : Lipschitz continuous convex function on X
- ♣ **Setup for MD ("Proximal Setup")** is given by

— norm $\|\cdot\|$ on E

— DGF (Distance-Generating Function) $\omega(\cdot) : X \rightarrow \mathbf{R}$ which should be continuously differentiable and strongly convex, modulus 1 w.r.t. $\|\cdot\|$, function on X :

$$\langle \nabla \omega(u) - \nabla \omega(v), u - v \rangle \geq \|u - v\|^2 \forall u, v \in X$$

♠ $\omega(\cdot)$ and $\|\cdot\|$ define the important parameter — ω -capacity of X

$$\Theta = \max_{u, v \in X} [V_v(u) := \omega(u) - \omega(v) - \langle \nabla \omega(v), u - v \rangle]$$

Note: With “Ball setup” $\omega(u) = \frac{1}{2} \langle u, u \rangle$, $\|u\| \equiv \|u\|_2 = \sqrt{\langle u, u \rangle}$ one has

$$\Theta = \frac{1}{2} \max_{u, v \in X} \|u - v\|_2^2$$

♣ As applied to (P) , MD generates search points x_t according to

$$x_1 \in X, \quad x_{t+1} = \text{Prox}_{x_t}(\gamma_t f'(x_t)) := \underset{y \in X}{\operatorname{argmin}} [\langle \gamma_t f'(x_t), y \rangle + V_{x_t}(y)], \quad (MD)$$

$$V_x(y) = \omega(y) - [\omega(x) + \langle y - x, \nabla \omega(x) \rangle]$$

where $\gamma_t > 0$ are stepsizes.

$$x_{t+1} = \text{Prox}_{x_t}(\gamma_t f'(x_t)) := \underset{y \in X}{\operatorname{argmin}} [\langle \gamma_t f'(x_t), y \rangle + V_{x_t}(y)] \quad (\text{MD})$$

Note:

- With Ball setup, (MD) becomes exactly the SD recurrence

$$x_{t+1} = \Pi_X(x_t - \gamma_t f'(x_t))$$

- In order for (MD) to be practical, a step should be easy to implement. Thus, X and $\omega(\cdot)$ should fit each other, meaning that auxiliary problems

$$\min_{y \in X} [\langle \zeta, y \rangle + \omega(y)]$$

should be easy to solve.

Why and how MD converges?

$$\boxed{\begin{aligned} \{\min_{x \in X} f(x), \omega(\cdot)\} &\Rightarrow x_{t+1} = \operatorname{argmin}_{y \in X} [\langle \gamma_t f'(x_t), y \rangle + V_{x_t}(y)] \\ V_x(y) &= \omega(y) - [\omega(x) + \langle y - x, \nabla \omega(x) \rangle] \end{aligned}}$$

We have seen that MD step ensures inequality

$$\forall u \in X : \gamma_t \langle f'(x_t), x_t - u \rangle \leq V_{x_t}(u) - V_{x_{t+1}}(u) + \frac{1}{2} \gamma_t^2 \|f'(x_t)\|_*^2$$

It follows that for positive integers $T_0 \leq T$ one has

$$\begin{aligned} \sum_{t=T_0}^T \gamma_t \underbrace{\langle f'(x_t), x_t - u \rangle}_{\geq f(x_t) - f(u)} &\leq V_{x_{T_0}}(u) - V_{x_{T+1}}(u) + \frac{1}{2} \sum_{t=T_0}^T \gamma_t^2 \|f'(x_t)\|_*^2 \leq \Theta + \frac{1}{2} \sum_{t=T_0}^T \gamma_t^2 \|f'(x_t)\|_*^2 \\ &\quad [\Theta = \max_{u,v \in X} V_u(v)] \end{aligned} \quad (!)$$

For MD, relation (!) plays the same crucial role that the inequality

$$\sum_{t=T_0}^T \gamma_t \langle f'(x_t), x_t - u \rangle \leq \frac{1}{2} \max_{x,y \in X} \|x - y\|_2^2 + \frac{1}{2} \sum_{t=T_0}^T \gamma_t^2 \|f'(x_t)\|_2^2$$

played for SD. Specifically, (!) implies that

$$\boxed{\epsilon_T \equiv \min_{t \leq T} f(x_t) - f_* \leq \frac{\Theta + \frac{1}{2} \sum_{t=T_0}^T \gamma_t^2 \|f'(x_t)\|_*^2}{\sum_{t=T_0}^T \gamma_t}}$$

$$\epsilon_T \equiv \min_{t \leq T} f(x_t) - f_* \leq \frac{\Theta + \frac{1}{2} \sum_{t=T_0}^T \gamma_t^2 \|f'(x_t)\|_*^2}{\sum_{t=T_0}^T \gamma_t}$$

As a result,

♣ [Convergence with “divergent series” stepsizes] *Whenever $0 < \gamma_t \rightarrow 0$ as $t \rightarrow \infty$ in such a way that $\sum_t \gamma_t = \infty$, one has $\epsilon_T \rightarrow 0$ as $T \rightarrow \infty$*

♣ [Optimal stepsize policy] *With stepsizes $\gamma_t = \frac{\sqrt{2\Theta}}{\|f'(x_t)\|_* \sqrt{t}}$, one has*

$$\epsilon_T \equiv \min_{t \leq T} f(x_t) - f_* \leq O(1) \frac{\sqrt{\Theta} L_{\|\cdot\|}(f)}{\sqrt{T}}$$

where $L_{\|\cdot\|}(f)$ is the Lipschitz constant of f w.r.t. the norm $\|\cdot\|$.

$$\{f_* = \min_{x \in X} f(x), \omega(\cdot) : X \rightarrow \mathbf{R}, \Theta = \max_{u,v \in X} [\omega(u) - \omega(v) - \langle \nabla \omega(v), u - v \rangle]\}$$

$$\Rightarrow x_{t+1} = \operatorname{argmin}_{y \in X} [\langle \gamma_t f'(x_t), y \rangle + V_{x_t}(y)], \gamma_t = \frac{\sqrt{\Theta}}{\|f'(x_t)\|_* \sqrt{t}}$$

$$\Rightarrow \min_{t \leq T} f(x_t) - f_* \leq O(1) \frac{\sqrt{\Theta} L_{\|\cdot\|}(f)}{\sqrt{T}}$$

♠ To get the usual SD, one uses

♣ **Ball setup** $\omega(u) = \frac{1}{2} \|u\|_2^2$, $\|\cdot\| = \|\cdot\|_2$ [$X \subset \{x : \|x\|_2 \leq R\} \Rightarrow \Theta \leq \frac{1}{2} R^2$]

♥ There are several other important setups:

♣ **Simplex setup:** $\|\cdot\| = \|\cdot\|_1$, $X \subset \Delta_n = \{x \in \mathbf{R}^n : x \geq 0, \sum_i x_i \leq 1\}$

$$\omega(x) = (1 + \delta) \sum_i (x_i + \delta/n) \ln(x_i + \delta/n), \delta = 10^{-16}$$

resulting in

$$\Theta \leq O(1) \ln(n + 1)$$

♣ ℓ_1/ℓ_2 **setup:** $X \subset \mathbf{R}^{k_1} \times \mathbf{R}^{k_2} \times \dots \times \mathbf{R}^{k_n}$,

$$\omega([x^1; \dots; x^n]) = O(1) \left[\sum_{i=1}^n \|x^i\|_2^{\pi_n} \right]^{2/\pi_n}, \quad \pi_n = 1 + \frac{1}{n}$$

$$\|[x^1; \dots; x^n]\| = \sum_i \|x^i\|_2$$

resulting in

$$X \subset \{x : \|x\| \leq R\} \Rightarrow \Theta \leq O(1) \ln(n+1) R^2$$

Note:

- When $k_i = 1$ for all i , $\|\cdot\|$ becomes $\|\cdot\|_1$ and $\omega(x)$ becomes strongly convex with modulus 1, w.r.t. $\|\cdot\|_1$, on the entire \mathbf{R}^n .
- When $n = 1$, $\|\cdot\|$ becomes $\|\cdot\|_2$, and $\omega(u)$ becomes $\frac{1}{2}\|u\|_2^2$

♣ **Nuclear norm setup:** $X \subset \mathbf{R}^{p \times q}$,

$$\omega(x) = O(1) \left[\sum_{i=1}^n \sigma_i^{\pi_n}(x) \right]^{2/\pi_n}$$

$$\left[n = \min[p, q], \pi_n = 1 + \frac{1}{n}, \sigma_i(x) : \text{singular values of } x \right]$$

$$\|x\| = \|x\|_{\text{nuc}} := \sum_i \sigma_i(x)$$

resulting in

$$X \subset \{x : \|x\| \leq R\} \Rightarrow \Theta \leq O(1) \ln(n+1) R^2$$

Justifying Simplex setup: It is easily seen that ω is strongly convex, modulus 1, w.r.t. $\|\cdot\|$, iff

$$\langle \nabla^2 \omega(x) h, h \rangle \geq \|h\|^2 \quad \forall x \in X \forall h$$

For $x \in \Delta_n$ and $\bar{\omega}(x) = \sum_i (x_i + n^{-1}\delta) \ln(x_i + n^{-1}\delta)$, setting $\bar{x}_i = x_i + n^{-1}\delta$, one has

$$\begin{aligned} \|h\|_1^2 &= \left[\sum_i |h_i| \right]^2 = \left[\sum_i (|h_i|/\sqrt{\bar{x}_i}) \sqrt{\bar{x}_i} \right]^2 \leq \left[\sum_i h_i^2 / \bar{x}_i \right] \left[\sum_i \bar{x}_i \right] \\ &\leq (1 + \delta) \left(\sum_i h_i^2 / \bar{x}_i \right) = (1 + \delta) \langle h, \nabla^2 \bar{\omega}(x) h \rangle, \end{aligned}$$

whence $\omega(x) := (1 + \delta)\bar{\omega}(x)$ is strongly convex, modulus 1 w.r.t. $\|\cdot\|_1$, on Δ_n .

Next, for $x, y \in \Delta_n$, setting $\bar{y}_i = y_i + \delta n^{-1}$, $\bar{x}_i = x_i + \delta n^{-1}$, we have

$$\begin{aligned} \omega(y) - \omega(x) - \langle \nabla \omega(x), y - x \rangle &= (1 + \delta) \left[\sum_i \bar{y}_i \ln \bar{y}_i - \sum_i \bar{x}_i \ln \bar{x}_i - \sum_i (1 + \ln \bar{x}_i) (\bar{y}_i - \bar{x}_i) \right] \\ &= (1 + \delta) \left[\sum_i \bar{y}_i \ln(\bar{y}_i / \bar{x}_i) + \sum_i [\bar{x}_i - \bar{y}_i] \right] \\ &\leq (1 + \delta) \left[\sum_i \bar{y}_i \ln((n + \delta)/\delta) + 1 \right] \leq O(1) \ln n. \end{aligned}$$

$$f_* = \min_{x \in X} f(x) \quad (P)$$

♣ Let us compare the convergence properties of MD with Simplex setup and SD (i.e., MD with Ball setup).

• Observe that in order to apply MD with Simplex setup, X should be a subset of the standard simplex. We can ensure this requirement by scaling and translating the original feasible domain. As a result, MD with Simplex setup becomes applicable to an *arbitrary* convex problem (P) with compact feasible domain X , and the efficiency estimate for the method becomes

$$\epsilon_T[\text{Simplex setup}] = \min_{t \leq T} f(x_t) - f_* \leq E_{\text{simplex}}(T) := O(1) \ln^{1/2}(n) \overbrace{\max_{x,y \in X} \|x - y\|_1 L_{\|\cdot\|_1}(f)}^{\text{Var}_{\|\cdot\|_1, X}(f)} / \sqrt{T} \quad (S)$$

while for SD the efficiency estimate is

$$\epsilon_T[\text{Ball setup}] = \min_{t \leq T} f(x_t) - f_* \leq E_{\text{ball}}(T) := O(1) \overbrace{\max_{x,y \in X} \|x - y\|_2 L_{\|\cdot\|_2}(f)}^{\text{Var}_{\|\cdot\|_2, X}(f)} / \sqrt{T} \quad (B)$$

The ratio of the right hand side bounds in the estimates is

$$\frac{E_{\text{simplex}}(T)}{E_{\text{ball}}(T)} = O(\sqrt{\ln n}) \cdot \underbrace{\left[\frac{\max_{x,y \in X} \|x - y\|_1}{\max_{x,y \in X} \|x - y\|_2} \right]}_A \cdot \underbrace{\left[\frac{L_{\|\cdot\|_1}(f)}{L_{\|\cdot\|_2}(f)} \right]}_B$$

$$\frac{E_{\text{simplex}}(T)}{E_{\text{ball}}(T)} = O(\sqrt{\ln n}) \cdot \underbrace{\left[\frac{\max_{x,y \in X} \|x - y\|_1}{\max_{x,y \in X} \|x - y\|_2} \right]}_A \cdot \underbrace{\left[\frac{L_{\|\cdot\|_1}(f)}{L_{\|\cdot\|_2}(f)} \right]}_B$$

- **Small (large)** ratio $\frac{E_{\text{simplex}}(T)}{E_{\text{ball}}(T)}$ means that *as far as theoretical accuracy guarantees are concerned, Simplex setup is much better (worse) than Ball setup.*
- The factor $O(\sqrt{\ln n})$ is “against” Simplex setup; however, in practice this factor is just a moderate absolute constant.
- Note that $\frac{\|u\|_1}{\|u\|_2}$ is always ≥ 1 and, depending on x , can be as large as \sqrt{n} . Therefore
 - factor A is always ≥ 1 (i.e., is “against” Simplex setup). Depending on the geometry of X , it can be as small as 1 and as large as \sqrt{n}
 - factor B is always ≤ 1 (i.e., is “in favour” of Simplex setup) and can be as small as $\frac{1}{\sqrt{n}}$. The actual value of B is

$$\frac{L_{\|\cdot\|_1}(f)}{L_{\|\cdot\|_2}(f)} = \frac{\max_{x \in X} \|f'(x)\|_\infty}{\max_{x \in X} \|f'(x)\|_2}$$

and depends on the “geometry” of f . For example,

— when all first order partial derivatives of f in X are of the same order (“ f is nearly equally sensitive to all variables”), we have

$$B = O\left(\frac{\|(a, \dots, a)^T\|_\infty}{\|(a, \dots, a)^T\|_2}\right) = O(n^{-1/2})$$

— when just $O(1)$ first order derivatives of f on X are of the same order, and the remaining derivatives are negligible small (“ f is sensitive to just $O(1)$ variables”), we have

$$B = O\left(\frac{\|(a, 0, \dots, 0)^T\|_\infty}{\|(a, 0, \dots, 0)^T\|_2}\right) = O(1)$$

♣ **Conclusion:** The performance ratio χ depends on the geometry of X and f .

$$\chi = \frac{E_{\text{simplex}}(T)}{E_{\text{ball}}(T)} = O(\sqrt{\ln n}) \cdot \underbrace{\left[\frac{\max_{x,y \in X} \|x - y\|_1}{\max_{x,y \in X} \|x - y\|_2} \right]}_A \cdot \underbrace{\left[\frac{L_{\|\cdot\|_1}(f)}{L_{\|\cdot\|_2}(f)} \right]}_B$$

$$1 \leq A \leq \sqrt{n} \quad 1 \geq B \geq 1/\sqrt{n}$$

Extreme example I: X is a ball. In this case, $A = \sqrt{n}$, and since $B \geq \frac{1}{\sqrt{n}}$, $\chi \geq 1$ – method with Ball setup (i.e., the classical SD) outperforms the method with Simplex setup by factor which varies from $O(\sqrt{\ln n})$ (f is nearly equally sensitive to all variables) to $O(\sqrt{n \ln n})$ (f is sensitive to just $O(1)$ variables).

Extreme example II: X is the unit simplex Δ_n . In this case, $A = O(1)$, and since $B \leq 1$ and $O(\sqrt{\ln n})$ in practice a moderate absolute constant, $\chi \leq O(1)$ – method with Simplex setup outperforms the classical SD by factor which varies from $O\left(\sqrt{\frac{n}{\ln n}}\right)$ (f is nearly equally sensitive to all variables) to $O\left(\sqrt{\frac{1}{\ln n}}\right)$ (f is sensitive to just $O(1)$ variables).

Conclusion: Flexibility in setup allows to adjust MD, to some extent, to the geometry of the problem to be solved. Let all flowers blossom!

Application example: Positron Emission Tomography Image Reconstruction

♣ The Maximum Likelihood estimate of tracer's density in PET is

$$\lambda_* = \operatorname{argmin}_{\lambda \geq 0} \left\{ \sum_{j=1}^n p_j \lambda_j - \sum_{i=1}^m y_i \ln \left(\sum_{j=1}^n p_{ij} \lambda_j \right) \right\}$$

$[y_i \geq 0 \text{ are observations, } p_{ij} \geq 0, p_j = \sum_i p_{ij}]$

The KKT optimality conditions read

$$\lambda_j \left(p_j - \sum_i y_i \frac{p_{ij}}{\sum_\ell p_{i\ell} \lambda_\ell} \right) = 0 \quad \forall j,$$

whence, taking sum over j ,

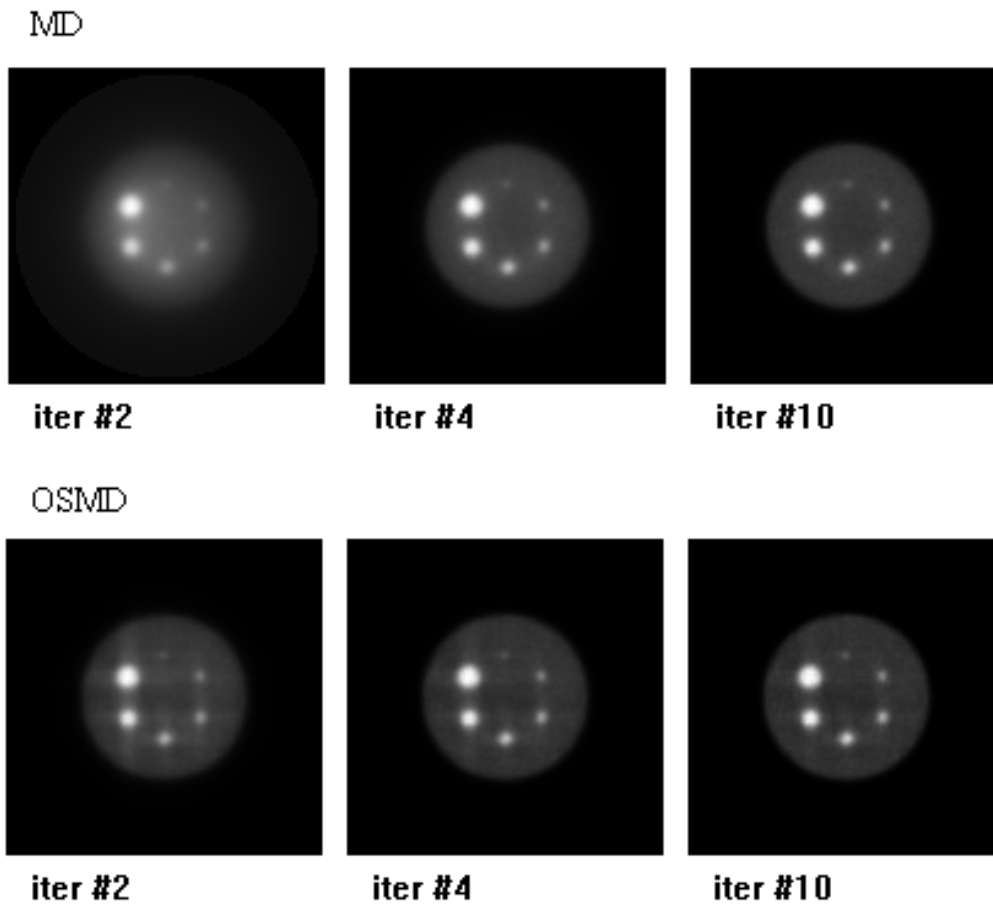
$$\sum_j p_j \lambda_j = B \equiv \sum_i y_i.$$

Thus, in fact (PET) is the problem of minimizing over a simplex. Passing to the variables $x_j = p_j B^{-1} \lambda_j$, we end up with the problem

$$\min_x \left\{ f(x) = - \sum_i y_i \ln \left(\sum_j q_{ij} x_j \right) : x \in \Delta_n \right\} \quad (\text{PET})$$

$[q_{ij} = B p_{ij} p_j^{-1}]$

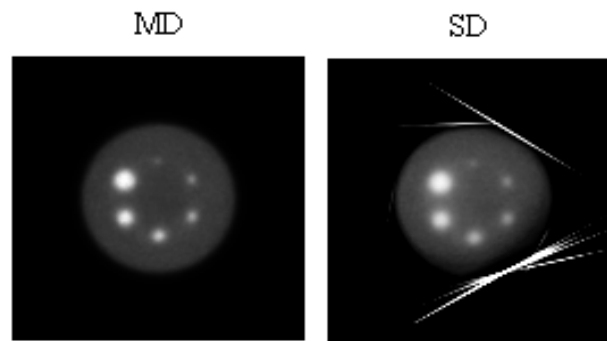
♣ Illustration: “Hot Spheres” phantom ($n = 515,871$)



Itr	1	2	3	4	5	6	7	8	9	10
$f(x_t)$	-4.295	-4.767	-5.079	-5.189	-5.168	-5.230	-5.181	-5.227	-5.189	-5.225

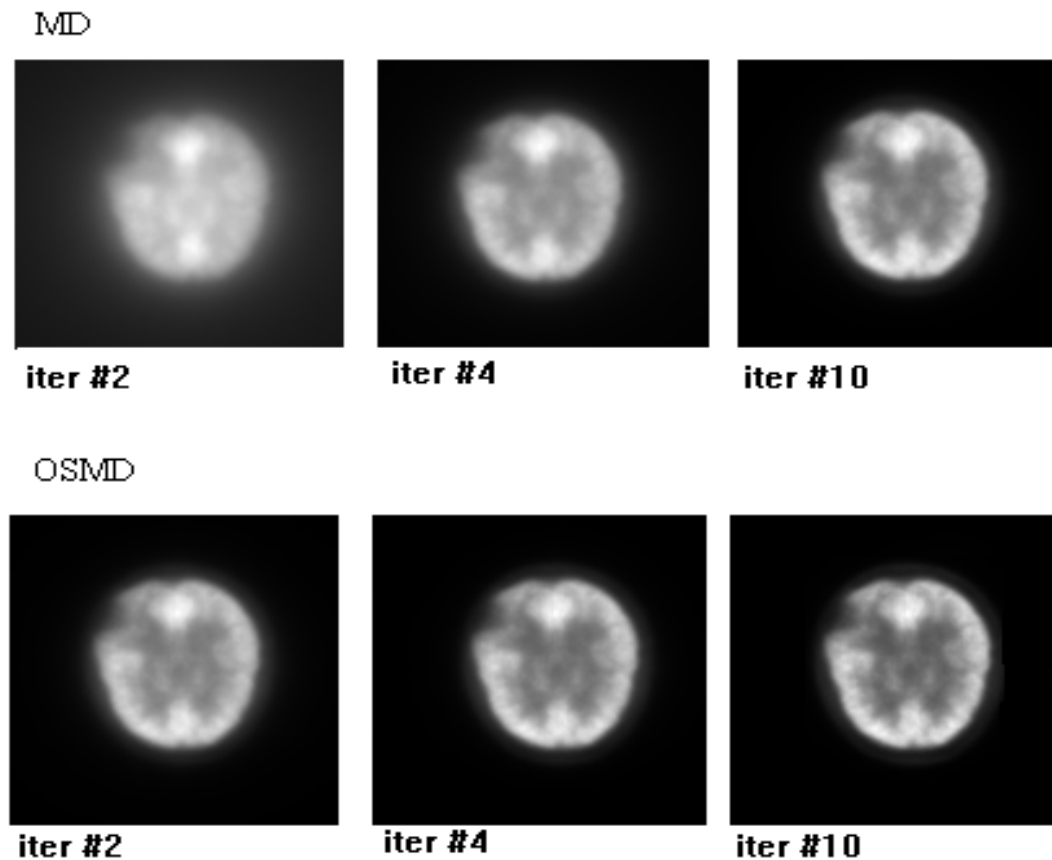
$[f_* \geq -5.283]$

Simplex setup. Progress in accuracy in 10 iterations by factor 21.4



Simplex setup (left) vs. Ball setup (right) progress in accuracy 21.4 vs. 5.26

♣ Illustration: Brain clinical data ($n = 2,763,635$)



Itr	1	2	3	4	5	6	7	8	9	10
$f(x_t)$	-1.463	-1.848	-2.001	-2.012	-2.015	-2.015	-2.016	-2.016	-2.016	-2.016

$[f_* \geq -2.050]$

Simplex setup. Progress in accuracy in 10 iterations by factor 17.5

Mirror-Level Algorithm

♣ Same as SD, the general Mirror Descent admits a version with memory – Mirror Level (ML) algorithm. The setup for ML is similar to the one of MD and is given by a norm $\|\cdot\|$ on E and a continuously differentiable and strongly convex, modulus 1 w.r.t. $\|\cdot\|$, DGF $\omega(\cdot) : X \rightarrow \mathbf{R}$.

♣ At step t of ML, we

— compute $f(x_t), f'(x_t)$ and build the current model of f

$$f_t(x) = \max_{\tau \leq t} [f(x_\tau) + \langle f'(x_\tau), x - x_\tau \rangle]$$

which underestimates the objective and is exact at the points x_1, \dots, x_t ;

— define the *best found so far value of the objective* $f^t = \min_{\tau \leq t} f(x_\tau)$

— define the current *lower bound* f_t on f_* by solving the auxiliary problem

$$f_t = \min_{x \in X} f_t(x)$$

The current *gap* $\Delta_t = f^t - f_t$ is an upper bound on the inaccuracy of the best found so far approximate solution;

— compute the current *level* $\ell_t = f_t + \lambda \Delta_t$ ($\lambda \in (0, 1)$ is a parameter)

— finally, we set

$$L_t = \{x \in X : f_t(x) \leq \ell_t\},$$

$$x_{t+1} = \text{Prox}_{x_t}^{L_t}(0) := \underset{x \in L_t}{\operatorname{argmin}} [\langle -\nabla \omega(x_t), x \rangle + \omega(x)]$$

and loop to step $t + 1$.

♠ With Ball setup,

$$\text{Prox}_{x_t}^{L_t}(0) = \underset{x \in L_t}{\operatorname{argmin}} \left[-x_t^T x + \frac{1}{2} x^T x \right] = \underset{x \in L_t}{\operatorname{argmin}} \frac{1}{2} \|x - x_t\|_2^2.$$

i.e., the method becomes exactly the BL algorithm.

Efficiency Estimate for ML

Fact: For every ϵ , $0 < \epsilon < \Delta_1$, the number N of steps of ML before a gap $\leq \epsilon$ is obtained (i.e., before an ϵ -solution is found) does not exceed the bound

$$N(\epsilon) = \frac{4\Theta L_{\|\cdot\|}^2(f)}{\lambda(1-\lambda)^2(2-\lambda)\epsilon^2}.$$

$$[\Theta = \max_{x,y \in X} \{V_x(y) := \omega(y) - \omega(x) - \langle y - x, \nabla \omega(x) \rangle\}]$$

In particular, for ℓ_1/ℓ_2 and Nuclear Norm setups one has

$$N(\epsilon) = O(\ln n) \frac{(\max_{x,y \in X} \|x - y\| L_{\|\cdot\|}(f))^2}{\lambda(1-\lambda)^2(2-\lambda)\epsilon^2}.$$

with $\|\cdot\|$ and n defined in the descriptions of the setups.

Claim: For every ϵ , $0 < \epsilon < \Delta_1$, the number N of steps of ML before a gap $\leq \epsilon$ is obtained (i.e., before an ϵ -solution is found) does not exceed the bound

$$N(\epsilon) = \frac{4\Theta L_{\|\cdot\|}^2(f)}{\lambda(1-\lambda)^2(2-\lambda)\epsilon^2}.$$

In particular, for ℓ_1/ℓ_2 and Nuclear Norm setups one has

$$N(\epsilon) = O(\ln n) \frac{(\max_{x,y \in X} \|x - y\| L_{\|\cdot\|}(f))^2}{\lambda(1-\lambda)^2(2-\lambda)\epsilon^2}.$$

with $\|\cdot\|$ and n defined in the descriptions of the setups.

♣ Similar to Claim result for BL was derived from the following fact:

Let $J = \{s, s+1, \dots, r\}$ be a segment of iterations of BL:

$$\Delta_r \geq (1-\lambda)\Delta_s.$$

Then the cardinality of J can be upper-bounded as $\text{Card}(J) \leq \frac{(\max_{x,y \in X} \|x-y\|_2 L_{\|\cdot\|_2}(f))^2}{(1-\lambda)^2 \Delta_r^2}$.

♠ Similar fact for ML reads:

(!) Let $J = \{s, s+1, \dots, r\}$ be a segment of iterations of ML: $\Delta_r \geq (1-\lambda)\Delta_s$.

Then the cardinality of J can be upper-bounded as $\text{Card}(J) \leq \frac{2\Theta L_{\|\cdot\|}^2(f)}{(1-\lambda)^2 \Delta_r^2}$.

Claim is derived from (!) in exactly the same fashion as the in the case of BL.

(!) Let $J = \{s, s+1, \dots, r\}$ be a segment of iterations of ML: $\Delta_r \geq (1-\lambda)\Delta_s$.

Then the cardinality of J can be upper-bounded as $\text{Card}(J) \leq \frac{2\Theta L_{\|\cdot\|}^2(f)}{(1-\lambda)^2 \Delta_r^2}$.

Proof. Same as in the case of BL, we observe that

- For t running through a segment of iterations J , the level sets $L_t = \{x \in X : f_t(x) \leq \ell_t\}$ have a point in common, namely, $v \in \text{Argmin}_{x \in X} f_r(x)$;
- When $t \in J$, the distances $\gamma_t = \|x_t - x_{t+1}\|$ are not too small: $\gamma_t \geq \frac{(1-\lambda)\Delta_r}{L_{\|\cdot\|}(f)}$.
- As we shall see in a while,

$$\begin{aligned} V_{x_{t+1}}(v) &\leq V_{x_t}(v) - \frac{1}{2}\gamma_t^2, \quad t \in J \\ [V_x(y) = \omega(y) - [\langle y - x, \nabla \omega(x) \rangle + \omega(x)] &\geq \frac{1}{2}\|y - x\|^2] \end{aligned} \quad (\#)$$

Thus, while t stays within J , $V_{x_t}(v)$ decrease from step to step by at least $\frac{1}{2}\gamma_t^2$.

Since $0 \leq V_x(y) \leq \Theta$ for all $x, y \in X$, $(\#)$ combines with the lower bound on γ_t , $t \in J$, to imply the desired upper bound on the cardinality of J

$$V_{x_{t+1}}(v) \leq V_{x_t}(v) - \frac{1}{2} \|x_t - x_{t+1}\|^2, \quad t \in J \quad (\#)$$

Proof of (#). Magic Inequality says that whenever $x \in X$, $\xi \in E$ and $x_+ = \operatorname{argmin}_{y \in X} [\langle \xi - \nabla \omega(x), y \rangle + \omega(y)]$,

it holds

$$\langle \xi, x_+ - u \rangle \leq V_x(u) - V_{x_+}(u) - V_x(x_+),$$

This fact admits modification as follows:

(\$) Let $Y \subset X$ be nonempty convex compact sets in Euclidean space E and $\omega(\cdot)$ be a DGF for X compatible with a norm $\|\cdot\|$ on E . Given $x \in X$ and $\xi \in E$, let

$$x_+ = \operatorname{argmin}_{y \in Y} [\langle \xi - \nabla \omega(x), y \rangle + \omega(y)]$$

Then

$$\forall u \in Y : \langle \xi, x_+ - u \rangle \leq V_x(u) - V_{x_+}(u) - V_x(x_+).$$

Applying (\$) to $\xi = 0$, $x = x_t$, $Y = L_t$ and $u = v$, we get (#).

Proof of Modification repeats the proof of plain Magic Inequality:

$$\begin{aligned} x_+ = \operatorname{argmin}_{y \in Y} [\langle \xi - \nabla \omega(x), y \rangle + \omega(y)] &\Rightarrow \forall u \in Y : \langle \xi - \nabla \omega(x) + \nabla \omega(x_+), u - x_+ \rangle \geq 0 \\ &\quad \text{[optimality conditions]} \\ \Leftrightarrow \forall u \in Y : \langle \xi, x_+ - u \rangle &\leq \langle \nabla \omega(x_+) - \nabla \omega(x), u - x_+ \rangle \\ &= [\omega(u) - \omega(x) - \langle \nabla \omega(x), u - x \rangle] \\ &\quad - [\omega(u) - \omega(x_+) - \langle \nabla \omega(x_+), u - x_+ \rangle] \\ &\quad - [\omega(x_+) - \omega(x) - \langle \nabla \omega(x), x_+ - x \rangle] \\ &= V_x(u) - V_{x_+}(u) - V_x(x_+) \end{aligned}$$

NERML – Non-Euclidean Restricted Memory Level algorithm

$$f_* = \min_{x \in X} f(x)$$

- ♣ NERML is a Mirror Descent extension of TPL (Truncated Proximal Level) method, NERML is a version of ML where bundle size is kept at at most a given level m .
- ♣ The setup for NERML, same as those for MD and ML, is given by a norm $\|\cdot\|$ on the Euclidean space E where X lives and a continuously differentiable strongly convex, modulus 1 w.r.t. $\|\cdot\|$, Distance Generating Function $\omega(\cdot) : X \rightarrow \mathbf{R}$.
- ♣ At every step, NERML calls First Order oracle. In $t = 1, 2, \dots$ steps, NERML builds t -th approximate solution $x^t \in X$ along lower bound f_t on the minimum value f_* of f on X , so that t -th . gap $\Delta_t = f(x^t) - f_t$ upper-bounds inaccuracy of x^t in terms of the objective. The efficiency estimate of NERML is given by the following

Fact: For every ϵ , $0 < \epsilon < \Delta_1$, the total number of NERML steps before a gap $\leq \epsilon$ is obtained (i.e., before an ϵ -solution is found) does not exceed the bound

$$N(\epsilon) = O(1)\Theta L_{\|\cdot\|}^2(f)\epsilon^{-2},$$

$$\left[\Theta = \max_{x,y \in X} \{V_x(y) = \omega(y) - \omega(x) - \langle y - x, \nabla \omega(x) \rangle\} \right]$$

NERML – Construction

- ♣ Execution of NERML is split into *phases*. Phase s is associated with
 - prox-center $c_s \in X$
 - s -th upper bound f^s on f_* , which is the best value of the objective observed before the phase begins
 - s -th lower bound f_s on f_* , which is the best lower bound on f_* observed before the phase begins
- f^s and f_s define
 - ◇ s -th optimality gap $\Delta_s = f^s - f_s$
 - ◇ s -th level $\ell_s = f_s + \lambda \Delta_s$, where $\lambda \in (0, 1)$ is parameter of the method,
 - ◇ s -th local DGF $\omega_s(x) = \omega(x) - \langle \nabla \omega(c_s), x \rangle + \omega(c_s)$
- current model $\tilde{f}^s(\cdot) \leq f(\cdot)$ of $f(\cdot)$, which is the maximum of $\leq m$ affine forms.
- ♠ To initialize the first phase, we choose $c_1 \in X$, compute $f(c_1), f'(c_1)$ and set

$$\tilde{f}^1(x) = f(c_1) + \langle f'(c_1), x - c_1 \rangle, \quad f^1 = f(c_1), \quad f_1 = \min_{x \in X} \tilde{f}^1(x).$$

- ♣ At the beginning of step $t = 1, 2, \dots$ of phase s , we have at our disposal
 - upper bound $f^{s,t-1} \leq f^s$ on f_* , which is the best found so far value of the objective,
 - lower bound $f_{s,t-1} \geq f_s$ on f_* ,
 - model $\tilde{f}^{s,t-1}(\cdot) \leq f(\cdot)$ of the objective which is the maximum of $\leq m$ affine forms
 - iterate $x_t \in X$ and set

$$H_{t-1} = \{x : \langle \alpha_{t-1}, x \rangle \geq \beta_{t-1}\}$$

such that

$$\begin{aligned} x \in X, f(x) \leq \ell_s &\Rightarrow x \in H_{t-1} & (a_t) \\ x_t = \operatorname{argmin}_x \{\omega_s(x) : x \in H_{t-1} \cap X\} & & (b_t) \end{aligned}$$

- ♠ To initialize the first step of phase s , we set

$$f^{s,0} = f^s, f_{s,0} = f_s, \tilde{f}^{s,0}(\cdot) = \tilde{f}^s(\cdot), \alpha_0 = 0, \beta_0 = 0 [\Rightarrow H_0 = E]$$

thus ensuring (a_1) , and set $x_1 = c_s$, thus ensuring (b_1) .

♠ **Step t phase s :** Given

- bounds $f^{s,t-1} \geq f_*$, $f_{s,t-1} \leq f_*$ • model $\tilde{f}^{s,t-1}(\cdot) \leq f(\cdot)$,
- x_t and $H_{t-1} = \{x : \langle \alpha_{t-1}, x \rangle \geq \beta_{t-1}\}$ such that

$$x \in X, f(x) \leq \ell_s \Rightarrow x \in H_{t-1} \quad (a_t) \ \& \ x_t = \operatorname{argmin}_x \{\omega_s(x) : x \in H_{t-1} \cap X\} \quad (b_t)$$

1. we compute $f(x_t)$, $f'(x_t)$ and set

$$g_t(x) = f(x_t) + \langle f'(x_t), x - x_t \rangle;$$

2. we define $\tilde{f}^{s,t}(\cdot)$ as the maximum of $g_t(\cdot)$ and affine forms associated with $\tilde{f}^{s,t-1}$ (dropping, if necessary, one of the latter forms to make $\tilde{f}^{s,t}$ the maximum of at most m forms). If $f(x_t) \leq \ell_s + 0.5(f^s - \ell_s)$ ("progress in upper bound"), we terminate phase s and set

$$f^{s+1} = f^{s,t}, \quad f_{s+1} = f_{s,t-1}, \quad \tilde{f}^{s+1}(\cdot) = \tilde{f}^{s,t}(\cdot),$$

otherwise

3. we compute $f_t = \min_x \{\tilde{f}^{s,t}(x) : x \in H_{t-1} \cap X\}$. Since $f(x) \geq \ell_s$ in $X \setminus H_{t-1}$, we have $f_* \geq \min[\ell_s, f_t]$, so that

$$f_{s,t} \equiv \max\{f_{s,t-1}, \min[\ell_s, f_t]\} \leq f_*.$$

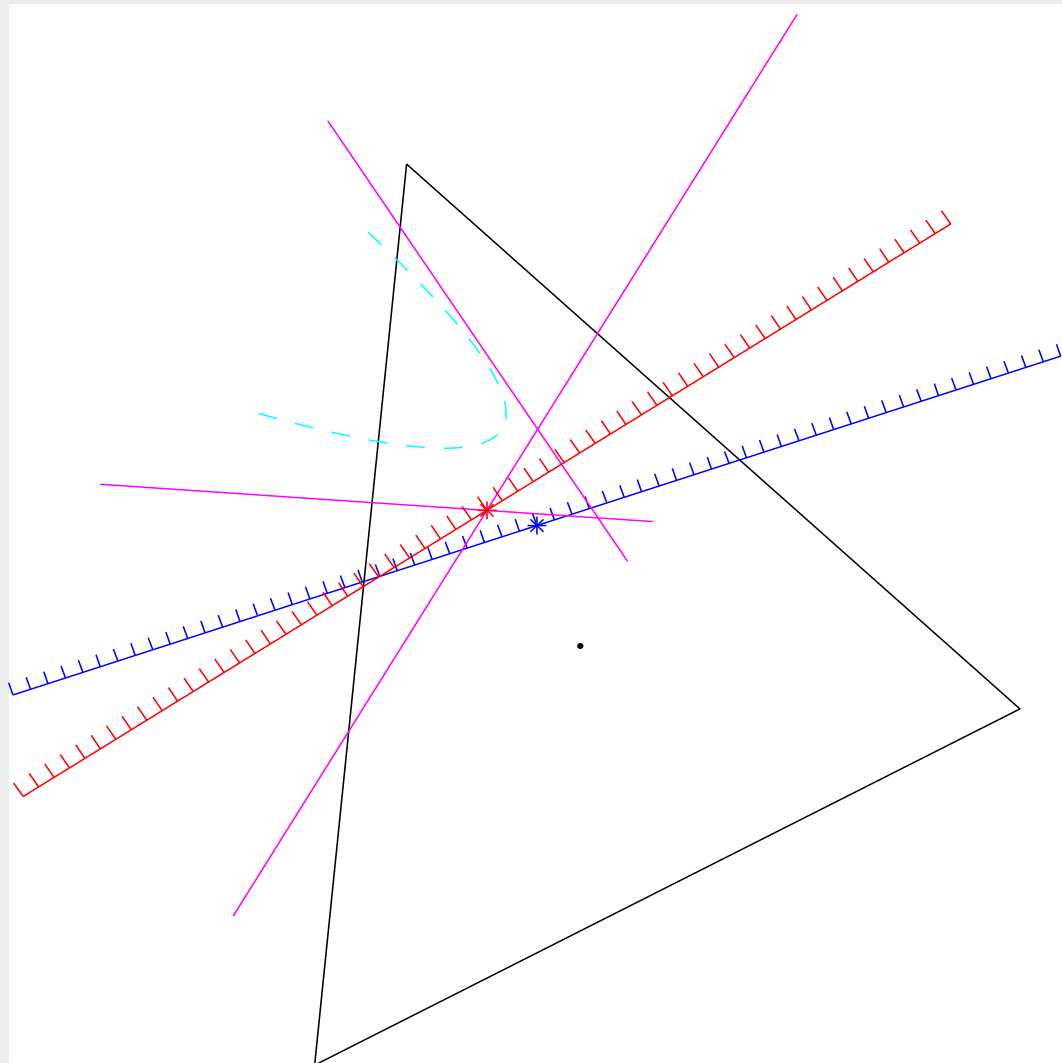
If $f_{s,t} \geq \ell_s - 0.5(\ell_s - f_s)$ ("progress in lower bound"), we terminate phase s and set

$$f^{s+1} = f^{s,t}, \quad f_{s+1} = f_{s,t}, \quad \tilde{f}^{s+1}(\cdot) = \tilde{f}^{s,t}(\cdot)$$

otherwise we set

$$\begin{aligned} x_{t+1} &= \operatorname{argmin}_x \left\{ \omega_s(x) : x \in X \cap H_{t-1}, \tilde{f}^{s,t}(x) \leq \ell_s \right\} \\ H_t &= \{x : \langle \nabla \omega_s(x_{t+1}), x - x_{t+1} \rangle \geq 0\} \end{aligned}$$

and loop to step $t + 1$ of phase s .



Step of NERML

$$x_{t+1} = \operatorname{argmin}_x \left\{ \omega_s(x) : x \in X \cap H_{t-1}, \tilde{f}^{s,t}(x) \leq \ell_s \right\} \quad (1)$$

$$H_t = \{x : \langle \nabla \omega_s(x_{t+1}), x - x_{t+1} \rangle \geq 0\} \quad (2)$$

Note: When passing to step $t + 1$, it is ensured that

$$x \in X, f(x) \leq \ell_s \Rightarrow x \in H_t \quad (a_{t+1})$$

$$x_{t+1} = \operatorname{argmin}_x \left\{ \omega_s(x) : x \in X \cap H_t, \tilde{f}^{s,t}(x) \leq \ell \right\} \quad (b_{t+1})$$

Indeed, x_{t+1} is the minimizer of $\omega_s(x)$ on the set

$$Y_t = X \cap H_{t-1} \cap \{x : \tilde{f}^{t,s}(x) \leq \ell_s\}$$

whence

$$\begin{aligned} & \langle \nabla \omega_s(x), x - x_{t+1} \rangle \geq 0 \quad \forall x \in Y_t \\ \Rightarrow Y_t & \subset H_t = \{x : \langle \nabla \omega_s(x_{t+1}), x - x_{t+1} \rangle \geq 0\} \quad (*) \end{aligned}$$

Thus,

$$(x \in X, f(x) \leq \ell_s) \underbrace{\Rightarrow}_{(a_t)} (x \in X \cap H_{t-1}, f(x) \leq \ell_s) \Rightarrow (x \in X \cap H_{t-1}, \tilde{f}^{s,t}(x) \leq \ell_s) \underbrace{\Rightarrow}_{(*)} x \in H_t$$

as required in (a_{t+1}) . (b_{t+1}) readily follows from the definition of H_t . □

Convergence of NERML

♣ The efficiency estimate for TLM was a nearly straightforward consequence of the following fact:

(*) *The number of steps of TLM at a phase s does not exceed*

$$N_s = \frac{4 \left(\max_{x,y} \|x - y\|_2 L_{\|\cdot\|_2}(f) \right)^2}{(1 - \lambda)^2 \Delta_s^2} + 1.$$

♣ For NERML, a similar fact is valid:

(!) *The number of steps of NERML at a phase s does not exceed*

$$N_s = \frac{8\Theta L_{\|\cdot\|}^2(f)}{(1 - \lambda)^2 \Delta_s^2} + 1.$$

♠ The same reasoning as in the case of TLM, with (!) playing the role of (*), yields

Corollary: *For every ϵ , $0 < \epsilon < \Delta_1$, the total number of NERML steps before a gap $\leq \epsilon$ is obtained (i.e., before an ϵ -solution is found) does not exceed the bound*

$$N(\epsilon) = c(\lambda)\Theta L_{\|\cdot\|}^2(f)\epsilon^{-2}.$$

Claim:

(!) The number of steps of NERML at a phase s does not exceed $N_s = \frac{8\Theta L_{\|\cdot\|}^2(f)}{(1-\lambda)^2 \Delta_s^2} + 1$.

Proof. Let phase s not be terminated in course of N steps. By construction, for $1 \leq t \leq N$ we have

$$\begin{aligned} & x_{t+1} \in H_{t-1} \cap X \text{ \& } x_t = \operatorname{argmin}_x \{ \omega_s(x) : x \in H_{t-1} \cap X \} \\ \Rightarrow \omega_s(x_{t+1}) & \geq \omega_s(x_t) + \underbrace{\langle \nabla \omega(x_t), x_{t+1} - x_t \rangle}_{\geq 0} + \frac{1}{2} \|x_{t+1} - x_t\|^2 \geq \omega_s(x_t) + \frac{1}{2} \|x_{t+1} - x_t\|^2 \quad (1) \end{aligned}$$

Further, when passing from x_t to $x_{t+1} = \operatorname{argmin}_x \left\{ \omega_s(x) : x \in H_{t-1} \cap X, \tilde{f}^{s,t}(x) \leq \ell_s \right\}$, the function $g_t(x) \equiv f(x_t) + \langle f'(x_t), x - x_t \rangle \leq \tilde{f}^{s,t}$ varies from the value $f(x_t) \geq f^{s,t}$ to a value $\leq \ell_s$ and thus decreases by at least $0.5(1-\lambda)\Delta_s$ (otherwise phase s would be terminated at step t due to progress in upper bound). Since $g_t(\cdot)$ is Lipschitz continuous, with constant $L_{\|\cdot\|}(f)$ w.r.t. $\|\cdot\|$, we conclude that

$$0.5(1-\lambda)\Delta_s \leq \|x_t - x_{t+1}\| L_{\|\cdot\|}(f) \Rightarrow \|x_t - x_{t+1}\| \geq \frac{0.5(1-\lambda)\Delta_s}{L_{\|\cdot\|}(f)}.$$

Applying (1), we arrive at

$$\omega_s(x_{t+1}) \geq \omega_s(x_t) + \frac{(1-\lambda)^2}{8L_{\|\cdot\|}^2(f)} \Delta_s^2, \quad 1 \leq t \leq N. \quad (2)$$

Since the function $\omega_s(x) = \omega(x) - \langle \nabla \omega(c_s), x - c_s \rangle + \omega(c_s)$ maps X into $[0, \Theta]$, (2) implies (!). \square

Implementation issues

♣ *How to solve auxiliary problems?* At a step of NERML, one should solve the auxiliary problems

$$f_t = \min_x \left\{ \tilde{f}^{s,t}(x) : x \in H_{t-1} \cap X \right\} \quad (L)$$

$$x_{t+1} = \operatorname{argmin}_x \left\{ \omega_s(x) : x \in H_{t-1} \cap X, \tilde{f}^{s,t}(x) \leq \ell_s \right\} \quad (N)$$

Formally, both (L) and (N) are problems of the same dimension as the problem of interest.

Question: Does it make sense to reduce the large-scale problem of interest to a *series* of equally large-scale auxiliary problems?

Answer: Yes, it does – (L), (N) can be easily reduced to a *low-dimensional* black-box-represented convex programs.

$$\min_x \left\{ \tilde{f}^{s,t}(x) : x \in H_{t-1} \cap X \right\} \quad (L)$$

♣ Assume that X is a simple polytope. Then (L) is an LP program and can be solved as such, unless the dimension of X is really large. In the latter case, we can solve (L) via Lagrange Duality. Indeed, the objective in (L) is the maximum of (at most) m affine functions $h_i(x)$, $i = 1, \dots, m$, while H_{t-1} is given by a single linear inequality $h_0(x) \leq 0$. Thus, (L) is the problem

$$\begin{aligned} f_t &= \min_{x \in X} \{ \max_{1 \leq i \leq m} h_i(x) : h_0(x) \leq 0 \} \\ &= \max_{\lambda} \{ F(\lambda) = \min_{x \in X} [\sum_{i=0}^m \lambda_i h_i(x)] : \lambda \geq 0, \sum_{i=1}^m \lambda_i = 1 \}. \end{aligned}$$

• In order to compute $F(\lambda)$ and $F'(\lambda)$ at a given λ , it suffices to minimize over X the linear function $h_{\lambda}(x) = \sum_{i=0}^m \lambda_i h_i(x)$. after a minimizer x_{λ} of $h_{\lambda}(\cdot)$ over X is found, one sets

$$F(\lambda) = h_{\lambda}(x_{\lambda}); \quad F'(\lambda) = (h_1(x_{\lambda}), \dots, h_m(x_{\lambda}))^T. \quad (*)$$

♣ Assuming problems $\min_{x \in X} [\langle \xi, x \rangle + \omega(x)]$ easily solvable, problem of minimizing linear objective over X is easily solvable as well. \Rightarrow it is easy to implement the First Order oracle for F

Thus, we can find f_t by solving the black-box-represented convex program

$$\max_{\lambda} \left\{ F(\lambda) : \lambda \geq 0, \sum_{i=1}^m \lambda_i = 1 \right\}$$

with dimension $m + 1$ (which is under our full control!) by, say, the Ellipsoid method.

♣ The second auxiliary problem

$$\begin{aligned} x_{t+1} &= \operatorname{argmin}_x \left\{ \omega_s(x) : x \in X \cap H_{t-1}, \tilde{f}^{s,t} \leq \ell_s \right\} \\ &= \operatorname{argmin}_{x \in X} \left\{ \omega(x) + \langle \xi_s, x \rangle : \tilde{h}_i(x) \leq 0, i = 1, \dots, m+1 \right\} \end{aligned}$$

also can be reduced to $m+1$ -dimensional black-box-represented convex program

$$\max_{\lambda \geq 0} \Phi(\lambda), \quad \Phi(\lambda) = \min_{x \in X} \left[\omega(x) + \langle \xi_s, x \rangle + \sum_{i=1}^{m+1} \lambda_i h_i(x) \right]$$

with First Order oracle readily given by the possibility to solve auxiliary problems

$$x_\lambda = \operatorname{argmin}_{x \in X} \left[\omega(x) + \langle \xi_s, x \rangle + \sum_{i=1}^{m+1} \lambda_i h_i(x) \right].$$

After $\lambda_* \in \operatorname{Argmin}_{\lambda \geq 0} \Phi(\lambda)$ is found by, e.g., the Ellipsoid method, we recover x_{t+1} as x_{λ_*} .
Note: $\omega(\cdot)$ is strongly convex, so that high-accuracy *approximate* solution to $\max_{\lambda \geq 0} \Phi(\lambda)$ results in high accuracy approximation to x_{t+1} .

⇒ With the outlined approach MD/ML/NERML become implementable under the *only* assumption that one can easily solve problems $\min_X [\langle \xi, x \rangle + \omega(x)]$. This indeed is so for

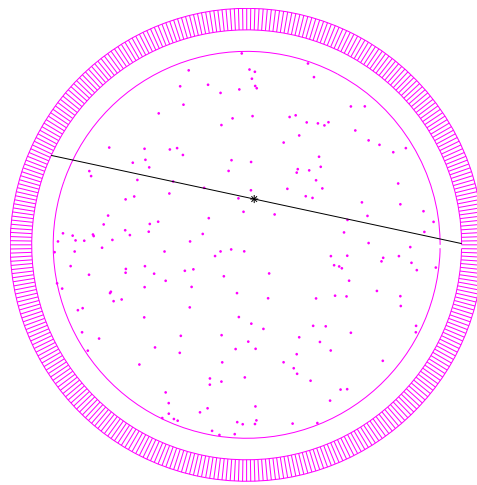
- Ball setup and simple X (ball, box, positive part of ball, standard simplex,...),
- Simplex setup and simple X (the entire simplex Δ_n , intersection of Δ_n and a box,...),
- Spectahedron setup with X comprised of block-diagonal matrices with diagonal blocks of size $O(1)$.

In all these cases, $(*)$ can be solved in $\leq O(n \ln n)$ a.o.

How It Works: PET Image Reconstruction via NERML

$$\min_x \left\{ f(x) = -\sum_{i=1}^m y_i \ln \left(\sum_{j=1}^n q_{ij} x_j \right) : x \in \Delta_n \right\} \quad (\text{PET}')$$

♣ We have simulated 2D PET scanner with a single ring of detectors:



Ring with 360 detectors, field of view and a LOR (ring's radius 1, field of view's radius 0.9)

- and field of view partitioned into pixels by 128×128 regular grid. With this setup,
- the design dimension of the problem is $n = 10,471$;
 - the number of log-terms in the objective is 39,784
 - the number of nonzero q_{ij} is 3,746,832 (the density of the matrix $[q_{ij}]$ is 0.009).
- ♣ The algorithm: plain NERML with Simplex setup, $m = 1$ and $\lambda = 0.95$.

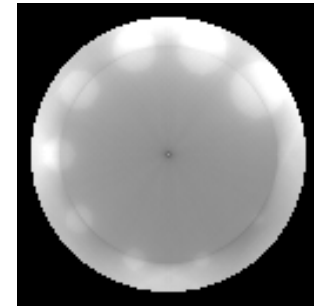
♣ Experiment 1: noiseless measurements (brighter pixels correspond to higher tracer's density):



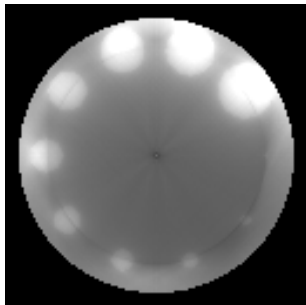
True image: 10 "hot spots"
 $f = f_* = 2.817$



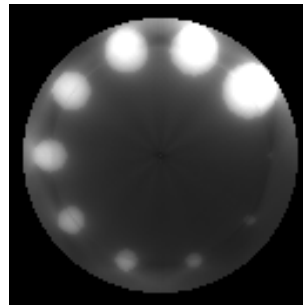
$x^1 = n^{-1}(1, \dots, 1)^T$
 $f = 3.247$



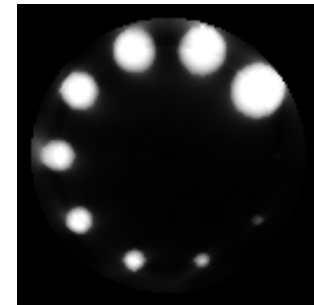
x^2 – some traces of 8 spots
 $f = 3.185$



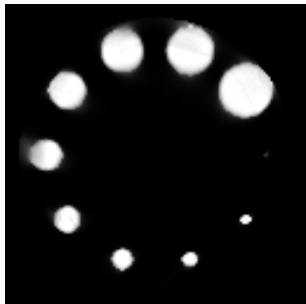
x^3 – traces of 8 spots
 $f = 3.126$



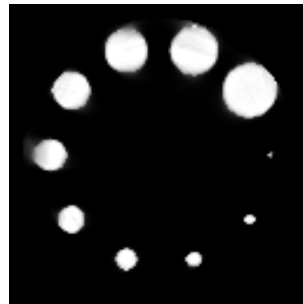
x^5 – some trace of 9-th spot
 $f = 3.016$



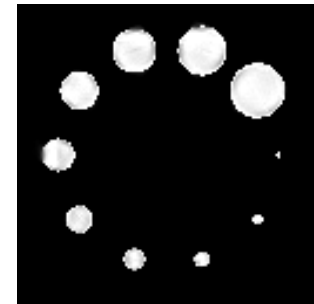
x^8 – 10-th spot still missing...
 $f = 2.869$



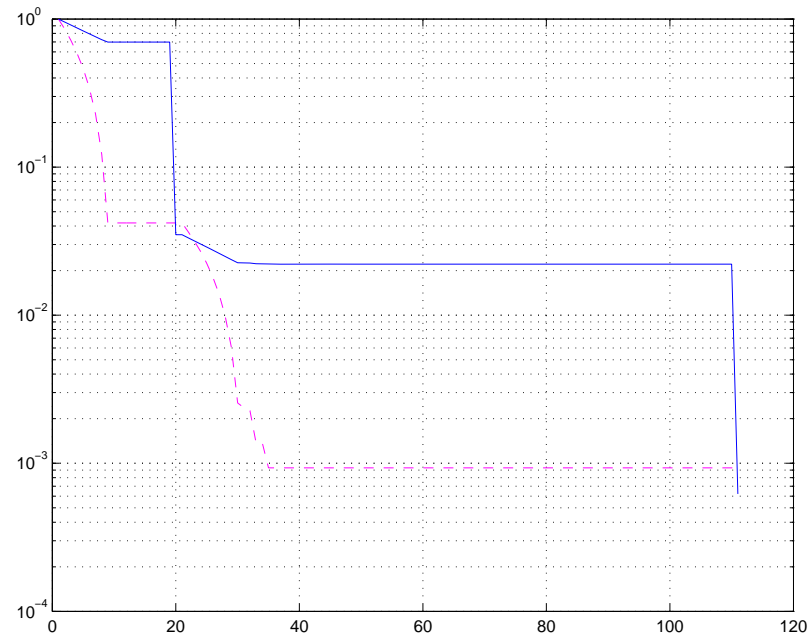
x^{24} – trace of 10-th spot
 $f = 2.828$



x^{27} – all 10 spots in place
 $f = 2.823$



x^{31} – that is it...
 $f = 2.818$



Progress in accuracy, noiseless measurements.

- solid line: Relative gap $\frac{\text{Gap}(t)}{\text{Gap}(1)}$ vs. step number t ; $\text{Gap}(t)$ is the difference between the best found so far value $f(x^t)$ of f and the current lower bound on f_* .
- In 111 steps, the gap was reduced by factor > 1600
- dashed line: Progress in accuracy $\frac{f(x^t) - f_*}{f(x^1) - f_*}$ vs. step number t
- In 111 steps, the accuracy was improved by factor > 1080
 - 111 steps of the NERML algorithm took 18'51" on a 350 MHz Pentium II laptop with 96 MB RAM.

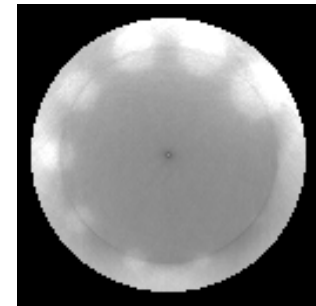
♣ Experiment 2: noisy measurements (at average, 40 LOR's per bright pixel, 63,092 LOR's totally):



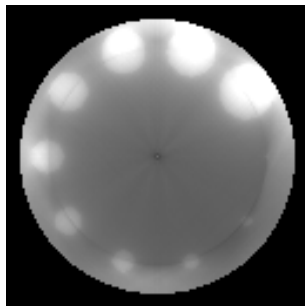
True image: 10 "hot spots"
 $f = -0.883$



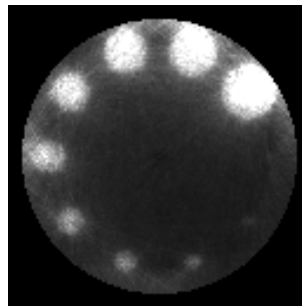
$x^1 = n^{-1}(1, \dots, 1)^T$
 $f = -0.452$



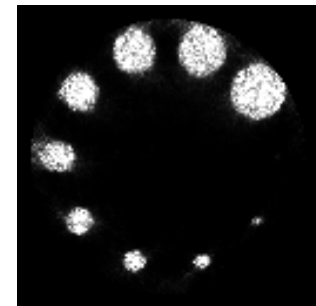
x^2 – light traces of 5 spots
 $f = -0.520$



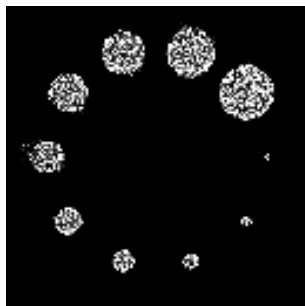
x^3 – traces of 8 spots
 $f = -0.585$



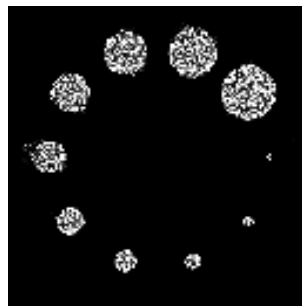
x^5 – 8 spots in place
 $f = -0.707$



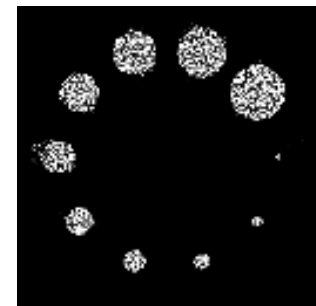
x^8 – 10th spot still missing...
 $f = -0.865$



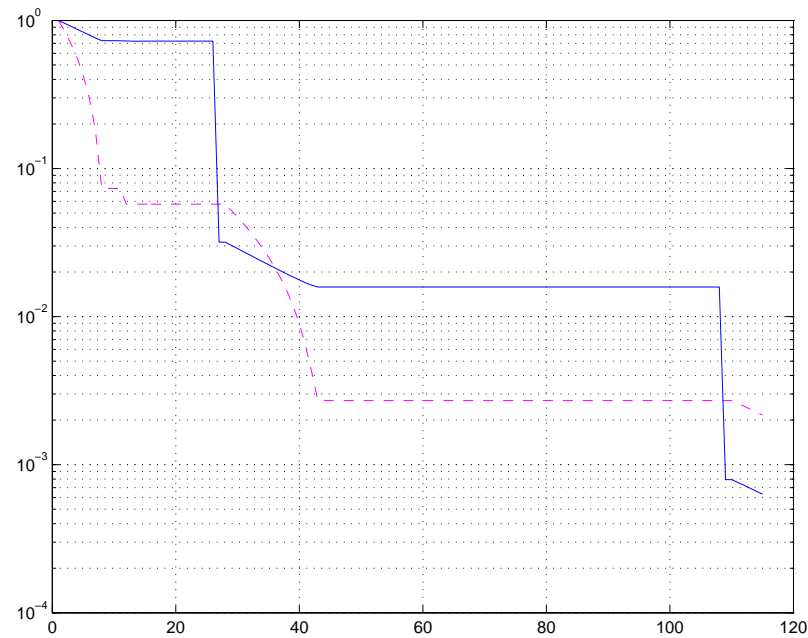
x^{12} – all 10 spots in place
 $f = -0.872$



x^{35} – all 10 spots in place
 $f = -0.886$



x^{43} ...
 $f = -0.896$



Progress in accuracy, noisy measurements.

solid line: Relative gap $\frac{\text{Gap}(t)}{\text{Gap}(1)}$ vs. step number t

- In 115 steps, the gap was reduced by factor 1580

dashed line: Progress in accuracy $\frac{f(x^t) - f}{f(x^1) - f}$ vs. step number t (f is the last lower bound on f_* built in the run)

- In 115 steps, the accuracy was improved by factor > 460

Mirror Descent Stochastic Approximation

♣ Consider the case when solving a convex program

$$f_* = \min_{x \in X} f(x)$$

[• $X \subset \mathbf{R}^n$: convex compact • $f : X \rightarrow \mathbf{R}$ convex and Lipschitz]

no precise first order information is available. Specifically, we have at our disposal

- *Proximal setup for X* – norm $\|\cdot\|$ and DGF $\omega(\cdot)$
- *Stochastic Oracle (SO) for f* as follows: at t -th call to the oracle, x_t being the input, the oracle returns

$$g(x_t, \xi_t) \in \mathbf{R}, G(x_t, \xi_t) \in \mathbf{R}^n$$

as random estimates of $f(x_t)$ and $f'(x_t)$, where ξ_1, ξ_2, \dots is a sequence of independent realizations of a *random variable* ξ ("oracle's noise").

♠ We assume that the SO is *unbiased*:

$$\mathbf{E}\{g(x, \xi)\} = f(x), \quad \mathbf{E}\{G(x, \xi)\} \in \partial f(x).$$

In addition, we assume that

$$\mathbf{E}\{\|G(x, \xi)\|_*^2\} \leq L^2 < \infty \quad \forall x \in X$$

Example: Our f is given as expectation:

$$f(x) = \int_{\Xi} F(x, \xi) dP(\xi),$$

where F is convex in x and efficiently computable.

When we cannot compute the expectation in a closed analytic form, but can instead sample from the distribution P , we, under mild regularity assumptions on F , have at our disposal unbiased Stochastic Oracle

$$g(x, \xi) = F(x, \xi), \quad G(x, \xi) = F'_x(x, \xi)$$

$f_* = \min_{x \in X} f(x)$
$\mathbf{E}\{g(x, \xi)\} = f(x), \mathbf{E}\{G(x, \xi)\} \in \partial f(x), \mathbf{E}\{\ G(x, \xi)\ _*^2\} \leq L^2 < \infty \quad \forall x \in X$
$\text{Prox}_x(\xi) = \operatorname{argmin}_{u \in X} \left[\langle \xi, u \rangle + \underbrace{\omega(u) - \omega(x) - \langle u - x, \nabla \omega(x) \rangle}_{V_x(u)} \right]$

♣ We can solve the problem with *Mirror Descent Stochastic Approximation* which is completely similar to MD:

$$x_1 \in X; x_{t+1} = \text{Prox}_{x_t}(\gamma_t G(x_t, \xi_t)), 1 \leq t \leq N;$$

$$x^N = \frac{1}{\gamma_1 + \dots + \gamma_N} \sum_{t=1}^N \gamma_t x_t.$$

Here $\gamma_t > 0$ are deterministic stepsizes, and $\|\cdot\|$ and the function ω underlying the prox-mapping are given by Proximal setup.

$$x_1 \in X; x_{t+1} = \text{Prox}_{x_t}(\gamma_t G(x_t, \xi_t)), 1 \leq t \leq N;$$

$$x^N = \frac{1}{\gamma_1 + \dots + \gamma_N} \sum_{t=1}^N \gamma_t x_t.$$

Fact: For the MD Stochastic Approximation one has

$$\mathbf{E}\{f(x^N) - f(x_*)\} \leq [\sum_{t=1}^N \gamma_t]^{-1} \mathbf{E}\{\sum_{t=1}^N \gamma_t [f(x_t) - f_*]\} \leq \frac{\Theta + \frac{1}{2} \sum_{t=1}^N \gamma_t^2 L^2}{\sum_{t=1}^N \gamma_t},$$

$$\Theta = \max_{x, y \in X} V_x(y)$$

that is, we get exactly the same efficiency estimate as in the case of precise First Order oracle, but now – for the **expected** inaccuracy of the approximate solution x^N – the weighted sum of the search points we have generated in course of $N = 1, 2, \dots$ steps.

• **Remark:** Euclidean version

$$x_{t+1} = \underset{u \in X}{\operatorname{argmin}} \| [x_t - \gamma_t G(x_t, \xi_t)] - u \|_2^2$$

of Mirror Descent Stochastic Approximation is called *Stochastic Subgradient Descent* and is extremely popular in today Machine Learning.

$$x_1 \in X; x_{t+1} = \text{Prox}_{x_t}(\gamma_t G(x_t, \xi_t)), 1 \leq t \leq N; x^N = \frac{1}{\gamma_1 + \dots + \gamma_N} \sum_{t=1}^N \gamma_t x_t.$$

Convergence Analysis of Mirror Descent Stochastic Approximation

♠ Let us carry out convergence analysis of the algorithm. Denoting by x_* a minimizer of f over X , we, as always, have

$$\sum_{t=1}^N \gamma_t \langle G(x_t, \xi_t), x_t - x_* \rangle \leq \Theta + \frac{1}{2} \sum_{t=1}^N \gamma_t^2 \|G(x_t, \xi_t)\|_*^2$$

Taking expectations of both sides and taking into account that x_t is a deterministic function of ξ_1, \dots, ξ_{t-1} , while ξ_1, \dots, ξ_N are independent, we get

$$\sum_{t=1}^N \gamma_t \mathbf{E}\{\langle f'(x_t), x_t - x_* \rangle\} \leq \Theta + \frac{1}{2} \sum_{t=1}^N \gamma_t^2 L^2,$$

whence also

$$\mathbf{E}\left\{\sum_{t=1}^N \gamma_t [f(x_t) - f(x_*)]\right\} \leq \Theta + \frac{1}{2} \sum_{t=1}^N \gamma_t^2 L^2$$

$$\sum_{t=1}^N \gamma_t \mathbf{E}\{f(x_t) - f(x_*)\} \leq \Theta + \frac{1}{2} \sum_{t=1}^N \gamma_t^2 L^2 \quad \& \quad x^N = \frac{1}{\gamma_1 + \dots + \gamma_N} \sum_{t=1}^N \gamma_t x_t$$

By convexity,

$$\mathbf{E}\{f(x^N) - f(x_*)\} \leq [\sum_{t=1}^N \gamma_t]^{-1} \mathbf{E}\{\sum_{t=1}^N \gamma_t [f(x_t) - f(x_*)]\} \leq \frac{\Theta + \frac{1}{2} \sum_{t=1}^N \gamma_t^2 L^2}{\sum_{t=1}^N \gamma_t},$$

as claimed. □

Online Optimization

- **Problem:** Assume on time horizon $1, 2, \dots, T$ you and nature (or adversary) play game as follows:
 - at time t you are at a point $x_t \in X$, where $X \subset \mathbf{R}^n$ is a once for ever fixed convex compact set.
 - at time t the nature/adversary selects a Lipschitz continuous convex function $f_t(x) : X \rightarrow \mathbf{R}$ and enforces you to pay the random amount

$$\phi_t(x_t, \xi_t)$$

where ξ_t is random variable, and ϕ_t, ξ_t are such that

$$\mathbf{E}_{\xi_t} \{\phi_t(x, \xi_t)\} = f_t(x), x \in X.$$

Besides this, the nature reports stochastic subgradient $G_t(x_t, \xi_t)$ of f_t at x_t :

$$g_t(x) := \mathbf{E}_{\xi_t} \{G_t(x, \xi_t)\} \in \partial f_t(x_t).$$

— you are allowed to use all accumulated so far information to select the next point $x_{t+1} \in X$, and then the process continues.

Important: The random variables $\xi_1, \xi_2, \dots, \xi_T$ are mutually independent.

- **Goal:** The performance of your policy for selecting x_1, \dots, x_T is the expectation

$$\mathbf{E}_{\xi_1, \dots, \xi_T} \{ \phi_1(x_1, \xi_1) + \phi_2(x_2, \xi_2) + \dots + \phi_T(x_T, \xi_T) \}$$

of your total payment. In Online Optimization, this performance is compared with the one of “ideal player” who knows the future – the sequence f_1, \dots, f_T , but not the realization of noises! – in advance, but *cannot move* – must ensure that $x_1 = x_2 = \dots = x_T$. Denoting the common value of x_t by x , the ideal player will select x by solving the problem

$$\min_{x \in X} \mathbf{E}_{\xi_1, \dots, \xi_T} \left\{ \sum_{t=1}^T \phi_t(x, \xi_t) \right\} = \min_{x \in X} \left\{ \sum_{t=1}^T f_t(x) \right\}.$$

The difference

$$\text{Regret} = \mathbf{E}_{\xi_1, \dots, \xi_T} \left\{ \sum_{t=1}^T \phi_t(x_t, \xi_t) \right\} - \min_{x \in X} \left\{ \sum_{t=1}^T f_t(x) \right\}$$

is called *regret*; the goal of Online Minimization is to select the policy for updating x_t which makes the regret as small as possible.

Note: The paradigm of Online Minimization is *different* from the one of usual optimization even when $f_t \equiv f$ is independent of t . With the usual approach, an algorithm is an offline process; it does not matter how nonoptimal are the search points — the only thing which matters is how nonoptimal is the resulting approximate solution. In contrast, in Online Optimization with fixed f , we “pay on the fly,” and what matters is how good at average, in terms of the objective, are the search points.

♣ **Mirror Descent Regret Minimization.** Let us fix Proximal setup for X — a norm $\|\cdot\|$ on the embedding X linear space E , and a DGF $\omega(x) : X \rightarrow \mathbb{R}$ which is continuously differentiable and strongly convex, modulus 1, w.r.t. $\|\cdot\|$. As always, we set

$$\Theta = \max_{u,v \in X} [V_v(u) := \omega(u) - \omega(v) - \langle \nabla \omega(v), u - v \rangle]$$

Assumption: $\mathbf{E}_{\xi_t} \{ \|G_t(x, \xi_t)\|_*^2 \} \leq L^2 \ \forall (x \in X, t \leq T)$.

♠ Consider the recurrence

$$x_{t+1} = \text{Prox}_{x_t}[\gamma G_t(x_t, \xi_t)] := \underset{u \in X}{\text{argmin}} [\langle \gamma G_t(x_t, \xi_t), y \rangle + V_{x_t}(y)], \ t = 1, \dots, T.$$

with *fixed* stepsize $\gamma > 0$.

Let $x_* \in \text{Argmin}_{x \in X} \sum_{t=1}^T f_t(x)$. By our standard argument, we have

$$\sum_{t=1}^T \gamma \langle G_t(x_t, \xi_t), x_t - x_* \rangle \leq \Theta + \frac{1}{2} \gamma^2 \sum_{t=1}^T \|G_t(x_t, \xi_t)\|_*^2.$$

Taking expectations and recalling that x_t is a deterministic function of ξ_1, \dots, ξ_{t-1} and therefore

$$\mathbf{E}_{\xi_t} \{ \langle G_t(x_t, \xi_t), x_t - x_* \rangle \} = \langle f'_t(x_t), x_t - x_* \rangle,$$

we get $\sum_{t=1}^T \mathbf{E} \{ \langle f'_t(x_t), x_t - x_* \rangle \} \leq \frac{\Theta}{\gamma} + \gamma T L^2$

$$\begin{aligned} \Rightarrow \text{Regret} &= \mathbf{E} \left\{ \sum_{t=1}^T [\phi_t(x_t, \xi_t) - f_t(x_*)] \right\} = \mathbf{E} \left\{ \sum_{t=1}^T [f_t(x_t) - f_t(x_*)] \right\} \\ &\leq \mathbf{E} \left\{ \sum_{t=1}^T \langle f'_t(x_t), x_t - x_* \rangle \right\} \leq \frac{\Theta}{\gamma} + \frac{\gamma}{2} T L^2 \end{aligned}$$

$$\mathbb{E} \left\{ \sum_{t=1}^T [f_t(x_t) - f_t(x_*)] \right\} \leq \frac{\Theta}{\gamma} + \frac{\gamma}{2} T L^2$$

Setting $\gamma = \frac{\sqrt{2\Theta}}{L\sqrt{T}}$, we get for the policy in question

$$\frac{1}{T} \text{Regret} \leq \frac{\sqrt{2\Theta} L}{\sqrt{T}}$$

Thus, with the MD policy *the average regret per step* $\frac{\text{Regret}}{T}$ *for large* T *can be made as small as* $O(1/\sqrt{T})$.

Note: In the above construction, the stepsize γ is the same for all $t \leq T$ and is “tuned” to the time horizon T we are interested in. With appropriate modification, the stepsize can be made varying in time in such a way that the average, per unit time, regrets on time horizons $T = 1, 2, \dots$ will go to zero as $T \rightarrow \infty$ at the rate $O(1/\sqrt{T})$.

Application Example: Prediction for Deterministic Boolean Sequence

[for in-depth treatment, see A. Rakhlin, K. Sridharan, *Statistical Learning and Sequential Prediction*, http://www.mit.edu/~rakhlin/courses/stat928/stat928_notes.pdf]

Situation: We observe a deterministic Boolean sequence $\xi^N = (\xi_1, \dots, \xi_N)$, $\xi_t \in \{0, 1\}$ on time horizon $1, \dots, N$.

Goal: To build *predictions* $\hat{\xi}_t$ which, given $\xi^{t-1} = (\xi_1, \dots, \xi_{t-1})$, predict (perhaps in randomized fashion) ξ_t , $t = 1, \dots, N$.

Performance of a collection $\Xi = \{\hat{\xi}_t, t \leq N\}$, of predictions is quantified by *average over time expected prediction error*

$$\text{Err}[\Xi] = \mathbf{E} \left\{ \frac{1}{N} \sum_{t=1}^N \chi(\hat{\xi}_t, \xi_t) \right\} \quad \left[\chi(\xi, \xi') = \begin{cases} 0, & \xi = \xi' \\ 1, & \xi \neq \xi' \end{cases} \right]$$

the expectation being taken over the random “driving factors,” if any, influencing $\hat{\xi}_t$ (these factors are present when the predictions indeed are randomized).

Note: We make no assumptions on the nature of Boolean sequence ξ^N !!

Basic Predictor: We allow for $\hat{\xi}_t$ to be randomized: the conditional, given what happened on time horizon $1, \dots, t-1$, probability for $\hat{\xi}_t$ to take value 1 is $x_t \in [0, 1]$. Note that

$$\mathbf{E}_{|t-1} \left\{ \chi(\hat{\xi}_t, \xi_t) \right\} = x_t[1 - \xi_t] + (1 - x_t)\xi_t = f_t(x_t) := |x_t - \xi_t| \quad (!)$$

where $\mathbf{E}_{|s}$ is the conditional, given realization of driving factors influencing $\hat{\xi}_1, \dots, \hat{\xi}_s$, expectation.

- To update x_t , we use “online subgradient descent,” – the recurrence

$$x_{t+1} = \Pi_{\Delta}[x_t - \gamma_t f'_t(x_t)], \quad f'_t(x) = \begin{cases} -1, & x < \xi_t \\ 0, & x = \xi_t \\ 1, & x > \xi_t \end{cases}$$

where $\Pi_{\Delta}(s) = \begin{cases} 0, & s < 0 \\ s, & 0 \leq s \leq 1 \\ 1, & s > 1 \end{cases}$ is the metric projection on $\Delta = [0, 1]$, $x_1 \in [0, 1]$ is once

for ever fixed, and γ_t are deterministic positive stepsizes satisfying $\gamma_1 \geq \gamma_2 \geq \dots \geq \gamma_N$.

Note: The resulting sequence x_1, \dots, x_N is deterministic! $\Rightarrow \text{Err}[\Xi] = \frac{1}{N} \sum_{t=1}^N f_t(x_t)$ by (!).

Performance Analysis: Let us fix $\bar{x} \in [0, 1]$ and set $d_t = \frac{1}{2}(x_t - \bar{x})^2$. By the standard argument, noting that $f_t(x)$ is convex, we have

$$\begin{aligned}
 & \gamma_t f'_t(x_t)(x_t - \bar{x}) \leq d_t - d_{t+1} + \frac{1}{2}\gamma_t^2 \\
 \Rightarrow & f_t(x_t) - f_t(\bar{x}) \leq f'_t(x_t)(x_t - \bar{x}) \leq \frac{d_t - d_{t+1}}{\gamma_t} + \frac{1}{2}\gamma_t \\
 \Rightarrow & \sum_{t=1}^N [f_t(x_t) - f_t(\bar{x})] \leq \sum_{t=1}^N \frac{d_t - d_{t+1}}{\gamma_t} + \frac{1}{2} \sum_{t=1}^N \gamma_t \\
 = & \frac{1}{2} \sum_{t=1}^N \gamma_t + \frac{d_1}{\gamma_1} + d_2 \underbrace{\left[\frac{1}{\gamma_2} - \frac{1}{\gamma_1} \right]}_{\geq 0} + d_3 \underbrace{\left[\frac{1}{\gamma_3} - \frac{1}{\gamma_2} \right]}_{\geq 0} + \dots + d_N \underbrace{\left[\frac{1}{\gamma_N} - \frac{1}{\gamma_{N-1}} \right]}_{\geq 0} - \frac{1}{\gamma_N} d_{N+1} \\
 \leq & \frac{1}{2} \sum_{t=1}^N \gamma_t + \frac{1}{2} \left[\frac{1}{\gamma_1} + \left[\frac{1}{\gamma_2} - \frac{1}{\gamma_1} \right] + \left[\frac{1}{\gamma_3} - \frac{1}{\gamma_2} \right] + \dots + \left[\frac{1}{\gamma_N} - \frac{1}{\gamma_{N-1}} \right] \right] \text{ [since } 0 \leq d_t \leq 1/2 \text{]} \\
 = & \frac{1}{2} \sum_{t=1}^N \gamma_t + \frac{1}{2} \frac{1}{\gamma_N} \\
 \Rightarrow & \text{Err}[\Xi] = \frac{1}{N} \sum_{t=1}^N f_t(x_t) \leq \frac{1}{N} \sum_{t=1}^n f_t(\bar{x}) + \frac{1}{2N} \left[\sum_{t=1}^N \gamma_t + \frac{1}{\gamma_N} \right]
 \end{aligned}$$

• Let us set $\gamma_t = \frac{\alpha}{\sqrt{t}}$ with some $\alpha > 0$. Then $\sum_{t=1}^N \gamma_t \leq \alpha \int_0^N s^{-1/2} ds = 2\alpha N^{1/2}$ and $\frac{1}{\gamma_N} = \alpha^{-1} \sqrt{N}$, and we get $\text{Err}[\Xi] \leq \frac{1}{N} \sum_{t=1}^n f_t(\bar{x}) + \frac{1}{2} \left[2\alpha + \frac{1}{\alpha} \right] N^{-1/2}$, which with $\alpha = 1/\sqrt{2}$ yields

$$\text{Err}[\Xi] \leq \frac{1}{N} \sum_{t=1}^N |\bar{x} - \xi_t| + \sqrt{2/N}. \quad (\#)$$

$$\text{Err}[\Xi] \leq \underbrace{\frac{1}{N} \sum_{t=1}^n |\bar{x} - \xi_t|}_{E(\bar{x})} + \sqrt{2/N}. \quad (\#)$$

♠ Now let λ be the fraction of ones in ξ^N . Note that $E(1) = 1 - \lambda$ and $E(0) = \lambda$, so that $(\#)$ (which holds true for every $\bar{x} \in [0, 1]$) implies that

$$\text{Err}[\Xi] \leq \min[\lambda, 1 - \lambda] + \sqrt{2/N}. \quad (!)$$

Conclusions: • When N is large, upper bound $(!)$ on average, over time horizon $1, \dots, N$, expected prediction error is close to $\min[\lambda, 1 - \lambda]$. The latter quantity *always is* $\leq 1/2$.

• Bound $1/2$ is not interesting: we can arrive at $\text{Err}[\Xi] = 1/2$ when “predicting” by flipping a perfect coin, not using observations at all.

• **However:** In the “asymmetric case” $\min[\lambda, 1 - \lambda] < 1/2$, we get a *nontrivial* upper bound on the average expected prediction error – *and this is with no assumptions on ξ^N except for asymmetry!*

♠ **Fact:** *When all we know about ξ^N is that the fraction of ones in the sequence is a given λ , then, for every $\epsilon > 0$, no prediction can guarantee average expected error $\leq \min[\lambda, 1 - \lambda] - \epsilon$, provided that N is large enough!*

Modification

♠ The accuracy guaranties we have obtained so far, as applied to the sequence $0, 1, 0, 1, 0, 1, 0, 1, \dots$ results in trivial bound $1/2$, in spite of the fact that *were we sure that the presented fragment is “representative” for the entire sequence*, every normal person would predict without errors at all.

Question: *How to modify the approach to predict well sequences like $0, 1, 0, 1, 0, 1, \dots$?*

Model: There exist m “states” $0, 1, \dots, m-1$. We observe, one entry at a time, a sequence $\zeta_0, \zeta_1, \zeta_2, \dots, \zeta_N$, with $\zeta_t \in S = \{0, 1, \dots, m-1\}$, and we know in advance that for every state v , there are exactly two known to us states $p_0(v)$ and $p_1(v)$, with $p_0(v) \neq p_1(v)$, such that if $\zeta_t = v$, then $\zeta_{t+1} \in \{p_0(v), p_1(v)\}$.

Goal: for $t \geq 1$, to predict ζ_t given $\zeta_0, \zeta_1, \dots, \zeta_{t-1}$.

Illustration: A Boolean sequence ξ^N can be treated as follows: take $m = 2^\kappa$ for $\kappa \geq 0$ and say that at time $t \geq \kappa$ the sequence is in state v , if the bits $\xi_{t-\kappa+1}, \xi_{t-\kappa+2}, \dots, \xi_t$ form the binary representation of v . This definition of states says loud and clear what are $p_0(v)$ and $p_1(v)$; e.g., with $\kappa = 3$,

$$p_0(5 = \overline{101}) = \overline{010} = 2, \quad p_1(5 = \overline{101}) = \overline{011} = 3$$
$$[\text{for } \iota_s \in \{0, 1\}, \quad 0 \leq s < \kappa, \quad \overline{\iota_0 \dots \iota_{\kappa-1}} = 2^{\kappa-1}\iota_0 + 2^{\kappa-2}\iota_1 + \dots + 2\iota_{\kappa-2} + \iota_{\kappa-1}]$$

Example: the sequence $0, 1, 0, 1, 0, 1, \dots$ when considered with $\kappa = 2$ alternates between the states $v = 1 = \overline{01}$ and $v = 2 = \overline{10}$ from the set $S = \{0, 1, 2, 3\}$.

Note: in our original treatment of Boolean sequence ξ^N , we used $\kappa = 0$

Modified Predictor we are about to design for the model just outlined is our Basic Predictor used separately for every one of the states. Specifically:

- We introduce “counters of visits” of states – the quantities $\tau(t, v) = \text{Card}\{\tau \leq t : \zeta_\tau = v\}$, $v \in S$, and denote by $t(v, \tau)$ the instant t when the sequence ζ^N for τ -th time visits state v
- The prediction $\hat{\zeta}_t$, $t \geq 1$, is made when we are at state ζ_{t-1} and is selected at random in $\{p_0(\zeta_{t-1}), p_1(\zeta_{t-1})\}$, with conditional, by what happened prior to time t , probability of the second choice equal to $x_{\zeta_{t-1}, \tau(t-1, \zeta_{t-1})}$, with $x_{v, \tau}$ given by the recurrence
 - $x_{v, 1} \in [0, 1]$ are once for ever fixed
 - $x_{v, \tau+1} = \Pi_{[0, 1]}[x_{v, \tau} - \gamma_\tau f'_{v, \tau}(x_{v, \tau})]$, where

$$f_{v, \tau}(x) = |x - \delta_{v, \tau}|, \quad \delta_{v, \tau} = \begin{cases} 0, & \zeta_{t(v, \tau)+1} = p_0(v) \\ 1, & \zeta_{t(v, \tau)+1} = p_1(v) \end{cases}, \quad \gamma_\tau = \frac{1}{\sqrt{2\tau}},$$

and f' stands for a subgradient of f .

In other words,

- we associate with states v Boolean sequences $\xi_{1v}, \xi_{2v}, \xi_{3v}, \dots$ with $\xi_{\tau v}$ being 0 or 1 depending on whether ζ^N at τ -th visit to state v moved from this state to $p_0(v)$ or to $p_1(v)$;
- when predicting ζ_t at state $v = \zeta_{t-1}$ visited for τ -th time, the prediction is $p_0(v)$ or $p_1(v)$ depending on what, according to Basic Predictor as applied to the observed so far part $\xi_{1v}, \xi_{2v}, \dots, \xi_{\tau v}$ of the Boolean sequence associated with the state v , is the next term, $\xi_{(\tau+1), v}$, in this sequence.

Illustration: Assume we observe a Boolean sequence $\xi_0, \xi_1, \xi_2 \dots$ and select $\kappa = 2$, resulting in four states:

$$\overline{00} = 0, \overline{01} = 1, \overline{10} = 2, \overline{11} = 3$$

of the induces ζ -sequence, and in

v	$p_0(v)$	$p_1(v)$
0	0 $\dots \overline{00} \rightarrow \dots 0\overline{00}$	1 $\dots \overline{00} \rightarrow \dots 0\overline{01}$
1	2 $\dots \overline{01} \rightarrow \dots 0\overline{10}$	3 $\dots \overline{01} \rightarrow \dots 0\overline{11}$
2	0 $\dots \overline{10} \rightarrow \dots 1\overline{00}$	1 $\dots \overline{10} \rightarrow \dots 1\overline{01}$
3	2 $\dots \overline{11} \rightarrow \dots 1\overline{10}$	3 $\dots \overline{11} \rightarrow \dots 1\overline{11}$

where, say, the last row says that when the current state of ζ -sequence is $3 = \overline{11}$, the next state can be either $2 = \overline{10}$, or $3 = \overline{11}$, depending on whether the corresponding fragment in ξ -sequence is 110 or 111.

v	$p_0(v)$	$p_1(v)$
0	0	1
1	2	3
2	0	1
3	2	3

♠ This is how we predict Boolean ξ -sequence

$$\xi_0\xi_1\xi_2\xi_3\xi_4\xi_5\xi_6\ldots = 0100110\ldots$$

which gives rise to ζ -sequence

$$\zeta_0, \zeta_1, \zeta_2, \zeta_3, \zeta_4, \zeta_5, \ldots = 1, 2, 0, 1, 3, 2, \ldots = \overline{01}, \overline{10}, \overline{00}, \overline{01}, \overline{11}, \overline{10}, \ldots$$

- First, we predict ξ_2 being at state $\zeta_0 = 1$ (first visit) \Rightarrow probability to predict 1 is $x_{1,1}$ (parameter of construction $\in [0, 1]$). We have $\zeta_1 = 2 = p_0(\zeta_0) \Rightarrow \delta_{1,1} = 0 \Rightarrow x_{1,2} = \Pi_{[0,1]}[x_{1,1} - \gamma_1 \text{sign}[x_{1,1} - \delta_{1,1}]] = \Pi_{[0,1]}[x_{1,1} - \gamma_1]$
- Next, we predict ξ_3 being at state $\zeta_1 = 2$ (first visit) \Rightarrow probability to predict 1 is $x_{2,1}$ (parameter of construction $\in [0, 1]$). We have $\zeta_2 = 0 = p_0(\zeta_1) \Rightarrow \delta_{2,1} = 0 \Rightarrow x_{2,2} = \Pi_{[0,1]}[x_{2,1} - \gamma_1 \text{sign}[x_{2,1} - \delta_{2,1}]] = \Pi_{[0,1]}[x_{2,1} - \gamma_1]$
- Next, we predict ξ_4 being at state $\zeta_2 = 0$ (first visit) \Rightarrow probability to predict 1 is $x_{0,1}$ (parameter of construction $\in [0, 1]$). We have $\zeta_3 = 1 = p_1(\zeta_2) \Rightarrow \delta_{0,1} = 1 \Rightarrow x_{0,2} = \Pi_{[0,1]}[x_{0,1} - \gamma_1 \text{sign}[x_{0,1} - \delta_{0,1}]] = \Pi_{[0,1]}[x_{0,1} + \gamma_1]$
- Next, we predict ξ_5 being at state $\zeta_3 = 1$ (second visit) \Rightarrow probability to predict 1 is $x_{1,2}$ (has already been built). We have $\zeta_4 = 3 = p_1(\zeta_3) \Rightarrow \delta_{1,2} = 1 \Rightarrow x_{1,3} = \Pi_{[0,1]}[x_{1,2} - \gamma_2 \text{sign}[x_{1,2} - \delta_{1,2}]] = \Pi_{[0,1]}[x_{1,2} + \gamma_2]$

.....

♠ **Performance Analysis.** Given ζ^N , let $\mathcal{N}_v = \{t, 1 \leq t \leq N : \zeta_{t-1} = v\}$, $N_v = \text{Card}(\mathcal{N}_v)$.

Note: $\{\mathcal{N}_v : v \in S\}$ is partition of $\{1, 2, \dots, N\}$ into non-overlapping subsets.

• For $v \in \Upsilon = \{v : \mathcal{N}_v \neq \emptyset\}$, let ξ_τ^v , $1 \leq \tau \leq N_v$, be the Boolean sequence with τ -th entry equal to 0 or 1 depending on whether the τ -th visit to state v (this visit happens at time $t(v, \tau)$) results in transition to the state $p_0(v)$ (i.e., $\zeta_{t(v, \tau)+1} = p_0(v)$) or to the state $p_1(v)$ (i.e., $\zeta_{t(v, \tau)+1} = p_1(v)$).

Observation: By construction, the just described Modified Predictor is our Basic Predictor run separately on every Boolean sequence $\{\xi_\tau^v\}_{\tau=1}^{N_v}$, $v \in \Upsilon$

\Rightarrow The expected value of the ratio $\frac{\# \text{ of wrong predictions on time horizon } 1, 2, \dots, N}{N}$ is upper-bounded by

$$\overline{\text{Err}} := \frac{1}{N} \sum_{v \in \Upsilon} [N_v \min [\lambda_v, 1 - \lambda_v] + \sqrt{2N_v}] ,$$

where

$$\lambda_v = \frac{\text{Card}\{t \in \{1, \dots, N\} : \zeta_{t-1} = v, \zeta_t = p_1(v)\}}{\text{Card}\{t \in \{1, \dots, N\} : \zeta_{t-1} = v\}} = \frac{\text{Card}\{t \in \{1, \dots, N\} : \zeta_{t-1} = v, \zeta_t = p_1(v)\}}{N_v}$$

Back to Predicting Boolean Sequence

♠ Assume we want to predict terms ξ_t of Boolean sequence $\xi^N = (\xi_1, \dots, \xi_N)$ on time horizon $\{1, \dots, N\}$.

Goal: To compare the performance guarantees given by Basic and Modified Predictors when the latter is applied to the sequence of states $\{\zeta_t = \overline{\xi_{t-\kappa+1} \dots \xi_t}\}$

Note: To make the comparison possible, we assume that ξ^N is augmented from the left by Boolean entries $\xi_{1-\kappa}, \xi_{2-\kappa}, \dots, \xi_0$, thus making well defined the states for times $t \in \{0, 1, \dots, N\}$, as required for Modified Predictor.

- For Basic Predictor, the performance guarantee on time horizon $1, 2, \dots, N$ is

$$\begin{aligned} \text{Err} &\leq \text{Err}_I := \phi(\bar{\lambda}) + \sqrt{2/N} \\ &\left[\phi(\lambda) = \min[\lambda, 1 - \lambda], \bar{\lambda} = \frac{\text{Card}\{t, 1 \leq t \leq N : \xi_t = 1\}}{N} \right] \end{aligned} \quad (I)$$

- For Modified Predictor, the performance guarantee on time horizon $1, 2, \dots, N$ is

$$\begin{aligned} \text{Err} &\leq \text{Err}_E := \frac{1}{N} \sum_{v \in \gamma} [N_v \phi(\lambda_v) + \sqrt{2N_v}] \\ &[N_v = \text{Card}\{t, 1 \leq t \leq N : \overline{\xi_{t-\kappa} \dots \xi_{t-1}} = v\}, \lambda_v = \text{Card}\{t, 1 \leq t \leq N : \overline{\xi_{t-\kappa} \dots \xi_{t-1}} = v, \xi_t = 1\} / N_v] \end{aligned} \quad (E)$$

Note:

- $N = \sum_{v \in \gamma} N_v, \bar{\lambda} = \frac{1}{N} \sum_{v \in \gamma} N_v \lambda_v$, ϕ is concave

\Rightarrow The leading term $\frac{1}{N} \sum_{v \in \gamma} N_v \phi(\lambda_v)$ in (E) is \leq the leading term $\phi(\bar{\lambda})$ in (I)

- The extra term $\sqrt{2/N}$ in (I) is \geq the extra term $\frac{1}{N} \sum_{v \in \gamma} \sqrt{2N_v}$ in (E). However, the latter term is $\leq \sqrt{2^\kappa} \sqrt{2/N}$, and both these terms are small when N is large

\Rightarrow We can expect that for κ fixed and N large, Modified Predictor is better than Basic one.

Note: Similar argument shows that for large N , one can expect Modified Predictor to be the better the larger is κ .

How It Works

♠ Empirical expected average, over time horizon $1, \dots, N = 1024$, prediction error (data over 1000 simulations):

ξ^N	$\kappa = 0$	$\kappa = 1$	$\kappa = 2$	$\kappa = 3$	$\kappa = 4$	$\kappa = 5$
RAND	0.499/0.499	0.487/0.484	0.496/0.564	0.513/0.596	0.499/0.627	0.513/0.691
QR[e]	0.518/0.542	0.463/0.499	0.141/0.215	0.143/0.232	0.145/0.247	0.148/0.263
QR[π]	0.515/0.540	0.301/0.346	0.306/0.369	0.170/0.259	0.143/0.256	0.145/0.270
QR[$\sqrt{2}$]	0.518/0.543	0.184/0.235	0.187/0.255	0.190/0.275	0.194/0.294	0.166/0.281

Numbers in cells: empirical average prediction error and its theoretical upper bound
 κ : parameter of Modified Predictor [$\kappa = 0$ corresponds to Basic Predictor]

Generation of ξ^N :

- RAND: MATLAB pseudo-random generator: to get ξ_t , you generate $v_t = \text{rand}(1,1)$ and set $\xi_t = 0$ or $\xi_t = 1$ depending on whether or not $v_t < 0.5$
- QR[α]: we generate sequence $v_t = t\alpha - \text{floor}(t\alpha)$ and set ξ_t to 0 or to 1 depending on whether or not $v_t < 1/2$.

Mirror Descent for Convex-Concave Saddle Point Problems

♣ Convex-Concave Saddle Point problem is

$$SV = \min_{x \in X} \max_{y \in Y} \phi(x, y) \quad (\text{SP})$$

where:

- $X \subset E_x, Y \subset E_y$ are nonempty closed and bounded convex sets in Euclidean spaces E_x, E_y
- $\phi(x, y) : Z := X \times Y \rightarrow \mathbf{R}$ is the *cost function* which is Lipschitz continuous, convex in $x \in X$ and concave in $y \in Y$.
- ♣ *Solutions* to (SP) are, by definition, *saddle points* of ϕ on $X \times Y$, that is, points $(x_*, y_*) \in X \times Y$ where ϕ achieves its minimum in $x \in X$ and its maximum in $y \in Y$:

$$\forall (x \in X, y \in Y) : \phi(x, y_*) \geq \phi(x_*, y_*) \geq \phi(x_*, y).$$

$$SV = \min_{x \in X} \max_{y \in Y} \phi(x, y) \quad (\text{SP})$$

♠ **Fact:** (SP) gives rise to two optimization problems:

$$\begin{aligned} (P) : \quad \text{Opt}(P) &= \min_{x \in X} \left[\bar{\phi}(x) := \max_{y \in Y} \phi(x, y) \right] \\ &= \min_{x \in X} \max_{y \in Y} \phi(x, y) \\ (D) : \quad \text{Opt}(D) &= \max_{y \in Y} \left[\underline{\phi}(y) := \min_{x \in X} \phi(x, y) \right] \\ &= \max_{y \in Y} \min_{x \in X} \phi(x, y) \end{aligned}$$

- We always have $\text{Opt}(P) \geq \text{Opt}(D)$ [“weak duality”]
- ϕ has saddle points on $X \times Y$ *iff* both (P) and (D) are solvable *with equal optimal values*: $\text{Opt}(P) = \text{Opt}(D)$, that is,

$$\min_{x \in X} \max_{y \in Y} \phi(x, y) = \max_{y \in Y} \min_{x \in X} \phi(x, y)$$

[“strong duality”]. In this case the saddle points are exactly the pairs $(x \in \text{Argmin}_X \bar{\phi}, y \in \text{Argmax}_Y \underline{\phi})$.

$$\begin{aligned}
(P) : \quad \text{Opt}(P) &= \min_{x \in X} \left[\bar{\phi}(x) := \max_{y \in Y} \phi(x, y) \right] \\
&= \min_{x \in X} \max_{y \in Y} \phi(x, y) \\
(D) : \quad \text{Opt}(D) &= \max_{y \in Y} \left[\underline{\phi}(y) := \min_{x \in X} \phi(x, y) \right] \\
&= \max_{y \in Y} \min_{x \in X} \phi(x, y)
\end{aligned}$$

- Under our standing assumption (X, Y are nonempty convex compacts, ϕ is Lipschitz continuous convex-concave), *both (P) and (D) are solvable with equal optimal values, that is, saddle points do exist.*

♠ It is natural to quantify the (in)accuracy of an approximate saddle point $(x, y) \in Z := X \times Y$ by its *saddle point residual*

$$\epsilon_{\text{Sad}}(x, y) = \bar{\phi}(x) - \underline{\phi}(y) = [\bar{\phi}(x) - \text{Opt}(P)] + [\text{Opt}(D) - \underline{\phi}(y)]$$

This residual always is nonnegative and is zero iff (x, y) is a saddle point of ϕ .

♣ **Vector field associated with a saddle point problem.** Under our standing assumptions, we can associate with a convex-concave saddle point problem

$$\min_{x \in X} \max_{y \in Y} \phi(x, y)$$

vector field

$$F(z = [x; y]) = [F_x(x, y); F_y(x, y)] : Z := X \times Y \rightarrow E_z := E_x \times E_y$$

with

$$F_x(x, y) \in \partial_x \phi(x, y), \quad F_y(x, y) \in \partial_y [-\phi(x, y)]$$

♠ **Assumption:** From now on, we assume that the vector field $F : Z \rightarrow E_z$ is bounded.

$$F(z = [x; y]) = [F_x(x, y); F_y(x, y)] : Z := X \times Y \rightarrow E_z := E_x \times E_y$$

$$F_x(x, y) \in \partial_x \phi(x, y), \quad F_y(x, y) \in \partial_y [-\phi(x, y)]$$

♠ **Facts:**

- F is monotone:

$$\forall (z, z' \in Z := X \times Y) : \langle F(z) - F(z'), z - z' \rangle \geq 0$$

Indeed, setting $z = (x, y)$, $z' = (x', y')$, we have

$$\begin{aligned} \langle F(z) - F(z'), z - z' \rangle &= \langle F_x(x, y) - F_x(x', y'), x - x' \rangle + \langle F_y(x, y) - F_y(x', y'), y - y' \rangle \\ &\geq [\phi(x, y) - \phi(x', y)] + [\phi(x', y') - \phi(x, y')] + [(-\phi)(x, y) - (-\phi)(x, y')] + [(-\phi)(x', y') - (-\phi)(x', y)] \\ &= 0 \end{aligned}$$

- Saddle points of ϕ on $Z = X \times Y$ are exactly the points $z_* \in Z$ such that

$$\langle F(z), z - z_* \rangle \geq 0 \quad \forall z \in Z.$$

♠ **Note:** When Y is a singleton, convex-concave saddle point problem $\min_{x \in X} \max_{y \in Y} \phi(x, y)$ becomes the problem of minimizing a convex function over X . “Convex minimization” versions of the above facts read: *For a Lipschitz continuous convex function $f(x) : X \rightarrow \mathbf{R}$*

- The field $f'(\cdot)$ of subgradients of f is monotone: $\langle f'(x) - f'(y), x - y \rangle \geq 0, x, y \in X$
- Minimizers of f on X are exactly the points $x_* \in X$ such that $\langle f'(x), x - x_* \rangle \geq 0 \quad \forall x \in X$.

$$SV = \min_{x \in X} \max_{y \in Y} \phi(x, y) \quad (\text{SP})$$

- $X \subset E_x$, $Y \subset E_y$ are nonempty closed and bounded convex sets in Euclidean spaces E_x, E_y
- $\phi(x, y) : Z := X \times Y \rightarrow \mathbf{R}$ is the *cost function* which is Lipschitz continuous, convex in $x \in X$ and concave in $y \in Y$.

♣ Problems (SP) arise in a wide spectrum of applications. Our major interest in these problems stems from the fact that *numerous "complex" and nonsmooth convex functions $f(x)$ admit saddle point representation:*

$$f(x) = \max_{y \in Y} \phi(x, y)$$

with convex-concave and smooth functions ϕ , which allows to reduce a nonsmooth minimization problem

$$\min_{x \in X} f(x)$$

to a smooth convex-concave saddle point problem

$$\min_{x \in X} \max_{y \in Y} \phi(x, y)$$

and this "gain in smoothness" possesses dramatic potential as far as computationally cheap First Order methods are concerned.

Examples of saddle point reformulations:

- Maximum of smooth convex functions:

$$f(x) := \max_{1 \leq i \leq m} f_i(x) = \max_{y \in Y} [\phi(x, y) := \sum_i y_i f_i(x)]$$

$$[Y = \{y \geq 0, \sum_i y_i = 1\}]$$

When f_i are smooth, so is ϕ ; when f_i are linear, ϕ is just bilinear.

- Norm-type functions:

$$\|Ax - b\| = \max_{y: \|y\|_* \leq 1} [\phi(x, y) = \langle y, Ax - b \rangle]$$

- Maximal eigenvalue of a symmetric matrix:

$$\lambda_{\max}(x) = \max_{y \in Y} [\phi(x, y) = \text{Tr}(xy)], \quad Y = \{y \succeq 0 : \text{Tr}(y) = 1\}$$

Note: Smooth/bilinear saddle point representations admit fully algorithmic calculus. For example,

General case:

$$f_i(x) = \max_{y_i \in Y_i} \phi_i(x, y_i), \quad \lambda_i \geq 0$$

$$\Rightarrow \sum_i \lambda_i f_i(x) = \max_{y=[y_1; \dots; y_k] \in Y_1 \times \dots \times Y_k} \underbrace{\left[\sum_i \lambda_i \phi_i(x, y_i) \right]}_{\phi(x, [y_1; \dots; y_k])}$$

Bilinear case:

$$f_i(x) = \max_{y_i \in Y_i} [\langle a_i, x \rangle + \langle b_i, y_i \rangle + \langle x, A_i y_i \rangle], \quad \lambda_i \geq 0$$

$$\Rightarrow \sum_i \lambda_i f_i(x) = \max_{y=[y_1; \dots; y_k] \in Y_1 \times \dots \times Y_k} \left[\sum_i \langle \lambda_i a_i, x \rangle + \langle \lambda_i b_i, y_i \rangle + \langle x, \lambda_i A_i y_i \rangle \right]$$

$$= \max_{y=[y_1; \dots; y_k] \in Y_1 \times \dots \times Y_k} \left[\langle \sum_i \lambda_i a_i, x \rangle + \langle [\lambda_1 b_1; \dots; \lambda_k b_k], y \rangle + \langle x, [\lambda_1 A_1, \dots, \lambda_k A_k] y \rangle \right]$$

$$\text{SV} = \min_{x \in X} \max_{y \in Y} \phi(x, y) \quad (\text{SP})$$

$$\Rightarrow F(z = [x; y]) = [F_x(x, y) \in \partial_x \phi(x, y); F_y(x, y) \in \partial_y [-\phi(x, y)]].$$

- $X \subset E_x$, $Y \subset E_y$ are nonempty closed and bounded convex sets in Euclidean spaces E_x, E_y
- $\phi(x, y) : Z := X \times Y \rightarrow \mathbf{R}$ is the *cost function* which is Lipschitz continuous, convex in $x \in X$ and concave in $y \in Y$.

♠ (SP) can be solved by MD. Indeed, let $\|\cdot\|$ be a norm on $E = E_x \times E_y$ and $\omega(\cdot)$ be a DGF for $Z = X \times Y$ which is compatible with $\|\cdot\|$. Consider the process

$$z_1 \in Z; z_{t+1} = \text{Prox}_{z_t}(\gamma_t F(z_t)); z^t = \left[\sum_{\tau=1}^t \gamma_\tau \right]^{-1} \sum_{\tau=1}^t \gamma_\tau z_\tau$$

$$[z_\tau = [x_\tau; y_\tau]]$$

♣ **Fact I:** One has

$$\epsilon_{\text{Sad}}(x^t, y^t) \leq \frac{\Theta + \frac{1}{2} \sum_{\tau=1}^T \gamma_\tau^2 \|F(z_\tau)\|_*^2}{\sum_{\tau=1}^T \gamma_\tau}, \quad [\Theta = \max_{z, z' \in Z} V_z(z')]$$

with all consequences related to the rate of convergence, stepsize policies, etc.

$$z_1 \in Z; z_{t+1} = \text{Prox}_{z_t}(\gamma_t F(z_t)); z^t = [\sum_{\tau=1}^t \gamma_\tau]^{-1} \sum_{\tau=1}^t \gamma_\tau z_\tau$$

$$[z_\tau = [x_\tau; y_\tau]]$$

Proof of Fact I: As always, we have

$$\forall u = [\xi; \eta] \in Z : \sum_{\tau=1}^t \gamma_\tau \langle F(z_\tau), z_\tau - u \rangle \leq \Theta + \frac{1}{2} \sum_{\tau=1}^T \gamma_\tau^2 \|F(z_\tau)\|_*^2$$

and

$$\begin{aligned} \langle F(z_\tau), z_\tau - u \rangle &= \langle \phi'_x(x_\tau, y_\tau), x_\tau - \xi \rangle + \langle -\phi'_y(x_\tau, y_\tau), y_\tau - \eta \rangle \\ &\geq [\phi(x_\tau, y_\tau) - \phi(\xi, y_\tau)] + [-\phi(x_\tau, y_\tau) + \phi(x_\tau, \eta)] \\ &= \phi(x_\tau, \eta) - \phi(\xi, y_\tau) \end{aligned}$$

\Rightarrow setting $\Gamma_t = \sum_{\tau=1}^t \gamma_\tau$ and $\lambda_\tau = \gamma_\tau / \Gamma_t$, we get

$$\underbrace{\sum_{\tau=1}^t \lambda_\tau [\phi(x_\tau, \eta) - \phi(\xi, y_\tau)]}_{\geq \phi(x^t, \eta) - \phi(\xi, y^t)} \leq \frac{\Theta + \frac{1}{2} \sum_{\tau=1}^t \gamma_\tau^2 \|F(z_\tau)\|_*^2}{\sum_{\tau=1}^T \gamma_\tau}.$$

$$\Rightarrow \forall ([\xi; y] \in X \times Y) : \phi(x^t, \eta) - \phi(\xi, y^t) \leq \frac{\Theta + \frac{1}{2} \sum_{\tau=1}^t \gamma_\tau^2 \|F(z_\tau)\|_*^2}{\sum_{\tau=1}^T \gamma_\tau}.$$

The supremum of the left hand side in $\xi \in X, \eta \in Y$ is $\epsilon_{\text{Sad}}(x^t, y^t)$, and we arrive at the required result

$$\epsilon_{\text{Sad}}(x^t, y^t) \leq \frac{\Theta + \frac{1}{2} \sum_{\tau=1}^T \gamma_\tau^2 \|F(z_\tau)\|_*^2}{\sum_{\tau=1}^T \gamma_\tau},$$

Mirror-Prox Scheme

Saddle Point Mirror Descent for $\min_{x \in X} \max_{y \in Y} \phi(x, y)$:

$$z_1 \in Z = X \times Y; z_{t+1} = \text{Prox}_{z_t}(\gamma_t F(z_t)); z^N = [\gamma_1 + \dots + \gamma_N]^{-1} \sum_{t=1}^N \gamma_t z_t$$

♣ Consider the *extragradient* Saddle Point MD:

$$z_1 \in Z = X \times Y; z_t \mapsto w_t = \text{Prox}_{z_t}(\gamma_t F(z_t)); w_t \mapsto z_{t+1} = \text{Prox}_{z_t}(\gamma_t F(w_t));$$
$$z^t = \left[\sum_{\tau=1}^t \gamma_\tau \right]^{-1} \sum_{\tau=1}^t \gamma_\tau w_\tau$$

♣ **Fact II:** Let F be Lipschitz:

$$\|F(z) - F(z')\|_* \leq L \|z - z'\|.$$

Then the constant stepsizes

$$\gamma_t \equiv \gamma = \frac{1}{L}$$

ensure that

$$\epsilon_{\text{Sad}}(z^t) \leq \frac{\Theta}{t\gamma} = \frac{\Theta L}{t}, t = 1, 2, \dots \quad [1/t \text{ rate!!!}]$$

$$z_1 \in Z; w_t = \text{Prox}_{z_t}(\gamma_t F(z_t)); z_{t+1} = \text{Prox}_{z_t}(\gamma_t F(w_t));$$

$$z^t = \left[\sum_{\tau=1}^t \gamma_\tau \right]^{-1} \sum_{\tau=1}^t \gamma_\tau w_\tau$$

Proof of Fact II: Magic Inequality states

$$(a) \quad \forall u \in Z : \langle \gamma_t F(w_t), z_{t+1} - u \rangle \leq V_{z_t}(u) - V_{z_{t+1}}(u) - V_{z_t}(z_{t+1})$$

$$(b) \quad \forall v \in Z : \langle \gamma_t F(z_t), w_t - v \rangle \leq V_{z_t}(v) - V_{w_t}(v) - V_{z_t}(w_t)$$

Applying (b) to $v = z_{t+1}$, we get

$$\langle \gamma_t F(z_t), w_t - z_{t+1} \rangle \leq V_{z_t}(z_{t+1}) - V_{w_t}(z_{t+1}) - V_{z_t}(w_t),$$

while (a) implies

$$\begin{aligned} \langle \gamma_t F(w_t), w_t - u \rangle &\leq V_{z_t}(u) - V_{z_{t+1}}(u) - V_{z_t}(z_{t+1}) + \gamma_t \langle F(w_t), w_t - z_{t+1} \rangle \\ \Rightarrow \langle \gamma_t F(w_t), w_t - u \rangle &\leq V_{z_t}(u) - V_{z_{t+1}}(u) - V_{z_t}(z_{t+1}) + \gamma_t \langle F(z_t), w_t - z_{t+1} \rangle + \gamma_t \langle F(w_t) - F(z_t), w_t - z_{t+1} \rangle \end{aligned}$$

Taken together, these inequalities imply that

$$\begin{aligned} \langle \gamma_t F(w_t), w_t - u \rangle &\leq V_{z_t}(u) - V_{z_{t+1}}(u) + [\gamma_t \langle F(w_t) - F(z_t), w_t - z_{t+1} \rangle - V_{w_t}(z_{t+1}) - V_{z_t}(w_t)] \\ &\leq V_{z_t}(u) - V_{z_{t+1}}(u) + \left[\frac{1}{2} \gamma_t^2 \|F(z_t) - F(w_t)\|_*^2 - V_{z_t}(w_t) \right] \end{aligned}$$

Now let F be Lipschitz: $\|F(z) - F(z')\|_* \leq L\|z - z'\|$. Since $V_{z_t}(w_t) \geq \frac{1}{2}\|w_t - z_t\|^2$, we get

$$\langle \gamma_t F(w_t), w_t - u \rangle \leq V_{z_t}(u) - V_{z_{t+1}}(u) + \frac{1}{2} \|w_t - z_t\|^2 [L^2 \gamma_t^2 - 1],$$

and we end up with

$$\gamma_t \equiv \gamma = \frac{1}{L} \forall t \Rightarrow \gamma \langle F(w_t), w_t - u \rangle \leq V_{z_t}(u) - V_{z_{t+1}}(u) \quad \forall u \in Z,$$

whence by the same argument as in the end of proof of Fact I we have

$$\epsilon_{\text{Sad}}(z^t) \leq \frac{\Theta}{t\gamma} = \frac{\Theta L}{t}, \quad t = 1, 2, \dots \quad [1/t \text{ rate!!!}]$$

♣ **Conclusion:** *When the objective of a convex optimization problem*

$$\text{Opt} = \min_{x \in X} f(x)$$

with convex compact X admits saddle point representation:

$$f(x) = \max_{y \in Y} \phi(x, y)$$

*with convex-concave **smooth** (with Lipschitz continuous gradient) ϕ and convex compact Y , we can solve the problem at the rate $O(1/t)$, provided we can equip X and Y with “computationally cheap” proximal setup (i.e., with norms and DGF’s resulting in easy-to-compute prox-mappings).*

Stochastic Saddle Point Mirror Descent and Acceleration by Randomization

♠ Consider a convex-concave saddle point problem

$$\begin{aligned} \text{SV} &= \min_{x \in X} \max_{y \in Y} \phi(x, y) & (\text{SP}) \\ \Rightarrow F(z = (x, y)) &= [F_x(x, y) \in \partial_x \phi(x, y); F_y(x, y) \in \partial_y [-\phi(x, y)]] \end{aligned}$$

- $X \subset E_x, Y \subset E_y$: nonempty closed and bounded convex sets in Euclidean spaces E_x, E_y
- $\phi : X \times Y \rightarrow \mathbf{R}$: Lipschitz continuous and convex-concave

♠ $Z = X \times Y$ is equipped with Proximal setup – a norm $\|\cdot\|$ on $E = E_x \times E_y$ and a compatible with this norm DGF $\omega : Z \rightarrow \mathbf{R}$.

♠ Assume that the field F is given by Stochastic Oracle:

When calling the oracle at step t , the query point being $z_t = (x_t, y_t)$, the oracle returns a random estimate $G(z_t, \xi_t)$ of $F(z_t)$ which is unbiased and “stochastically bounded”:

$$\forall z \in Z = X \times Y : \mathbf{E}\{G(z, \xi)\} = F(z) \ \& \ \mathbf{E}\{\|G(z, \xi)\|_*^2\} \leq L^2.$$

As always, ξ_1, ξ_2, \dots are independent realizations of a random variable ξ .

$$\text{SV} = \min_{x \in X} \max_{y \in Y} \phi(x, y) \quad (\text{SP})$$

$$F(x, y) = [F_x(x, y) \in \partial_x \phi(x, y); F_y(x, y) \in \partial_y [-\phi(x, y)]]$$

$$G(z, \xi) : \mathbf{E}_\xi \{G(z, \xi)\} = F(z) \ \& \ \mathbf{E}_\xi \{\|G(z, \xi)\|_*^2\} \leq L^2 \ \forall z = [x; y] \in Z = X \times Y$$

♠ Stochastic Saddle Point Mirror Descent for (SP) is the recurrence

$$z_1 \in Z; z_{t+1} = \text{Prox}_{z_t}(\gamma_t G(z_t, \xi_t)); z^t = \left[\sum_{\tau=1}^t \gamma_\tau \right]^{-1} \sum_{\tau=1}^t \gamma_\tau z_\tau. \quad [\gamma_\tau > 0]$$

Theorem: [Lecture Notes, Theorem 5.3.6] *For the above recurrence one has*

$$\mathbf{E} \{ \epsilon_{\text{Sad}}(z^t) \} \leq \frac{7}{2} \cdot \frac{2\Theta + L^2 \sum_{\tau=1}^t \gamma_\tau^2}{\sum_{\tau=1}^t \gamma_\tau}.$$

$$[\Theta = \max_{u, v \in Z} \{V_u(v) := \omega(v) - \omega(u) - \langle v - u, \nabla \omega(u) \rangle\}]$$

In particular, given a number N of iterations and setting

$$\gamma_t = \frac{\sqrt{2\Theta}}{L\sqrt{N}}, \ 1 \leq t \leq N,$$

we ensure that

$$\mathbf{E} \{ \epsilon_{\text{Sad}}(z^N) \} \leq \frac{7\sqrt{2\Theta}L}{\sqrt{N}}.$$

Note: Similar results hold true for Mirror Prox.

♣ **Application: Matrix Game.** *Matrix Game* problem is as follows:

$$\begin{aligned} \text{SV} &= \min_{x \in \Delta_n} \max_{y \in \Delta_m} y^T A x & (\text{MG}) \\ [\Delta_p &= \{u \in \mathbf{R}^p : u \geq 0, \sum_i u_i = 1\}] \end{aligned}$$

Interpretation: Two players are playing an antagonistic game; the first selects a $j \in \{1, \dots, n\}$, the second selects an $i \in \{1, \dots, m\}$. The loss of the first player (i.e., the profit of the second player) is A_{ij} , where A is a given $m \times n$ matrix. Naturally, the first player wants to reduce his losses, and the second player wants to increase his profit.

- When players make their choices simultaneously, there is *no* natural definition of “equilibrium,” unless the matrix has a “saddle point” – some entry A_{i_*, j_*} is minimal in its column and is maximal in its row.
- In the general case, the concept of a solution to the game, going back to von Neumann and Morgenstern, is to look what happens when the players repeat the matrix game many times, drawing their choices at random independently of each other and across the time. Denoting by $x \in \Delta_n$ the probability distribution from which the first player draws his choices, and by $y \in \Delta_m$ similar distribution for the second player, the expected loss of the first player (expected profit of the second player) will be

$$y^T A x$$

Thus, (MG) can be thought of as the problem of finding *the best randomized* policies of the players (called their *mixed strategies*); if both players are interested in their long run losses and profits, sticking to the mixed strategies given by a saddle point of the *bilinear* (and thus convex-concave) game (MG) will be optimal policies for every one of them.

$$SV = \min_{x \in \Delta_n} \max_{y \in \Delta_m} y^T A x \quad (\text{MG})$$

$$[\Delta_p = \{u \in \mathbf{R}^p : u \geq 0, \sum_i u_i = 1\}]$$

(MG) is just a primal-dual pair of LP programs:

$$\begin{aligned} \text{Opt}(P) &= \min_{x \in \Delta_n} \max_i \text{Row}_i^T[A] x \\ \text{Opt}(D) &= \max_{y \in \Delta_m} \min_j \text{Col}_j^T[A] y \end{aligned}$$

where $\text{Row}_i^T[A]$ is i -th row, and $\text{Col}_j[A]$ is j -th column in A .

⇒ (MG) can be solved by interior point LP methods.

$$\begin{aligned} \text{SV} &= \min_{x \in \Delta_n} \max_{y \in \Delta_m} y^T A x & (\text{MG}) \\ [\Delta_p &= \{u \in \mathbf{R}^p : u \geq 0, \sum_i u_i = 1\}] \end{aligned}$$

♠ In the large-scale case, (MG) can be solved by Mirror Prox; with appropriate setup, MP yields the efficiency estimate

$$\epsilon_{\text{Sad}}(x^N, y^N) \leq O(1) \sqrt{\ln(n) \ln(m)} \max_{i,j} |A_{ij}| / N$$

The complexity of a step is $O(m + n)$ plus the complexity of two matrix-vector multiplications:

$$\Delta_n \ni x \mapsto Ax, \quad \Delta_m \ni y \mapsto A^T y$$

needed to compute the associated with (MG) vector field

$$F(x, y) = \left[\begin{array}{c|c} & A^T \\ \hline -A & \end{array} \right] \begin{bmatrix} x \\ y \end{bmatrix}.$$

When A is a general-type dense matrix, the arithmetic complexity of finding an ϵ -solution to the problem is therefore

$$\mathcal{C}_{\text{determ}}(\epsilon) = O(1) \sqrt{\ln(m) \ln(n)} \mathit{mn} \frac{\max_{i,j} |A_{ij}|}{\epsilon} \text{ flop.}$$

Can we do better?

♣ **Observation:** Computing matrix-vector multiplication

$$\mathbf{R}^p \ni u \mapsto Bu \in \mathbf{R}^q$$

is easy to randomize:

— the vector $v = \text{abs}[u]/\|u\|_1$ (abs acts coordinatewise) is a probabilistic vector (non-negative entries summing up to 1). Treating v as a probability distribution on $\{1, 2, \dots, p\}$, we draw at random an index j from this distribution and return

$$\eta = \|u\|_1 \text{sign}(u_j) \text{Col}_j(B),$$

thus ensuring that $\mathbf{E}\{\eta\} = Bu$.

— generating a realization of η is cheap:

— drawing j costs $O(p)$ flop: in $O(p)$ flop one computes the “cumulative distribution”

$$U_j = \|u\|_1^{-1} \sum_{k < j} |u_k|, \quad 1 \leq j \leq p,$$

of the probabilistic vector, generates $\zeta \sim \text{Uniform}[0, 1]$ and needs $O(\ln(p))$ comparisons to find by Bisection j such that

$$U_{j-1} < \zeta \leq U_j$$

— after j is generated, computing η takes just $O(q)$ flop

\Rightarrow *arithmetic cost of computing η is $O(1)(p + q)$*

• Whatever be a norm $\|\cdot\|$, the noise of our oracle is under control:

$$\|\eta\| \leq \|u\|_1 \max_j \|\text{Col}_j[B]\|.$$

The situation is especially nice when $\|u\|_1$ can be bounded in advance.

$$\text{SV} = \min_{x \in \Delta_n} \max_{y \in \Delta_m} y^T A x \quad (\text{MG})$$

$$[\Delta_p = \{u \in \mathbf{R}^p : u \geq 0, \sum_i u_i = 1\}] \Rightarrow F(x, y) = \left[\begin{array}{c|c} & A^T \\ \hline -A & \end{array} \right] \begin{bmatrix} x \\ y \end{bmatrix}$$

♠ Applying the above approach to (MG), we get a cheap randomized oracle for F ; a call to this oracle costs just $O(m+n)$ flop, vs. the cost $O(mn)$ of the precise computation of F .

\Rightarrow Utilizing the cheap stochastic oracle in MD, we get an algorithm for solving (MG) which ensures

$$\mathbf{E} \{ \epsilon_{\text{Sad}}(x^N, y^N) \} \leq O(1) \sqrt{\ln(m) \ln(n)} \left(\frac{\max_{i,j} |A_{ij}|}{\sqrt{N}} \right),$$

with $O(m+n)$ flop per step.

\Rightarrow For every $\epsilon > 0, \delta \in (0, 1)$, one can build in $(1 - \delta)$ -reliable fashion an ϵ -solution to (MG) at the cost of

$$\mathcal{C}_{\text{rand}}(\epsilon) = C(\delta) \ln(n) \ln(m) (m+n) / \chi^2 \text{ flop} \\ [\chi = \epsilon / \max_{i,j} |A_{ij}|: \text{relative accuracy}]$$

which for fixed δ, χ and large m, n is by orders of magnitude better than the best known “deterministic cost”

$$\mathcal{C}_{\text{determ}}(\epsilon) = O(1) \cdot \sqrt{\ln(m) \ln(n)} mn / \chi \text{ flop.}$$

of ϵ -solution to (MG).

$$\mathcal{C}_{\text{rand}}(\epsilon) = C(\delta) \ln(n) \ln(m)(m+n)/\chi^2 \text{ flop}$$

[$\chi = \epsilon / \max_{i,j} |A_{ij}|$: relative accuracy]

Note: Our algorithm exhibits *sublinear time behavior*: for fixed χ and large m, n , *reliable design of ϵ -solution requires inspection of a negligibly small, going to 0 as m, n grow, randomly selected fraction of the data.*

An “ad hoc” algorithm with this property (in retrospect, pretty similar to Stochastic MD Approximation) was discovered in 1995 by Grigoriadis and Khachiyan.

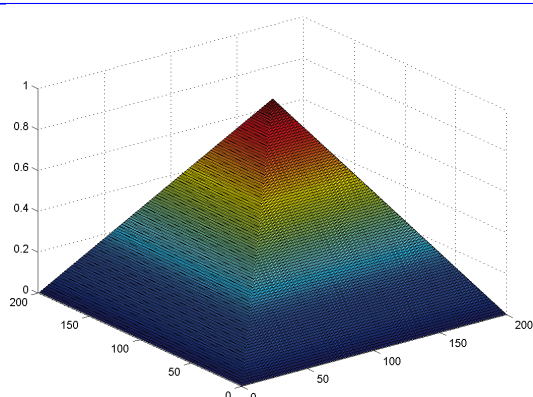
♣ **Illustration:** There are N houses in a city, i -th with wealth w_i . Every evening, Burglar selects a house i to be attacked, and Policeman selects his location at a house j . When the burglary starts, the probability for Policeman to react to alarm and to prevent the burglary is $\exp\{-\theta d(i, j)\}$, where $d(i, j)$ is the distance between locations i and j , so that the expected profit of Burglar is $A_{ij} = w_i[1 - \exp\{-\theta d(i, j)\}]$. Our goal is to solve in mixed strategies the resulting game

$$\max_{y \in \Delta_N} \min_{x \in \Delta_N} y^T A x.$$

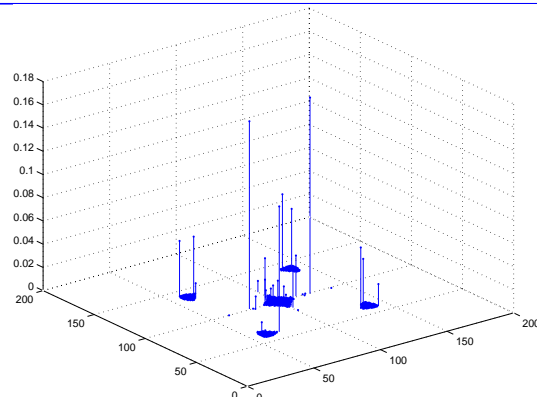
♠ Assuming an $n \times n$ equidistant grid of houses with wealth decreasing from the downtown to outskirts, the resulting $(N := n^2) \times N$ matrix game was solved by the state-of-the-art commercial LP Interior Point Method (IPM) `mosekopt`, by the Deterministic Mirror Prox and by the randomized MD seeking $\epsilon_{\text{Sad}} < 0.001$, with CPU limit of 5,300 sec. Here are the results:

N	IPM			DMP			RMD		
	Steps	CPU	Gap	Steps	CPU	Gap	Steps	CPU	Gap
1600	21	120	6.0e-9	78	6	1.0e-3	10556	264	1.0e-3
6400	21	6930	1.1e-8	80	31	1.0e-3	10408	796	1.0e-3
14400	not tested			95	171	1.0e-3	9422	1584	1.0e-3
40000	out of memory			15	5533	0.022	10216	4931	1.0e-3

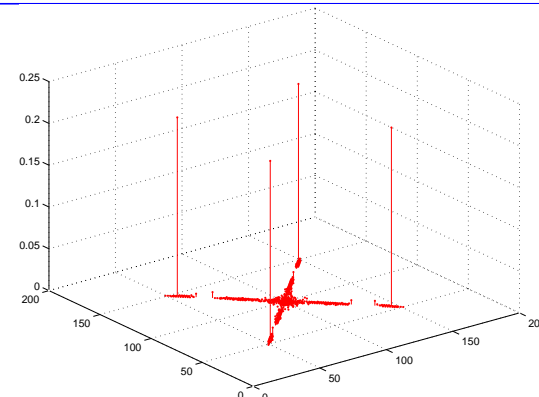
Policeman vs. Burglar, N houses



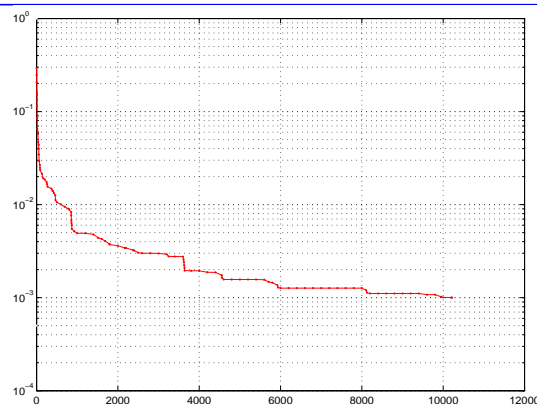
Wealth



Policeman



Burglar



ϵ_{Sad} vs. iteration count

Policeman vs. Burglar, $N = 40,000$. RMD with 10,216 steps (4931 sec)

Smooth Convex Minimization: Nesterov's Fast Gradient Method

♣ Problem of interest: Composite minimization

$$\text{Opt} = \min_{x \in X} \{\phi(x) = \Psi(x) + f(x)\}$$

- X : closed convex nonempty subset in Euclidean space E
 (X, E) is equipped with proximal setup $(\omega(\cdot), \|\cdot\|)$
- $\Psi : X \rightarrow \mathbf{R}$: convex and continuous
- $f : X \rightarrow \mathbf{R}$: represented by FO oracle convex function
with Lipschitz continuous gradient:
 $\forall x, y \in X : \|\nabla f(x) - \nabla f(y)\|_* \leq L_f \|x - y\|$

♠ **Main Assumption:** We are able to compute *composite prox-mappings*, i.e., solve auxiliary problems

$$\min_{x \in X} \{\omega(x) + \langle h, x \rangle + \alpha \Psi(x)\} \quad [\alpha \geq 0]$$

♥ **Example:** LASSO problem

- $$\min_{x \in X} \left\{ \overbrace{\lambda \|x\|_E}^{\Psi(x)} + \overbrace{\frac{1}{2} \|A(x) - b\|_2^2}^{f(x)} \right\}$$
- $\|\cdot\|_E$:
 - (a) block ℓ_1/ℓ_2 norm $\sum_{j=1}^n \|x^j\|_2$ on $E = \mathbf{R}^{k_1} \times \dots \times \mathbf{R}^{k_n}$ (ℓ_1 case)
 - (b) nuclear norm on the space E of block diagonal matrices of a given block diagonal structure (*nuclear norm case*)
 - $A(\cdot) : E \rightarrow \mathbf{R}^m$: linear mapping
 - X : either the unit $\|\cdot\|_E$ -ball, or the entire E

♥ For properly chosen proximal setup, Main Assumption is satisfied: *computing composite prox mapping*

$$\min_{x \in X} \{ \omega(x) + \langle h, x \rangle + \alpha \Psi(x) \} \quad [\alpha \geq 0]$$

takes $O(\dim E)$ a.o. in the case of (a) and reduces to computing singular value decomposition of a matrix from E in the case of (b).

Example: $\|\cdot\|_E$ is $\|\cdot\|_1$ norm on \mathbf{R}^n (“sparse recovery”).

- With Ball setup $\|\cdot\| = \|\cdot\|_2$, $\omega(\cdot) = \frac{1}{2}\|\cdot\|_2^2$ computing composite prox-mapping reduces to solving the problem

$$\min_x \left\{ \sum_i [h_i x_i + \beta |x_i| + \frac{1}{2} x_i^2] : x \in X \right\} \quad [\beta \geq 0]$$

The problem is trivial when $X = \mathbf{R}^n$ or X is a box $a \leq x \leq b$. When X is the unit $\|\cdot\|_p$ -ball, $1 \leq p < \infty$, the problem still is easy – it reduces to *one-dimensional* Lagrange dual problem

$$\max_{\lambda \geq 0} \left[\underline{L}(\lambda) := \min_{x \in \mathbf{R}^n} \underbrace{\sum_i [h_i x_i + \beta |x_i| + \frac{1}{2} x_i^2 + \lambda |x_i|^p]}_{\text{easy to compute}} - \lambda \right]$$

- When $X = E$ or $X = \{x \in E : \|x\|_E \leq 1\}$, computing composite prox-mapping remains easy when the Ball proximal setup is replaced with ℓ_1/ℓ_2 one.

Nesterov's Fast Gradient algorithm for Composite Minimization

♣ Problem:

$$\begin{aligned} \text{Opt} &= \min_{x \in X \subseteq E} \{\phi(x) := \Psi(x) + f(x)\} \\ &\bullet \Psi, f: \text{convex and} \\ &\forall x, y \in X : \|\nabla f(x) - \nabla f(y)\|_* \leq L_f \|x - y\| \end{aligned} \tag{CP}$$

♠ **Assumptions:** L_f is known and (CP) is solvable with an optimal solution x_* .

♠ The algorithm is described in terms of proximal setup $(\omega(\cdot), \|\cdot\|)$ for X and auxiliary sequence

$$\{L_t \in (0, L_f]\}_{t=0}^{\infty}$$

which can be adjusted on-line.

Recall that DGF ω defines Bregman distance

$$V_x(y) = \omega(y) - \omega(x) - \langle \nabla \omega(x), y - x \rangle \quad [x, y \in X]$$

$$\text{Opt} = \min_{x \in X \subseteq E} \{\phi(x) := \Psi(x) + f(x)\}$$

♣ **Algorithm:**

♠ **Initialization:** Set

$$A_0 = 0, y_0 = x_\omega = \operatorname{argmin}_X \omega, \psi_0(x) = V_{x_\omega}(x)$$

and select $y_0^+ \in X$ such that $\phi(y_0^+) \leq \phi(y_0)$.

♠ **Step** $t = 0, 1, 2, \dots$: Given $\psi_t(\cdot) = \omega(\cdot) + \alpha \Psi(\cdot) + \langle \text{affine form} \rangle$ [$\alpha \geq 0$], $y_t^+ \in X$, $A_t \in \mathbf{R}_+$, and L_t , $0 < L_t \leq L_f$,

• Compute $z_t = \operatorname{argmin}_{x \in X} \psi_t(x)$ (reduces to computing composite prox-mapping)

• Find the positive root a_{t+1} of the equation $L_t a_{t+1}^2 = A_t + a_{t+1}$ and set

$$A_{t+1} = A_t + a_{t+1}, \tau_t = a_{t+1}/A_{t+1} \in (0, 1]$$

• Set $x_{t+1} = \tau_t z_t + (1 - \tau_t) y_t^+$ and compute $f(x_{t+1}), \nabla f(x_{t+1})$

• Compute $\hat{x}_{t+1} = \operatorname{argmin}_{x \in X} \left\{ \langle \nabla f(x_{t+1}), x \rangle + \Psi(x) + \frac{1}{a_{t+1}} V_{z_t}(x) \right\}$ (reduces to computing composite prox-mapping)

• Set

$$\begin{aligned} y_{t+1} &= \tau_t \hat{x}_{t+1} + (1 - \tau_t) y_t^+ \\ \psi_{t+1}(x) &= \psi_t(x) + a_{t+1} [f(x_{t+1}) + \langle \nabla f(x_{t+1}), x - x_{t+1} \rangle + \Psi(x)] \end{aligned}$$

and select somehow $y_{t+1}^+ \in X$ such that $\phi(y_{t+1}^+) \leq \phi(y_{t+1})$.

• Finally, select $L_{t+1} \in (0, L_f]$.

Step t is completed; go to step $t + 1$.

♣ **Theorem** [Yu. Nesterov '83, '07] Assume that the sequence $\{L_t \in (0, L_f]\}$ is such that

$$\frac{V_{z_t}(\widehat{x}_{t+1})}{A_{t+1}} + \langle \nabla f(x_{t+1}), y_{t+1} - x_{t+1} \rangle + f(x_{t+1}) \geq f(y_{t+1})$$

(this for sure is the case when $L_t \equiv L_f$). Then

$$\phi(y_t^+) - \text{Opt} \leq A_t^{-1} V_{x_\omega}(x_*) \leq \frac{4L_f}{t^2} V_{x_\omega}(x_*), \quad t = 1, 2, \dots$$

♠ **Illustration:** As applied to a solvable LASSO problem

$$x_* = \operatorname{argmin}_x \left\{ \phi(x) := \lambda \|x\|_E + \frac{1}{2} \|A(x) - b\|_2^2 \right\}$$

with $\|\cdot\|_E$ either (a) block ℓ_1/ℓ_2 norm on $E = \underbrace{\mathbf{R}^{k_1} \times \dots \times \mathbf{R}^{k_n}}_{n \text{ factors}}$, or (b) nuclear norm on

$E = \mathbf{R}^{p \times q}$ with $n = \min[p, q]$, the Fast Gradient method with appropriate proximal setup in $t = 1, 2, \dots$ steps ensures

$$\phi(y_t^+) \leq \text{Opt} + O(\ln(n+1)) \frac{\|A\|_{E,2}^2}{t^2} \|x_*\|_E^2$$

where $\|A\|_{E,2} = \max\{\|A(x)\|_2 : \|x\|_E \leq 1\}$

♣ **Note:** $O(1/t^2)$ rate of convergence is, seemingly, the best one can expect from oracle-based methods in the large scale case.

The precise statement is as follows:

♡ Let n be a positive integer. Consider Least Squares problems

$$\text{Opt} = \min_x \|Ax - b\|_2^2 \quad (QP)$$

with $n \times n$ symmetric matrices A .

For every positive reals R, L and every number $t \leq n/4$ of steps, for every t -step solution algorithm \mathcal{B} operating with the “multiplication oracle” $u \mapsto Au$ one can find an instance of (QP) such that

- the spectral norm of A does not exceed L ,
- $\text{Opt} = 0$, and the $\|\cdot\|_2$ -norm of some optimal solution does not exceed R ,
- the approximate solution y generated by \mathcal{B} , as applied to the instance, after t calls to the oracle, satisfies

$$\|Ay - b\|_2^2 \geq O(1) \frac{L^2 R^2}{t^2}$$

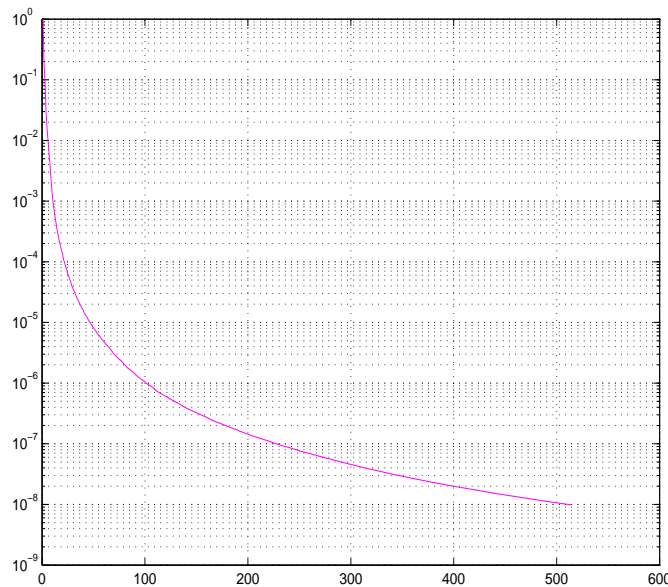
How it Works: Fast Composite Minimization for LASSO

♣ Test problem:

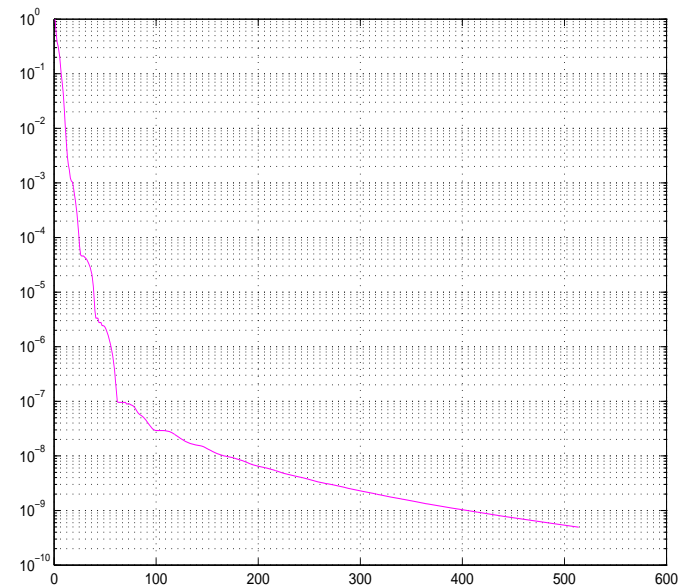
$$\text{Opt} = \min_x \left\{ \phi(x) := 0.01\|x\|_1 + \frac{1}{2}\|Ax - b\|_2^2 \right\}$$

with 4096×2048 randomly generated matrix A .

Method	Setup	Iterations	CPU, sec	Nonoptimality
IPM	—	11	103.1	<1.e-12
FGr	Ball setup	512	36.3	2.4e-6
FGr	ℓ_1/ℓ_2 setup	512	36.5	1.2e-7



Ball setup



ℓ_1 setup

Progress in accuracy $\frac{\phi(y_t^+) - \text{Opt}}{\phi(y_0^+) - \text{Opt}}$ vs. t

Platform: 2×3.40 GHz CPU, 16.0 GB RAM, 64-bit Windows 7

Prehistory of Fast Gradients

♠ Nesterov's Fast Gradient Algorithm hardly can be treated as intuitive, and its justification, while short, is a miraculous purely algebraic manipulation. We believe that the construction *is* a miracle, and as such it should be learned and used, but not “explained.” This being said, the “prehistory predecessors” of this magic algorithm are quite understandable.

Situation and goal: *Convex function $f : \mathbf{R}^n \rightarrow \mathbf{R}$ has Lipschitz continuous with constant 1 gradient:*

$$\|f'(x) - f'(y)\|_2 \leq \|x - y\|_2 \quad \forall x, y$$

and achieves its minimum at some point x_ . We want to design a First Order algorithm which ensures that*

$$f(x_k) - f(x_*) \leq O(1/k^2), \quad k = 1, 2, \dots$$

Step 0: Quadratic case. Assume that f is quadratic. Then the “method of choice” is Conjugate Gradients which, on a closest inspection, indeed converges at the rate $O(1/k^2)$, and it is easy to understand *simple* reasons for that.

- Let the starting point be $x_0 = 0$. Then k -th iterate of CG is the minimizer of f on the linear span of the gradients

$$g_t = f'(x_t)$$

at the iterates with $t < k$. As a result,

A. *The gradients $g_k = f'(x_k)$ along the CG trajectory are mutually orthogonal, and g_k is orthogonal to x_k ;*

B. $f(x_{k+1}) \leq f(x_k) - \frac{1}{2}\|g_k\|_2^2$.

Indeed, for every function h with Lipschitz continuous, with constant 1, gradient it holds

$$h(x - h'(x)) \leq h(x) - \frac{1}{2}\|h'(x)\|_2^2.$$

and for CG we clearly have $f(x_{k+1}) \leq f(x_k - g_k)$.

- Let $V_k = f(x_k) - f(x_*)$, and let λ_k be positive reals. We have

$$\begin{aligned}
\sum_{t=1}^k \lambda_t V_t &\leq \sum_{t=1}^k \lambda_t \langle g_t, x_t - x_* \rangle && \text{[by convexity]} \\
&= \sum_{t=1}^k \lambda_t \langle g_t, -x_* \rangle && \text{[} g_t \text{ and } x_t \text{ are orthogonal!]} \\
&= \langle \sum_{t=1}^k \lambda_t g_t, -x_* \rangle \\
&\leq \frac{1}{2} \left\| \sum_{t=1}^k \lambda_t g_t \right\|_2^2 + \frac{1}{2} \|x_*\|_2^2 && \text{[Cauchy Inequality]} \\
&= \frac{1}{2} \sum_{t=1}^k \lambda_t^2 \|g_t\|_2^2 + \frac{1}{2} \|x_*\|_2^2 && \text{[} g_1, \dots, g_k \text{ are mutually orthogonal!]} \\
&\leq \sum_{t=1}^k \lambda_t^2 [V_t - V_{t+1}] + \frac{1}{2} \|x_*\|_2^2 && \text{[since } f(x_{t+1}) \leq f(x_t) - \frac{1}{2} \|g_t\|_2^2] \\
&= \sum_{t=1}^k [\lambda_t^2 - \lambda_{t-1}^2] V_t - \lambda_k^2 V_{k+1} + \frac{1}{2} \|x_*\|_2^2 && \text{[here } \lambda_0 = 0]
\end{aligned}$$

From now on let $\lambda_t > 0$, $t \geq 1$, be given by the recurrence

$$\lambda_t^2 - \lambda_{t-1}^2 = \lambda_t \quad [\lambda_0 = 0]$$

- Then the above computation as applied with λ_t 's just specified yields

$$\lambda_k^2 V_{k+1} \leq \frac{1}{2} \|x_*\|_2^2$$

and, as is immediately seen, $\lambda_t \geq t/2$ for all t

$$\Rightarrow f(x_{k+1}) - \min f \leq \frac{2 \|x_*\|_2^2}{k^2}$$

$$\begin{aligned}
\sum_{t=1}^k \lambda_t V_t &\leq \sum_{t=1}^k \lambda_t \langle g_t, x_t - x_* \rangle && \text{[by convexity]} \\
&= \sum_{t=1}^k \lambda_t \langle g_t, -x_* \rangle && \text{[} g_t \text{ and } x_t \text{ are orthogona!]} \\
&= \langle \sum_{t=1}^k \lambda_t g_t, -x_* \rangle \\
&\leq \frac{1}{2} \left\| \sum_{t=1}^k \lambda_t g_t \right\|_2^2 + \frac{1}{2} \|x_*\|_2^2 && \text{[Cauchy Inequality]} \\
&= \frac{1}{2} \sum_{t=1}^k \lambda_t^2 \|g_t\|_2^2 + \frac{1}{2} \|x_*\|_2^2 && \text{[} g_1, \dots, g_k \text{ are mutually orthogonal!]} \\
&\leq \sum_{t=1}^k \lambda_t^2 [V_t - V_{t+1}] + \frac{1}{2} \|x_*\|_2^2 && \text{[since } f(x_{t+1}) \leq f(x_t) - \frac{1}{2} \|g_t\|_2^2] \\
&= \sum_{t=1}^k [\lambda_t^2 - \lambda_{t-1}^2] V_t - \lambda_k^2 V_{k+1} + \frac{1}{2} \|x_*\|_2^2 && \text{[here } \lambda_0 = 0]
\end{aligned}$$

Step 1. From Quadratic to Smooth Convex Case via 2D minimization. Looking at the above computation, observe that it still goes through if all that we ensure is

- orthogonality of $g_k = f'(x_k)$ to x_k and to $\sum_{t=1}^{k-1} \lambda_t g_t$, $k = 1, 2, \dots$
- inequality $f(x_{k+1}) \leq f(x_k) - \frac{1}{2} \|g_k\|_2^2$, $k = 1, 2, \dots$

Note:

- To ensure **a**, it suffices to define x_k as the minimizer of f on (any) linear subspace containing the vector $\sum_{t=1}^{k-1} \lambda_t g_t$
- To ensure **b**, it suffices to ensure that $f(x_{k+1}) \leq f(x_k - g_k)$.

\Rightarrow We arrive at $O(1/k^2)$ algorithm as follows:

Set $x_0 = 0$ and for $k = 1, 2, \dots$

— given x_{k-1} , set $\hat{x}_k = x_{k-1} - g_{k-1}$;

— define x_k as the minimizer of f on the linear span of \hat{x}_k and $\sum_{t=1}^{k-1} \lambda_t g_t$.

The required 2D minimization can be carried out (at nearly no cost) by Center of Gravity or by Ellipsoid Algorithm.

Note: Historically, this result was first obtained circa '79 with different selection of λ_k 's; the above elegant rule is part of Nesterov's breakthrough (1982)

Step 2: From 2D minimization to Line Search. Consider the following modification of the previous algorithm: *Set $x_0 = 0$, and for $k = 1, 2, \dots$*

— *given x_{k-1} , set $\hat{x}_k = x_{k-1} - g_{k-1}$*

— *specify x_k as the minimizer of f on the line $\hat{x}_k + \mathbf{R} \left[\hat{x}_k + \sum_{t=1}^{k-1} \lambda_t g_t \right]$.*

For this algorithm, $f(x_{t+1}) \leq f(\hat{x}_{t+1}) \leq f(x_t) - \frac{1}{2} \|g_t\|_2^2$, i.e.,

$$\frac{1}{2} \|g_t\|_2^2 \leq V_t - V_{t+1}, \quad (+)$$

g_t is orthogonal to $\hat{x}_t + \sum_{s=1}^{t-1} \lambda_s g_s$:

$$\langle g_t, \hat{x}_t \rangle = -\langle g_t, \sum_{s=1}^{t-1} \lambda_s g_s \rangle \quad (!)$$

and $x_t = \hat{x}_t + \gamma_t [\hat{x}_t + \sum_{s=1}^{t-1} \lambda_s g_s]$ for some $\gamma_t \in \mathbf{R}$, whence

$$-x_t = -(1 + \gamma_t) \hat{x}_t - \gamma_t \sum_{s=1}^{t-1} \lambda_s g_s \quad (*)$$

Now,

$$\begin{aligned} & -V_t \geq \langle g_t, x_* - x_t \rangle \text{ [by convexity]} \\ \Leftrightarrow & -V_t \geq \langle g_t, x_* \rangle + \langle g_t, -x_t \rangle \\ & = \langle g_t, x_* \rangle + (1 + \gamma_t) \langle g_t, -\hat{x}_t \rangle - \gamma_t \langle g_t, \sum_{s=1}^{t-1} \lambda_s g_s \rangle \text{ [by (*)]} \\ \Rightarrow & -V_t \geq \langle g_t, x_* \rangle + \langle g_t, \sum_{s=1}^{t-1} \lambda_s g_s \rangle \text{ [by (!)]} \\ \Rightarrow & \lambda_t V_t + \langle \lambda_t g_t, x_* \rangle + \langle \lambda_t g_t, \sum_{s=1}^{t-1} \lambda_s g_s \rangle \leq 0 \\ \Leftrightarrow & \lambda_t V_t + \langle \lambda_t g_t, x_* \rangle + \frac{1}{2} \left\| \sum_{s=1}^t \lambda_s g_s \right\|_2^2 - \frac{1}{2} \left\| \sum_{s=1}^{t-1} \lambda_s g_s \right\|_2^2 - \frac{1}{2} \lambda_t^2 \|g_t\|_2^2 \leq 0 \\ \Rightarrow & \sum_{t=1}^k \lambda_t V_t + \langle \sum_{t=1}^k \lambda_t g_t, x_* \rangle + \frac{1}{2} \left\| \sum_{t=1}^k \lambda_t g_t \right\|_2^2 - \frac{1}{2} \sum_{t=1}^k \lambda_t^2 \|g_t\|_2^2 \leq 0 \text{ [summing up over } t] \\ \Rightarrow & \sum_{t=1}^k \lambda_t V_t - \frac{1}{2} \left\| \sum_{t=1}^k \lambda_t g_t \right\|_2^2 - \frac{1}{2} \|x_*\|_2^2 + \frac{1}{2} \left\| \sum_{t=1}^k \lambda_t g_t \right\|_2^2 \leq \frac{1}{2} \sum_{t=1}^k \lambda_t^2 \|g_t\|_2^2 \text{ [due to } \langle a, b \rangle \geq -\frac{1}{2} \|a\|_2^2 - \frac{1}{2} \|b\|_2^2] \\ & \leq \sum_{t=1}^k \lambda_t^2 [V_t - V_{t+1}] \text{ [by (+)]} \\ \Rightarrow & \sum_{t=1}^k \lambda_t V_t \leq \frac{1}{2} \|x_*\|^2 + \sum_{t=1}^k \lambda_t^2 [V_t - V_{t+1}] \end{aligned}$$

The concluding inequality is exactly what led us to $V_{k+1} \leq \frac{\|x_*\|_2^2}{2\lambda_k^2} \leq \frac{2\|x_*\|_2^2}{k^2}$

♣ The above *algebraic manipulation* results in $O(1/k^2)$ algorithm

$$\begin{aligned} x_{k-1} \mapsto \hat{x}_k &:= x_{k-1} - g_{k-1} \mapsto x_k := \hat{x}_k + \gamma_k [\hat{x}_k + \sum_{t=1}^{k-1} \lambda_t g_t] \\ \gamma_k &\in \operatorname{Argmin}_{\gamma \in \mathbf{R}} f(\hat{x}_k + \gamma [\hat{x}_k + \sum_{t=1}^{k-1} \lambda_t g_t]) \\ &\left[g_t = f'(x_t), \lambda_t^2 - \lambda_{t-1}^2 = \lambda_t, \lambda_0 = 0, x_0 = 0 \right] \end{aligned}$$

Nesterov's breakthrough (1982) was in replacing the line search for identifying γ_k with *explicit formula* for γ_k . *This required completely new justification of the algorithm and paved road to important extensions, including*

- *passing from unconstrained to constrained minimization,*
- *passing from Euclidean to general proximal algorithms,*
- *passing from smooth convex to composite convex minimization,*
- *...*

Beyond the Scope of Proximal Algorithms: Conditional Gradients

$$\text{Opt} = \min_{x \in X} f(x)$$

♣ **Fact:** All considered so far “computationally cheap” large scale alternatives to IPM’s were *proximal type* First Order methods

♠ **But:** *In order to be computationally cheap, a proximal type method should operate with problems on Favorable Geometry domains X (those allowing for Proximal setup $(\|\cdot\|, \omega(\cdot))$ with moderate ω -capacity Θ , in order to have a reasonable iteration count) admitting easy to compute prox-mappings (“Simple Geometry,” otherwise an iteration becomes expensive).*

- ♠ Both Favorable and Simple Geometry requirements can be violated. For example,
- when X is a box, Favorable Geometry is missing
 - when X is a nuclear norm ball in $\mathbf{R}^{n \times n}$ or a spectahedron (the set of $\succeq 0$ matrices with unit trace) in \mathbf{S}^n , we do have Favorable Geometry, but computing the associated prox-mapping requires singular value decomposition of $n \times n$ matrix (or the eigenvalue decomposition of a symmetric $n \times n$ matrix), and both these computations require

$$O(n^3) = O((\dim X)^{3/2}) \text{ a.o.}$$

While much cheaper than the cost $O((\dim X)^3) = O(n^6)$ a.o. of an IPM iteration, $O(n^3)$ a.o. prox-mapping for large n becomes prohibitively time consuming.

Note: nuclear norm balls/spectahedrons arise naturally in many important applications, including, but not reducing to, low rank matrix recovery, multi-class classification in Machine Learning and high dimensional Statistics (and more generally – large scale Semidefinite programming).

♠ Another important example of generic problem with *Complex Geometry* is *Total Variation based Image Reconstruction*

$$\min_{x \in \mathbf{R}^{m \times n}} \left\{ \lambda \cdot \text{TV}(x) + \frac{1}{2} \|A(x) - b\|_2^2 \right\},$$

where $x = [x_{ij}] \in \mathbf{R}^{m \times n}$ is an $(m \times n)$ -pixel image, and $\text{TV}(x)$ is the *Total Variation*:

$$\text{TV}(x) = \sum_{i=1}^{m-1} \sum_{j=1}^n |x_{i+1,j} - x_{i,j}| + \sum_{i=1}^m \sum_{j=1}^{n-1} |x_{i,j+1} - x_{i,j}|$$

— the ℓ_1 -norm of the discrete gradient of $x = [x_{ij}]$. Restricted to the space $\mathbf{M}_0^{m,n}$ of $m \times n$ images *with zero mean*, TV becomes a norm.

For the unit TV-ball, no DGF compatible with the TV norm and leading to easy-to-compute prox mapping is known...

Linear Minimization Oracle

♣ **Observation:** When $X \subset E$ admits a proximal setup with easy-to-compute prox-mapping, X definitely admits a computationally cheap **Linear Minimization Oracle (LMO)** — a procedure which, given on input a linear form $\langle \eta, \cdot \rangle$, returns

$$x[\eta] \in \operatorname{Argmin}_{x \in X} \langle \eta, x \rangle$$

Indeed, the optimization program

$$\min_{x \in X} \langle \eta, x \rangle$$

is the “limiting case,” as $\theta \rightarrow +0$, of the programs

$$\min_{x \in X} \{ \theta \omega(x) + \langle \eta, x \rangle \}.$$

♠ **Fact:** Admitting a cheap LMO is a **much weaker** requirement than admitting proximal setup with cheap prox-mapping, and *there are important domains X with Complex Geometry admitting relatively cheap Linear Minimization Oracle.*

Examples:

A: Nuclear Norm ball $X = \{x \in \mathbf{R}^{m \times n} : \|x\|_{\text{nuc}} \leq 1\}$. Here computing $x[\eta]$ reduces to finding the left and the right *leading* singular vectors of $\eta \in \mathbf{R}^{m \times n}$, i.e., to solving the problem

$$\max_{\|u\|_2 \leq 1, \|v\|_2 \leq 1} u^T \eta v.$$

For large m, n , this is incomparably easier than finding full singular value decomposition of η required to compute prox-mapping.

B: Spectahedron $X = \{x \in \mathbf{S}^n : x \succeq 0, \text{Tr}(x) = 1\}$. Here computing $x[\eta]$ reduces to finding the leading eigenvector of $-\eta$, i.e., to solving the problem

$$\min_{\|u\|_2=1} u^T \eta u.$$

For large n , this is incomparably easier than finding full eigenvalue decomposition of η required to compute prox-mapping.

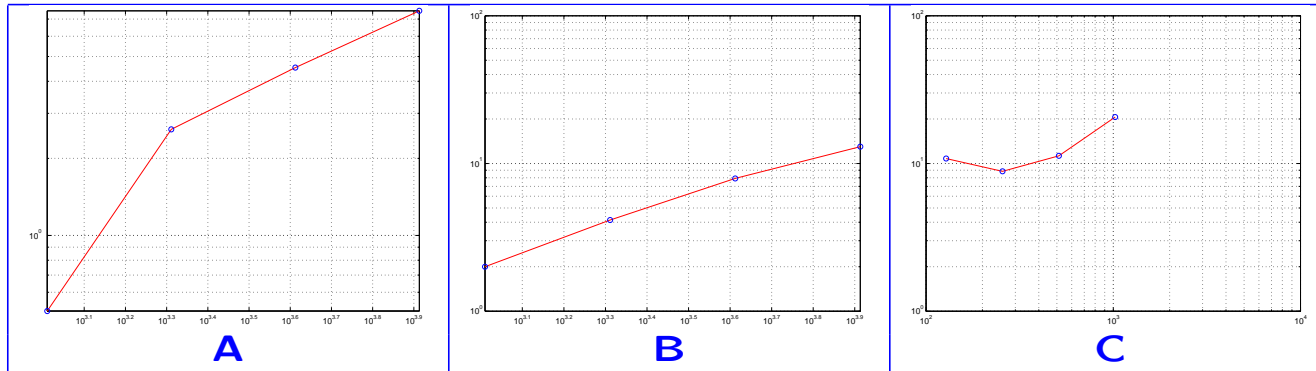
Examples (continued):

C: Unit TV-ball $X = \{x \in \mathbb{M}_0^{m,n} : \text{TV}(x) \leq 1\}$: For $\eta \in \mathbb{M}_0^{m,n}$, a point $x[\eta] \in \text{Argmin}_{x \in X} \text{Tr}(\eta x^T)$ is readily given by the optimal Lagrange multipliers for the *capacitated network flow problem*

$$\max_{t,f} \{t : \Gamma f = t\eta, \|f\|_\infty \leq 1\}$$

Γ : incidence matrix of the network with nodes (i, j) , $1 \leq i \leq m$, $1 \leq j \leq n$, and arcs $(i, j) \rightarrow (i+1, j)$, $(i, j) \rightarrow (i, j+1)$

♠ Illustration:



A: CPU ratio "full svd" / "finding leading singular vectors" for $n \times n$ matrix vs. n

n	1024	2048	4096	8192
CPU ratio	0.5	2.6	4.5	7.5

Full svd for $n = 8192$ takes 475.6 sec!

B: CPU ratio "full evd" / "finding leading eigenvector" for $n \times n$ symmetric matrix vs. n

n	1024	2048	4096	8192
CPU ratio	2.0	4.1	7.9	13.0

Full evd for $n = 8192$ takes 142.1 sec!

C: CPU ratio "metric projection" / "LMO computation" for TV ball in $M_0^{n,n}$ vs. n

n	129	256	512	1024
CPU ratio	10.8	8.8	11.3	20.6

Metric projection onto TV ball for $n = 1024$ takes 1062.1 sec!

Platform: 2×3.40 GHz CPU, 16.0 GB RAM, 64-bit Windows 7

Conditional Gradient Algorithm

$$\begin{aligned} \text{Opt} &= \min_{x \in X} f(x) \\ [\bullet \ X \subset E: \text{convex compact set} \ \bullet \ f : X \rightarrow \mathbb{R}: \text{convex}] & \quad (CM) \end{aligned}$$

W.l.o.g. we assume that X linearly spans the embedding Euclidean space E .

♣ When X is given by Linear Minimization oracle and f is smooth, (CM) can be solved by Conditional Gradient (CndG), a.k.a. Frank-Wolfe, algorithm given by the recurrence

$$\begin{aligned} x_1 \in X, \ x_{t+1} &\in X : f(x_{t+1}) \leq f\left(x_t + \frac{2}{t+1}(x_t^+ - x_t)\right), \\ &\left[x_t^+ = x[\nabla f(x_t)] \in \text{Argmin}_{y \in X} \langle \nabla f(x_t), y \rangle \right] \\ f_*^t &= \max_{\tau \leq t} [f(x_\tau) + \langle \nabla f(x_\tau), x_\tau^+ - x_\tau \rangle] \leq \text{Opt} \end{aligned}$$

♠ **Theorem:** Let $f : X \rightarrow \mathbb{R}$ be convex and (κ, L) -smooth:

$$\begin{aligned} \forall x, y \in X : f(y) &\leq f(x) + \langle \nabla f(x), y - x \rangle + \frac{L}{\kappa} \|x - y\|_X^\kappa \\ &\left[\begin{array}{l} \bullet \ L < \infty, \ \kappa \in (1, 2]: \text{ parameters} \\ \bullet \ \|\cdot\|_X: \text{ norm with the unit ball } \frac{1}{2}[X - X] \end{array} \right] \end{aligned}$$

When solving (CP) by CndG, one has for $t = 2, 3, \dots$

$$f(x_t) - \text{Opt} \leq f(x_t) - f_t^* \leq \frac{2^{2\kappa}}{\kappa(3 - \kappa)} \cdot \frac{L}{(t + 1)^{\kappa-1}}$$

$$\forall x, y \in X : f(y) \leq f(x) + \langle \nabla f(x), y - x \rangle + \frac{L}{\kappa} \|x - y\|_X^\kappa \quad (!)$$

[• $L < \infty, \kappa \in (1, 2]$: parameters]

Note: A *sufficient* condition for (!) is Hölder continuity of $\nabla f(x)$:

$$\|\nabla f(x) - \nabla f(y)\|_{X,*} \leq L \|x - y\|_X^{\kappa-1} \quad \forall x, y \in X$$

For convex f and $\kappa = 2$, this condition is also *necessary* for (!).

$$\forall x, y \in X : f(y) \leq f(x) + \langle \nabla f(x), y - x \rangle + \frac{L}{\kappa} \|x - y\|_X^\kappa$$

♣ Typically, the CndG rate of convergence $O(1/T^{\kappa-1})$ is **not** the best we can hope for. For example, when $\kappa = 2$ and X is either

- the unit $\|\cdot\|_p$ ball in \mathbf{R}^n with $1 \leq p \leq 2$, or
- the unit nuclear norm ball in $\mathbf{R}^{n \times n}$,

Nesterov's Fast Gradient method converges at the rate

$$O(1) \ln(n+1) L^2 / t^2,$$

and CndG only at the rate $O(1)L/t$. In fact,

♥ *In Favorable Geometry case, the only, if any, disadvantage of proximal algorithms as compared to CndG is the necessity to compute prox mappings, which could be expensive for problems with Complex Geometry.*

♠ *Beyond the case of Favorable Geometry, CndG can be optimal.*

Fact: *Let X be n -dimensional box:*

$$X = \{x \in \mathbf{R}^n : \|x\|_\infty \leq 1\}.$$

Then for every $t \leq n$, $L < \infty$, $\kappa \in (1, 2]$, and every utilizing local oracle t -step method \mathcal{B} for minimizing (κ, L) -smooth convex functions over X there exists a function f in the family such that for the approximate minimizer $x_{\mathcal{B}}$ of f generated by \mathcal{B} it holds

$$f(x_{\mathcal{B}}) - \min_X f \geq \frac{O(1)}{\ln(n)} \frac{L}{t^{\kappa-1}}$$

\Rightarrow *When minimizing smooth convex functions, represented by a local oracle, over an n -dimensional box, t -step CndG cannot be accelerated by more than $O(\ln(n))$ factor, provided $t \leq n$.*

- The result remains true when replacing n -dimensional box X with its matrix analogy

$$\{x \in \mathbf{R}^{n \times n} : \text{spectral norm of } x \text{ is } \leq 1\}$$

- When minimizing (κ, L) -smooth functions over n -dimensional $\|\cdot\|_p$ -balls with $2 \leq p \leq \infty$, the rate-of-convergence advantages of proximal algorithms over CndG rapidly deteriorate as p grows and disappears (up to $O(\ln(n))$ -factor) when p becomes as large as $O(\ln(n))$.

Proof of Theorem

$$(a) \quad f(y) \leq f(x) + \langle \nabla f(x), y - x \rangle + \frac{L}{\kappa} \|y - x\|_X^\kappa$$

$$(b) \quad f(x_{t+1}) \leq f(x_t + \gamma_t(x_t^+ - x_t)),$$

$$\gamma_t = \frac{2}{t+1}, \quad x_t^+ \in \text{Argmin}_{y \in X} \langle \nabla f(x_t), y \rangle$$

$$f_*^t := \max_{\tau \leq t} \underbrace{[f(x_\tau) + \langle \nabla f(x_\tau), x_\tau^+ - x_\tau \rangle]}_{\leq \min_X f}$$

$$? \Rightarrow ? \quad f(x_t) - f_*^t \leq \frac{\frac{2^{\kappa+1}L}{\kappa(3-\kappa)} \gamma_t^{\kappa-1}}{\gamma_t^{\kappa-1}} \quad (!_t), t \geq 2$$

Let

$$\epsilon_t = f(x_t) - f_*^t, \quad e_t = \langle \nabla f(x_t), x_t - x_t^+ \rangle$$

$$\bullet \quad f_*^t \geq f(x_t) + \langle \nabla f(x_t), x_t^+ - x_t \rangle \Rightarrow e_t \geq \epsilon_t$$

We have

$$(c) \quad \|x_t - x_t^+\|_X \leq 2$$

$$\begin{aligned} \Rightarrow \quad f(x_{t+1}) &\leq f(x_t + \gamma_t(x_t^+ - x_t)) \quad [\text{by (b)}] \\ &\leq f(x_t) + \gamma_t \langle \nabla f(x_t), x_t^+ - x_t \rangle + \frac{L}{\kappa} [2\gamma_t]^\kappa \\ &\quad [\text{by (a), (c)}] \end{aligned}$$

$$\begin{aligned} &= f(x_t) - \gamma_t e_t + \frac{2^\kappa L}{\kappa} \gamma_t^\kappa \\ &\leq f(x_t) - \gamma_t \epsilon_t + \frac{2^\kappa L}{\kappa} \gamma_t^\kappa \quad [\text{since } e_t \geq \epsilon_t] \end{aligned}$$

$$\begin{aligned} \Rightarrow \quad \epsilon_{t+1} = f(x_{t+1}) - f_*^{t+1} &\leq f(x_{t+1}) - f_*^t \\ &\quad [\text{since } f_*^{t+1} \geq f_*^t] \\ &\leq \epsilon_t (1 - \gamma_t) + \frac{2^\kappa L}{\kappa} \gamma_t^\kappa \end{aligned}$$

$$\begin{aligned}
& [0 \leq] \quad \epsilon_{t+1} \leq \epsilon_t(1 - \gamma_t) + \frac{2^\kappa L}{\kappa} \gamma_t^\kappa \quad (*_t) \\
? \Rightarrow ? \quad & \epsilon_t \leq \frac{2^{\kappa+1} L}{\kappa(3-\kappa)} \gamma_t^{\kappa-1}, t \geq 2 \quad [\gamma_t = \frac{2}{t+1}] \quad (!_t)
\end{aligned}$$

- By $(*_2)$, we have $\epsilon_2 \leq \frac{2^\kappa L}{\kappa} \Rightarrow \epsilon_2 \leq \frac{2^{\kappa+1} L}{\kappa(3-\kappa)} (2/3)^{\kappa-1}$ due to $1 < \kappa \leq 2 \Rightarrow (!_2)$ holds true.
- Assuming $(!_t)$ true for some $t \geq 2$, we have

$$\begin{aligned}
\epsilon_{t+1} & \leq \frac{2^{\kappa+1} L}{\kappa(3-\kappa)} \gamma_t^{\kappa-1} (1 - \gamma_t) + \frac{2^\kappa L}{\kappa} \gamma_t^\kappa \quad [\text{by } (*_t) \text{ and } (!_t)] \\
& = \frac{2^{\kappa+1} L}{\kappa(3-\kappa)} \left[\gamma_t^{\kappa-1} - \frac{\kappa-1}{2} \gamma_t^\kappa \right] \\
& = \frac{2^{\kappa+1} L}{\kappa(3-\kappa)} 2^{\kappa-1} \left[(t+1)^{1-\kappa} + (1-\kappa)(t+1)^{-\kappa} \right] \\
& \leq \frac{2^{\kappa+1} L}{\kappa(3-\kappa)} 2^{\kappa-1} (t+2)^{1-\kappa} \quad [\text{by convexity of } (t+1)^{1-\kappa}] \\
& = \frac{2^{\kappa+1} L}{\kappa(3-\kappa)} \gamma_{t+1}^{\kappa-1} \Rightarrow (!_{t+1}) \text{ holds true.}
\end{aligned}$$

Thus, $(!_t)$ holds true for all t , Q.E.D.

Conditional Gradient Algorithm for Norm-regularized Smooth Convex Minimization

Source: Harchaoui, Z., Juditsky, A., Nemirovski, A. Conditional Gradient Algorithms for Norm-Regularized Smooth Convex Optimization. *Mathematical Programming* 152:1-2 (2015), 75–112. <https://www2.isye.gatech.edu/~nemirovs/HarchaouiJudNem.pdf>

♣ “As is”, CndG is applicable only to minimizing *smooth* convex functions on *bounded* and closed convex domains.

Question: *How to apply CndG to Composite Minimization problem*

$$\text{Opt} = \min_{x \in \mathbf{K}} \{ \lambda \|x\| + f(x) \}$$

[

•

\mathbf{K} : closed convex cone in Euclidean space E

$\|\cdot\|$: norm on E

$\lambda > 0$: penalty

$f : \mathbf{K} \rightarrow \mathbf{R}$: convex function with Lipschitz continuous gradient:

$$\|\nabla f(x) - \nabla f(y)\|_* \leq L_f \|x - y\|, \quad x, y \in \mathbf{K}$$

]

♠ **Main Assumption:** *We have at our disposal LMO oracle for the intersection of the unit $\|\cdot\|$ -ball with the cone \mathbf{K} . Given on input a linear form $\langle \eta, \cdot \rangle$ on E , the oracle returns*

$$x[\eta] \in \text{Argmin}_x \{ \langle \eta, x \rangle : x \in \mathbf{K}, \|x\| \leq 1 \}$$

Examples:

A. $E = \mathbf{R}^{m \times n}$, $\|\cdot\| = \|\cdot\|_{\text{nuc}}$, $\mathbf{K} = E$

B. $E = \mathbf{S}^n$, $\|\cdot\| = \|\cdot\|_{\text{nuc}}$, $\mathbf{K} = \mathbf{S}_+^n = \{x \in E : x \succeq 0\}$

C. $E = \mathbf{M}_0^{m,n}$, $\|\cdot\| = \text{TV}(\cdot)$, $\mathbf{K} = E$.

♣ We can reformulate the problem of interest as

$$\begin{aligned} \text{Opt} &= \min_{[x;r] \in \mathbf{K}^+} \{ \phi(x, r) := \lambda r + f(x) \} \\ \mathbf{K}^+ &= \{ [x; r] \in E^+ := E \times \mathbf{R} : x \in \mathbf{K}, \|x\| \leq r \} \end{aligned}$$

♠ **Assumption:** *There exists $D_* < \infty$ such that*

$$y := [x; r] \in \mathbf{K}^+ \ \& \ r > D_* \Rightarrow \phi(y) > \phi(0),$$

and we are given a finite upper bound D^+ on D_ .*

Note: *The efficiency estimate for the forthcoming method depends on D_* , and not on D^+ !*

♠ **Algorithm:**

- **Initialization:** Set $y_1 = 0 \in \mathbf{K}^+$
- **Step** $t = 1, 2, \dots$ Given $y_t = [x_t; r_t] \in \mathbf{K}^+$,
 - compute $\nabla f(x_t)$
 - compute $x_t^+ = x[\nabla f(x_t)] \in \text{Argmin}_x \{ \langle \nabla f(x_t), x \rangle : x \in \mathbf{K}, \|x\| \leq 1 \}$
 - set $\Delta_t = \text{Conv} \{ y_t, 0, D^+[x_t^+; 1] \} \subset \mathbf{K}^+$ and find $y_{t+1} \in \mathbf{K}^+ : \phi(y_{t+1}) \leq \min_{y \in \Delta_t} \phi(y)$

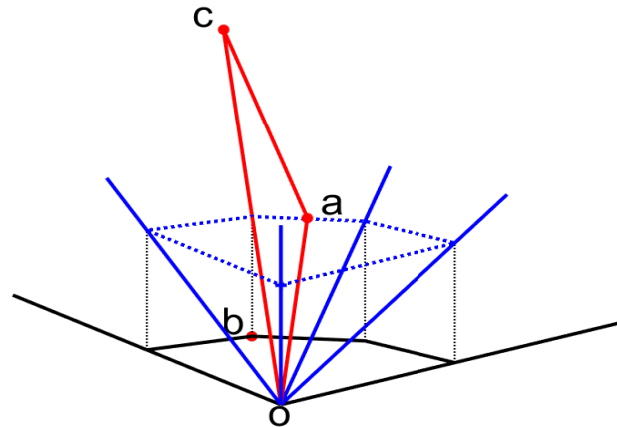
Step t is completed; pass to step $t + 1$.

$$\min_{x \in \mathbf{K}} [\lambda \|x\| + f(x)] \Leftrightarrow \min_{[x;r] \in \mathbf{K}^+} [\phi(x,r) = \lambda r + f(x)]$$

$$[\mathbf{K}^+ = \{[x;r] : x \in \mathbf{K}, r \geq \|x\|\}]$$

♠ **Algorithm:**

- **Initialization:** Set $y_1 = 0 \in \mathbf{K}^+$
- **Step** $t = 1, 2, \dots$ Given $y_t = [x_t; r_t] \in \mathbf{K}^+$,
 - compute $\nabla f(x_t)$
 - compute $x_t^+ = x[\nabla f(x_t)] \in \text{Argmin}_x \{\langle \nabla f(x_t), x \rangle : x \in \mathbf{K}, \|x\| \leq 1\}$
 - set $\Delta_t = \text{Conv} \{y_t, 0, D^+[x_t^+; 1]\} \subset \mathbf{K}^+$ and find $y_{t+1} \in \mathbf{K}^+ : \phi(y_{t+1}) \leq \min_{y \in \Delta_t} \phi(y)$



Geometry of step

- \mathbf{K} : quadrant on the XY plane • black polygon: the set $\{x \in \mathbf{K} : \|x\| \leq 1\}$
- blue polygon: intersection of \mathbf{K}^+ with the hyperplane $r = 1$
- a: current iterate y_t • b: $x_t^+ \in \text{argmin}_{x \in \mathbf{K}, \|x\| \leq 1} \langle \nabla f(y_t), x \rangle$ • c: $D_+ \cdot [x_t^+; 1]$
- $y_{t+1} \in \mathbf{K}^+ : \phi(y_{t+1}) \leq \min_{y \in \Delta_t} \phi(y)$, Δ_t : triangle with vertices o,a,c

Note: One can set $y_{t+1} \in \text{Argmin}_{y \in \Delta_t} \phi(y)$. With this policy, a step requires minimizing ϕ over a 2D triangle Δ_t , which can be done within machine precision in $O(1)$ steps (e.g., by the Ellipsoid method).

$$\text{Opt} = \min_{[x;r] \in \mathbf{K}^+} \{\phi(x, r) := \lambda r + f(x)\}$$

$$\mathbf{K}^+ = \{[x; r] \in E^+ := E \times \mathbf{R} : x \in \mathbf{K}, \|x\| \leq r\}$$

♣ **Theorem:** For the outlined algorithm,

$$\phi(y_t) - \text{Opt} \leq \frac{8L_f D_*^2}{t + 14}, \quad t = 2, 3, \dots$$

♠ **Bundle Implementation:** We can set

$$y_{t+1} \in \text{Argmin}_y \{\phi(y) : y \in \text{Conv}\{0 \cup Y_t\}\} \quad (*)$$

$$Y_t \subset \mathbf{K}^+ : \text{finite set containing } y_t = [x_t; r_t] \text{ and } D^+[x_t^+; 1], \text{ with}$$

$$x_t^+ \in \text{Argmin}_x \{\langle \nabla f(x_t), x \rangle : x \in \mathbf{K}, \|x\| \leq 1\}$$

For example, we can comprise Y_t of $y_t, D^+[x_t^+; 1]$ and several of the previous iterates y_1, \dots, y_{t-1} .

♥ Bundle approach is especially attractive when

$$f(x) = \Psi(Ax + b)$$

for easy to compute Ψ , like $\Psi(u) = \frac{1}{2}u^T u$. Here computing $f, \nabla f$ at a convex (or linear) combination $x = \sum \lambda_i x_i$ of points x_i with already computed Ax_i becomes cheap: $Ax = \sum_i \lambda_i (Ax_i)$.

⇒ the FO oracle for $(*)$ is computationally cheap

$$y_{t+1} \in \operatorname{Argmin}_y \{ \phi(y) : y \in \operatorname{Conv}\{0 \cup Y_t\} \} \quad (*)$$

$Y_t \subset \mathbf{K}^+$: finite set containing $y_t = [x_t; r_t]$ and $D^+[x_t^+; 1]$, with

$$x_t^+ \in \operatorname{Argmin}_x \{ \langle \nabla f(x_t), x \rangle : x \in \mathbf{K}, \|x\| \leq 1 \}$$

• For example, with $f(x) = \frac{1}{2} \|Ax - b\|_2^2$, solving $(*)$ reduces to solving $k_t = \operatorname{Card}(Y_t)$ -dimensional convex quadratic problem

$$\min_{\lambda \in \mathbf{R}^{k_t}} \left\{ \frac{1}{2} \lambda^T Q_t \lambda + 2 q_t^T \lambda : \lambda \geq 0, \sum_j \lambda_j \leq 1 \right\}, \quad (!)$$

$$Q_t = [x_i^T A^T A x_j]_{i,j}$$

where x_j , $1 \leq j \leq k_t$, are the x -components of the points from Y_t .

\Rightarrow Assuming that Y_t is a set of moderate cardinality (say, few tens) obtained from Y_{t-1} by discarding several “old” points and adding the new points $y_t = [x_t; r_t], D^+[x_t^+; 1]$, updating

$$[Q_{t-1}, q_{t-1}] \mapsto [Q_t, q_t]$$

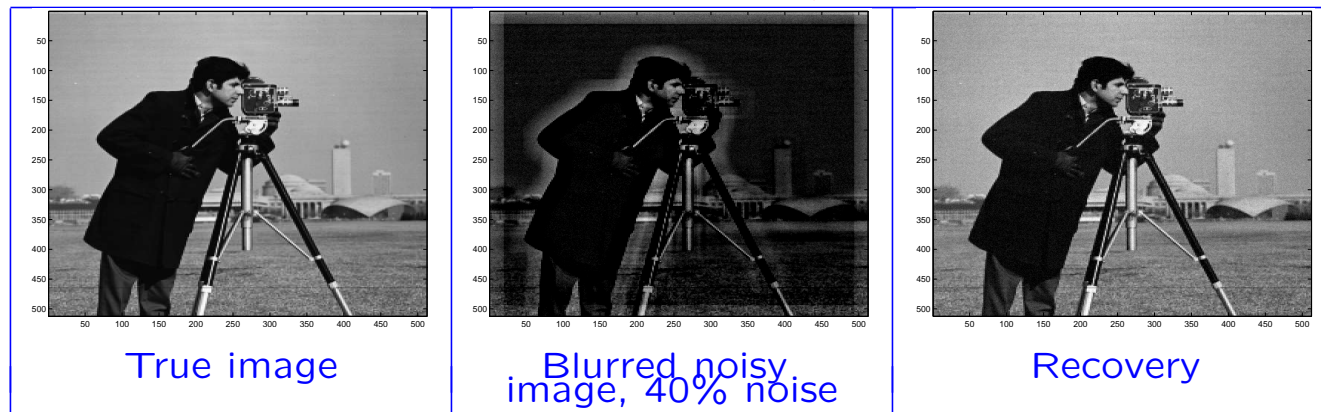
basically reduces to computing matrix-vector products Ax_t and Ax_t^+ . After Q_t, q_t are computed, $(!)$ can be solved “in no time” by an IPM.

Note: Ax_t is computed anyway when computing $\nabla f(x_t)$.

How It Works: TV-based Image Reconstruction



Bundle CndG, 256×256 image (65,536 variables)
Recovery in 13 CndG iterations, CPU time 50.0 sec
Error removal: 98.5%, $\phi(y_{13})/\phi(0) < 4.6e-5$



Bundle CndG, 512×512 image (262,144 variables)
Recovery in 18 CndG iterations, CPU time 370.3 sec
Error removal: 98.2%, $\phi(y_{18})/\phi(0) < 1.3e-4$

Platform: 2×3.40 GHz CPU with 16.0 GB RAM and
64-bit operating system

♠ **Note:** We used 15-element bundle, adding to it at step t the points $y_t = [x_t; r_t], D^+[x_t^+; 1]$ and $[\nabla f(x_t); \text{TV}(\nabla f(x_t))]$ and removing (up to) 3 old points according to “first in — first out.” *Adding $[\nabla f(x_t); \text{TV}(\nabla f(x_t))]$ to the bundle dramatically accelerated the algorithm.*

How It Works: Low Rank Matrix Completion

♠ Problem:

$$\text{Opt} = \min_{x \in \mathbf{R}^{n \times n}} \{0.1\|x\| + \|x - a\|_F^2\}$$

$$\left[\begin{array}{lll} \bullet \|\cdot\|: \text{nuclear norm} & \bullet \|\cdot\|_F: \text{Frobenius norm} & \bullet a = \bar{x} + \xi \\ \text{Rank}(\bar{x}) \approx \sqrt{n}, \|\bar{x}\| \approx \sqrt{2n/\pi}, \|\xi\|_F \approx 0.1\|\bar{x}\|_F \text{ with i.i.d. Gaussian } \xi_{ij} \end{array} \right]$$

- Required relative inaccuracy **0.01**

n	Method	CPU, sec	Iterations	Relative inaccuracy
128	CndG	4.5	42	<1.3e-6
	IPM	2675.0	31	<1.e-10
1024	CndG	44.2	31	<0.008
	IPM	not tested		
4096	CndG	1997.7	87	<0.01
	IPM	not tested		
8192 [†]	CndG	1364.5	36	<0.01
	IPM	not tested		

[†] Rank(\bar{x}) = 32

Platform: 2 × 3.40 GHz CPU with 16.0 GB RAM and 64-bit operating system

Note: CPU time in 8192×8192 example is less than needed to compute just 3 full svd's of a 8192 × 8192 matrix \Rightarrow *The time taken by 36 steps of CndG is less than needed to perform just 3 steps of the simplest proximal algorithm, or just 2 steps of Nesterov's Fast Gradient method for Composite minimization!*

Conditional Gradients for Nonsmooth Convex Minimization

Source: Cox, B., Juditsky, A., Nemirovski, A. Dual subgradient algorithms for large-scale nonsmooth learning problems. *Mathematical Programming Series B* **148:1-2** (2014), 143-180.

<https://www2.isye.gatech.edu/~nemirovs/CoxJudNem.pdf>

♠ **Situation and goal:** Given convex compact domain X represented by Linear Minimization Oracle, we want to solve convex program

$$\text{Opt} = \min_{x \in X} f(x)$$

where f is a Lipschitz continuous convex function.

Difficulty: Since X is given by LMO, it is problematic to use proximal algorithms; and since f can be nonsmooth, Conditional Gradient cannot be applied directly.

Remedy: Use *Fenchel-type representation*

$$f(x) = \max_{y \in Y} [x^T [Ay + a] - \phi(y)]$$

[• Y : convex set • $\phi(\cdot) : Y \rightarrow \mathbf{R}$: convex function]

Note: Fenchel-type representation is a special case of what we called *saddle point representation*

$$f(x) = \max_{y \in Y} \phi(x, y) \quad [\phi : \text{convex-concave}]$$

Note: Whenever $f : \mathbf{R}^n \rightarrow \mathbf{R} \cup \{+\infty\}$ is a proper (i.e., with a nonempty domain) convex lower semicontinuous function, it admits *Fenchel* (a.k.a. *Legendre*) *representation*

$$f(x) = \sup_{y \in \mathbf{R}^n} [x^T y - f_*(y)]$$

$$\left[\begin{array}{l} f_*(y) = \sup_{x \in \mathbf{R}^n} [y^T x - f(x)]: \text{Fenchel dual of } f \\ f_* \text{ is convex proper lower semicontinuous, } [f_*]_* = f \end{array} \right]$$

$$\left[\begin{array}{l} f(x) = \sup_{y \in \mathbb{R}^n} [x^T y - f_*(y)] \\ f_*(y) = \sup_{x \in \mathbb{R}^n} [y^T x - f(x)]: \text{ Fenchel dual of } f \\ f_* \text{ is convex proper lower semicontinuous along with } f, \text{ and } [f_*]_* = f \end{array} \right]$$

Note: Fenchel dual “exists in the nature,” but, aside of a handful of simple cases, is not available in closed form or in the form allowing for a cheap FO oracle.

In contrast, *Fenchel type* representations typically are readily available.

Example A. When $f(x) = \|Bx - b\|$, computing $f_*(y)$ reduces to solving a nontrivial convex problem

$$f_*(y) = \sup_x [y^T x - \|Bx - b\|],$$

while Fenchel-type representation is immediate:

$$f(x) = \max_{y: \|y\|_* \leq 1} y^T (Bx - b) = \max_{y: \|y\|_* \leq 1} [x^T \underbrace{[B^T y]}_{Ay} - \underbrace{b^T y}_{\phi(y)}]$$

Example B. When summing up two convex functions with known Fenchel duals, the Fenchel dual of the sum is given by difficult to compute “inf-convolution”:

$$[f + h]_*(y) = \inf_v [f_*(v) + h_*(y - v)]$$

In contrast, when summing up two convex functions with known Fenchel-type representations, a Fenchel-type representation of the sum is immediate:

$$\begin{aligned} f_i(x) &= \sup_{y_i \in Y_i} [x^T [A_i y_i + a_i] - g_i(y_i)], \quad 1 \leq i \leq m \\ \Rightarrow \sum_i f_i(x) &= \sup_{y=[y_1; \dots; y_m] \in \underbrace{Y_1 \times \dots \times Y_m}_Y} \left[\underbrace{\sum_i x^T [A_i y_i + a_i]}_{x^T [Ay + a]} - \underbrace{\sum_i g_i(y_i)}_{\phi(y)} \right] \end{aligned}$$

$$\text{Opt} = \min_{x \in X} f(x) \quad (P)$$

Assumption: We know Fenchel-type representation of f :

$$f(x) = \max_{y \in Y} [x^T [Ay + a] - \phi(y)]$$

where convex compact set Y admits a computation-friendly proximal setup, and ϕ is a Lipschitz continuous convex function given by First Order oracle.

\Rightarrow Problem of interest (P) is the primal problem associated with the convex-concave saddle point problem

$$\text{Opt} = \min_{x \in X} \max_{y \in Y} [x^T [Ay + a] - \phi(y)] .$$

The dual problem, in minimization form, is

$$[-\text{Opt} =] \min_{y \in Y} \left[g(y) := -\min_{x \in X} x^T [Ay + a] + \phi(y) \right] \quad (D)$$

and LMO for X induces First Order oracle for G : given $y \in Y$ and computing

$$x_y \in \text{Argmin}_{x \in X} x^T [Ay + a],$$

we have

$$\begin{aligned} g(y) &= -x_y^T [Ay + a] + \phi(y) \\ g'(y) &:= -A^T x_y + \phi'(y) \text{ is a subgradient of } g \text{ at } y \end{aligned}$$

\Rightarrow we can solve (D) by proximal-type First Order algorithm!

$$\begin{aligned}\text{Opt} &= \min_{x \in X} \left\{ f(x) = \max_{y \in Y} [x^T [Ay + a] - \phi(y)] \right\} \quad (P) \\ -\text{Opt} &= \min_{y \in Y} \left\{ g(y) = -\min_{x \in X} x^T [Ay + a] + \phi(y) \right\} \quad (D)\end{aligned}$$

Question: *How to recover a good approximate solution to (P) from information accumulated when solving (D)?*

Answer: *Use accuracy certificates!*

Accuracy Certificates

Let Z be a convex compact set, $F(\cdot)$ be a vector field on Z . Consider an N -step algorithm which operates with Z and F by generating sequence of search points $z_i \in Z$, $i \leq N$ along with the sequence $F(z_i)$, $i \leq N$, of the values of F along the search points.

- Collection $\mathcal{F} = \{z_i \in Z, F(z_i)\}_{i=1}^N$ is called the *the execution protocol* of the algorithm
- An *accuracy certificate* for execution protocol \mathcal{F} is an N -dimensional vector λ of nonnegative weights λ_i summing up to 1
- The *resolution* of (\mathcal{F}, λ) on Z is defined as

$$\text{Res}(\mathcal{F}, \lambda|Z) = \max_{z \in Z} \left[\sum_{i=1}^N \lambda_i \langle F(z_i), z_i - z \rangle \right]$$

Observation: Every one of considered so far deterministic proximal First Order algorithms for convex minimization and convex-concave saddle point problems worked with some vector field F on a convex compact set Z and in N steps generated some execution protocol $\mathcal{F} = \{z_i \in Z, F(z_i)\}_{i=1}^N$ and accuracy certificate λ . When specifying approximate solution as

$$z^N = \sum_i \lambda_i z_i,$$

the resolution $\text{Res}(\mathcal{F}, \lambda|Z)$ was an upper bound on inaccuracy of z^N resulting in efficiency estimates we got.

Example: Subgradient/Mirror Descent for convex minimization problem $\min_{z \in Z} f(z)$ works with subgradient vector field $F(z) = f'(z)$ of the objective and ensures that

$$\forall z \in Z : \sum_{i=1}^N \gamma_i \langle F(z_i), z_i - z \rangle \leq \Theta + \sum_{i=1}^N \gamma_i^2 \|F(z_i)\|_*^2$$

[Θ : capacity of X w.r.t. DGF in question]

$$\Rightarrow \text{Res}(\mathcal{F}, \lambda|Z) := \max_{z \in Z} \sum_i \lambda_i \langle F(z_i), z_i - z \rangle \leq \mathcal{R} := \frac{\Theta + \sum_{i=1}^N \gamma_i^2 \|F(z_i)\|_*^2}{\sum_{i=1}^N \gamma_i} \quad (!)$$

$$\left[\lambda_i = \gamma_i / \sum_{j=1}^N \gamma_j \right]$$

Our efficiency estimate for SD/MD was yielded by (!) combined with the relation

$$f(\sum_i \lambda_i z_i) - f(z_*) \leq \sum_i \lambda_i [f(z_i) - f(z_*)] \leq \sum_i \lambda_i \langle F(z_i), z_i - z_* \rangle \leq \text{Res}(\mathcal{F}, \lambda|Z). \quad (!!)$$

where $z_* \in \text{Argmin}_Z f$.

Note:

- SD/MD ensures (!) *independently of what is the origin of the vector field F the method works with*
- (!!) holds independently of where the execution protocol with $F = f'$ and the accuracy certificate come from.

♠ *In retrospect, all we cared about when designing algorithms like SD, MD, or their bundle versions, or Mirror Prox, etc., was generating execution protocol and accuracy certificate with as small as possible guaranteed resolution.*

$$\begin{aligned}\text{Opt} &= \min_{x \in X} \{f(x) = \max_{y \in Y} [x^T[Ay + a] - \phi(y)]\} \quad (P) \\ -\text{Opt} &= \min_{y \in Y} \{g(y) = -\min_{x \in X} x^T[Ay + a] + \phi(y)\} \quad (D)\end{aligned}$$

♠ **Fact:** Assume we are solving (D) by First Order method producing in N steps execution protocol

$$\mathcal{G} = \{y_i \in Y, g'(y_i) = -A^T x_{y_i} + \phi'(y_i)\}_{i=1}^N$$

$$x_{y_i} \in \text{Argmin}_{x \in X} x^T[Ay_i + a]$$

and accuracy certificate λ . Let us set

$$x^N = \sum_{i=1}^N \lambda_i x_{y_i}, \quad y^N = \sum_{i=1}^N \lambda_i y_i.$$

Then x^N is feasible for (P) and solves (P) within accuracy $\text{Res} := \text{Res}(\mathcal{G}, \lambda|Y)$.

Proof of Fact: Let $x \in X$ and $y \in Y$. We have

$$\begin{aligned}
 \text{Res} &\geq \sum_i \lambda_i \langle -A^T x_{y_i} + \phi'(y_i), y_i - y \rangle = \sum_i \lambda_i \langle x_{y_i}, A[y - y_i] \rangle + \underbrace{\sum_i \lambda_i \langle \phi'(y_i), y_i - y \rangle}_{\geq \sum_i \lambda_i \phi(y_i) - \phi(y)} \\
 &\geq \sum_i \lambda_i \langle x_{y_i}, Ay + a \rangle - \sum_i \lambda_i \underbrace{\langle x_{y_i}, Ay_i + a \rangle}_{\leq \langle x, Ay_i + a \rangle} + \underbrace{\sum_i \lambda_i \phi(y_i)}_{\geq \phi(y^N)} - \phi(y) \\
 &\geq \sum_i \lambda_i \langle x_{y_i}, Ay + a \rangle - \sum_i \lambda_i \langle x, Ay + a \rangle + \phi(y^N) - \phi(y) \\
 &= \langle x^N, Ay + a \rangle - \langle x, Ay^N + a \rangle + \phi(y^N) - \phi(y) \\
 &\Rightarrow \langle x^N, Ay + a \rangle - \phi(y) \leq \text{Res} + \langle x, Ay^N + a \rangle - \phi(y^N)
 \end{aligned}$$

The resulting inequality holds true for all $x \in X$ and $y \in Y$, implying that

$$\begin{aligned}
 f(x^N) &= \max_{y \in Y} [\langle x^N, Ay + a \rangle - \phi(y)] \leq \text{Res} + \min_{x \in X} [\langle x, Ay^N + a \rangle - \phi(y^N)] \\
 &\leq \text{Res} + \max_{y \in Y} \min_{x \in X} [\langle x, Ay + a \rangle - \phi(y)] = \text{Res} + \text{Opt}.
 \end{aligned}$$