.

Mini-Course on Convex Programming Algorithms

Arkadi Nemirovski

Arik.Nemirovski@isye.gatech.edu

School of Industrial and Systems Engineering

Georgia Institute of Technology

Atlanta Georgia USA

Rio-De-Janeiro, Brazil

July 2013

Skolkovo, Russia

December 2016

Lecture I.

From Linear to Conic Programming

- Convex Programming: solvable case in Optimization
- Convex Programming in structure-revealing form: Conic Programming
- Calculus of conic programs
- Conic duality
- Illustration: semidefinite relaxations of difficult problems
- Illustration: Lyapunov Stability Analysis and S-Lemma

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 still underlies the majority of real life applications of Optimization, especially large-scale ones.

♣ 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 *polynomial time* solution algorithms.
As applied to an instance of a generic problem, like Linear Programming

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

a polynomial time algorithm solves it to a *whatever high required accuracy* ϵ, *in terms of global optimality*, in a number of arithmetic operations which is *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.

$\Rightarrow$ 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 have a reasonable a priory bound on how long it will take to reach 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*

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

normalized by the restriction

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

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

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

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 $\mathrm{int}B$*, 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

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

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

♣ The low (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. 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.

♣ 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 nonlinear 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
- 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

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

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:

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

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

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

with another *vector* inequality

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

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

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

$\mathbf{K}$: closed, pointed and convex cone in $\mathbb{R}^m$ with a nonempty interior.

Requirements on $\mathbf{K}$ ensure that $\geq_{\mathbf{K}}$ obeys the usual rules for inequalities:

- $\geq_{\mathbf{K}}$ is a *partial order*:

$$\begin{array}{ll} x \geq_{\mathbf{K}} x \, \forall x & \text{[reflexivity]} \\ (x \geq_{\mathbf{K}} y \ \& \ y \geq_{\mathbf{K}} x) \Rightarrow x = y & \text{[antisymmetry]} \\ (x \geq_{\mathbf{K}} y, y \geq_{\mathbf{K}} z) \Rightarrow x \geq_{\mathbf{K}} z & \text{[transitivity]} \end{array}$$

- $\geq_{\mathbf{K}}$ is compatible with linear operations: the validity of $\geq_{\mathbf{K}}$ inequality is preserved when we multiply both sides by the same nonnegative real and add to it another valid $\geq_{\mathbf{K}}$-inequality;

- in a sequence of $\geq_{\mathbf{K}}$-inequalities, one can pass to limits:

$$\begin{array}{c} a_i \geq_{\mathbf{K}} b_i, \, i = 1, 2, ... \ \& \ a_i \to a \ \& b_i \to b \\ \Downarrow \\ a \geq_{\mathbf{K}} b \end{array}$$

- one can define the strict version $>_{\mathbf{K}}$ of $\geq_{\mathbf{K}}$:

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

Arithmetics of $>_{\mathbf{K}}$ and $\geq_{\mathbf{K}}$ inequalities is completely similar to the arithmetics of the usual coordinate-wise $\geq$ and $>$.

♣ LP problem:

$$\min_x \left\{c^T x : Ax - b \geq 0\right\} \Leftrightarrow \min_x \left\{c^T x : Ax - b \in \mathbb{R}^m_+\right\}$$

♣ General Conic problem:

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

- (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.

♣ (??) A general convex cone has no more structure than a general convex function. Why conic reformulation is "structure-revealing"?

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

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

♠ Three Magic Families of cones:

- *$\mathcal{LP}$: Nonnegative orthants* $\mathbb{R}^m_+$ – direct products of m nonnegative rays $\mathbb{R}_+ = \{s \in \mathbb{R} : s \geq 0\}$ giving rise to Linear Programming programs
$$\min_s \left\{c^T x : a_\ell^T x - b_\ell \geq 0, 1 \leq \ell \leq q\right\}.$$

- *$\mathcal{CQP}$: Direct products of Lorentz cones* $\mathbf{L}^p_+ = \{u \in \mathbb{R}^p : u_p \geq \left(\sum_{i=1}^{p-1} u_i^2\right)^{1/2}\}$ 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
$$\min_x \left\{c^T x : \underbrace{\lambda_{\min}(\mathcal{A}^\ell(x)) \geq 0}_{\Leftrightarrow \mathcal{A}^\ell(x) \succeq 0}, 1 \leq \ell \leq q\right\}.$$
where $\mathbf{S}^p$ is the space of $p \times p$ real symmetric matrices, $\mathcal{A}_\ell(x) \in \mathbf{S}^p$ are affine in x and $\lambda_{\min}(S)$ is the minimal eigenvalue of $S \in \mathbf{S}^p$.

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

What can be reduced to $\mathcal{LP}/\mathcal{CQP}/\mathcal{SDP}$?
Calculus of Conic programs

♣ Let $\mathcal{K}$ be a family of regular cones closed w.r.t. taking direct products.

♠ **Definition:** • A *$\mathcal{K}$-representation of a set $X \subset \mathbb{R}^n$* is a representation

$$X = \{x \in \mathbb{R}^n : \exists u \in \mathbb{R}^m : Ax + Bu - b \in \mathbf{K}\} \qquad (*)$$

where $\mathbf{K} \in \mathcal{K}$.

• X is called *$\mathcal{K}$-representable*, if X admits a $\mathcal{K}$-r.

♡ **Note:** *Minimizing a linear objective $c^T x$ over a $\mathcal{K}$-representable set X reduces to a conic program on a cone from $\mathcal{K}$.*

Indeed, given $(*)$, problem $\min\limits_{x \in X} c^T x$ is equivalent to

$$\mathrm{Opt} = \min_{x,u} \left\{ c^T x : Ax + Bu - b \in \mathbf{K} \right\}$$

♠ **Definition:** • *A $\mathcal{K}$-representation of a function $f : \mathbb{R}^n \to \mathbb{R} \cup \{+\infty\}$ is a $\mathcal{K}$-representation of the epigraph of f:*

$$\begin{aligned} \mathrm{Epi}\{f\} &:= \{(x,t) : t \geq f(x)\} \\ &= \{x,t : \exists v : Px + pt + Qv - q \in \mathbf{K}\}, \quad \mathbf{K} \in \mathcal{K} \end{aligned}$$

• f is called *$\mathcal{K}$-representable*, if f admits a $\mathcal{K}$-r.

♡ **Note:** • *A level set of a $\mathcal{K}$-r. function is $\mathcal{K}$-r.:*

$$\begin{array}{rcl}\mathrm{Epi}\{f\} & := & \{(x,t) : t \geq f(x)\} \\ & = & \{x,t : \exists v : Px + pt + Qu - q \in \mathbf{K}\}\end{array}$$
$$\Rightarrow \{x : f(x) \leq c\} = \{x : \exists v : Px + Qu - [q - cp] \in \mathbf{K}\}$$

• *Minimization of a $\mathcal{K}$-r. function f over a $\mathcal{K}$-r. set X reduces to a conic program on a cone from $\mathcal{K}$:*

$$\left.\begin{array}{rcl} x \in X & \Leftrightarrow & \exists u : Ax + Bu - b \in \mathbf{K}_X \\ t \geq f(x) & \Leftrightarrow & \exists v : Px + pt + Qv - q \in \mathbf{K}_f \end{array}\right\} \Rightarrow$$

$$\begin{array}{c} \min_{x \in X} f(x) \\ \Updownarrow \\ \min\limits_{t,x,u,v} \left\{ t : [Ax + Bu - b; Px + pt + Qv - q] \in \underbrace{\mathbf{K}_X \times \mathbf{K}_f}_{\in \mathcal{K}} \right\} \end{array}$$

♣ Investigating "expressive abilities" of generic Magic conic problems reduces to answering the question

What are $\mathcal{LP}/\mathcal{CQP}/\mathcal{SDP}$-r. functions/sets?

♠ **"Built-in" restriction is Convexity:** *A $\mathcal{K}$-representable set/function must be convex.*

♠ **Good news:** *Convexity, essentially, is the only restriction: for all practical purposes, all convex sets/functions arising in applications are $\mathcal{SDP}$-r. Quite rich families of convex functions/sets are $\mathcal{LP}/\mathcal{CQP}$-r.*

♡ **Note:** Nonnegative orthants are direct products of (1-dimensional) Lorentz cones, and Lorentz cones are intersections of semidefinite cones and properly selected linear subspaces $\Rightarrow \mathcal{LP} \subset \mathcal{CQP} \subset \mathcal{SDP}$.

♣ *Let $\mathcal{K}$ be a family of regular cones closed w.r.t. taking direct products and passing from a cone $\mathbf{K}$ to its dual cone*

$$\mathbf{K}_* = \{\lambda : \langle \lambda, \xi \rangle \geq 0 \ \forall \xi \in \mathbf{K}\}$$

Note: $\mathbf{K}_*$ is regular cone provided $\mathbf{K}$ is so, and

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

♠ **Fact:** *$\mathcal{K}$-representable sets/functions admit fully algorithmic calculus: all basic convexity-preserving operations with functions/sets, as applied to $\mathcal{K}$-r. operands, produce $\mathcal{K}$-r. results, and the resulting $\mathcal{K}$-r.'s are readily given by $\mathcal{K}$-r.'s of the operands.*
"Calculus rules" are independent of what $\mathcal{K}$ is.
$\Rightarrow$ *Starting with "raw materials"* (characteristic for $\mathcal{K}$ elementary $\mathcal{K}$-r. sets/functions) *and applying calculus rules, we can recognize $\mathcal{K}$-representability and get explicit $\mathcal{K}$-r.'s of sets/functions of interest.*

♣ **Basics of "calculus of $\mathcal{K}$-representability":**

♠ **[Sets:]** If $X_1, ..., X_k$ are $\mathcal{K}$-r. sets, so are their

- *intersections,*
- *direct products,*
- *images under affine mappings,*
- *inverse images under affine mappings.*

♠ **[Functions:]** If $f_1, ..., f_k$ are $\mathcal{K}$-r. functions, so are their

- *linear combinations with nonnegative coefficients,*
- *superpositions with affine mappings.*

Moreover, *if $F, f_1, ..., f_k$ are $\mathcal{K}$-r. functions, so is the superposition $F(f_1(x), ..., f_k(x))$ provided that F is monotonically nondecreasing in its arguments.*

♠ More advanced convexity-preserving operations preserve $\mathcal{K}$-representability under (pretty mild!) regularity conditions. This includes

- **for sets:** taking *conic hulls* and *convex hulls of (finite) unions* and passing from a set to its *recessive cone*, or *polar*, or *support function*
- **for functions:** *partial minimization*, *projective transformation*, and *taking Fenchel dual*.

♠ **Note:** *Calculus rules are simple and algorithmic*
$\Rightarrow$ *Calculus can be run on a compiler* [used in `cvx`].

Illustration

$$\min c^T x + d^T y$$

$$y \geq 0,\ Ax + By \leq b$$

$$2y_1^{-\frac{7}{2}} y_2^{-3} y_3^{\frac{-1}{5}} + 3y_2^{-\frac{3}{2}} y_4^{-\frac{2}{3}} \leq e^T x + 4y_1^{\frac{1}{5}} y_2^{\frac{2}{5}} y_3^{\frac{2}{5}} + 5y_3^{\frac{1}{3}} y_4^{\frac{2}{5}}$$

$$\left[\begin{array}{c|c|c|c} x_1 - x_2 & x_3 + x_2 & & \\ \hline x_3 + x_2 & x_2 - x_4 & x_5 - 6 & \\ \hline & x_5 - 6 & x_6 + x_7 & -x_8 \\ \hline & & -x_8 & x_5 \end{array}\right] \succeq 0$$

$$\left[\begin{array}{ccccc} x_1 & x_2 & x_3 & x_4 & x_5 \\ x_2 & x_6 & x_7 & x_8 & x_9 \\ x_3 & x_7 & x_{10} & x_{11} & x_{12} \\ x_4 & x_8 & x_{11} & x_{13} & x_{14} \\ x_5 & x_9 & x_{12} & x_{14} & x_{15} \end{array}\right] \succeq 0$$

$$\text{Det}\left(\left[\begin{array}{ccccc} x_1 & x_2 & x_3 & x_4 & x_5 \\ x_2 & x_6 & x_7 & x_8 & x_9 \\ x_3 & x_7 & x_{10} & x_{11} & x_{12} \\ x_4 & x_8 & x_{11} & x_{13} & x_{14} \\ x_5 & x_9 & x_{12} & x_{14} & x_{15} \end{array}\right]\right) \geq 1$$

Sum of 2 largest singular values of $\left[\begin{array}{ccc} x_1 & x_2 & x_3 \\ x_4 & x_5 & x_6 \\ x_7 & x_8 & x_9 \\ x_{10} & x_{11} & x_{12} \\ x_{13} & x_{14} & x_{15} \end{array}\right]$ is ≤ 6

$$1 - \sum_{i=1}^{6} [x_i - x_{i+1}] s^i \leq 0,\ \tfrac{3}{2} \leq s \leq 6$$

$$\sum_{i=1}^{4} x_{2i} \cos(i\phi) - \sum_{i=1}^{4} x_i \sin(i\phi) \leq 1,\ \tfrac{\pi}{3} \leq \phi \leq \tfrac{\pi}{2}$$

- the blue part of the problem is in $\mathcal{LP}$
- the blue-magenta part of the problem is in $\mathcal{CQP}$ and can be approximated, *in a polynomial time fashion*, by $\mathcal{LP}$
- the entire problem is in $\mathcal{SDP}$

and the reductions to $\mathcal{LP}/\mathcal{CQP}/\mathcal{SDP}$ are "fully algorithmic."

Conic Duality

♣ Conic Programming admits nice Duality Theory completely similar to LP Duality.

Primal problem:

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

$\Leftrightarrow$ [passing to primal slack $\xi = Ax - b$]

$$\min_\xi \left\{ e^T \xi : \xi \in [\mathcal{L} - b] \cap \mathbf{K} \right\}$$

$$[\mathcal{L} = \mathrm{Im} A,\ A^T e = c,\ \mathrm{Ker} A = \{0\}]$$

Dual problem:

$$\max_y \left\{ b^T y : y \in [\mathcal{L}^\perp + e] \cap \mathbf{K}_* \right\}$$

$$\Leftrightarrow \quad \max_y \left\{ b^T y : A^T y = c, y \geq_{\mathbf{K}_*} 0 \right\}$$

[$\mathbf{K}_*$: cone dual to $\mathbf{K}$]

Thus,

- the dual problem is conic along with primal
- the duality is completely symmetric

Note: Cones from Magic Families are self-dual, so that the dual of a Linear/Conic Quadratic/Semidefinite program is of exactly the same type.

Derivation of the Dual Problem

♣ **Primal problem:**

$$\mathrm{Opt}(P) = \min_x \left\{ c^T x : \begin{array}{l} A_i x - b_i \in \mathbf{K}^i, i \le m \\ Rx = r \end{array} \right\} \quad (P)$$

♠ **Goal:** find a systematic way to bound $\mathrm{Opt}(P)$ *from below*.

♠ **Simple observation:** *When $y_i \in \mathbf{K}^i_*$, the scalar inequality $y_i^T A_i x \ge y_i^T b_i$ is a consequence of the constraint $A_i x - b_i \in \mathbf{K}^i$. If y is a vector of the same dimension as r, the scalar inequality $y^T Rx \ge y^T r$ is a consequence of the constraint $Rx = r$.*

$\Rightarrow$ Whenever $y_i \in \mathbf{K}^i_*$ for all i and y is a vector of the same dimension as r, the scalar linear inequality

$$[\textstyle\sum_i A_i^T y_i + R^T y]^T x \ge \sum_i b_i^T y_i + r^T y$$

is a consequence of the constraints in (P)

$\Rightarrow$ Whenever $y_i \in \mathbf{K}^i_*$ for all i and y is a vector of the same dimension as r such that

$$\textstyle\sum_i A_i^T y_i + R^T y = c,$$

the quantity $\sum_i b_i^T y_i + r^T y$ is a lower bound on $\mathrm{Opt}(P)$.

• The *Dual problem*

$$\mathrm{Opt}(D) = \max_{y_i, y} \left\{ \sum_i b_i^T y_i + r^T y : \begin{array}{l} y_i \in \mathbf{K}^i_*, i \le m \\ \sum_i A_i^T y_i + R^T y = c \end{array} \right\} \quad (D)$$

is just the problem of maximizing this lower bound on $\mathrm{Opt}(P)$.

♣ **Definition:** A conic problem

$$\min_x \left\{ c^T x : \begin{array}{l} A_i x - b_i \in \mathbf{K}^i,\ i \le m \\ Ax \le b \end{array} \right\} \qquad (C)$$

is called *strictly feasible*, if there exists a feasible solution $\bar{x}$ where all the constraints are satisfied *strictly*: $A_i\bar{x} - b_i \in \mathrm{int}\mathbf{K}^i$ for all i, $A\bar{x} < b$, and is called *essentially strictly feasible*, if there exists a *feasible* solution $\bar{x}$ where all *non-polyhedral* constraints are satisfied strictly: $A\bar{x} \le b$ and $A_i\bar{x} - b_i \in \mathrm{int}\mathbf{K}^i$ for all $i \le m$.

♣ **Conic Programming Duality Theorem.** *Consider a conic problem*

$$\mathrm{Opt}(P) = \min_x \left\{ c^T x : \begin{array}{l} A_i x - b_i \in \mathbf{K}^i, i \le m \\ Rx = r \end{array} \right\} \quad (P)$$

along with its dual

$$\mathrm{Opt}(D) = \max_{y_i, y} \left\{ \sum_i b_i^T y_i + r^T y : \begin{array}{l} y_i \in \mathbf{K}^i_*, i \le m \\ \sum\limits_i A_i^T y_i + R^T y = c \end{array} \right\} \quad (D)$$

Then:

♠ [Symmetry] *Duality is symmetric: the dual problem is conic, and its dual is (equivalent to) the primal problem;*

♠ [Weak duality] *One has* $\mathrm{Opt}(D) \le \mathrm{Opt}(P)$;

♠ [Strong duality] *Let one of the problems be essentially strictly feasible and bounded. Then the other problem is solvable, and*

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

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

$$\begin{array}{|c|}\hline \min_x \left\{c^T x : Ax - b \in \mathbf{K}\right\} \qquad (P) \\ \Leftrightarrow \quad \min_\xi \left\{e^T \xi : \xi \in [\mathcal{L} - b] \cap \mathbf{K}\right\} \\ \hline \end{array}$$

$$\Updownarrow$$

$$\begin{array}{|c|}\hline \max_y \left\{b^T y : y \in [\mathcal{L}^\perp + e] \cap \mathbf{K}_*\right\} \\ \Leftrightarrow \quad \max_y \left\{b^T y : A^T y = c, y \geq_{\mathbf{K}_*} 0\right\} \qquad (D) \\ \left[\begin{array}{c} \mathcal{L} = \mathrm{Im} A, \ A^T e = c, \\ \mathbf{K}_* = \{y : y^T \xi \geq 0 \ \forall \xi \in \mathbf{K}\} \end{array}\right] \\ \hline \end{array}$$

Conic Programming Optimality Conditions:

Let both (P) and (D) be essentially strictly feasible. Then a pair (x, y) of primal and dual feasible solutions is comprised of optimal solutions to the respective problems if and only if

- [Zero Duality Gap]

$$\mathrm{DualityGap}(x, y) := c^T x - b^T y = 0$$

[Indeed,

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

]

and if and only if

- [Complementary Slackness]

$$[Ax - b]^T y = 0$$

[Indeed,

$$\begin{array}{rcl} [Ax - b]^T y & = & (A^T y)^T x - b^T y = c^T x - b^T y \\ & = & \mathrm{DualityGap}(x, y) \end{array}$$

]

$$\begin{array}{|rc|l|}\hline
 & \min\limits_x \left\{c^T x : Ax - b \in \mathbf{K}\right\} & (P)\\
\Leftrightarrow & \min\limits_\xi \left\{e^T \xi : \xi \in [\mathcal{L} - b] \cap \mathbf{K}\right\} & \\
\hline
\end{array}$$

$$\Updownarrow$$

$$\begin{array}{|rc|l|}\hline
 & \max\limits_y \left\{b^T y : y \in [\mathcal{L}^\perp + e] \cap \mathbf{K}_*\right\} & \\
\Leftrightarrow & \max\limits_y \left\{b^T y : A^T y = c, y \geq_{\mathbf{K}_*} 0\right\} & (D)\\
 & \left[\begin{array}{c} \mathcal{L} = \mathrm{Im} A,\ A^T e = c, \\ \mathbf{K}_* = \{y : y^T \xi \geq 0\ \forall \xi \in \mathbf{K}\}\end{array}\right] & \\
\hline
\end{array}$$

♣ Conic Duality, same as the LP one, is

- *fully algorithmic:* to write down the dual, given the primal, is a purely mechanical process
- *fully symmetric:* the dual problem "remembers" the primal one

♡ Cf. Lagrange Duality:

$$\min_x \{f(x) : g_i(x) \le 0,\ i = 1, ..., m\} \quad (P)$$

$$\Downarrow$$

$$\max_{y \ge 0} \underline{L}(y) \quad (D)$$

$$\left[\underline{L}(y) = \min_x \left\{ f(x) + \sum_i y_i g_i(x) \right\}\right]$$

- Dual "exists in the nature", but is given implicitly; its objective, typically, is not available in a closed form
- Duality is asymmetric: given $\underline{L}(\cdot)$, we, typically, cannot recover f and g_i...

♣ Conic Duality in the case of Magic cones:

- powerful tool to process problem, to some extent, "on paper", which in many cases provides extremely valuable insight and/or allows to end up with a problem much better suited for numerical processing
- is heavily exploited by efficient polynomial time algorithms for Magic conic problems

Illustration: Semidefinite Relaxation

♣ Consider a quadratically constrained quadratic program

$$\mathrm{Opt} = \min_{x\in\mathbb{R}^n}\{f_0(x) : f_i(x) \le 0, 1 \le i \le m\}$$
$$\left[f_i(x) = x^T A_i x + 2b_i^T x + c_i,\ 0 \le i \le m\right] \qquad (QP)$$

Note: (QP) is "as difficult as a problem can be:" e.g., the Boolean constraints on variables: $x_i \in \{0,1\}$ can be modeled as quadratic equalities $x_i^2 - x_i = 0$ and thus can be modeled as pairs of simple quadratic inequalities.

♠ **Question:** *How to lower-bound* Opt*?*

$$\mathrm{Opt} = \min_{x\in\mathbb{R}^n}\{f_0(x) : f_i(x) \le 0, 1 \le i \le m\} \quad (QP)$$
$$\left[f_i(x) = x^T A_i x + 2b_i^T x + c_i,\ 0 \le i \le m\right]$$

How to lower-bound Opt*?*

♠ **Answer, I: Semidefinite Relaxation.** Associate with x the symmetric matrix

$$X[x] = [x;1][x;1]^T = \begin{bmatrix} xx^T & x \\ x^T & 1 \end{bmatrix}$$

and rewrite (QP) equivalently as

$$\mathrm{Opt} = \min_X \left\{ \mathrm{Tr}(Q_0X) : \begin{array}{l} \mathrm{Tr}(Q_iX) \le 0, 1 \le i \le m \\ X = X[x] \text{ for some } x \end{array} \right\} \quad (QP')$$
$$\left[Q_i = \begin{bmatrix} A_i & b_i \\ b_i^T & c_i \end{bmatrix}\right]$$

(QP') has just *linear* in X objective and constraints. The "domain restriction"

$$\text{"}X = X[x] \text{ for some } x\text{"}$$

says that

- $X \in \mathbb{R}^{(n+1)\times(n+1)}$ *is symmetric positive semidefinite and* $X_{n+1,n+1} = 1$ (nice convex constraints)
- X *is of rank 1* (highly nonconvex constraint)

Removing the "troublemaking" rank restriction, we end up with *semidefinite relaxation* of (QP) – the problem

$$\mathrm{SDP} = \min_X \left\{ \mathrm{Tr}(Q_0X) : \begin{array}{c} \mathrm{Tr}(Q_iX) \le 0, 1 \le i \le M \\ X \succeq 0, X_{n+1,n+1} = 1 \end{array} \right\}$$

$$\mathrm{Opt} = \min_{x\in\mathbb{R}^n}\{f_0(x) : f_i(x) \le 0, 1 \le i \le m\} \qquad (QP)$$
$$\left[f_i(x) = x^T A_i x + 2b_i^T x + c_i,\ 0 \le i \le m\right]$$

$$\Updownarrow$$

$$\mathrm{Opt} = \min_X \left\{ \mathrm{Tr}(Q_0 X) : \begin{array}{c} \mathrm{Tr}(Q_i X) \le 0, 1 \le i \le m \\ X = \left[\begin{array}{cc} xx^T & x \\ x^T & 1 \end{array}\right] \text{ for some } x \end{array} \right\} \qquad (QP')$$
$$\left[Q_i = \left[\begin{array}{cc} A_i & b_i \\ b_i^T & c_i \end{array}\right]\right]$$

$$\Downarrow$$

$$\mathrm{SDP} = \min_X \left\{ \mathrm{Tr}(Q_0 X) : \begin{array}{c} \mathrm{Tr}(Q_i X) \le 0, 1 \le i \le M \\ X \succeq 0, X_{n+1,n+1} = 1 \end{array} \right\} \qquad (SDP)$$

♠ **Probabilistic Interpretation of** (SDP)**:**

Assume that instead of solving (QP) in deterministic variables x, we are solving the problem in *random vectors* ξ and want to minimize the *expected value* of the objective under the restriction that *the constraints are satisfied at average.*

Since f_i are quadratic, the expectations of the objective and the constraints are affine functions of the *moment matrix* $X = \mathbf{E}\left\{\left[\begin{array}{cc} \xi\xi^T & \xi \\ \xi^T & 1 \end{array}\right]\right\}$ which can be an arbitrary symmetric positive semidefinite matrix X with $X_{n+1,n+1} = 1$. *It is immediately seen that the "randomized" version of* (QP) *is exactly* (SDP).

♣ With outlined interpretation, *an optimal solution to (SDP) gives rise to* (various) *randomized solutions to the problem of interest*.

In good cases, *we can extract from these randomized solutions feasible solutions to the problem of interest with reasonable approximation guarantees in terms of optimality.*

We can, e.g.,

— use X_* to generate a sample $\xi^1, ..., \xi^N$ of, say, $N = 100$ random solutions to (QP),

— "correct" ξ^t to get *feasible* solutions x^t to (QP).

The approach works when the correction is easy, e.g., when at some known point $\bar{x}$ the constraints of (QP) are satisfied *strictly*. Here we can take as x^t the closest to ξ^t *feasible* solution from the segment $[\bar{x}, \xi^t]$.

— select from the resulting N feasible solutions x^t to (QP) the best in terms of the objective.

♡ When applicable, the outlined approach can be combined with *local improvement* – N runs of any traditional algorithm for nonlinear optimization as applied to (QP), $x^1, ..., x^N$ being the starting points of the runs.

♣ **Example: Quadratic Maximization over the box**

$$\text{Opt} = \max_x\{x^T Lx : x_i^2 \le 1,\ 1 \le i \le n\} \qquad (QP)$$

$$\Rightarrow \text{SDP} = \max_X\{\text{Tr}(XL) : X \succeq 0, X_{ii} \le 1\ \forall i\} \quad (SDP)$$

Note: When $L \succeq 0$ or L has zero diagonal, Opt and SDP remain intact when the inequality constraints are replaced with their equality versions.

♠ **MAXCUT:** The combinatorial problem *"given n-node graph with arcs assigned nonnegative weights $a_{ij} = a_{ij}$, $1 \le i, j \le n$, split the nodes into two non-overlapping subsets to maximize the total weight of the arcs linking nodes from different subsets"* is equivalent to (QP) with $L_{ij} = \begin{cases} \sum_k a_{ik}, & j = i \\ -a_{ij} & , j \ne i \end{cases}$

♠ **Theorem of Goemans and Williamson** '94:

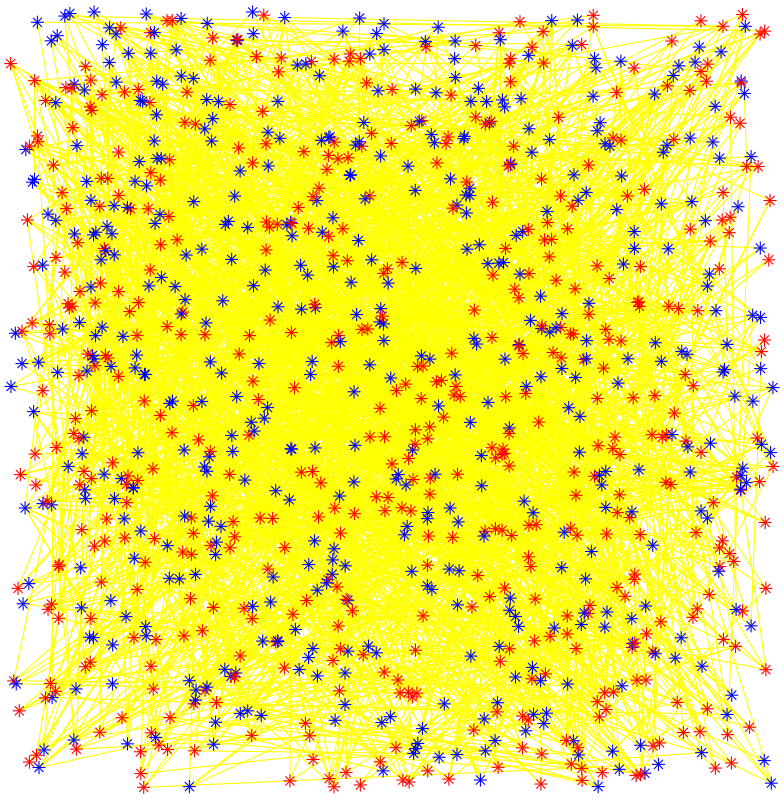
$$\text{Opt} \le \text{SDP} \le 1.1383 \cdot \text{Opt} \qquad (!)$$

Note: To approximate Opt within 4% is NP-hard...

Sketch of the proof of (!): treat an optimal solution X_* of (SDP) as the covariance matrix of zero mean Gaussian random vector ξ and look at

$$\mathbf{E}\{\text{sign}[\xi]^T L \text{sign}[\xi]\}.$$

Illustration: MAXCUT, 1024 nodes, 2614 arcs.



Suboptimal cut, weight $\geq 0.9196 \cdot \mathrm{SDP} \geq 0.9196 \cdot \mathrm{Opt}$

[Slightly better than Goemans-Williamson guarantee:
weight $\geq 0.8785 \cdot \mathrm{SDP} \geq 0.8785 \cdot \mathrm{Opt}$]

$$\text{Opt} = \max_x\{x^T Lx : x_i^2 \le 1,\ 1 \le i \le n\} \qquad (QP)$$
$$\Rightarrow \text{SDP} = \max_X\{\text{Tr}(XL) : X \succeq 0, X_{ii} \le 1\ \forall i\} \quad (SDP)$$

♠ **Nesterov's $\pi/2$ Theorem.** Matrix L arising in MAXCUT is $\succeq 0$ (and possesses additional properties). What can be said about (SDP) under the only restriction $L \succeq 0$?

Answer [Nesterov'98]: $\text{Opt} \le \text{SDP} \le \frac{\pi}{2} \cdot \text{Opt}$.

Sketch of the proof: similar to Goemans-Williamson.

Illustration: L: randomly built positive semidefinite 1024×1024 matrix. Relaxation combined with local improvement yields a feasible solution $\bar{x}$ with

$$\bar{x}^T L \bar{x} \ge 0.7867 \cdot \text{SDP} \ge 0.7867 \cdot \text{Opt}$$

$$\mathrm{Opt} = \max_x\{x^TLx : x_i^2 \le 1,\ 1 \le i \le n\} \qquad (QP)$$
$$\Rightarrow \mathrm{SDP} = \max_X\{\mathrm{Tr}(XL) : X \succeq 0, X_{ii} \le 1\ \forall i\} \quad (SDP)$$

♠ **The case of indefinite L:** When L is an arbitrary symmetric matrix, one has

$$\mathrm{Opt} \le \mathrm{SDP} \le O(1)\ln(n)\mathrm{Opt}.$$

This is a particular case of the following result [Nem.,Roos,Terlaky '98]: *The SDP relaxation*

$$\mathrm{SDP} = \max_X\{\mathrm{Tr}(XL) : \mathrm{Tr}(XQ_i) \le 1, i \le m\}$$

of the problem

$$\mathrm{Opt} = \max_x\left\{x^TLx : x^TQ_ix \le 1, i \le m\right\} \qquad (P)$$
$$[Q_i \succeq 0\ \forall i, \textstyle\sum_i Q_i \succ 0]$$

satisfies $\mathrm{Opt} \le \mathrm{SDP} \le O(1)\ln(m)\mathrm{Opt}$.

Illustration, A: Problem (QP) with randomly selected indefinite 1024×1024 matrix L. Relaxation combined with local improvement yields a feasible solution $\bar{x}$ with

$$\bar{x}^TL\bar{x} \ge 0.7649 \cdot \mathrm{SDP} \ge 0.7649 \cdot \mathrm{Opt}$$

Illustration, B: Problem (P) with randomly selected indefinite 1024×1024 matrix L and 64 randomly selected positive semidefinite matrices Q_i of rank 64. Relaxation yields a feasible solution $\bar{x}$ with

$$\bar{x}^TL\bar{x} \ge 0.9969 \cdot \mathrm{SDP} \ge 0.9969 \cdot \mathrm{Opt}$$

$$\boxed{\begin{array}{c}\mathrm{Opt} = \min\limits_{x\in\mathbb{R}^n}\{f_0(x) : f_i(x) \le 0, 1 \le i \le m\} \\ \left[f_i(x) = x^T A_i x + 2b_i^T x + c_i,\ 0 \le i \le m\right]\end{array}} \quad (QP)$$

$$\Downarrow$$

$$\boxed{\mathrm{SDP} = \min_X \left\{\mathrm{Tr}(Q_0 X) : \begin{array}{c}\mathrm{Tr}(Q_i X) \le 0, 1 \le i \le M \\ X \succeq 0, X_{n+1,n+1} = 1\end{array}\right\} \quad (SDP)}$$

$$\left[Q_i = \begin{bmatrix} A_i & b_i \\ b_i^T & c_i \end{bmatrix}\right]$$

♣ **Dual of** (SDP)**:** Denoting $y_i \ge 0$ the Lagrange multipliers for the scalar inequality constraints $-\mathrm{Tr}(Q_i X) \ge 0$, $1 \le i \le m$, by $Y \succeq 0$ the Lagrange multiplier for the constraint $X \succeq 0$, and by z the multiplier for the equality constraint $X_{n+1,n+1} = 1$, the aggregation of constraints yields the inequality

$$\mathrm{Tr}\left(\left[Y - \sum_{i=1}^m y_i Q_i + \left[\begin{array}{c|c} & \\ \hline & z \end{array}\right]\right] X\right) \ge z$$

To yield a lower bound on SDP, the left hand side of this inequality should be $\mathrm{Tr}(Q_0 X)$ *identically in* X, so that the dual problem is

$$\max_{y_i, Y, z}\left\{z : \begin{array}{c} Y \succeq 0, y_i \ge 0, 1 \le i \le m \\ Y - \sum_{i=1}^m y_i Q_i + \left[\begin{array}{c|c} & \\ \hline & z \end{array}\right] = Q_0 \end{array}\right\}$$

or, equivalently

$$\max_{y_1,\ldots,y_m,z}\left\{z : \begin{array}{c}\left[\begin{array}{c|c} A_0 + \sum_{i=1}^m y_i A_i & b_0 + \sum_{i=1}^m y_i b_i \\ \hline b_0^T + \sum_{i=1}^m y_i b_i^T & c_0 + \sum_{i=1}^m c_i y_i - z \end{array}\right] \succeq 0 \\ y_i \ge 0, 1 \le i \le m\end{array}\right\}$$

$$\mathrm{SDP} = \min_{X} \left\{ \mathrm{Tr}(Q_0 X) : \begin{array}{c} \mathrm{Tr}(Q_i X) \le 0, 1 \le i \le M \\ X \succeq 0, X_{n+1,n+1} = 1 \end{array} \right\} \qquad (P)$$

$$\mathrm{SDP} = \max_{y_i, z} \left\{ z : \left[\begin{array}{c|c} A_0 + \sum_{i=1}^m y_i A_i & b_0 + \sum_{i=1}^m y_i b_i \\ \hline b_0^T + \sum_{i=1}^m y_i b_i^T & c_0 + \sum_{i=1}^m c_i y_i - z \end{array} \right] \succeq 0 \right\} \qquad (D)$$

$$y_i \ge 0, 1 \le i \le m$$

Note: Our primal problem (SDP) has a "massive" matrix variable X ($(n+1)(n+2)/2$ scalar variables) and "large" semidefinite constraint $X \succeq 0$. The dual problem has equally large semidefinite constraint, but just $(m+1)$ variables.

$\Rightarrow$ *For some solution algorithms, the dual problem is better suited than the primal one!*

$$\boxed{\begin{array}{c}\mathrm{Opt} = \min\limits_{x\in\mathbb{R}^n}\{f_0(x) : f_i(x) \le 0, 1 \le i \le m\} \\ \left[f_i(x) = x^T A_i x + 2b_i^T x + c_i,\ 0 \le i \le m\right]\end{array}} \quad (QP)$$

$$\Downarrow$$

$$\boxed{\mathrm{SDP} = \max\limits_{y_i,z}\left\{z : \left[\begin{array}{c|c} A_0 + \sum_{i=1}^m y_i A_i & b_0 + \sum_{i=1}^m y_i b_i \\ \hline b_0^T + \sum_{i=1}^m y_i b_i^T & c_0 + \sum_{i=1}^m c_i y_i - z \end{array}\right] \succeq 0,\ y_i \ge 0, 1 \le i \le m\right\} (D)}$$

♠ (D) can be derived independently by *Lagrange relaxation* of (QP). Specifically, the Lagrange function

$$\mathcal{L}(x,y) = f_0(x) + \sum_{i=1}^m y_i f_i(x)$$

of (QP). *where the Lagrange multipliers* y_i *are restricted to be nonnegative*, underestimates $f_0(x)$ on the feasible set of (QP), whence

$$y \ge 0 \Rightarrow \mathrm{Opt} \ge \underline{\mathcal{L}}(y) := \inf_x\{f_0(x) + \sum_{i=1}^m y_i f_i(x)\}$$

It is immediately seen that (D) *is the problem*

$$\mathrm{SDP} = \max_{y_1,\ldots,y_m\ge 0} \underline{\mathcal{L}}(y).$$

Illustration: Lyapunov Stability Analysis

♣ Consider an *uncertain* time varying linear dynamical system

$$\frac{d}{dt}x(t) = A(t)x(t) \qquad \text{(ULS)}$$

• $x(t) \in \mathbb{R}^n$: state at time t,

• $A(t) \in \mathbb{R}^{n\times n}$: known to take all values in a given *uncertainty set* $\mathcal{U} \subset \mathbb{R}^{n\times n}$.

♠ (ULS) is called *stable*, if all trajectories of the system converge to 0 as $t \to \infty$:

$$A(t) \in \mathcal{U}\ \forall t \geq 0,\ \tfrac{d}{dt}x(t) = A(t)x(t) \Rightarrow \lim_{t\to\infty} x(t) = 0.$$

♣ **Question:** *How to certify stability?*

♠ Standard *sufficient* stability condition is *the existence of Lyapunov Stability Certificate – a matrix $X \succ 0$ such that the function $L(x) = x^T Xx$ for some $\alpha > 0$ satisfies*

$$\tfrac{d}{dt}L(x(t)) \leq -\alpha L(x(t)) \text{ for all trajectories}$$

and thus goes to 0 exponentially fast along the trajectories:

$$\begin{aligned}
&\tfrac{d}{dt}L(x(t)) \leq -\alpha L(x(t)) \Rightarrow \tfrac{d}{dt}\left[\exp\{\alpha t\}L(x(t))\right] \leq 0\\
\Rightarrow\ &\exp\{\alpha t\}L(x(t)) \leq L(x(0)),\ t \geq 0\\
\Rightarrow\ &L(x(t)) \leq \exp\{-\alpha t\}L(x(0))\\
\Rightarrow\ &\|x(t)\|_2^2 \leq \frac{\lambda_{\max}(X)}{\lambda_{\min}(X)}\exp\{-\alpha t\}\|x(0)\|_2^2
\end{aligned}$$

• For a *time-invariant* system, this condition is necessary and sufficient for stability.

♠ **Question:** *When $\alpha > 0$ is such that $\frac{d}{dt}L(x(t)) \leq -\alpha L(x(t))$ for all trajectories $x(t)$ satisfying $\frac{d}{dt}x(t) = A(t)x(t)$ with $A(t) \in \mathcal{U}$ for all t ?*

♡ **Answer:** We should have

$$\begin{array}{rcl}\frac{d}{dt}\left(x^T(t)Xx(t)\right) & = & (\frac{d}{dt}x(t))^T Xx(t) + x^T(t)X\frac{d}{dt}x(t) \\ & = & x^T(t)A^T(t)Xx(t) + x^T(t)XAx(t) \\ & = & x^T(t)\left[A^T(t)X + XA(t)\right]x(t) \\ & \leq & -\alpha x^T(t)Xx(t)\end{array}$$

Thus,

$\frac{d}{dt}L(x(t)) \leq -\alpha L(x(t))$ for all trajectories

$\Leftrightarrow$ $x^T(t)\left[A^T(t)X + XA(t)\right]x(t) \leq -\alpha x^T(t)Xx(t)$ for all trajectories

$\Leftrightarrow$ $x^T(t)[A^T(t)X + XA(t) + \alpha X]x(t) \leq 0$ for all trajectories

$\Leftrightarrow$ $A^TX + XA \preceq -\alpha X \ \forall A \in \mathcal{U}$

$\Rightarrow X \succ 0$ is LSC for a given $\alpha > 0$ iff X solves semi-infinite LMI

$$A^TX + XA \preceq -\alpha X \ \forall A \in \mathcal{U}$$

$\Rightarrow$ *Uncertain linear dynamical system*

$$\frac{d}{dt}x(t) = A(t)x(t), \ A(t) \in \mathcal{U}$$

admits an LSC iff the semi-infinite system of LMI's

$$X \succeq I, \ A^TX + XA \preceq -I \ \forall A \in \mathcal{U}$$

in matrix variable X is solvable.

♠ **But:** *SDP is about finite, and not semi-infinite, systems of LMI's. Semi-infinite systems of LMI's typically are heavily computationally intractable...*

$$X \succeq I,\ A^T X + X A \preceq -I\ \forall A \in \mathcal{U} \qquad (!)$$

♠ **Solvable case I:** *Scenario* (a.k.a. *polytopic*) *uncertainty* $\mathcal{U} = \mathrm{Conv}\{A_1, ..., A_N\}$. Here (!) is equivalent to the finite system of LMI's

$$X \succeq I,\ A_k^T X + X A_k \preceq -I,\ 1 \le k \le N$$

♠ **Solvable case II:** *Unstructured Norm-Bounded uncertainty*

$$\mathcal{U} = \{A = \bar{A} + B\Delta C :\ \|\Delta\|_{2,2} \le \rho\},$$

- $\|\cdot\|_{2,2}$: spectral norm of a matrix.

♡ **Example:** We close *open loop time invariant system*

$$\begin{array}{rcll} \frac{d}{dt}x(t) & = & Px(t) + Bu(t) & \text{[state equations]} \\ y(t) & = & Cx(t) & \text{[observed output]} \end{array}$$

with *linear feedback*

$$u(t) = Ky(t),$$

thus arriving at the *closed loop* system

$$\frac{d}{dt}x(t) = [P + BKC]x(t)$$

and want to certify stability of the closed loop system when the feedback matrix K is subject to time-varying norm-bounded perturbations:

$$K = K(t) \in \mathcal{V} = \{\bar{K} + \Delta : \|\Delta\|_{2,2} \le \rho\}.$$

This is exactly the same as to certify stability of the system

$$\frac{d}{dt}x(t) = A(t)x(t),\ A(t) \in \mathcal{U} = \{\underbrace{P + B\bar{K}C}_{\bar{A}} + B\Delta C\}$$

with unstructured norm-bounded uncertainty.

- **Observation:** The semi-infinite system of LMI's

$$X \succeq I \;\&\; A^T X + X A^T \preceq -I \;\forall (A = \bar{A} + B\Delta C : \|\Delta\|_{2,2} \le \rho)$$

is of the generic form

$$\begin{cases} (A): & \text{finite system of LMI's in variables } x \\ \hline & \text{semi-infinite LMI} \\ (!): & A(x) + L^T(x)\Delta R + R^T\Delta^T L(x) \succeq 0 \;\forall (\Delta : \|\Delta\|_{2,2} \le \rho) \\ & A(x),\ L(x): \text{ affine in } x \end{cases}$$

♠ **Fact:** [S.Boyd et al, early 90's] *Assuming w.l.o.g. that* $R \neq 0$, *the semi-infinite LMI (!) can be equivalently represented by the usual LMI*

$$\left[\begin{array}{c|c} A(x) - \lambda R^T R & \rho L^T(x) \\ \hline \rho L(x) & \lambda I \end{array}\right] \succeq 0 \quad (!!)$$

in variables x, λ, *meaning that* x *satisfies (!) if and only* x *can be augmented by properly selected* λ *to satisfy (!!).*

♣ Key argument when proving Fact:

S-Lemma: *A homogeneous quadratic inequality*

$$x^T Bx \geq 0 \qquad (B)$$

is a consequence of strictly feasible homogeneous quadratic inequality

$$x^T Ax \geq 0 \qquad (A)$$

if and only if (B) can be obtained by taking weighted sum, with nonnegative weights, of (A) and identically true homogeneous quadratic inequality:

$$\exists(\lambda \geq 0 \ \& \ C : \underbrace{x^T Cx \geq 0 \, \forall x}_{\Leftrightarrow C \succeq 0}) : x^T Bx \equiv \lambda x^T Ax + x^T Cx$$

or, which is the same, if and only if

$$\exists \lambda \geq 0 : B \succeq \lambda A.$$

Immediate corollary: *A quadratic inequality*

$$x^T Bx + 2b^T x + \beta \geq 0$$

is a consequence of strictly feasible quadratic inequality $x^T Ax + 2a^T x + \alpha \geq 0$ if and only if

$$\exists \lambda \geq 0 : \left[\begin{array}{c|c} B - \lambda A & b^T - \lambda a^T \\ \hline b - \lambda a & \beta - \lambda \alpha \end{array}\right] \succeq 0$$

$\Rightarrow$ *We can efficiently optimize a quadratic function over the set given by a single strictly feasible quadratic constraint.*

♣ **S-Lemma:** *A homogeneous quadratic inequality*

$$x^T Bx \geq 0 \qquad (B)$$

is a consequence of strictly feasible homogeneous quadratic inequality

$$x^T Ax \geq 0 \qquad (A)$$

if and only if (B) *can be obtained by taking weighted sum, with nonnegative weights, of* (A) *and identically true homogeneous quadratic inequality:*

$$\exists(\lambda \geq 0 \ \& \ C : \underbrace{x^T Cx \geq 0 \, \forall x}_{\Leftrightarrow C \succeq 0}) : x^T Bx \equiv \lambda x^T Ax + x^T Cx$$

or, which is the same, if and only if

$$\exists \lambda \geq 0 : B \succeq \lambda A.$$

♠ **Note:** The "if" part of the claim is evident and remains true when we replace (A) with a *finite system* of quadratic inequalities:

Let a system of homogeneous quadratic inequalities

$$x^T A_i x \geq 0, \ 1 \leq i \leq m,$$

and a "target" inequality $x^T Bx \geq 0$ *be given. If the target inequality can be obtained by taking weighted sum, with nonnegative coefficients, of the inequalities of the system and an identically true homogeneous quadratic inequality,* or, equivalently, If there exist $\lambda_i \geq 0$ such that

$$B \succeq \textstyle\sum_i \lambda_i A_i,$$

then the target inequality is a consequence of the system.

$$\exists \lambda_i \geq 0 : B \succeq \sum_{i=1}^m \lambda_i A_i \qquad (!)$$
$$\Rightarrow x^T B x \geq 0 \text{ is a consequence of } x^T A_i x \geq 0, 1 \leq i \leq m$$

• If instead of homogeneous *quadratic* inequalities we were speaking about homogeneous *linear* ones, similar *sufficient* condition for the target inequality to be a consequence of the system would be also *necessary* (Homogeneous Farkash Lemma).

• The power of S-Lemma is in the claim that *when $m = 1$, the sufficient condition (!) for the target inequality $x^T B x \geq 0$ to be a consequence of the system $x^T A_i x \geq 0$, $1 \leq i \leq m$, is also necessary*, provided the "system" $x^T A_1 x \geq 0$ is strictly feasible.

The "necessity" part of S-Lemma *fails to be true* when $m > 1$.

Proof of the "only if" part of S-Lemma

• **Situation:** We are given two symmetric matrices A, B such that

(I): $$\exists \bar{x} : \bar{x}^T A \bar{x} > 0$$

and

(II): $$x^T A x \geq 0 \text{ implies } x^T B x \geq 0$$

or, equivalently,

(I-II): $$\text{Opt} := \min_x \{x^T B x : x^T A x \geq 0\} \geq 0$$
and the constraint $x^T A x \geq 0$ is strictly feasible

• **Goal:** To prove that

(III): $$\exists \lambda \geq 0 : B \succeq \lambda A$$

or, equivalently, that

(III′): $$\text{SDP} := \min_X \{\text{Tr}(BX) : \text{Tr}(AX) \geq 0,\ X \succeq 0\} \geq 0.$$

Equivalence of (III) and (III′): By (I), semidefinite program in (III′) is strictly feasible. Since the program is homogeneous, its optimal value is either 0, or $-\infty$. By Conic Duality, the optimal value is finite (i.e., 0) if and only if the dual problem

$$\max_{\lambda, Y} \{0 : B = \lambda A + Y, \lambda \geq 0, Y \succeq 0\}$$

is solvable, which is exactly (III).

• Given that $x^TAx \geq 0$ implies $x^TBx \geq 0$ we should prove that

$$\mathrm{Tr}(BX) \geq 0 \text{ } \textit{whenever} \text{ } \mathrm{Tr}(AX) \geq 0 \text{ } \textit{and} \text{ } X \succeq 0$$

• Let $X \succeq 0$ be such that $\mathrm{Tr}(AX) \geq 0$, and let us prove that $\mathrm{Tr}(BX) \geq 0$.

There exists *orthogonal* U such that $U^TX^{1/2}AX^{1/2}U$ is diagonal

$\Rightarrow$ For every vector ξ with ± 1 entries:

$$\begin{aligned}[X^{1/2}U\xi]^TA[X^{1/2}U\xi] &= \xi^T\underbrace{[U^TX^{1/2}AX^{1/2}U]}_{\text{diagonal}}\xi \\ &= \mathrm{Tr}(U^TX^{1/2}AX^{1/2}U) \\ &= \mathrm{Tr}(AX) \geq 0\end{aligned}$$

$\Rightarrow$ For every vector ξ with ± 1 entries:

$$0 \leq [X^{1/2}U\xi]^TB[X^{1/2}U\xi] = \xi^T[U^TX^{1/2}BX^{1/2}U]\xi$$

$\Rightarrow$ [Taking average over ± 1 vectors ξ]

$$0 \leq \mathrm{Tr}(U^TX^{1/2}BX^{1/2}U) = \mathrm{Tr}(BX)$$

Thus, $\mathrm{Tr}(BX) \geq 0$, as claimed.

Lecture II.

Interior Point Methods for $\mathcal{LP}$ and $\mathcal{SDP}$

- Primal-dual pair of $\mathcal{SDP}$ programs
- Logarithmic Barrier for the Semidefinite cone and primal-dual central path
- Tracing the primal-dual central path
- Commutative scalings
- How to start path tracing

♣ **Interior Point Methods** (IPM's) are state-of-the-art theoretically and practically efficient polynomial time algorithms for solving well-structured convex optimization programs, primarily Linear, Conic Quadratic and Semidefinite ones.

♠ Modern IPMs were first developed for $\mathcal{LP}$, and the words "Interior Point" are aimed at stressing the fact that instead of traveling along the vertices of the feasible set, as in the Simplex algorithm, the methods work in the interior of the feasible domain.

♠ **Note:** "intrinsic nature" of IPM's for $\mathcal{LP}$, $\mathcal{CQP}$ and $\mathcal{SDP}$ is the same – these methods are given by applying general *self-concordance-based* theory of polynomial time IPM's for Convex Programming to the case of conic problems on *self-dual homogeneous cones*, i.e., self-dual cones such that the group of one-to-one affine mappings preserving the cone acts transitively on its interior.

♡ The $\mathcal{LP}$ and $\mathcal{SDP}$ cases are more than "intrinsically similar" – they admit *fully unified treatment* (including the notational aspects).

$\Rightarrow$ In the sequel, we focus on the $\mathcal{SDP}$ case.

Primal-Dual Pair of $\mathcal{SDP}$ Programs

♣ Consider an $\mathcal{SDP}$ program in the form

$$\mathrm{Opt}(P) = \min_x \left\{ c^T x : \mathcal{A}x := \sum_{j=1}^n x_j A_j \succeq B \right\} \qquad (P)$$

where A_j, B are $m \times m$ block diagonal symmetric matrices of a given block-diagonal structure ν (i.e., with a given number and given sizes of diagonal blocks).

♠ (P) can be thought of as a conic problem on the self-dual and regular positive semidefinite cone $\mathbf{S}^\nu_+$ in the space $\mathbf{S}^\nu$ of symmetric block diagonal $m \times m$ matrices with block-diagonal structure ν.

• $\mathbf{S}^\nu$ is equipped with the Frobenius inner product:

$$\langle A, B \rangle = \mathrm{Tr}(AB^T) = \mathrm{Tr}(AB) = \textstyle\sum_{i,j=1}^m A_{ij} B_{ij}.$$

♠ **Note:** In the diagonal case (where all diagonal blocks are of size 1), (P) becomes an $\mathcal{LP}$ program with m linear inequality constraints and n variables.

$$\mathrm{Opt}(P) = \min_x \left\{ c^T x : \mathcal{A}x := \sum_{j=1}^n x_j A_j \succeq B \right\} \quad (P)$$

♠ **Standing Assumption A:** *The mapping $x \mapsto \mathcal{A}x$ has trivial kernel, or, equivalently, the matrices $A_1, ..., A_n$ are linearly independent.*

♠ The problem dual to (P) is

$$\mathrm{Opt}(D) = \max_{S \in \mathbf{S}^\nu} \{ \mathrm{Tr}(BS) : S \succeq 0,\ \mathrm{Tr}(A_j S) = c_j \ \forall j \} \quad (D)$$

♠ **Standing Assumption B:** *Both (P) and (D) are strictly feasible* ($\Rightarrow$ both problems are solvable with equal optimal values).

$$\mathrm{Opt}(P) = \min_x \left\{ c^T x : \mathcal{A}x := \sum_{j=1}^n x_j A_j \succeq B \right\} \qquad (P)$$
$$\mathrm{Opt}(D) = \max_{S \in \mathbf{S}^\nu} \left\{ \mathrm{Tr}(BS) : S \succeq 0,\ \mathrm{Tr}(A_j S) = c_j \ \forall j \right\} \qquad (D)$$

♠ Let $C \in \mathbf{S}^\nu$ satisfy the equality constraint in (D), so that

$$\begin{array}{rcl} \mathrm{Tr}(C[\mathcal{A}x - B]) & = & \sum_j x_j \mathrm{Tr}(CA_j) - \mathrm{Tr}(CB) \\ & = & c^T x - \mathrm{Tr}(CB) \end{array} .$$

Passing in (P) from x to the *primal slack* $X = \mathcal{A}x - B$, we can rewrite (P) equivalently as the problem

$$\mathrm{Opt}(\mathcal{P}) = \min_{X \in \mathbf{S}^\nu} \left\{ \mathrm{Tr}(CX) : X \in [\mathcal{L}_P - B] \cap \mathbf{S}^\nu_+ \right\} \qquad (\mathcal{P})$$
$$\mathcal{L}_P = \mathrm{Im}(\mathcal{A}) = \mathrm{Lin}\{A_1, ..., A_n\}$$
$$[\mathrm{Opt}(\mathcal{P}) = \mathrm{Opt}(P) - \mathrm{Tr}(CB)],$$

while (D) is the problem

$$\mathrm{Opt}(D) = \max_{S \in \mathbf{S}^\nu} \left\{ \mathrm{Tr}(BS) : S \in [\mathcal{L}_D + C] \cap \mathbf{S}^\nu_+ \right\} \qquad (D)$$
$$\mathcal{L}_D = \mathcal{L}_P^\perp = \{S : \mathrm{Tr}(A_j S) = 0,\ 1 \le j \le n\}$$

♠ At a primal-dual feasible pair (x, S), the Duality Gap $c^T x - \mathrm{Tr}(BS)$, expressed in terms of S and $X = \mathcal{A}x - B$, is

$$\begin{array}{l} [\mathrm{Tr}(C[\mathcal{A}x - B]) + \mathrm{Tr}(CB)] - \mathrm{Tr}(BS) = \mathrm{Tr}(CX) + \mathrm{Tr}(CB) - \mathrm{Tr}(BS) \\ \quad = [\mathrm{Tr}(CX) + \mathrm{Tr}(CB) - \mathrm{Opt}(P)] + [\mathrm{Opt}(D) - \mathrm{Tr}(BS)] \\ \quad = [\mathrm{Tr}(CX) - \mathrm{Opt}(\mathcal{P})] + [\mathrm{Opt}(D) - \mathrm{Tr}(BS)] \end{array}$$

On the other hand, whenever X, S are feasible for $(\mathcal{P})$, (D), we have $X + B \in \mathcal{L}_P$, $S - C \in \mathcal{L}_D = \mathcal{L}_P^\perp$

$$\begin{array}{l} \Rightarrow 0 = \mathrm{Tr}([X + B][S - C]) = \mathrm{Tr}(XS) - [\mathrm{Tr}(XC) + \mathrm{Tr}(BC) - \mathrm{Tr}(BS)] \\ \quad = \mathrm{Tr}(XS) - [\mathrm{Tr}(CX) + \mathrm{Tr}(CB) - \mathrm{Tr}(BS)] \end{array} .$$

⇒ *Whenever $X = \mathcal{A}x - B$ is feasible for $(\mathcal{P})$ and S is feasible for D, we have*

$$\begin{array}{rcl} \mathrm{DualityGap}(X, S) & := & \overbrace{[\mathrm{Tr}(CX) - \mathrm{Opt}(\mathcal{P})] + [\mathrm{Opt}(D) - \mathrm{Tr}(BS)]}^{= c^T x - \mathrm{Tr}(BS)} \\ & = & \mathrm{Tr}(XS). \end{array}$$

$$\begin{array}{rl}
\mathrm{Opt}(P) = \min\limits_x \left\{c^T x : \mathcal{A}x := \sum_{j=1}^n x_j A_j \succeq B\right\} & (P)\\
\Leftrightarrow \mathrm{Opt}(\mathcal{P}) = \min\limits_X \left\{\mathrm{Tr}(CX) : X \in [\mathcal{L}_P - B] \cap \mathbf{S}^\nu_+\right\} & (\mathcal{P})\\
\mathrm{Opt}(D) = \max\limits_S \left\{\mathrm{Tr}(BS) : S \in [\mathcal{L}_D + C] \cap \mathbf{S}^\nu_+\right\} & (D)\\
\left[\mathcal{L}_P = \mathrm{Im}\mathcal{A},\ \mathcal{L}_D = \mathcal{L}_P^\perp\right] &
\end{array}$$

Since (P) and (D) are strictly feasible, both problems are solvable with equal optimal values, and a pair of feasible solutions X to $(\mathcal{P})$ and S to $(\mathcal{D})$ is comprised of optimal solutions to the respective problems iff

$$\mathrm{DualityGap}(X, S) = \mathrm{Tr}(XS) = 0.$$

Claim: *For positive semidefinite matrices X, S, $\mathrm{Tr}(XS) = 0$ if and only if $XS = SX = 0$.*

Proof:

- **Standard Fact of Linear Algebra:** *For every matrix $A \succeq 0$ there exists exactly one matrix $B \succeq 0$ such that $A = B^2$; B is denoted $A^{1/2}$.*
- **Standard Fact of Linear Algebra:** *Whenever A, B are matrices such that the product AB makes sense and is a square matrix, $\mathrm{Tr}(AB) = \mathrm{Tr}(BA)$.*
- **Standard Fact of Linear Algebra:** *Whenever $A \succeq 0$ and QAQ^T makes sense, we have $QAQ^T \succeq 0$.*

• Standard Facts of LA $\Rightarrow$ Claim:

$0 = \mathrm{Tr}(XS) = \mathrm{Tr}(X^{1/2}X^{1/2}S) = \mathrm{Tr}(X^{1/2}SX^{1/2})$

$\Rightarrow$ All diagonal entries in the positive semidefinite matrix $X^{1/2}SX^{1/2}$ are zeros

$\Rightarrow X^{1/2}SX^{1/2} = 0$

$\Rightarrow (S^{1/2}X^{1/2})^T(S^{1/2}X^{1/2}) = 0$

$\Rightarrow S^{1/2}X^{1/2} = 0$

$\Rightarrow SX = S^{1/2}[S^{1/2}X^{1/2}]X^{1/2} = 0$

$\Rightarrow XS = (SX)^T = 0.$ $\square$

$$\begin{array}{rl}
\mathrm{Opt}(P) = \min_x \left\{ c^T x : \mathcal{A}x := \sum_{j=1}^n x_j A_j \succeq B \right\} & (P) \\
\Leftrightarrow \mathrm{Opt}(\mathcal{P}) = \min_X \left\{ \mathrm{Tr}(CX) : X \in [\mathcal{L}_P - B] \cap \mathbf{S}_+^\nu \right\} & (\mathcal{P}) \\
\mathrm{Opt}(D) = \max_S \left\{ \mathrm{Tr}(BS) : S \in [\mathcal{L}_D + C] \cap \mathbf{S}_+^\nu \right\} & (D)
\end{array}$$

$$\left[\mathcal{L}_P = \mathrm{Im}\mathcal{A},\ \mathcal{L}_D = \mathcal{L}_P^\perp\right]$$

Theorem: *Assuming* (P), (D) *strictly feasible, feasible solutions* X *for* $(\mathcal{P})$ *and* S *for* (D) *are optimal for the respective problems if and only if*

$$XS = SX = 0$$

("$\mathcal{SDP}$ Complementary Slackness").

Logarithmic Barrier for the Semidefinite Cone $\mathbf{S}^\nu_+$

$$\begin{array}{rl}
\mathrm{Opt}(P) = \min\limits_x \left\{c^T x : \mathcal{A}x := \sum_{j=1}^n x_j A_j \succeq B\right\} & (P)\\
\Leftrightarrow \mathrm{Opt}(\mathcal{P}) = \min\limits_X \left\{\mathrm{Tr}(CX) : X \in [\mathcal{L}_P - B] \cap \mathbf{S}^\nu_+\right\} & (\mathcal{P})\\
\mathrm{Opt}(D) = \max\limits_S \left\{\mathrm{Tr}(BS) : S \in [\mathcal{L}_D + C] \cap \mathbf{S}^\nu_+\right\} & (D)\\
\left[\mathcal{L}_P = \mathrm{Im}\mathcal{A},\ \mathcal{L}_D = \mathcal{L}_P^\perp\right]
\end{array}$$

♣ A crucial role in building IPMs for (P), (D) is played by the *logarithmic barrier for the positive semidefinite cone:*

$$K(X) = -\ln \mathrm{Det}(X) : \mathrm{int}(\mathbf{S}^\nu_+) \to \mathbb{R}$$

♠ **Facts:** *$K(X)$ is a smooth function on its domain $\mathbf{S}^\nu_{++} = \{X \in \mathbf{S}^\nu : X \succ 0\}$. The first- and the second order directional derivatives of this function taken at a point $X \in \mathrm{dom}K$ along a direction $H \in \mathbf{S}^\nu$ are given by*

$$\begin{array}{rcl}
\frac{d}{dt}\Big|_{t=0} K(X+tH) & = & -\mathrm{Tr}(X^{-1}H)\\
& & \left[\Leftrightarrow \nabla K(X) = -X^{-1}\right]\\
\frac{d^2}{dt^2}\Big|_{t=0} K(X+tH) & = & \mathrm{Tr}(H[X^{-1}HX^{-1}])\\
& = & \mathrm{Tr}([X^{-1/2}HX^{-1/2}]^2)
\end{array}$$

In particular, K is strongly convex:

$$X \in \mathrm{Dom}K, 0 \neq H \in \mathbf{S}^\nu \Rightarrow \frac{d^2}{dt^2}\Big|_{t=0} K(X+tH) > 0$$

Proof:

$$
\begin{aligned}
&\tfrac{d}{dt}\big|_{t=0}[-\ln\det(X+tH)]\\
&=\tfrac{d}{dt}\big|_{t=0}[-\ln\det(X[I+tX^{-1}H])]\\
&=\tfrac{d}{dt}\big|_{t=0}[-\ln\det(X)-\ln\det(I+tX^{-1}H)]\\
&=\tfrac{d}{dt}\big|_{t=0}[-\ln\det(I+tX^{-1}H)]\\
&=-\tfrac{d}{dt}\big|_{t=0}[\det(I+tX^{-1}H)]\ \text{[chain rule]}\\
&=-\operatorname{Tr}(X^{-1}H)
\end{aligned}
$$

$$
\begin{aligned}
&\tfrac{d}{dt}\big|_{t=0}\left[-\operatorname{Tr}([X+tG]^{-1}H)\right]\\
&=\tfrac{d}{dt}\big|_{t=0}\left[-\operatorname{Tr}([X[I+tX^{-1}G]]^{-1}H)\right]\\
&=-\operatorname{Tr}\left(\left[\tfrac{d}{dt}\big|_{t=0}[I+tX^{-1}G]^{-1}\right]X^{-1}H\right)\\
&=\operatorname{Tr}(X^{-1}GX^{-1}H)
\end{aligned}
$$

In particular, when $X \succ 0$ and $H \in \mathbf{S}^{\nu}$, $H \neq 0$, we have

$$
\begin{aligned}
&\tfrac{d^2}{dt^2}\big|_{t=0}K(X+tH)=\operatorname{Tr}(X^{-1}HX^{-1}H)\\
&=\operatorname{Tr}(X^{-1/2}[X^{-1/2}HX^{-1/2}]X^{-1/2}H)\\
&=\operatorname{Tr}([X^{-1/2}HX^{-1/2}]X^{-1/2}HX^{-1/2})\\
&=\langle X^{-1/2}HX^{-1/2},X^{-1/2}HX^{-1/2}\rangle>0.
\end{aligned}
$$

Additional properties of $K(\cdot)$:

- $\nabla K(tX) = -[tX]^{-1} = -t^{-1}X^{-1} = t^{-1}\nabla K(X)$
- The mapping $X \mapsto -\nabla K(X) = X^{-1}$ maps the domain $\mathbf{S}^\nu_{++}$ of K onto itself and is self-inverse:

$$S = -\nabla K(X) \Leftrightarrow X = -\nabla K(S) \Leftrightarrow XS = SX = I$$

- The function $K(X)$ is an *interior penalty* for the positive semidefinite cone $\mathbf{S}^\nu_+$: whenever points $X_i \in \mathrm{Dom}K = \mathbf{S}^\nu_{++}$ converge to a boundary point of $\mathbf{S}^\nu_+$, one has $K(X_i) \to \infty$ as $i \to \infty$.

Primal-Dual Central Path

$$\begin{array}{rl}
\mathrm{Opt}(P) = \min\limits_x \left\{c^T x : \mathcal{A}x := \sum_{j=1}^n x_j A_j \succeq B\right\} & (P)\\
\Leftrightarrow \mathrm{Opt}(\mathcal{P}) = \min\limits_X \left\{\mathrm{Tr}(CX) : X \in [\mathcal{L}_P - B] \cap \mathbf{S}^\nu_+\right\} & (\mathcal{P})\\
\mathrm{Opt}(D) = \max\limits_S \left\{\mathrm{Tr}(BS) : S \in [\mathcal{L}_D + C] \cap \mathbf{S}^\nu_+\right\} & (D)\\
K(X) = -\ln \mathrm{Det}(X)
\end{array}$$

Let

$$\mathcal{X} = \{X \in \mathcal{L}_P - B : X \succ 0\}$$
$$\mathcal{S} = \{S \in \mathcal{L}_D + C : S \succ 0\}.$$

be the (nonempty!) sets of strictly feasible solutions to $(\mathcal{P})$ and (D), respectively. Given *path parameter* $\mu > 0$, consider the functions

$$P_\mu(X) = \mathrm{Tr}(CX) + \mu K(X) : \mathcal{X} \to \mathbb{R}$$
$$D_\mu(S) = -\mathrm{Tr}(BS) + \mu K(S) : \mathcal{S} \to \mathbb{R}\ .$$

Fact: *For every $\mu > 0$, the function $P_\mu(X)$ achieves its minimum at $\mathcal{X}$ at a unique point $X_*(\mu)$, and the function $D_\mu(S)$ achieves its minimum on $\mathcal{S}$ at a unique point $S_*(\mu)$. These points are related:*

$$X_*(\mu) = \mu S_*^{-1}(\mu) \Leftrightarrow S_*(\mu) = \mu X_*^{-1}(\mu)$$
$$\Leftrightarrow X_*(\mu)S_*(\mu) = S_*(\mu)X_*(\mu) = \mu I$$

Thus, we can associate with $(\mathcal{P})$, (D) the primal-dual central path – the curve

$$\{X_*(\mu), S_*(\mu)\}_{\mu>0},$$

where for every $\mu > 0$ $X_*(\mu)$ is a strictly feasible solution to $(\mathcal{P})$, and $S_*(\mu)$ is a strictly feasible solution to (D), and $X_*(\mu)S_*(\mu) = \mu I$.

Duality Gap on the Central Path

$$\begin{array}{rl}
\mathrm{Opt}(P) = \min\limits_x \left\{c^T x : \mathcal{A}x := \sum_{j=1}^n x_j A_j \succeq B\right\} & (P)\\
\Leftrightarrow \mathrm{Opt}(\mathcal{P}) = \min\limits_X \left\{\mathrm{Tr}(CX) : X \in [\mathcal{L}_P - B] \cap \mathbf{S}^\nu_+\right\} & (\mathcal{P})\\
\mathrm{Opt}(D) = \max\limits_S \left\{\mathrm{Tr}(BS) : S \in [\mathcal{L}_D + C] \cap \mathbf{S}^\nu_+\right\} & (D)
\end{array}$$

$$\Rightarrow \left\{\begin{array}{l} X_*(\mu) \in [\mathcal{L}_P - B] \cap \mathbf{S}^\nu_{++} \\ S_*(\mu) \in [\mathcal{L}_D + C] \cap \mathbf{S}^\nu_{++} \end{array}\right\} : X_*(\mu)S_*(\mu) = \mu I$$

Observation: *On the primal-dual central path, the duality gap is*

$$\mathrm{Tr}(X_*(\mu)S_*(\mu)) = \mathrm{Tr}(\mu I) = \mu m.$$

Therefore the sum of non-optimalities of the strictly feasible solution $X_*(\mu)$ to $(\mathcal{P})$ and the strictly feasible solution $S_*(\mu)$ to (D) in terms of the respective objectives is equal to μm and goes to 0 as $\mu \to +0$.

$\Rightarrow$ Our ideal goal would be to move along the primal-dual central path, pushing the path parameter μ to 0 and thus approaching primal-dual optimality, while maintaining primal-dual feasibility.

♠ Our ideal goal is not achievable – how could we move along a curve? A *realistic* goal could be to move in a neighborhood of the primal-dual central path, staying close to it. A good notion of "closeness to the path" is given by the *proximity measure* of a triple $\mu > 0, X \in \mathcal{X}, S \in \mathcal{S}$ to the point $(X_*(\mu), S_*(\mu))$ on the path:

$$\begin{aligned}
\mathrm{dist}(X,S,\mu) &= \sqrt{\mathrm{Tr}(X[X^{-1}-\mu^{-1}S]X[X^{-1}-\mu^{-1}S])} \\
&= \sqrt{\mathrm{Tr}(X^{1/2}[X^{1/2}[X^{-1}-\mu^{-1}S]X^{1/2}]\left[X^{1/2}[X^{-1}-\mu^{-1}S]\right])} \\
&= \sqrt{\mathrm{Tr}([X^{1/2}[X^{-1}-\mu^{-1}S]X^{1/2}]\left[X^{1/2}[X^{-1}-\mu^{-1}S]X^{1/2}\right])} \\
&= \sqrt{\mathrm{Tr}([X^{1/2}[X^{-1}-\mu^{-1}S]X^{1/2}]^2)} \\
&= \sqrt{\mathrm{Tr}([I-\mu^{-1}X^{1/2}SX^{1/2}]^2)}.
\end{aligned}$$

Note: We see that $\mathrm{dist}(X,S,\mu)$ is well defined and $\mathrm{dist}(X,S,\mu) = 0$ iff $X^{1/2}SX^{1/2} = \mu I$, or, which is the same,

$$SX = X^{-1/2}[X^{1/2}SX^{1/2}]X^{1/2} = \mu X^{-1/2}X^{1/2} = \mu I,$$

i.e., iff $X = X_*(\mu)$ and $S = S_*(\mu)$.

Note: We have

$$\begin{aligned}
\mathrm{dist}(X,S,\mu) &= \sqrt{\mathrm{Tr}(X[X^{-1}-\mu^{-1}S]X[X^{-1}-\mu^{-1}S])} \\
&= \sqrt{\mathrm{Tr}([I-\mu^{-1}XS][I-\mu^{-1}XS])} \\
&= \sqrt{\mathrm{Tr}(\left[[I-\mu^{-1}XS][I-\mu^{-1}XS]\right]^T)} \\
&= \sqrt{\mathrm{Tr}([I-\mu^{-1}SX][I-\mu^{-1}SX])} \\
&= \sqrt{\mathrm{Tr}(S[S^{-1}-\mu^{-1}X]S[S^{-1}-\mu^{-1}X])},
\end{aligned}$$

$\Rightarrow$ The proximity is defined in a symmetric w.r.t. X, S fashion.

Fact: *Whenever $X \in \mathcal{X}$, $S \in \mathcal{S}$ and $\mu > 0$, one has*

$$\mathrm{Tr}(XS) \le \mu[m + \sqrt{m}\,\mathrm{dist}(X, S, \mu)]$$

Indeed, we have seen that

$$d := \mathrm{dist}(X, S, \mu) = \sqrt{\mathrm{Tr}([I - \mu^{-1}X^{1/2}SX^{1/2}]^2)}.$$

Denoting by λ_i the eigenvalues of $X^{1/2}SX^{1/2}$, we have

$$\begin{aligned}
& d^2 = \mathrm{Tr}([I - \mu^{-1}X^{1/2}SX^{1/2}]^2) = \textstyle\sum_i [1 - \mu^{-1}\lambda_i]^2 \\
\Rightarrow\ & \textstyle\sum_i |1 - \mu^{-1}\lambda_i| \le \sqrt{m}\sqrt{\sum_i [1 - \mu^{-1}\lambda_i]^2} = \sqrt{m}d \\
\Rightarrow\ & \textstyle\sum_i \lambda_i \le \mu[m + \sqrt{m}d] \\
\Rightarrow\ & \mathrm{Tr}(XS) = \mathrm{Tr}(X^{1/2}SX^{1/2}) = \textstyle\sum_i \lambda_i \le \mu[m + \sqrt{m}d]
\end{aligned}$$

Corollary. *Let us say that a triple (X, S, μ) is close to the path, if $X \in \mathcal{X}$, $S \in \mathcal{S}$, $\mu > 0$ and* $\mathrm{dist}(X, S, \mu) \le 0.1$. *Whenever (X, S, μ) is close to the path, one has*

$$\mathrm{Tr}(XS) \le 2\mu m,$$

that is, if (X, S, μ) is close to the path, then X is at most $2\mu m$-nonoptimal strictly feasible solution to $(\mathcal{P})$, and S is at most $2\mu m$-nonoptimal strictly feasible solution to (D).

Tracing the Central Path

♣ **The goal:** To follow the central path, staying close to it and pushing μ to 0 as fast as possible.

♣ **Question.** *Assume we are given a triple* $(\bar{X}, \bar{S}, \bar{\mu})$ *close to the path. How to update it into a triple* (X_+, S_+, μ_+)*, also close to the path, with* $\mu_+ < \mu$*?*

♠ **Conceptual answer:** Let us choose μ_+, $0 < \mu_+ < \bar{\mu}$, and try to update $\bar{X}, \bar{S}$ into

$$X_+ = \bar{X} + \Delta X, \; S_+ = \bar{S} + \Delta S$$

in order to make the triple (X_+, S_+, μ_+) close to the path. Our goal is to ensure that

$$\begin{array}{rcccccl} X_+ = \bar{X} + \Delta X & \in & \mathcal{L}_P - B & \& & X_+ & \succ & 0 \quad (a) \\ S_+ = \bar{S} + \Delta S & \in & \mathcal{L}_D + C & \& & S_+ & \succ & 0 \quad (b) \\ & & G_{\mu_+}(X_+, S_+) \approx 0 & & & & (c) \end{array}$$

where $G_\mu(X, S) = 0$ expresses equivalently the *augmented slackness* condition $XS = \mu I$. For example, we can take

$$\begin{array}{l} G_\mu(X, S) = S - \mu^{-1} X^{-1}, \text{ or} \\ G_\mu(X, S) = X - \mu^{-1} S^{-1}, \text{ or} \\ G_\mu(X, S) = XS + SX = 2\mu I, \text{ or...} \end{array}$$

$$
\begin{array}{rcl}
X_+ = \bar{X} + \Delta X & \in & \mathcal{L}_P - B \quad \& \quad X_+ \succ 0 \quad (a)\\
S_+ = \bar{S} + \Delta S & \in & \mathcal{L}_D + C \quad \& \quad S_+ \succ 0 \quad (b)\\
G_{\mu_+}(X_+, S_+) \approx 0 & & (c)
\end{array}
$$

♠ Since $\bar{X} \in \mathcal{L}_P - B$ and $\bar{X} \succ 0$, (a) amounts to $\Delta X \in \mathcal{L}_P$, which is a system of linear equations on ΔX, and to $\bar{X} + \Delta X \succ 0$. Similarly, (b) amounts to the system $\Delta S \in \mathcal{L}_D$ of linear equations on ΔS, and to $\bar{S} + \Delta S \succ 0$. To handle the troublemaking *nonlinear in* $\Delta X, \Delta S$ condition (c), we *linearize* G_{μ_+} *in* ΔX *and* ΔS*:*

$$
\begin{array}{l}
G_{\mu_+}(X_+, S_+) \approx G_{\mu_+}(\bar{X}, \bar{S})\\
\quad + \left.\frac{\partial G_{\mu_+}(X,S)}{\partial X}\right|_{(X,S)=(\bar{X},\bar{S})} \Delta X + \left.\frac{\partial G_{\mu_+}(X,S)}{\partial S}\right|_{(X,S)=(\bar{X},\bar{S})} \Delta S
\end{array}
$$

and enforce the linearization, as evaluated at ΔX, ΔS*, to be zero.* We arrive at the *Newton system*

$$
\left\{
\begin{array}{l}
\Delta X \in \mathcal{L}_P\\
\Delta S \in \mathcal{L}_D\\
\frac{\partial G_{\mu_+}}{\partial X}\Delta X + \frac{\partial G_{\mu_+}}{\partial S}\Delta S = -G_{\mu_+}
\end{array}
\right. \qquad (N)
$$

(the value and the partial derivatives of $G_{\mu_+}(X, S)$ are taken at the point $(\bar{X}, \bar{S})$).

♠ We arrive at conceptual *primal-dual path-following method* where one iterates the updatings

$$(X_i, S_i, \mu_i) \mapsto (X_{i+1} = X_i + \Delta X_i, S_{i+1} = S_i + \Delta S_i, \mu_{i+1})$$

with $\mu_{i+1} \in (0, \mu_i)$ and $\Delta X_i, \Delta S_i$ solving the Newton system

$$\begin{cases} \Delta X_i \in \mathcal{L}_P \\ \Delta S_i \in \mathcal{L}_D \\ \frac{\partial G^{(i)}_{\mu_{i+1}}}{\partial X} \Delta X_i + \frac{\partial G^{(i)}_{\mu_{i+1}}}{\partial S} \Delta S_i = -G^{(i)}_{\mu_{i+1}}, \end{cases} \qquad (N_i)$$

$G^{(i)}_\mu(X, S) = 0$ represents equivalently the augmented complementary slackness condition $XS = \mu I$ and the value and the partial derivatives of $G^{(i)}_{\mu_{i+1}}$ are evaluated at (X_i, S_i).

♠ Being initialized at a close to the path triple (X_0, S_0, μ_0), this conceptual algorithm should

- be well-defined: (N_i) should remain solvable, X_i should remain strictly feasible for $(\mathcal{P})$, S_i should remain strictly feasible for (D), and
- maintain closeness to the path: for every i, (X_i, S_i, μ_i) should remain close to the path.

♡ Under these limitations, we want to push μ_i to 0 as fast as possible.

Example: Primal Path-Following Method

$$\mathrm{Opt}(P) = \min_x \left\{ c^T x : \mathcal{A}x := \sum_{j=1}^n x_j A_j \succeq B \right\} \qquad (P)$$

$$\Leftrightarrow \mathrm{Opt}(\mathcal{P}) = \min_X \left\{ \mathrm{Tr}(CX) : X \in [\mathcal{L}_P - B] \cap \mathbf{S}^\nu_+ \right\} \qquad (\mathcal{P})$$

$$\mathrm{Opt}(D) = \max_S \left\{ \mathrm{Tr}(BS) : S \in [\mathcal{L}_D + C] \cap \mathbf{S}^\nu_+ \right\} \qquad (D)$$

$$\left[\mathcal{L}_P = \mathrm{Im}\mathcal{A},\ \mathcal{L}_D = \mathcal{L}_P^\perp\right]$$

♣ Let us choose

$$G_\mu(X,S) = S + \mu \nabla K(X) = S - \mu X^{-1}$$

Then the Newton system becomes

$$\begin{array}{rcl} \Delta X_i \in \mathcal{L}_P & \Leftrightarrow & \Delta X_i = \mathcal{A}\Delta x_i \\ \Delta S_i \in \mathcal{L}_D & \Leftrightarrow & \mathcal{A}^* \Delta S_i = 0 \\ & & \mathcal{A}^* U = [\mathrm{Tr}(A_1 U); ...; \mathrm{Tr}(A_n U)] \end{array} \qquad (N_i)$$

$$(!) \quad \Delta S_i + \mu_{i+1} \nabla^2 K(X_i) \Delta X_i = -[S_i + \mu_{i+1} \nabla K(X_i)]$$

♠ Substituting $\Delta X_i = \mathcal{A}\Delta x_i$ and applying $\mathcal{A}^*$ to both sides in (!), we get

$$(*) \quad \mu_{i+1} \underbrace{[\mathcal{A}^* \nabla^2 K(X_i) \mathcal{A}]}_{\mathcal{H}} \Delta x_i = -\big[\underbrace{\mathcal{A}^* S_i}_{=c} + \mathcal{A}^* \nabla K(X_i)\big]$$

$$\Delta X_i = \mathcal{A}\Delta x_i$$

$$S_{i+1} = \mu_{i+1} \left[\nabla K(X_i) - \nabla^2 K(X_i) \mathcal{A}\Delta x_i\right]$$

The mappings $h \mapsto \mathcal{A}h$, $H \mapsto \nabla^2 K(X_i) H$ have trivial kernels

$\Rightarrow \mathcal{H}$ is nonsingular

$\Rightarrow (N_i)$ has a unique solution given by

$$\Delta x_i = -\mathcal{H}^{-1} \left[\mu_{i+1}^{-1} c + \mathcal{A}^* \nabla K(X_i)\right]$$

$$\Delta X_i = \mathcal{A}\Delta x_i$$

$$S_{i+1} = S_i + \Delta S_i = \mu_{i+1} \left[\nabla K(X_i) - \nabla^2 K(X_i) \mathcal{A}\Delta x_i\right]$$

$$\mathrm{Opt}(P) = \min_x \left\{ c^T x : \mathcal{A}x := \sum_{j=1}^n x_j A_j \succeq B \right\} \qquad (P)$$
$$\Leftrightarrow \mathrm{Opt}(\mathcal{P}) = \min_X \left\{ \mathrm{Tr}(CX) : X \in [\mathcal{L}_P - B] \cap \mathbf{S}^\nu_+ \right\} \qquad (\mathcal{P})$$
$$\mathrm{Opt}(D) = \max_S \left\{ \mathrm{Tr}(BS) : S \in [\mathcal{L}_D + C] \cap \mathbf{S}^\nu_+ \right\} \qquad (D)$$
$$\left[\mathcal{L}_P = \mathrm{Im}\mathcal{A},\ \mathcal{L}_D = \mathcal{L}_P^\perp\right]$$
$$\Rightarrow \left\{ \begin{array}{l} \Delta x_i = -\mathcal{H}^{-1}\left[\mu_{i+1}^{-1} c + \mathcal{A}^* \nabla K(X_i)\right] \\ \qquad \left[\mathcal{H} = \mathcal{A}^* \nabla^2 K(X_i) \mathcal{A}\right] \\ \Delta X_i = \mathcal{A} \Delta x_i \\ S_{i+1} = S_i + \Delta S_i = \mu_{i+1}\left[\nabla K(X_i) - \nabla^2 K(X_i) \mathcal{A} \Delta x_i\right] \end{array} \right.$$

♠ $X_i = \mathcal{A}x_i - B$ for a (uniquely defined by X_i) strictly feasible solution x_i to (P). Setting

$$F(x) = K(\mathcal{A}x - B),$$

we have $\mathcal{A}^* \nabla K(X_i) = \nabla F(x_i),\ \mathcal{H} = \nabla^2 F(x_i)$

$\Rightarrow$ *The above recurrence can be written solely in terms of* x_i *and* F*:*

$$(\#) \quad \left\{ \begin{array}{l} \mu_i \mapsto \mu_{i+1} < \mu_i \\ x_{i+1} = x_i - [\nabla^2 F(x_i)]^{-1}\left[\mu_{i+1}^{-1} c + \nabla F(x_i)\right] \end{array} \right.$$
$$X_{i+1} = \mathcal{A}x_{i+1} - B$$
$$S_{i+1} = \mu_{i+1}\left[\nabla K(X_i) - \nabla^2 K(X_i)\mathcal{A}[x_{i+1} - x_i]\right]$$

Recurrence $(\#)$ is called the *primal path-following method.*

$$\mathrm{Opt}(P) = \min_x \left\{ c^T x : \mathcal{A}x := \sum_{j=1}^n x_j A_j \succeq B \right\} \qquad (P)$$
$$\Leftrightarrow \mathrm{Opt}(\mathcal{P}) = \min_X \left\{ \mathrm{Tr}(CX) : X \in [\mathcal{L}_P - B] \cap \mathbf{S}^\nu_+ \right\} \qquad (\mathcal{P})$$
$$\mathrm{Opt}(D) = \max_S \left\{ \mathrm{Tr}(BS) : S \in [\mathcal{L}_D + C] \cap \mathbf{S}^\nu_+ \right\} \qquad (D)$$
$$\left[\mathcal{L}_P = \mathrm{Im}\mathcal{A},\ \mathcal{L}_D = \mathcal{L}_P^\perp\right]$$

♠ The primal path-following method can be explained as follows:

• The barrier $K(X) = -\ln \mathrm{Det} X$ induces the barrier $F(x) = K(\mathcal{A}x - B)$ for the interior P^o of the feasible domain of (P).

• The primal central path

$$X_*(\mu) = \mathrm{argmin}_{X = \mathcal{A}x - B \succ 0} \left[\mathrm{Tr}(CX) + \mu K(X)\right]$$

induces the path

$$x_*(\mu) \in P^o:\ X_*(\mu) = \mathcal{A}x_*(\mu) + \mu F(x).$$

Observing that

$$\mathrm{Tr}(C[\mathcal{A}x - B]) + \mu K(\mathcal{A}x - B) = c^T x + \mu F(x) + \mathrm{const},$$

we have

$$x_*(\mu) = \mathrm{argmin}_{x \in P^o} F_\mu(x),\ F_\mu(x) = c^T x + \mu F(x).$$

• The method works as follows: given $x_i \in P^o, \mu_i > 0$, we

— replace μ_i with $\mu_{i+1} < \mu_i$

— convert x_i into x_{i+1} by applying to the function $F_{\mu_{i+1}}(\cdot)$ a single step of the *Newton minimization method*

$$x_i \mapsto x_{i+1} - [\nabla^2 F_{\mu_{i+1}}(x_i)]^{-1} \nabla F_{\mu_{i+1}}(x_i)$$

$$\begin{array}{rl}
\mathrm{Opt}(P) = \min\limits_x \left\{c^T x : \mathcal{A}x := \sum_{j=1}^n x_j A_j \succeq B\right\} & (P)\\
\Leftrightarrow \mathrm{Opt}(\mathcal{P}) = \min\limits_X \left\{\mathrm{Tr}(CX) : X \in [\mathcal{L}_P - B] \cap \mathbf{S}_+^\nu\right\} & (\mathcal{P})\\
\mathrm{Opt}(D) = \max\limits_S \left\{\mathrm{Tr}(BS) : S \in [\mathcal{L}_D + C] \cap \mathbf{S}_+^\nu\right\} & (D)\\
\left[\mathcal{L}_P = \mathrm{Im}\mathcal{A},\ \mathcal{L}_D = \mathcal{L}_P^\perp\right] &
\end{array}$$

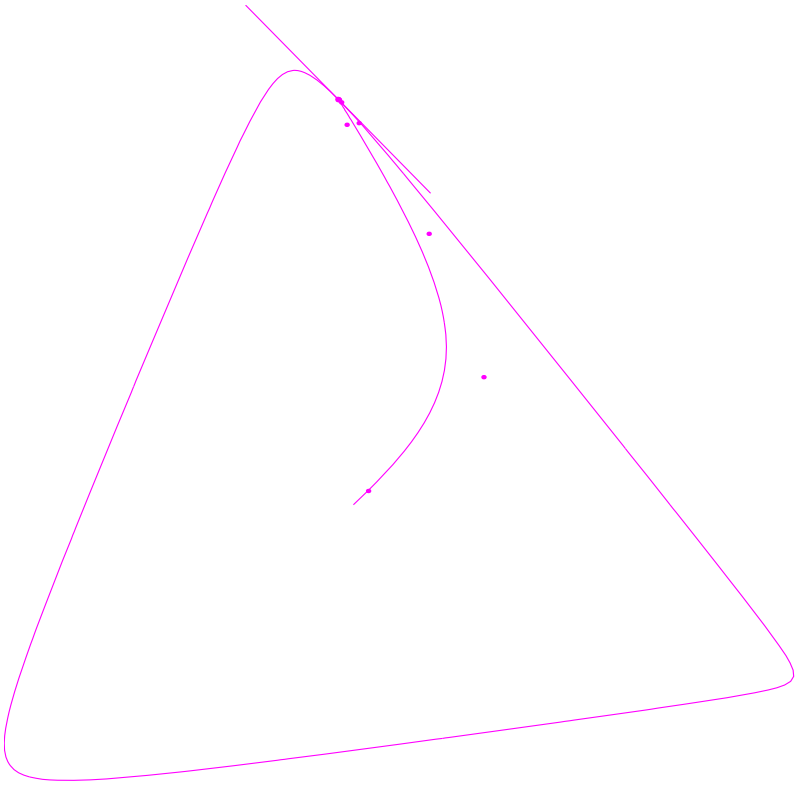
Theorem. *Let $(X_0 = \mathcal{A}x_0 - B, S_0, \mu_0)$ be close to the primal-dual central path, and let (P) be solved by the Primal path-following method where the path parameter μ is updated according to*

$$\mu_{i+1} = \left(1 - \frac{0.1}{\sqrt{m}}\right)\mu_i. \qquad (*)$$

Then the method is well defined and all triples $(X_i = \mathcal{A}x_i - B, S_i, \mu_i)$ are close to the path.

♠ With rule $(*)$ it takes $O(\sqrt{m})$ steps to reduce the path parameter μ by an absolute constant factor. Since the method stays close to the path, the duality gap $\mathrm{Tr}(X_i S_i)$ of i-th iterate does not exceed $2m\mu_i$.

⇒ *The number of steps to make the duality gap $\le \epsilon$ does not exceed $O(1)\sqrt{m}\ln\left(1 + \frac{2m\mu_0}{\epsilon}\right)$.*



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	Gap	Itr#	Objective	Gap
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

Duality gap along the iterations

♣ The Primal path-following method is yielded by Conceptual Path-Following Scheme when the Augmented Complementary Slackness condition is represented as

$$G_\mu(X,S) := S + \mu\nabla K(X) = 0.$$

Passing to the representation

$$G\mu(X,S) := X + \mu\nabla K(S) = 0,$$

we arrive at the *Dual path-following method* with the same theoretical properties as those of the primal method. The Primal and the Dual path-following methods imply the best known so far complexity bounds for $\mathcal{LP}$ and $\mathcal{SDP}$.

♠ In spite of being "theoretically perfect", Primal and Dual path-following methods in practice are inferior as compared to the methods based on less straightforward and more symmetric forms of the Augmented Complementary Slackness condition.

Commutative Scalings

♠ The Augmented Complementary Slackness condition is

$$XS = SX = \mu I \qquad (*)$$

Fact: For $X, S \in \mathbf{S}^\nu_{++}$, $(*)$ is equivalent to

$$XS + SX = 2\mu I$$

Indeed, if $XS = SX = \mu I$, then clearly $XS + SX = 2\mu I$. On the other hand,

$$\begin{array}{l} X, S \succ 0, XS + SX = 2\mu I \\ \Rightarrow S + X^{-1}SX = 2\mu X^{-1} \\ \Rightarrow X^{-1}SX = 2\mu X^{-1} - S \\ \Rightarrow X^{-1}SX = [X^{-1}SX]^T = XSX^{-1} \\ \Rightarrow X^2S = SX^2 \end{array}$$

We see that $X^2S = SX^2$. Since $X \succ 0$, X is a polynomial of X^2, whence X and S commute, whence $XS = SX = \mu I$. □

Fact: *Let $Q \in \mathbf{S}^\nu$ be nonsingular, and let $X, S \succ 0$. Then $XS = \mu I$ if and only if*

$$QXSQ^{-1} + Q^{-1}SXQ = 2\mu I$$

Indeed, it suffices to apply the previous fact to the matrices $\widehat{X} = QXQ \succ 0$, $\widetilde{S} = Q^{-1}SQ^{-1} \succ 0$. □

♠ In practical path-following methods, at step i the Augmented Complementary Slackness condition is written down as

$$G_{\mu_{i+1}}(X,S) := Q_i X S Q_i^{-1} + Q_i^{-1} S X Q_i - 2\mu_{i+1} I = 0$$

with properly chosen varying from step to step non-singular matrices $Q_i \in \mathbf{S}^\nu$.

Explanation: Let $Q \in \mathbf{S}^\nu$ be nonsingular. The *Q-scaling* $X \mapsto QXQ$ is a one-to-one linear mapping of $\mathbf{S}^\nu$ 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.*

$\Rightarrow$ Given a primal-dual pair of semidefinite programs

$$\mathrm{Opt}(\mathcal{P}) = \min_X \left\{ \mathrm{Tr}(CX) : X \in [\mathcal{L}_P - B] \cap \mathbf{S}^\nu_+ \right\} \quad (\mathcal{P})$$

$$\mathrm{Opt}(\mathcal{D}) = \max_S \left\{ \mathrm{Tr}(BS) : S \in [\mathcal{L}_D + C] \cap \mathbf{S}^\nu_+ \right\} \quad (\mathcal{D})$$

and a nonsingular matrix $Q \in \mathbf{S}^\nu$, one can pass in $(\mathcal{P})$ from variable X to variables $\widehat{X} = QXQ$, while passing in $(\mathcal{D})$ from variable S to variable $\widetilde{S} = Q^{-1}SQ^{-1}$. The resulting problems are

$$\mathrm{Opt}(\mathcal{P}) = \min_{\widehat{X}} \left\{ \mathrm{Tr}(\widetilde{C}\widehat{X}) : \widehat{X} \in [\widehat{\mathcal{L}}_P - \widehat{B}] \cap \mathbf{S}^\nu_+ \right\} \quad (\widehat{\mathcal{P}})$$

$$\mathrm{Opt}(\mathcal{D}) = \max_{\widetilde{S}} \left\{ \mathrm{Tr}(\widehat{B}\widetilde{S}) : \widetilde{S} \in [\widetilde{\mathcal{L}}_D + \widetilde{C}] \cap \mathbf{S}^\nu_+ \right\} \quad (\widetilde{\mathcal{D}})$$

$$\left[\begin{array}{c} \widehat{B} = QBQ, \widehat{\mathcal{L}}_P = \{QXQ : X \in \mathbf{L}_P\}, \\ \widetilde{C} = Q^{-1}CQ^{-1}, \widetilde{\mathcal{L}}_D = \{Q^{-1}SQ^{-1} : S \in \mathcal{L}_D\} \end{array} \right]$$

$$\mathrm{Opt}(\mathcal{P}) = \min_{\widehat{X}} \left\{ \mathrm{Tr}(\widetilde{C}\widehat{X}) : \widehat{X} \in [\widehat{\mathcal{L}}_P - \widehat{B}] \cap \mathbf{S}^\nu_+ \right\} \quad (\widehat{\mathcal{P}})$$

$$\mathrm{Opt}(\mathcal{D}) = \max_{\widetilde{S}} \left\{ \mathrm{Tr}(\widehat{B}\widetilde{S}) : \widetilde{S} \in [\widetilde{\mathcal{L}}_D + \widetilde{C}] \cap \mathbf{S}^\nu_+ \right\} \quad (\widetilde{\mathcal{D}})$$

$$\left[\begin{array}{c} \widehat{B} = QBQ, \widehat{\mathcal{L}}_P = \{QXQ : X \in \mathbf{L}_P\}, \\ \widetilde{C} = Q^{-1}CQ^{-1}, \widetilde{\mathcal{L}}_D = \{Q^{-1}SQ^{-1} : S \in \mathcal{L}_D\} \end{array} \right]$$

$\widehat{\mathcal{P}}$ and $\widetilde{\mathcal{D}}$ 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*

$$(X, S) \mapsto (\widehat{X} = QXQ, \widetilde{S} = Q^{-1}SQ^{-1})$$

$\mathcal{Q}$ preserves closeness to the path, etc.

♡ Writing down the Augmented Complementary Slackness condition as

$$QXSQ^{-1} + Q^{-1}SXQ = 2\mu I \qquad (!)$$

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 Augmented Complementary Slackness condition for the latter pair in the simplest primal-dual symmetric form

$$\widehat{X}\widetilde{S} + \widetilde{S}\widehat{X} = 2\mu I,$$

- "scale back" to the original primal-dual variables X, S, thus arriving at (!).

Note: In the $\mathcal{LP}$ case $\mathbf{S}^\nu$ is comprised of diagonal matrices, so that (!) is exactly the same as the "unscaled" condition $XS = \mu I$.

$$G_{\mu_{i+1}}(X,S) := Q_i X S Q_i^{-1} + Q_i^{-1} S X Q_i - 2\mu_{i+1} I = 0 \qquad (!)$$

With (!), the Newton system becomes

$$\Delta X \in \mathcal{L}_P,\ \Delta S \in \mathcal{L}_D$$
$$Q_i \Delta X S_i Q_i^{-1} + Q_i^{-1} S_i \Delta X Q_i + Q_i X_i \Delta S Q_i^{-1} + Q_i^{-1} \Delta S X_i Q_i = 2\mu_{i+1} I - Q_i X_i S_i Q_i^{-1} - Q_i^{-1} S_i X_i Q_i$$

♣ Theoretical analysis of path-following methods simplifies a lot when the scaling (!) is *commutative*, meaning that the matrices $\widehat{X}_i = Q_i X_i Q_i$ and $\widehat{S}_i = Q_i^{-1} S_i Q_i^{-1}$ commute.

Popular choices of commuting scalings are:

- $Q_i = S_i^{1/2}$ ("XS-method," $\widetilde{S} = I$)
- $Q_i = X_i^{-1/2}$ ("SX-method, $\widehat{X} = I$)
- $Q_i = \left(X^{-1/2}(X^{1/2} S X^{1/2})^{-1/2} X^{1/2} S\right)^{1/2}$ (famous *Nesterov-Todd* method, $\widehat{X} = \widetilde{S}$).

$$\begin{array}{rl}
\mathrm{Opt}(P) = \min_x \left\{ c^T x : \mathcal{A}x := \sum_{j=1}^n x_j A_j \succeq B \right\} & (P) \\
\Leftrightarrow \mathrm{Opt}(\mathcal{P}) = \min_X \left\{ \mathrm{Tr}(CX) : X \in [\mathcal{L}_P - B] \cap \mathbf{S}^\nu_+ \right\} & (\mathcal{P}) \\
\mathrm{Opt}(D) = \max_S \left\{ \mathrm{Tr}(BS) : S \in [\mathcal{L}_D + C] \cap \mathbf{S}^\nu_+ \right\} & (D) \\
\left[\mathcal{L}_P = \mathrm{Im}\mathcal{A},\ \mathcal{L}_D = \mathcal{L}_P^\perp \right] &
\end{array}$$

Theorem *Let a strictly feasible primal-dual pair (P), (D) of semidefinite programs be solved by a primal-dual path-following method based on commutative scalings. Assume that the method is initialized by a close to the path triple $(X_0, S_0, \mu_0 = \mathrm{Tr}(X_0 S_0)/m)$ and let the policy for updating μ be*

$$\mu_{i+1} = \left(1 - \frac{0.1}{\sqrt{m}}\right) \mu_i.$$

The the trajectory is well defined and stays close to the path, and $\mathrm{Tr}(X_i S_i) = \mu_i m$ for all i.
As a result, every $O(\sqrt{m})$ steps of the method reduce the duality gap by an absolute constant factor, and it takes $O(1)\sqrt{m} \ln\left(1 + \frac{m\mu_0}{\epsilon}\right)$ steps to make the duality gap $\leq \epsilon$.

♠ To improve the practical performance of primal-dual path-following methods, in actual computations

• the path parameter is updated in an on line adjustable fashion more "aggressive" than

$$\mu \mapsto \left(1 - \frac{0.1}{\sqrt{m}}\right)\mu;$$

• the method is allowed to travel in a wider neighborhood of the primal-dual central path than the neighborhood given by our "close to the path" restriction $\mathrm{dist}(X, S, \mu) \leq 0.1$;

• instead of updating $X_{i+1} = X_i + \Delta X_i$, $S_{i+1} = S_i + \Delta S_i$, one uses the more flexible updating

$$X_{i+1} = X_i + \alpha_i \Delta X_i,\ S_{i+1} = S_i + \alpha_i \Delta S_i$$

with α_i given by appropriate line search.

♣ The constructions and the complexity results we have presented are incomplete — 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 problem of interest.

How to Start Path Tracing: Infeasible Start Primal-Dual Path-Following Method

♣ Standard implementations of primal-dual path-following methods for $\mathcal{LP}/\mathcal{CQP}/\mathcal{SDP}$ are *infeasible start* methods based on *self-dual embedding*. In the $\mathcal{SDP}$ case, these methods are as follows:

♠ We start with the strictly feasible primal-dual pair of problems

$$\begin{array}{rl}
\mathrm{Opt}(P) = \min\limits_x \left\{c^T x : \mathcal{A}x := \sum_{j=1}^n x_j A_j \succeq B\right\} & (P)\\
\Leftrightarrow \mathrm{Opt}(\mathcal{P}) = \min\limits_X \left\{\mathrm{Tr}(CX) : X \in [\mathcal{L} - B] \cap \mathbf{S}^\nu_+\right\} & (\mathcal{P})\\
\mathrm{Opt}(D) = \max\limits_S \left\{\mathrm{Tr}(BS) : S \in [\mathcal{L}^\perp + C] \cap \mathbf{S}^\nu_+\right\} & (D)
\end{array}$$

$$[\mathcal{L} = \mathrm{Im}\mathcal{A}]$$

Note: When shifting B along $\mathcal{L}$ and C along $\mathcal{L}^\perp$, $(\mathcal{P})$, (D) remain "essentially intact:" the feasible sets remain the same, and the objectives are shifted by constants.

⇒ *We lose nothing when assuming $B \in \mathcal{L}^\perp$, $C \in \mathcal{L}$, in which case the duality gap at a primal-dual feasible pair (X, S) of solutions becomes*

$$\mathrm{DualityGap}(X, S) := \mathrm{Tr}(XS) = \mathrm{Tr}(CX) - \mathrm{Tr}(BS).$$

$$\mathrm{Opt}(\mathcal{P}) = \min_X \left\{ \mathrm{Tr}(CX) : X \in [\mathcal{L} - B] \cap \mathbf{S}^\nu_+ \right\} \quad (\mathcal{P})$$
$$\mathrm{Opt}(D) = \max_S \left\{ \mathrm{Tr}(BS) : S \in [\mathcal{L}^\perp + C] \cap \mathbf{S}^\nu_+ \right\} \quad (D)$$
$$[\mathcal{L} = \mathrm{Im}\mathcal{A}]$$

♠ Consider the system of conic constraints in variables $X, S \in \mathbf{S}^\nu$ and scalar variables τ, σ:

$$\begin{array}{rcl} X + \tau B & \in & \mathcal{L} + P \\ S - \tau C & \in & \mathcal{L}^\perp + D \\ \mathrm{Tr}(CX) - \mathrm{Tr}(BS) + \sigma & = & d \\ X \in \mathbf{S}^\nu_+,\ S \in \mathbf{S}^\nu_+,\ \tau \geq 0, \sigma \geq 0 & & \end{array} \quad (C)$$

where the data P, D, d are such that

(i) we can easily find a *strictly feasible* solution $\widehat{Y} = (\widehat{X}, \widehat{S}, \widehat{\sigma}, \widehat{\tau} = 1)$ to (C)

(ii) *The feasible set $\mathcal{Y}$ of (C) is unbounded, and whenever a sequence $\{Y_i = (X_i, S_i, \tau_i, \sigma_i) \in \mathcal{Y}\}_{i=1}^\infty$ goes to ∞, one has*

$$\tau_i \to \infty,\ i \to \infty$$

♠ Assume we have a mechanism for building a sequence $\{Y_i = (X_i, S_i, \tau_i, \sigma_i) \in \mathcal{Y}\}_{i=1}^\infty$ which goes to ∞. Setting

$$\bar{X}_i = \tau_i^{-1} X_i,\ \ \bar{S}_i = \tau_i^{-1} S_i$$

and taking into account that $\tau_i \to \infty$ by (ii), we conclude that *as $i \to \infty$, the primal-dual infeasibility of $(\bar{X}_i, \bar{S}_i)$ and the duality gap $\mathrm{Tr}(C\bar{X}_i) - \mathrm{Tr}(B\bar{S}_i)$ go to zero at the rate $O(1/\tau_i)$.*

$$\begin{array}{rcl} X+\tau B & \in & \mathcal{L}+P \\ S-\tau C & \in & \mathcal{L}^\perp+D \\ \langle C,X\rangle-\langle B,S\rangle+\sigma & = & d \\ X\in\mathbf{S}^\nu_+,\ S\in\mathbf{S}^\nu_+,\ \tau\geq 0,\sigma\geq 0 & & \end{array} \qquad (C)$$

♠ The outlined idea can be implemented as follows.

A. *We select $P\in\mathbf{S}^\nu_+$, $D\in\mathbf{S}^\nu_+$ in such a way that $P\succ B$ and $D\succ -C$, select somehow $\widehat{\sigma}>0$ and set*

$$d=\mathrm{Tr}(C[P-B])-\mathrm{Tr}(B[D+C])+\widehat{\sigma},$$
$$\widehat{Y}=(\widehat{X}=P-B,\widehat{S}=C+D,\widehat{\sigma},\widehat{\tau}=1)$$

With some moderate effort (heavily exploiting strict primal-dual feasibility of the primal-dual problem in question), it can be verified that this choice ensures (i) and (ii).

$$\begin{array}{rcl} X+\tau B & \in & \mathcal{L}+P \\ S-\tau C & \in & \mathcal{L}^\perp+D \\ \mathrm{Tr}(CX)-\mathrm{Tr}(BS)+\sigma & = & d \\ X\in\mathbf{S}^\nu_+,\ S\in\mathbf{S}^\nu_+,\ \tau\geq 0,\sigma\geq 0 & & \end{array} \qquad (C)$$

B. (C) is of the form

$$Y:=\mathrm{Diag}\{X,S,\sigma,\tau\}\in\mathcal{M}\cap\mathbf{S}^{\widehat{\nu}}_+$$

where $\widehat{\nu}$ is some block-diagonal structure, and $\mathcal{M}$ is an affine plane in $\mathbf{S}^{\widehat{\nu}}$. Denoting by $\widehat{K}(\cdot)$ the log Det barrier for $\mathbf{S}^{\widehat{\nu}}$ and setting $\widehat{C}=-\nabla\widehat{K}(\bar{Y})$, the primal central path of the auxiliary conic problem

$$\min_Y\left\{\mathrm{Tr}(\widehat{C}Y):Y\in\mathcal{M}\cap\mathbf{S}^{\widehat{\nu}}_+\right\} \qquad (M)$$

passes through $\widehat{Y}$ as $\mu=1$. *We can trace this path, starting with $\mu=1$ and $\widehat{Y}$ and staying close to it, pushing μ to ∞ rather than to 0.*

• Since the feasible set of (M) is unbounded by (ii), it can be seen that in this fashion we "run to ∞ along the feasible $\mathcal{Y}$ of (C)," thus enforcing the approximate primal-dual solutions $(\tau_i^{-1}X_i,\tau_i^{-1}S_i)$ to the problems of interest to approach primal-dual feasibility and primal-dual optimality at the rate which, on a closest inspection, is

$$\exp\{-O(1)i/\sqrt{m}\}$$

[m: size of matrices from $\mathbf{S}^\nu$] .

Lecture III.

Mirror Descent for Large-Scale Deterministic and Stochastic Convex Optimization

- Proximal Setup
- Basic Mirror Descent
- Favorable Geometry Case
- Stochastic Case
- Utilizing Problem's Structure: Mirror Prox
- Application: $O(1/t)$ Nonsmooth Convex Minimization
- Acceleration by Randomization

♣ **Problem of Primary Interest:** *Convex Minimization*

$$\mathrm{Opt} = \min_{x \in X} f(x) \qquad (P)$$

- X: convex compact subset of Euclidean space E
- $f : X \to \mathbb{R}$: convex Lipschitz continuous

♠ f is represented by a *First Order Oracle:*

- given on input $x \in X$, FOO returns the value $f(x)$ and a subgradient $f'(x)$ of f at x
- the vector field $x \mapsto f'(x)$ is assumed to be *bounded* on X

♣ **Mirror Descent for** (P), *milestones:*

- *Subgradient Descent* ("Euclidean prototype"): N. Shor, 1967:

$$X \ni x_\tau \mapsto x_{\tau+1} = \mathrm{Proj}_X(x_\tau - \gamma_\tau f'(x_\tau))$$

  - $\gamma_\tau > 0$: stepsizes • $\mathrm{Proj}_X(y) = \mathrm{argmin}_{z \in X} \|y - z\|_2$
- *General Mirror Descent scheme*: Nem., 1979
- *Modern Proximal Point form*: A. Beck & M. Teboulle, 2003

Proximal Setup

$$\mathrm{Opt} = \min_{x \in X} f(x) \qquad (P)$$

- X: compact subset of Euclidean space E

♣ **Setup for MD** ("proximal setup") is given by

- a *norm* $\|\cdot\|$ on E
- a *Distance Generating Function (DGF)*

$$\omega(x) : X \to \mathbb{R}$$

which should be

- *convex and continuous on* X
- *admitting a continuous on*

$$X^o = \{x \in X : \partial\omega(x) \neq \emptyset\}$$

selection $\omega'(x)$ *of subgradients*

- *compatible with* $\|\cdot\|$, that is, *strongly convex, modulus 1, w.r.t.* $\|\cdot\|$:

$$\langle \omega'(x) - \omega'(x'), x - x' \rangle \geq \|x - x'\|^2 \ \forall x, x' \in X^o$$

♠ **Example:** *Euclidean setup:*

$$E = \mathbb{R}^n,\ \|x\| = \|x\|_2,\ \omega(x) = \tfrac{1}{2} x^T x$$

♣ **Standing Assumption:** *From now on, if otherwise is not explicitly stated,* X *is assumed to be bounded.*

♣ **Proximal setup** $\|\cdot\|, \omega(\cdot)$ for $X \subset E$ induces:

- *ω-center of X* $x_\omega = \mathrm{argmin}_{x \in X} \omega(x)$
- *Bregman distance*

$$V_x(y) = \omega(y) - \omega(x) - \langle \omega'(x), y - x \rangle,$$

$x \in X^o, y \in X$. By strong convexity of $\omega(\cdot)$,

$$V_x(y) \geq \tfrac{1}{2}\|y - x\|^2$$

- *ω-radius of X*

$$\Omega = \Omega[X, \omega(\cdot)] = \sqrt{2[\max_{x \in X} \omega(x) - \min_{x \in X} \omega(x)]}$$

For $x \in X$ one has

$$\tfrac{1}{2}\|x - x_\omega\|^2 \leq V_{x_\omega}(x) \leq \omega(x) - \omega(x_\omega) \leq \tfrac{1}{2}\Omega^2$$
$$\Rightarrow \|x - x_\omega\| \leq \Omega \ \forall x \in X$$

- *prox-mapping*

$$[x \in X^o, \xi \in E] \mapsto \mathrm{Prox}_x(\xi) := \mathrm{argmin}_{z \in X} [\langle \xi, z \rangle + V_x(z)] \in X^o$$

♠ With Euclidean setup,

$$V_x(y) = \tfrac{1}{2}\|x - y\|_2^2, \ \mathrm{Prox}_x(\xi) = \mathrm{Proj}_X(x - \xi)$$

$\Rightarrow$ Subgradient Descent is the recurrence

$$x_{\tau+1} = \mathrm{Prox}_{x_\tau}(\gamma_\tau f'(x_\tau))$$

Basic Mirror Descent

- X: convex compact subset of Euclidean space E
- $\|\cdot\|, \omega(\cdot)$: proximal setup for (E, X)

♣ **MD** works with a *sequence of vector fields*

$$\{g_\tau(\cdot) : X \to E\}_\tau$$

represented by *an oracle*.

At call $\tau = 1, 2, ...$, the query point being x_τ, the oracle returns the vector $g_\tau(x_\tau) \in E$.

- *In most of applications, the sequence* $\{g_\tau(\cdot)\}_\tau$ *is just stationary:* $g_\tau(\cdot) \equiv g(\cdot)$.

♠ MD is the recurrence

$$x_1 = x_\omega := \mathrm{argmin}_X\, \omega(\cdot);\ x_{\tau+1} = \mathrm{Prox}_{x_\tau}(\gamma_\tau g_\tau(x_\tau))$$

- $x_\tau \in X^o$: search points
- $\gamma_\tau > 0$: stepsizes

$$x_1 = x_\omega := \mathrm{argmin}_X\, \omega;\; x_{\tau+1} = \mathrm{Prox}_{x_\tau}(\gamma_\tau g_\tau(x_\tau))$$

♣ **Main Property of MD:** *Under Boundedness Assumption*

$$\sup_{x\in X,\tau} \|g_\tau(x)\|_* \le L < \infty$$

- $\|\xi\|_* = \max\{\langle \xi, x\rangle : \|x\| \le 1\}$ is the conjugate of $\|\cdot\|$

the residual

$$\varepsilon_t := \max_{z\in X} \sum_{\tau\le t} \lambda_\tau^t \langle g_\tau(x_\tau), x_\tau - z\rangle,\;\; \lambda_\tau^t = \frac{\gamma_\tau}{\sum_{s\le t}\gamma_s}$$

obeys the bound

$$\varepsilon_t \le \frac{\Omega^2 + \sum_{\tau\le t}\gamma_\tau^2 \|g_\tau(x_\tau)\|_*^2}{2\sum_{\tau\le t}\gamma_\tau},\; t = 1, 2, ...$$

- *In particular,*

$$\frac{\Omega}{L\sqrt{t}} \le \gamma_\tau \le \frac{\Omega}{\|g_\tau(x_\tau)\|_*\sqrt{t}}, \qquad 1 \le \tau \le t$$

(e.g., $\gamma_\tau = \frac{\Omega}{L\sqrt{t}}$, or $\gamma_\tau = \frac{\Omega}{\|g_\tau(x_\tau)\|_*\sqrt{t}}$, $1 \le \tau \le t$) implies

$$\varepsilon_t \le \Omega L/\sqrt{t}.$$

♠ **Fact:** *When $g_\tau(\cdot)$ come from problem "with convex structure," the residual ε_t upper-bounds inaccuracy of the approximate solution*

$$x^t := \sum_{\tau\le t} \lambda_\tau^t x_\tau$$

to the problem.

Example 1: Convex Minimization $\mathrm{Opt} = \min_X f$. Applying MD to $\{g_\tau(\cdot) \equiv f'(\cdot)\}_\tau$ and assuming w.l.o.g. the Lipschitz constant $L_{\|\cdot\|}(f)$ of f taken w.r.t. $\|\cdot\|$ to upper-bound $\|f'(\cdot)\|_*$, one has

$$
\begin{array}{rcl}
 & & f(x^t) - \mathrm{Opt} \le \varepsilon_t : \\
\varepsilon_t & = & \max\limits_{z \in X} \sum_{\tau \le t} \lambda_\tau^t \langle f'(x_\tau), x_\tau - z \rangle \\
 & \ge & \max\limits_{z \in X} \sum_{\tau \le t} \lambda_\tau^t [f(x_\tau) - f(z)] \\
 & \ge & \max\limits_{z \in X} [f(\sum_{\tau \le t} \lambda_\tau^t x_\tau) - f(z)] \\
 & = & f(x^t) - \mathrm{Opt}
\end{array}
$$

$\Rightarrow$ *For every t, t-step MD with appropriate stepsizes ensures*

$$f(x^t) - \mathrm{Opt} \le \Omega L_{\|\cdot\|}(f)/\sqrt{t}$$

Example 1.A: Convex Online Minimization. *When $g_\tau(x) = f'_\tau(x)$, with convex functions $f_\tau(\cdot) : X \to \mathbb{R}$ satisfying $\|f'_\tau(x)\|_* \le L < \infty$ for all $x \in X, \tau$, t-step MD with stepsizes $\gamma_\tau = \frac{\Omega}{L\sqrt{t}}$, $1 \le \tau \le t$, ensures that*

$$\frac{1}{t} \sum_{\tau \le t} f_\tau(x_\tau) \le \frac{\Omega L}{\sqrt{t}} + \min_{x \in X} \frac{1}{t} \sum_{\tau \le t} f_\tau(x)$$

Example 2: Convex-Concave Saddle Point problem

$$\text{SadVal} = \min_{u\in U}\max_{v\in V} f(u,v).$$

♣ **Situation:**

- $X = U \times V \subset E_u \times E_v =: E$ with compact convex U,V
- $f(u,v) : X \to \mathbb{R}$: convex in $x \in U$, concave in $v \in V$, Lipschitz continuous

♠ f, U, V give rise to two convex optimization problems:

$$\begin{array}{rcll} \text{Opt}(P) & = & \min_{u\in U}\left[\overline{f}(u) := \max_{v\in V} f(u,v)\right] & (P)\\ \text{Opt}(D) & = & \max_{v\in V}\left[\underline{f}(v) := \min_{u\in U} f(u,v)\right] & (D)\end{array}$$

with equal optimal values:

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

and to *vector field*

$$g(\underbrace{[u;v]}_{x}) = \left[\begin{array}{c} g_u(u,v) \in \partial_u f(u,v)\\ g_v(u,v) \in \partial_v(-f(u,v))\end{array}\right] : \underbrace{U\times V}_{X} \to E$$

♠ Optimal solutions u_*, v_* to (P), (D) are exactly the *saddle points* of f on $U \times V$:

$$f(u,v_*) \geq f(u_*,v_*) \geq f(u_*,v)\ \forall (u\in U, v\in V):$$

Mirror Descent for Saddle Point Problems

$$\begin{array}{rcll}\mathrm{Opt}(P) & = & \min_{u\in U}\left[\overline{f}(u) := \max_{v\in V} f(u,v)\right] & (P)\\ \mathrm{Opt}(D) & = & \max_{v\in V}\left[\underline{f}(v) := \min_{u\in U} f(u,v)\right] & (D)\end{array}$$

$$\Rightarrow g(u;v) = [f'_u(u,v); -f'_v(u,v)] : U\times V \to E$$

♣ **Fact:** *Applying MD to $g_\tau(\cdot)\equiv g(\cdot)$, the residual*

$$\varepsilon_t = \max_{z\in X}\sum_{\tau\le t}\lambda^t_\tau\langle g(x_\tau), x_\tau - z\rangle,\ \ \lambda^t_\tau = \gamma_\tau/\sum_{s\le t}\gamma_s$$

upper-bounds the saddle point inaccuracy ("duality gap") of the approximate solution

$$x^t = [u^t; v^t] := \sum\nolimits_{\tau\le t}\lambda^t_\tau x_\tau$$

to (P,D):

$$[\overline{f}(u^t) - \mathrm{Opt}(P)] + [\mathrm{Opt}(D) - \underline{f}(v^t)] = \overline{f}(u^t) - \underline{f}(v^t) \le \varepsilon_t$$

$$\begin{array}{l}\forall [u;v]\in U\times V : \varepsilon_t \ge \sum_{\tau\le t}\lambda^t_\tau\langle g(x_\tau), x_\tau - [u;v]\rangle\\ = \sum_{\tau\le t}\lambda^t_\tau[\langle f'_u(u_\tau,v_\tau), u_\tau - u\rangle + \langle -f'_v(u_\tau,v_\tau), v_\tau - v\rangle]\\ \ge \sum_{\tau\le t}\lambda^t_\tau[f(u_\tau,v_\tau) - f(u,v_\tau) - f(u_\tau,v_\tau) + f(u_\tau,v)]\\ = \sum_{\tau\le t}\lambda^t_\tau[f(u_\tau,v) - f(u,v_\tau)] \ge f(u^t,v) - f(u,v^t)\\ \Rightarrow \varepsilon_t \ge \max_{u\in U, v\in V}[f(u^t,v) - f(u,v^t)] = \overline{f}(u^t) - \underline{f}(v^t).\end{array}$$

$$\begin{array}{rcll}\mathrm{Opt}(P) & = & \min_{u\in U}\left[\overline{f}(u) := \max_{v\in V} f(u,v)\right] & (P)\\ \mathrm{Opt}(D) & = & \max_{v\in V}\left[\underline{f}(v) := \min_{u\in U} f(u,v)\right] & (D)\end{array}$$
$$\Rightarrow g(u;v) = [f'_u(u,v); -f'_v(u,v)] : U\times V \to E$$

♠ Assuming that $\|\cdot\|$ respects representation $E = E_u \times E_v$: $\|[u;v]\| \equiv \|[u;-v]\|$, we can ensure that $\|g(\cdot)\|_* \le L_{\|\cdot\|}(f)$.

$\Rightarrow$ *t-step MD with properly chosen stepsizes ensures*

$$[\overline{f}(u^t) - \mathrm{Opt}(P)] + [\mathrm{Opt}(D) - \underline{f}(v^t)] \le \Omega L_{\|\cdot\|}(f)/\sqrt{t}.$$

♠ Similar results hold true for other "problems with convex structure:"

- variational inequalities with monotone operators
- convex Nash equilibrium problems

Reason for Main Property

♣ **Fact:** With $V_x(z) = \omega(z) - \omega(x) - \langle \omega'(x), z - x \rangle$ one has

$$x_+ = \mathrm{Prox}_x(\xi) := \mathrm{argmin}_{z \in X} \left[\langle \xi, z \rangle + V_x(z) \right] \quad (1)$$

$$\Rightarrow \forall (z \in X) : \langle \xi, x_+ - z \rangle \leq V_x(z) - V_{x_+}(z) - V_x(x_+) \quad (2)$$

Proof: rearrange terms in the optimality conditions for (1):

$$\langle \xi + \omega'(x_+) - \omega'(x), z - x_+ \rangle \geq 0 \, \forall z \in X$$

♣ **Fact:** (2) implies that

$$\forall (z \in X) : \langle \xi, x - z \rangle \leq V_x(z) - V_{x_+}(z) + \tfrac{1}{2} \|\xi\|_*^2 \quad (3)$$

Proof: by (2),

$$\langle \xi, x - z \rangle \leq V_x(z) - V_{x_+}(z) + [\langle \xi, x - x_+ \rangle - V_x(x_+)],$$

and

$$\langle \xi, x - x_+ \rangle - V_x(x_+) \leq \|\xi\|_* \|x - x_+\| - \tfrac{1}{2} \|x - x_+\|^2 \leq \tfrac{1}{2} \|\xi\|_*^2.$$

♠ By (3), $x_1 = \mathrm{argmin}_X \, \omega$; $x_{\tau+1} = \mathrm{Prox}_{x_\tau}(\gamma_\tau g_\tau)$ yields

$$\gamma_\tau \langle g_\tau, x_\tau - x \rangle \leq V_{x_\tau}(z) - V_{x_{\tau+1}}(z) + \tfrac{1}{2} \gamma_\tau^2 \|g_\tau\|_*^2 \; \forall (z \in X, \tau)$$

$$\Rightarrow \sum_{\tau \leq t} \gamma_\tau \langle g_\tau, x_\tau - z \rangle \leq \frac{1}{2} \Omega^2 + \frac{1}{2} \sum_{\tau \leq t} \gamma_\tau^2 \|g_\tau\|_*^2 \; \forall z \in X$$

♠ Dividing by $\sum_{\tau \leq t} \gamma_\tau$ and maximizing in $z \in X$, we get

$$\varepsilon_t := \mathrm{max}_{z \in X} \left[\textstyle\sum_{\tau \leq t} \lambda_\tau^t \langle g_\tau, x_\tau - z \rangle \right] \leq \frac{\Omega^2 + \sum_{\tau \leq t} \gamma_\tau^2 \|g_\tau\|_*^2}{2 \sum_{\tau \leq t} \gamma_\tau}$$

Role of Symmetry

$$\varepsilon_t \le \Omega[\sup_{x\in X,\tau} \|g_\tau(x)\|_*]/\sqrt{t} \qquad (*)$$

♣ When X is "nearly symmetric," the MD efficiency estimate can be improved. Assume that

- *X contains $\|\cdot\|$-ball of radius $\theta\Omega$*
- *The vector fields $\{g_\tau(\cdot)\}_\tau$ are uniformly semi-bounded:*

$$M := \sup_{x,x'\in X,\tau} \langle g_\tau(x), x'-x\rangle < \infty$$

Then *for every $t \ge 4/\theta^2$, the t-step MD with the stepsizes*

$$\gamma_\tau = \frac{\Omega}{\|g_\tau(x_\tau)\|_*\sqrt{t}},\ 1 \le \tau \le t$$

ensures that

$$\varepsilon_t \le 2\theta^{-1}M/\sqrt{t} \qquad (!)$$

♠ **Note:** *When $\theta = O(1)$,*

- (!) *can only be better than* $(*)$
- When $g_\tau(\cdot) \equiv g(\cdot)$ comes from

$$\min_{u\in U}\max_{v\in V} f(u,v),$$

we have

$$M \le \max_{U\times V} f - \min_{U\times V} f$$

$\Rightarrow$ (!) *becomes*

$$\varepsilon_t \le O(1)\left[\max_{U\times V} f - \min_{U\times V} f\right]/\sqrt{t}$$

$O(1/\sqrt{t})$ – good or bad?

♣ The MD convergence rate $O(1/\sqrt{t})$ is slow. However, *this is the best possible rate one can expect when solving nonsmooth large-scale convex problems represented by FO oracles*, or any other oracles providing *local* information.

♠ **Bad news:** *Consider Convex Minimization problem*

$$\mathrm{Opt}(f) = \min_x \{f(x) : \|x\| \le R\} \qquad (P_f)$$

where $\|\cdot\|$ *is either the norm* $\|\cdot\|_p$ *on* $E = \mathbb{R}^n$ $(p = 1, 2)$, *or the nuclear norm on* $\mathbb{R}^{n\times n}$. *Let*

$$\mathcal{F}_{\|\cdot\|}(L) = \{f : E \to \mathbb{R} : \ f \text{ is convex, } L_{\|\cdot\|}(f) \le L\},$$

and assume that when solving (P_f), $f \in \mathcal{F}_{\|\cdot\|}(L)$ *is learned via calls, one per step, to a FO (or any local) oracle. Then for every* $t \le n$ *and any* t*-step algorithm* $\mathcal{B}$ *one has*

$$\sup_{f\in\mathcal{F}_{\|\cdot\|}(L)} [f(x_{\mathcal{B}}(f)) - \mathrm{Opt}(f)] \ge 0.01LR/\sqrt{t}$$

- $x_{\mathcal{B}}(f)$: solution generated in t steps by $\mathcal{B}$ as applied to (P_f)

$$\mathrm{Opt}(f) = \min_{x \in X} f(x), \ X \subset X_R := \{x \in E : \|x\| \leq R\} \quad (P_f)$$

$\|\cdot\|$: $\|\cdot\|_p$ norm on $E = \mathbb{R}^n$ ($p = 1, 2$), or nuclear norm on $\mathbb{R}^{n \times n}$.

♠ **Relatively good news:** With appropriate proximal setup, t-step MD as applied to (P_f) ensures

$$f(x^t) - \mathrm{Opt}(f) \leq O\left(L_{\|\cdot\|}(f) R / \sqrt{t}\right)$$

- hidden factor: $O(1)$ for $\|\cdot\| = \|\cdot\|_2$, otherwise $O(1)\sqrt{\ln(n+1)}$

Note:

- *Rate of convergence is (nearly) dimension-independent*
- *When X is simple, computational effort per MD step in the large scale case is by order of magnitudes smaller than in all known polynomial time Convex Optimization techniques, like Interior Point methods*

$\Rightarrow$ *When solving problems with convex structure to low or medium accuracy, MD could be the method of choice...*

Favorable Geometry Case

$$\varepsilon_t \le \Omega[X,\omega] \sup_{x\in X,\tau} \|g_\tau(x)\|_*/\sqrt{t}$$

♣ **Question:** *How to choose a good proximal setup?*

- In general, the answer depends on the geometry of X and on a priori information on $\{g_\tau(\cdot)\}_\tau$
- There is, however, a *favorable geometry* case when the answer is clear:
  - Assuming w.l.o.g. that X linearly spans E, $X^+ = \frac{1}{2}[X - X]$ is the unit ball of norm $\|\cdot\|_X$ given solely by X.
  - A *Favorable Geometry case* is the one where X admits a d.-g.f. $\omega_X(\cdot)$ such that $\|\cdot\|_X, \omega_X(\cdot)$ is a valid proximal setup *with "moderate"* $\Omega_X := \Omega[X,\omega_X]$ ($O(1)$, or $O(1)\ln^{O(1)}(\dim X)$).

$$\varepsilon_t \leq \Omega[X,\omega] \sup_{x\in X,\tau} \|g_\tau(x)\|_*/\sqrt{t}$$

♠ **Observation:** *Let $\omega_X(\cdot)$ complement $\|\cdot\|_X$ to a proximal setup. Then for every proximal setup $\|\cdot\|$, $\omega(\cdot)$ for X and every $\{g_\tau(\cdot)\}_\tau$ one has*

$$\sup_{x\in X,\tau} \|g_\tau(x)\|_{X,*} \leq \Omega[X,\omega] \sup_{x\in X,\tau} \|g_\tau(x)\|_* \qquad (!)$$

whence

$$\Omega_X \sup_{x\in X,\tau} \|g_\tau(x)\|_{X,*} \leq \Omega_X\Omega[X,\omega] \sup_{x\in X,\tau} \|g_\tau(x)\|_*$$

$\Rightarrow$ Passing from $\|\cdot\|,\omega(\cdot)$ to $\|\cdot\|_X,\omega_X(\cdot)$ spoils MD efficiency at worst by factor $\Omega_X = \Omega[X,\omega_X]$. Thus, with moderate Ω_X, the proximal setup $\|\cdot\|_X,\omega_X(\cdot)$ is nearly optimal.

♠ **Reason for** (!): For every $g \in E$ and every x with $\|x\|_X \leq 1$, so that $x = [u-v]/2$ with $u,v \in X$ we have:

$$\begin{array}{rcl} \langle g,x\rangle & = & \frac{1}{2}\left[\langle g,u-x_\omega\rangle + \langle g,x_\omega - v\rangle\right] \\ & \leq & \frac{1}{2}\|g\|_*[\|u-x_\omega\| + \|v-x_\omega\|] \\ & \leq & \Omega[X,\omega]\|g\|_* \\ \Rightarrow \|g\|_{X,*} & \leq & \Omega[X,\omega]\|g\|_* \end{array}$$

Favorable Geometry: Examples

♠ **Examples of Favorable Geometry domains X:**

$$X = B^1 \times ... \times B^K$$

with moderate K and *favorable geometry atoms* B^k:

- ℓ_1/ℓ_2 *balls* $B = \{y = [y^1; ...; y^n] : \sum_{j=1}^n \|y^j\|_2 \le 1\}$:

$$\begin{array}{rcl} \|y\|_B & = & \sum_{j=1}^n \|y^j\|_2 \\ \omega_B(y) & = & O(1)\sqrt{\ln(n+1)}\sum_{j=1}^n \|y^j\|_2^{\vartheta_n}, \\ & \vartheta_n = \min[2, 1+1/\ln(2n)] & \\ & \Rightarrow \Omega_B \le O(1)\sqrt{\ln(n+1)} & \end{array}$$

Note: $n = 1$ *implies Euclidean setup for* $\|\cdot\|_2$*-ball.*

- *Nuclear norm balls* $B = \{y \in \mathbb{R}^{p\times q} : \sum_{j=1}^n \sigma_j(y) \le 1\}$

[$\sigma_j(y)$: j'th singular value of y, $n = \min[p, q]$]

$$\begin{array}{rcl} \|y\|_B & = & \sum_{j=1}^n \sigma_j(y), \\ \omega_B(y) & = & O(1)\sqrt{\ln(n+1)}\sum_{j=1}^n \sigma_j^{\vartheta_n}(y) \text{ [nuclear norm]}, \\ & \Rightarrow \Omega_B \le O(1)\sqrt{\ln(n+1)} & \end{array}$$

♠ Induced proximal setup for X is, e.g.,

$$\begin{array}{rcl} \|(x_1, ..., x_K)\| & = & \max_k \|x_k\|_{B^k}, \\ \omega(x_1..., x_k) & = & \sum_k \omega_{B^k}(x_k) \\ \Rightarrow \Omega_X = \sqrt{\sum_k \Omega_{B^k}^2} & \le & O(1)\sqrt{K\ln(\dim X)} \end{array}$$

- $K = O(1) \Rightarrow$ *Favorable Geometry case* (remains true when $X \subset B^1 \times ... \times B^K$ and $\|\cdot\|_X$ is within $O(1)$ factor of $\|\cdot\|$).

♠ We have presented DGF's for (subsets of) *unit balls* of the ℓ_1/ℓ_2 and the nuclear norm compatible with the respective norms.

In some applications, there is a need in DGF *for the entire space* compatible with a given norm on the space. The most important examples are:

- The *Euclidean* DGF $\omega(y) = \frac{1}{2}y^T y$ compatible with the standard Euclidean norm $\|\cdot\|_2$ on $E = \mathbb{R}^n$,
- The *block* ℓ_1 DGF

$$\begin{array}{rcl}\omega(y = [y^1; ...; y^n]) & = & O(1)\ln(n+1)\left(\sum_{j=1}^n \|y^j\|_2^{\vartheta_n}\right)^{2/\vartheta_n} \\ & = & O(1)\ln(n+1)\|[\|y^1\|_2; ...; \|y^n\|_2]\|_{\vartheta_n}^2\end{array}$$

compatible with the ℓ_1/ℓ_2 norm

$$\|[y^1; ...; y^n]\| = \sum_{j=1}^n \|y^j\|_2$$

on $E = \mathbb{R}^{k_1} \times ... \times \mathbb{R}^{k_n}$,

Note: When $k_j = 1$ for all j, the ℓ_1/ℓ_2 norm on E becomes the norm $\|\cdot\|_1$ on $\mathbb{R}^n$

$\Rightarrow$ *We get at our disposal a DGF on $\mathbb{R}^n$ compatible with $\|\cdot\|_1$*

- The *nuclear norm* DGF

$$\begin{array}{rcl}\omega(y) & = & O(1)\ln(n+1)\left(\sum_{j=1}^n \sigma_j^{\vartheta_n}(y)\right)^{2/\vartheta_n} \\ & = & O(1)\ln(n+1)\|\sigma(y)\|_{\vartheta_n}^2\end{array}$$

compatible with the nuclear norm on the space of matrices $E = \mathbb{R}^{p\times q}$, $n = \min[p, q]$.

Favorable Geometry: Counter-Examples

♠ A domain with *intrinsically bad* geometry is the usual box

$$X = \{x \in \mathbb{R}^n : \|x\|_\infty \le 1\} \quad .$$

Here $\Omega[X, \omega] \ge \sqrt{n}$ *for all proximal setups with* $\|\cdot\| = \|\cdot\|_X = \|\cdot\|_\infty$.

♠ In fact, *large-scale* $\|\cdot\|_p$*-balls with* $p > 2$ *"are bad:"*

Let $p \ge 2$. *Consider Convex Minimization problem*

$$\mathrm{Opt}(f) = \min_x\{f(x) : x \in \mathbb{R}^n, \|x\|_p \le R\}, \quad (P_f)$$

$$f \in \mathcal{F}_{n,p}(L) = \{f : \mathbb{R}^n \to \mathbb{R} : \ f \text{ is convex, } L_{\|\cdot\|_p}(f) \le L\}$$

Assume that when solving (P_f), $f \in \mathcal{F}_{n,p}(L)$ *is learned via calls, one per step, to a FO (or any local) oracle. Then for every* $t \le n$ *and any* t*-step algorithm* $\mathcal{B}$ *one has*

$$\sup_{f \in \mathcal{F}_{n,p}(L)} [f(x_{\mathcal{B}}(f)) - \mathrm{Opt}(f)] \ge 0.01 LR/t^{1/p}$$

• $x_{\mathcal{B}}(f)$: solution generated in t steps by $\mathcal{B}$ as applied to (P_f)

$\Rightarrow$ *As* $p > 2$ *grows, our abilities to minimize oracle-represented nonsmooth convex functions over* $\|\cdot\|_p$*-balls at a dimension independent rate deteriorate and disappear at* $p = \infty$.

Favorable Geometry: Illustration

♣ The most attractive feature of MD is *ability to adjust itself, to some extent, to problem's geometry and to ensure, under favorable circumstances, (nearly) dimension independent rate of convergence*. For example:

- *When minimizing convex f over ℓ_2-ball*

$$\{x \in \mathbb{R}^n : \|x\|_2 \leq 1\},$$

MD with Euclidean setup ensures

$$f(x^t) - \min_{x\in X} f(x) \leq O(1)\frac{\max_X f - \min_X f}{\sqrt{t}}$$

- *When minimizing convex f over ℓ_1-ball*

$$\{x \in \mathbb{R}^n : \|x\|_1 \leq 1\}$$

,

MD with appropriate Non-Euclidean setup ensures

$$f(x^t) - \min_{x\in X} f(x) \leq O(1)\sqrt{\ln(n+1)}\frac{\max_X f - \min_X f}{\sqrt{t}},$$

and similarly for minimizing over nuclear norm ball in $\mathbb{R}^{n\times n}$.

- "Wrong setup" (Euclidean when minimizing over ℓ_1/nuclear norm ball, or ℓ_1/nuclear norm when minimizing over ℓ_2-ball) can spoil the efficiency by factor *as large as* $O(\sqrt{n/\ln(n)})$.

Computational Aspects

♣ When processing a vector field g on a domain X, a step of MD reduces to

(a) computing the value of g at a point,

(b) computing the value $\mathrm{Prox}_z(h)$ of the prox-mapping — solving the auxiliary convex program

$$\min_{x\in X}\left[\langle \eta, x\rangle + \omega(x)\right] \qquad (P)$$

♠ In the Favourable Geometry case, X is a direct product of Favourable geometry atoms, and ω is separable w.r.t. the atoms

$\Rightarrow (P)$ reduces to similar problems for individual atoms.

♠ When an atom B is ℓ_1/ℓ_2 ball, (P) is the problem

$$\min_{y^1,\dots,y^n}\left\{\sum_j[\langle \zeta^j, y^j\rangle + \|y^j\|_2^\theta] : \sum_j \|y^j\|_2 \le 1\right\}$$

At the optimum, y^j clearly are negative multiples of ζ^j, and the problem reduces to

$$\min_{s_1,\dots,s_n}\left\{\sum_j[s_j^\theta - \beta_j s_j] : s\ge 0, \sum_j s_j \le 1\right\},$$

that is, to minimizing a separable function under a single separable constraint. Invoking Lagrange duality, the problem reduces to a *univariate* convex program and can be easily solved within machine accuracy in $O(n)$ a.o.

$\Rightarrow$ *For an ℓ_1/ℓ_2 ball B, computing prox-mapping is easy – it takes just $O(\dim B)$ a.o.*

♠ When atom B is the unit nuclear norm ball in $\mathbb{R}^{m\times n}$ (w.l.o.g., $m \geq n$), (P) is the problem

$$\min_{y\in\mathbb{R}^{m\times n}} \left\{ \mathrm{Tr}(y^T\zeta) + \sum_{j=1}^n \sigma_j^\theta(y) : \sum_{j=1}^n \sigma_j(y) \leq 1 \right\} \quad (*)$$

Let $\zeta = u\mathrm{Diag}\{h\}v$ be the svd of ζ. It is easily seen that $(*)$ *admits an optimal solution of the form* $y = -u\mathrm{Diag}\{s\}v$ *with* $s \geq 0$. Computing this solution reduces to solving

$$\min_{s\in\mathbb{R}^n} \left\{ \sum_j [s_j^\theta - \beta_j s_j] : s \geq 0, \sum_j s_j \leq 1 \right\}$$

which takes just $O(n)$ a.o.

⇒ *For a nuclear norm ball $B \subset \mathbb{R}^{m\times n}$, computing prox-mapping reduces to computing the svd of a matrix from $\mathbb{R}^{m\times n}$ and can be quite time consuming in the large scale case.*

Stochastic Case

♣ **Situation:** Given $X \subset E$ and proximal setup $\|\cdot\|, \omega(\cdot)$, we want to process vector fields

$$g_\tau(x) : X \to E$$

represented by *Stochastic Oracle*. At τ-th call to SO, the query point being $x_\tau \in X$, the oracle returns *an estimate* $h_\tau(x_\tau; \xi_\tau) \in E$ of $g_\tau(x_\tau)$. Here $h_\tau(\cdot;\cdot)$ are deterministic functions, and $\xi_1, \xi_2, ...$ are *i.i.d.* disturbances.

♠ **Example:** Problem

$$\min_{x \in X} \left[f(x) = \mathbf{E}_{\xi \sim P} F(x, \xi) \right]$$

with convex in $x \in X$ integrant F. The associated vector field $g(x) = f'(x)$ is usually difficult to compute. However, assuming one can sample from P and F is easy to compute, we can set

$$h_\tau(x; \xi_\tau) = F'_x(x, \xi_\tau) \text{ with } \xi_1, \xi_2, ... \text{ drawn from } P$$

♠ **Standing Assumption:** *When processing* $\{g_\tau(\cdot)\}_\tau$, *for some* L, σ, μ *and all* $x \in X$, τ *it holds:*

$$\begin{array}{rcl} \|g_\tau(x)\|_* & \le & L, \\ \|\mathbf{E}_\xi\{\Delta_\tau(x;\xi)\}\|_* & \le & \mu, \\ \mathbf{E}_\xi\{\|\Delta_\tau(x;\xi)\|_*^2\} & \le & \sigma^2 \end{array}$$

- $\Delta_\tau(x;\xi) := h_\tau(x;\xi) - g_\tau(x)$: oracle's error

Stochastic Mirror Descent

- X: convex compact subset of Euclidean space E
- $\|\cdot\|, \omega(\cdot)$: proximal setup for (E, X)

$$\Rightarrow \Omega = \sqrt{2[\max_X \omega - \min_x \omega]}$$

- $\{g_\tau(x) : X \to E\}_\tau$: vector fields of interest,

$$\|g_\tau(x)\|_* \le L < \infty$$

- $\{h_\tau(x;\xi) = g_\tau(x) + \Delta_\tau(x;\xi) : X \times \Xi \to E\}_\tau$: SO

$$\left[\begin{array}{rcl} \|\mathbf{E}_{\xi\sim P}\Delta_\tau(x;\xi)\|_* & \le & \mu, \\ \mathbf{E}_{\xi\sim P}\{\|\Delta_\tau(x;\xi)\|_*^2\} & \le & \sigma^2 \end{array}\right]$$

♣ **Stochastic Mirror Descent** is the recurrence

$$\begin{array}{rcl} x_1 & = & x_\omega := \mathrm{argmin}_X\, \omega; \\ x_{\tau+1} & = & \mathrm{Prox}_{x_\tau}(\gamma_\tau h_\tau(x_\tau;\xi_\tau)) \\ x^t & = & \sum_{\tau\le t} \lambda_\tau^t x_\tau, \\ & & \lambda_\tau^t = \dfrac{\gamma_\tau}{\sum_{s\le t}\gamma_s} \end{array}$$

- $\xi_\tau \sim P$: independent
- $\gamma_\tau > 0$: deterministic stepsizes

$$\begin{array}{rcl} x_1 & = & x_\omega := \mathrm{argmin}_X\, \omega; \\ x_{\tau+1} & = & \mathrm{Prox}_{x_\tau}(\gamma_\tau[g_\tau(x_\tau)+\Delta_\tau(x_\tau;\xi_\tau)]) \\ x^t & = & \sum_{\tau\le t}\lambda_\tau^t x_\tau,\ \lambda_\tau^t=\frac{\gamma_\tau}{\sum_{s\le t}\gamma_s} \end{array}$$
$$\|g_\tau(x)\|_*\le L,$$
$$\|\mathbf{E}_{\xi\sim P}\Delta_\tau(x;\xi)\|_*\le\mu$$
$$\mathbf{E}_{\xi\sim P}\{\|\Delta_\tau(x;\xi)\|_*^2\}\le\sigma^2$$

♣ **Main Property of SMD:** *One has*

$$\mathbf{E}\left\{\varepsilon_t:=\max_{z\in X}\sum_{\tau\le t}\lambda_\tau^t\langle g(x_\tau),x_\tau-z\rangle\right\}\le\frac{\Omega^2+[L^2+2\sigma^2]\sum_{\tau\le t}\gamma_\tau^2}{\sum_{\tau\le t}\gamma_\tau}+2\mu\Omega$$

- *In particular,* $\gamma_\tau=\Omega/\sqrt{[L^2+2\sigma^2]t},\ 1\le\tau\le t,$ *yields*

$$\mathbf{E}\{\varepsilon_t\}\le\Theta/\sqrt{t}+2\mu\Omega,\ \Theta=2\Omega\sqrt{L^2+2\sigma^2}.$$

- *Strengthening the bound on the second moment of* $\|\Delta_\tau\|_*$ *to*

$$\mathbf{E}\{\exp\{\|\Delta_\tau\|_*^2/\sigma^2\}\}\le\exp\{1\},$$

large deviation probabilities obey an exponential bound:

$$\forall\theta>0:\mathrm{Prob}\left\{\varepsilon_t>[\Theta+\theta\Sigma]/\sqrt{t}+2\mu\Omega\right\}\le O(1)\mathrm{e}^{-\theta}$$
$$\left[\Sigma=4\Omega\sigma\right]$$

♣ When $g_\tau(\cdot) \equiv g(\cdot)$ is associated with a problem with convex structure, e.g.,

A. $\min_{x\in X} f(x) \Rightarrow g(x) = f'(x)$, or

B. $\min_{u\in U} \max_{v\in V} f(u,v)$

$$\Rightarrow g(u,v) = [f'_u(u,v); -f'_v(u,v)],$$

the residual ε_t upper-bounds inaccuracy of the approximate solution x^t to the problem of interest.

$\Rightarrow$ *t-step SMD allows to solve stochastic convex problems with expected inaccuracy $O(1/\sqrt{t})$.* For example,

- in the case of A, we get
$$\mathbf{E}\{f(x^t) - \min_X f\} \le 2\Omega\sqrt{L^2+2\sigma^2}/\sqrt{t} + 2\mu\Omega$$
- in the case of B, we get
$$\mathbf{E}\{[\overline{f}(u^t) - \min_U \overline{f}] + [\max_V \underline{f} - \underline{f}(v^t)]\}$$
$$\le 2\Omega\sqrt{L^2+2\sigma^2}/\sqrt{t} + 2\mu\Omega.$$

♠ **Note:** In typical stochastic problems, *in every dimension*, not only a large one, $O(1/\sqrt{t})$ is the *best rate allowed by Statistics*.

Stochastic Mirror Descent: Illustration

♣ Consider *Binary Classification problem* where we draw from a distribution P *examples*

$$\xi_\tau = (\eta_\tau, y_\tau) \in \mathbb{R}^N \times \{\pm 1\}$$

and want to build a *linear classifier*

$$y \sim \mathrm{sign}(\langle x, \eta \rangle) \quad .$$

♠ The problem can be modeled as

$$\begin{array}{rcl} \mathrm{Opt}(\rho) & = & \min_{\|x\| \le 1} [p_\rho(x) = p(\rho x)] \\ p(x) & := & \mathbf{E}\{\max[1 - y\langle x, \eta \rangle, 0]\} \end{array}$$

$$\left[\begin{array}{l} p(x): \text{ convex upper bound on the} \\ \text{probability for } x \text{ to mis-classify} \end{array} \right]$$

• Let $\|\cdot\|$ be (a) $\|\cdot\|_2$, or (b) $\|\cdot\|_1$, or (c) nuclear norm on $\mathbb{R}^N = \mathbb{R}^{m \times n}$

♠ Assuming $\mathbf{E}\{\|\eta\|_*^2\} \le R^2 < \infty$ and setting

$$h(x; \eta, y) \equiv -\rho y \cdot \left\{ \begin{array}{l} 1, 1 - y\langle \rho x, \eta \rangle > 0 \\ 0, \text{otherwise} \end{array} \right\} \cdot \eta,$$

$$g(x) := \mathbf{E}_{\eta, y}\{h(x; \eta, y)\} \in p'_\rho(x)$$

we satisfy Standing Assumption with

$$X = \{\|x\| \le 1\}, \ L = \rho R, \ \sigma = 2\rho R, \ \mu = 0.$$

⇒ *For every $t \ge 1$, drawing a t-element sample from P and applying t-step SMD with appropriate proximal setup, we get a linear classifier ρx^t, $\|x^t\| \le 1$, such that*

$$\mathbf{E}\{p(\rho x^t)\} \le \mathrm{Opt}(\rho) + \rho R t^{-1/2} \times \left\{ \begin{array}{ll} O(1), & \text{case (a)} \\ O(1)\sqrt{\ln(N)}, & \text{cases (b), (c)} \end{array} \right.$$

Utilizing Problem's Structure: Mirror Prox

$$\mathrm{Opt} = \min_{x\in X} f(x) \qquad (P)$$

♣ *Unimprovable or not, convergence rate $O(1/\sqrt{t})$ is slow. When we can do better?*

• One can use *bundle* versions of MD re-utilizing past information. In practice, this improves the convergence pattern at the price of *controlled* increase in the computational cost of a step. *Theoretical complexity bounds, however, remain intact.*

• When f is smooth: $\|f'(x)-f'(x')\|_* \le \mathcal{M}\|x-x'\|$, the MD efficiency improves to

$$f(x^t) - \min_X f \le \Omega^2\mathcal{M}/t \qquad .$$

This is of no actual interest: with Nesterov's optimal method for smooth convex minimization one achieves unimprovable in the large-scale case efficiency $O(1)\Omega^2\mathcal{M}/t^2$.

• When f is *strongly convex*, properly modified MD converges at the rate $O(1/t)$.

• *For a wide spectrum of "well-structured" f, rate $O(1/t)$ can be achieved by smooth saddle point reformulation of (P).*

Extra-Gradient MD – Mirror Prox

♣ **Situation:** X is a convex compact subset of Euclidean space E, $\|\cdot\|, \omega(\cdot)$ is a proximal setup, $g(\cdot) : X \to E$ is a vector field represented by an oracle.

- At τ-th call, $x_\tau \in X$ being the query point, the oracle returns an estimate

$$h(x_\tau; \xi_\tau) = g(x_\tau) + \Delta(x_\tau; \xi_\tau)$$

of $g(x_\tau)$, ξ_τ are i.i.d.,

$$\|\mathbf{E}_\xi\{\Delta(x;\xi)\}\|_* \le \mu, \ \ \mathbf{E}_\xi\{\|\Delta(x;\xi)\|_*^2\} \le \sigma^2, \ \forall x \in X$$

- $g(\cdot)$ satisfies

$$\|g(x) - g(x')\|_* \le \mathcal{M}\|x - x'\| + L \ \ \forall (x, x' \in X)$$

• **Note:** $L = \sigma = \mu = 0 \Leftrightarrow g(\cdot)$ is *Lipschitz* & *precisely* observed.

♣ **Mirror Prox** is the recurrence

$$\boxed{\begin{array}{rcl} x_1 & = & x_\omega; \\ x_\tau & \mapsto & w_\tau = \mathrm{Prox}_{x_\tau}(\gamma_\tau h(x_\tau; \xi_{2\tau-1})) \\ & \mapsto & x_{\tau+1} = \mathrm{Prox}_{x_\tau}(\gamma_\tau h(w_\tau; \xi_{2\tau})) \\ x^t & = & \sum_{\tau \le t} \lambda_\tau^t w_\tau, \ \lambda_\tau^t = \frac{\gamma_\tau}{\sum_{s \le t} \gamma_s} \end{array}}$$

with deterministic stepsizes $\gamma_\tau > 0$.

- $(X \subset E,\ \|\cdot\|, \omega) \Rightarrow \Omega$
- $g(\cdot) : X \to E$: $\|g(x) - g(x')\|_* \le \mathcal{M}\|x - x'\| + L$
- oracle $x \mapsto h(x;\xi) = g(x) + \Delta(x;\xi)$:
  $\|\mathbf{E}_\xi\{\Delta(x;\xi)\}\|_* \le \mu,\ \ \mathbf{E}_\xi\{\|\Delta(x;\xi)\|_*^2\} \le \sigma^2$
- Mirror Prox:

$$\begin{array}{rcl} x_\tau & \mapsto & w_\tau = \mathrm{Prox}_{x_\tau}(\gamma_\tau h(x_\tau;\xi_{2\tau-1})) \\ & \mapsto & x_{\tau+1} = \mathrm{Prox}_{x_\tau}(\gamma_\tau h(w_\tau;\xi_{2\tau})) \\ x^t & = & \sum_{\tau\le t}\lambda_\tau^t w_\tau,\ \lambda_\tau^t = \gamma_\tau/\sum_{s\le t}\gamma_s \end{array}$$

♣ **Main Property of MP:** *Let* $0 < \gamma_\tau \le \frac{1}{2\mathcal{M}}$. *Then*

$$\mathbf{E}\left\{\varepsilon_t := \max_{z\in X}\sum_{\tau\le t}\lambda_\tau^t\langle g(x_\tau), x_\tau - z\rangle\right\} \le \frac{\Omega^2 + [3L^2+7\sigma^2]\sum_{\tau\le t}\gamma_\tau^2}{\sum_{\tau\le t}\gamma_\tau} + 2\mu\Omega$$

- In particular,

$$\gamma_\tau = \min\left[\frac{1}{2\mathcal{M}}, \frac{\Omega}{\sqrt{[3L^2+7\sigma^2]t}}\right],\ \tau \le t,$$

yields

$$\mathbf{E}\{\varepsilon_t\} \le \frac{2\Omega^2\mathcal{M}}{t} + \frac{\Theta}{\sqrt{t}} + 2\mu\Omega,\quad \Theta = 2\Omega\sqrt{3L^2+7\sigma^2}$$

♠ **Note:** *In the smooth deterministic case*

$$L = \sigma = \mu = 0,$$

we get $O(1/t)$ *convergence!*

- $X \subset E$, $\|\cdot\|, \omega \Rightarrow \Omega$
- $g(\cdot) : X \to E$: $\|g(x) - g(x')\|_* \le \mathcal{M}\|x - x'\| + L$
- oracle $x \mapsto h(x;\xi) = g(x) + \Delta(x;\xi)$:
  $\|\mathbf{E}_\xi\{\Delta(x;\xi)\}\|_* \le \mu, \ \mathbf{E}_\xi\{\|\Delta(x;\xi)\|_*^2\} \le \sigma^2$
- Mirror Prox:

$$\begin{array}{rcl} x_\tau & \mapsto & w_\tau = \mathrm{Prox}_{x_\tau}(\gamma_\tau h(x_\tau;\xi_{2\tau-1})) \\ & \mapsto & x_{\tau+1} = \mathrm{Prox}_{x_\tau}(\gamma_\tau h(w_\tau;\xi_{2\tau})) \\ x^t & = & \sum_{\tau\le t}\lambda_\tau^t w_\tau, \ \lambda_\tau^t = \gamma_\tau/\sum_{s\le t}\gamma_s \end{array}$$

♠ *Stepsizes*

$$\gamma_\tau = \min\left[\frac{1}{2\mathcal{M}}, \frac{\Omega}{\sqrt{[3L^2+7\sigma^2]t}}\right], \tau \le t,$$

yield

$$\mathbf{E}\{\varepsilon_t\} \le \frac{2\Omega^2\mathcal{M}}{t} + \frac{\Theta}{\sqrt{t}} + 2\mu\Omega,$$
$$\Theta = 2\Omega\sqrt{3L^2+7\sigma^2}$$

.

- Strengthening the bound on the second moment of $\|\Delta\|_*$ to $\mathbf{E}\{\exp\{\|\Delta\|_*^2/\sigma^2\}\} \le \exp\{1\}$, large deviation probabilities obey an exponential bound:

$$\forall\theta > 0 : \mathrm{Prob}\left\{\varepsilon_t > \frac{2\Omega^2\mathcal{M}}{t} + \frac{\Theta+\theta\Sigma}{\sqrt{t}} + 2\mu\Omega\right\} \le O(1)\mathrm{e}^{-\theta}$$

$$\left[\Sigma = 9\Omega\sigma\right]$$

Application: $O(1/t)$ Nonsmooth Convex Minimization

$$\mathrm{Opt}(P) = \min_{u \in U} f(u) \qquad (P)$$

♣ **Corollary:** *Let (P) be a convex program with compact $U \subset E_U$ and with f such that*

$$f(u) = \max_{v \in V} \phi(u, v)$$

- V: compact convex subset of Euclidean space E_v
- $\phi(u, v)$: convex-concave with *Lipschitz continuous* gradient

so that (P) is the primal form of the saddle point problem

$$\min_{u \in U} \max_{v \in V} \phi(u, v) \qquad (SP)$$

The vector field $g(u, v) = [\phi'_u(u, v); -\phi'_v(u, v)]$ associated with (SP) is Lipschitz continuous. Equipping

- *$E := E_U \times E_V, X := U \times V$ — with a proximal setup $\|\cdot\|, \omega$,*
- *$g(\cdot)$ — with a precise deterministic oracle,*

t-step MP yields $(u^t, v^t) \in U \times V$ such that

$$f(u^t) - \mathrm{Opt}(P) \le O(1)\Omega\mathcal{M}/t$$

$$\mathcal{M} = \min\{M : \|g(x) - g(x')\|_* \le M\|x - x'\| \ \forall (x, x' \in X)\}$$

$$\min_{u\in U}\,[f(u) = \max_{v\in V}\,\phi(u,v)]$$

♣ **Fact:** *If $\phi(u,v)$ is*

- *convex-concave with Lipschitz continuous gradient,*
- *affine in u,*
- *strongly concave in v,*

then properly modified MP ensures $O(1/t^2)$ convergence rate.

♠ **Note:** The premise does *not* imply smoothness of f.

Smooth and Bilinear Saddle Point Representations

♣ **Fact:** *Representations* $f(u) = \max_{v\in V}\phi(u,v)$ *with smooth convex-concave, and even with bilinear* ϕ *are available for wide spectrum of convex functions* f. *Whenever it is the case,* f *can be minimized via MP at the rate* $O(1/t)$.

Examples:

A. $f(u) = \max\limits_{k\le K} f_k(u)$ with smooth convex f_k

$$\Rightarrow f(u) = \max_{v\ge 0, \sum_k v_k = 1} \sum_k v_k f_k(u)$$

B. $f(u) = \|Au - b\|$

$$\Rightarrow f(u) = \max_{\|v\|_* \le 1} \langle v, Ay - b\rangle$$

C. $f(u) = \|y\| + \frac{1}{2}\|Au - b\|_2^2$

$$\Rightarrow f(u) = \max_{\|v\|_*\le 1, w} \left[\langle u, v\rangle + \langle w, Au - b\rangle - \frac{1}{2}w^T w\right]$$

D. $f(u)$: sum of k largest eigenvalues of $\overbrace{Au - b}^{\mathcal{A}(u)} \in \mathbf{S}^n$

$$\Rightarrow f(u) = \max_v [\mathrm{Tr}(v\mathcal{A}(u)) : 0 \preceq v \preceq I_n, \mathrm{Tr}(v) = k]$$

E. $f(u) = \inf\limits_{b\in\mathbb{R}} \left[\frac{1}{N}\sum\limits_{i=1}^N \max[1 - y_i(\langle u, \eta_i\rangle + b), 0]\right]$

$$\Rightarrow f(u) = \max_{v\in V} \sum_{i=1}^N v_i[1 - y_i\langle u, \eta_i\rangle]$$

$$V = \{v : 0 \le v_i \le \tfrac{1}{N}\,\forall i, \sum_i y_i v_i = 0\} \subset \{v \in \mathbb{R}^N : \|v\|_1 \le 1\}$$

$O(1/t)$ Nonsmooth Convex Minimization: Comments

$$\mathrm{Opt}(P) = \min_{u \in U} f(u) \qquad (P)$$

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

Accelerating algorithms by utilizing problem's structure is an old and still challenging goal.

• A common way to utilize structure is via "structure-revealing" *conic* formulations ($\mathcal{LP}/\mathcal{CQP}/\mathcal{SDP}$) and Interior Point Methods. However, *in the large scale case IPM iteration can become prohibitively costly*.

• Utilizing structure *within the realm of oracle-oriented methods with computationally cheap iterations* is due to Nesterov (2003).

Nesterov's Smoothing (2003) uses saddle point representation of a nonsmooth f to approximate f by a *smooth* function which is further minimized by Nesterov's algorithm for smooth convex minimization. The resulting convergence rate is $O(1/t)$.

• MP offers another way to utilize saddle point representation to achieve the same $O(1/t)$ rate.

"Practical scopes" of these two approaches are nearly identical.

$O(1/t)$ Nonsmooth Convex Minimization: Examples

♣ **Problem of interest:**

$$\mathrm{Opt}(P) = \min_{\|u\|\le 1} \|Au - b\|_p, \ A : M \times N \qquad (P)$$

where $p = 2$ or $p = \infty$, and $\|\cdot\|$ is

(a) $\|\cdot\|_2$ on $\mathbb{R}^N$, or

(b) $\|\cdot\|_1$ on $\mathbb{R}^N$, or

(c) nuclear norm on $\mathbb{R}^N = \mathbb{R}^{m\times n}$, $m \le n$

♠ Bilinear saddle point reformulation is

$$\mathrm{SadVal} = \min_{u\in U} \max_{v\in V} \langle v, Au - b\rangle$$

$$U = \{\|u\| \le 1\}, V = \{\|v\|_q \le 1\}, q = \tfrac{p}{p-1} \in \{1, 2\}$$

and its domain is the product of two favorable geometry atoms.

♠ Applying t-step MP with appropriate setup, we get u^t with $\|u^t\| \le 1$ and

$$f(u^t) - \mathrm{Opt}(P) \le \kappa \|A\|_{\|\cdot\|,p}/t$$

$$\|A\|_{\|\cdot\|,p} = \max\{\|Au\|_p : \|u\| \le 1\}$$

$$\kappa = O(1)\ln^{\frac{1}{2}-\frac{1}{p}}(M+1) \times \begin{cases} 1, & \text{case (a)} \\ \sqrt{\ln(N+1)}, & \text{case (b)} \\ \sqrt{\ln(m+1)}, & \text{case (c)} \end{cases}$$

$$\mathrm{Opt}(P) = \min_{\|u\|\le 1} \|Au-b\|_p,\ A : M\times N,\ p\in\{2,\infty\} \qquad (P)$$

$\|\cdot\|$: (a) $\|\cdot\|_2$ on $\mathbb{R}^N$ | (b) $\|\cdot\|_1$ on $\mathbb{R}^N$ | (c) nuclear norm on $\mathbb{R}^N = \mathbb{R}^{m\times n}$

$$\Rightarrow \boxed{f(u^t) - \mathrm{Opt}(P) \le O(1)\ln(MN)\|A\|_{\|\cdot\|,p}/t}$$

♠ MP step reduces to computing $O(1)$ matrix-vector products involving A and A^*, plus

— $O(M+N)$ a.o. in cases (a), (b)

— computing svd's of two $m\times n$ matrices in case (c).

$\Rightarrow$ Except for case (c), *MP is computationally cheap...*

♠ **Note:** When solving a *Least Squares* problem

$$\text{(LS)}\quad \mathrm{Opt}(A,b) = \min_{\|u\|_2\le 1} \|Au-b\|_2 \qquad [A : n\times n]$$

with A represented by *multiplication oracle*

$$u, u' \mapsto Au, A^T u' \quad ,$$

the rate $O(1/t)$ is unimprovable in the large-scale case: *the worst-case, over (A,b) with $\|A\|_{2,2}\le 1$ and $\mathrm{Opt}(A,b)=0$, inaccuracy in terms of the objective of* (LS) *is, for every t-step algorithm, at least $O(1)/t$, provided $t\le n/4$.*

Acceleration by Randomization

♣ **Problem of interest:**

$$\mathrm{Opt} = \min_{\|u\|_1 \le 1} \|Au - b\|_p \qquad [A : m \times n,\ p \in \{2, \infty\}]$$

$$\Leftrightarrow (\ell_1): \quad \min_{\|u\|_1 \le 1} \max_{\|v\|_{p/(p-1)} \le 1} \langle v, Au - b \rangle$$

$$\Rightarrow \quad g(u, v) = [A^T v; b - Au] : X := U \times V \to \mathbb{R}^{m+n}$$

$$U = \{u : \|u\|_1 \le 1\},\ V = \{v : \|v\|_{p/(p-1)} \le 1\}.$$

♠ *Omitting from now on logarithmic in m, n factors,* MP solves (ℓ_1) within accuracy ε in

$$N(\varepsilon) = \|A\|_{1,p}/\varepsilon, \quad \|A\|_{1,p} = \max_{j \le n} \|\mathrm{Col}_j[A]\|_p$$

steps, with two multiplications of vectors from U and from V by A, A^T, plus $O(m + n)$ a.o. "overhead," per step.

⇒ *The arithmetic cost of ε-solution for a general-type A is*

$$\mathcal{C}_d(\varepsilon) = mn\|A\|_{1,p}/\varepsilon \text{ a.o.}$$

In fact, this is the best operation count achievable in the large-scale case with known so far *deterministic* algorithms.

• *For large m, n, matrix-vector multiplications can become too time consuming...*

♠ Matrix-vector multiplications are easy to randomize:

In order to compute Bu, $B \in \mathbb{R}^{M\times N}$, *we draw an index* $\jmath$ *at random according to*

$$\mathsf{Prob}\{\jmath = j\} = \mathsf{sign}(u_j)/\|u\|_1,\ 1 \le j \le N$$

and return the vector

$$h = \|u\|_1 \mathsf{sign}(u_\jmath)\mathsf{Col}_\jmath[B]$$

Note:

- $\mathbf{E}\{h\} = Bu$, $\|h\|_q \le \|u\|_1\|B\|_{1,q}$
- Generating h costs $O(1)(M+N)$ a.o. (assuming cost $O(1)$ of computing/extracting individual entry of B).

$$\mathrm{Opt} = \min_{\|u\|_1 \le 1} \|Au - b\|_p \qquad [A : m \times n]$$

$$\Leftrightarrow \quad (\ell_1) : \min_{\|u\|_1 \le 1} \max_{\|v\|_{p/(p-1)} \le 1} \langle v, Au - b \rangle$$

$$\Rightarrow \quad g(u, v) = [A^T v; b - Au] : X := U \times V \to \mathbb{R}^{m+n}$$
$$U = \{u : \|u\|_1 \le 1\},\ V = \{v : \|v\|_{p/(p-1)} \le 1\}.$$

♠ *When solving* (ℓ_1) *with* $p = \infty$ *by MP with the precise values of* $g(\cdot)$ *replaced with their cheap unbiased random estimates, we* $(1 - \delta)$*-reliably get* ε*-solution to* (ℓ_1) *in* $\mathsf{ln}(1/\delta) \left[\|A\|_{1,\infty}/\varepsilon\right]^2$ *steps, the total computational effort being*

$$\mathcal{C}_r = (m + n)\,\mathsf{ln}(1/\delta) \left[\|A\|_{1,\infty}/\varepsilon\right]^2 \text{ a.o.}$$

♠ The "deterministic" operation count is

$$\mathcal{C}_d = mn\|A\|_{1,\infty}/\varepsilon \qquad .$$

⇒ *With the relative accuracy* $\varepsilon/\|A\|_{1,\infty}$ *and* δ *fixed and* m, n *large, randomized algorithm by far outperforms its deterministic competitors.*

• In addition, *Randomized MP exhibits sublinear time behavior: when* m, n *are large,* ε*-solution is obtained, in a* $(1 - \delta)$*-reliable fashion, by inspecting negligibly small fraction of the* mn *data entries.*

♠ In the case of $p = \infty$, our construction basically recovers the *ad hoc* sublinear time algorithm for matrix games (Grigoriadis & Khachiyan, 1994).

♠ *In the case of* $p = 2$, randomization leads to iteration count

$$\ln(1/\delta)[\|A\|_{1,2}/\varepsilon]^2\Gamma^2[A],$$
$$\Gamma(A) = \sqrt{m}\|A\|_{1,\infty}/\|A\|_{1,2} \in [1, \sqrt{m}]$$

and operation count

$$\mathcal{C}_r = (m+n)\ln(1/\delta)[\|A\|_{1,2}/\varepsilon]^2\Gamma^2[A] \text{ a.o.}$$

vs. the "deterministic" operation count

$$\mathcal{C}_d = mn[\|A\|_{1,2}/\varepsilon] \text{ a.o.}$$

- with $\Gamma[A]$ like $O(1)\ln(mn)$, everything is as when $p = \infty$
- with $\Gamma[A]$ as large as $O(\sqrt{m})$, randomization is really bad.

♠ **However:** Preprocessing

$$[A, b] \Rightarrow [\bar{A}, \bar{b}] = \mathbf{F}\mathrm{Diag}\{\chi\}[A, b]$$

with $m \times m$ DFT matrix $\mathbf{F}$ and $\chi \sim \mathrm{Uniform}(\{-1; 1\}^m)$ yields *equivalent* problem and ensures $(1-\delta)$-reliably

$$\Gamma[\bar{A}] \leq \sqrt{\ln(mn/\delta)}.$$

⇒ *With randomization and preprocessing, the operation count is*

$$\mathcal{C}_r = mn + (m+n)\ln^2(1/\delta)[\|A\|_{1,2}/\varepsilon]^2$$

which for small and fixed $\varepsilon/\|A\|_{1,2}$ *and large* m, n *is negligibly small as compared to*

$$\mathcal{C}_d = mn[\|A\|_{1,2}/\varepsilon] \text{ a.o.}$$

How it Works: Policeman vs. Burglar

♣ **Problem:** There are n houses in a city, i-th with wealth w_i. Every evening, *Burglar* chooses a house i to be attacked, and Policeman chooses his post near a house j. After burglary starts, Policeman becomes aware where it happens, and his probability to catch Burglar is

$$\exp\{-\theta \mathrm{dist}(i,j)\},$$

$\mathrm{dist}(i,j)$: distance between houses i and j

Burglar seeks to maximize his expected profit

$$w_i(1-\exp\{-\theta \mathrm{dist}(i,j)\}),$$

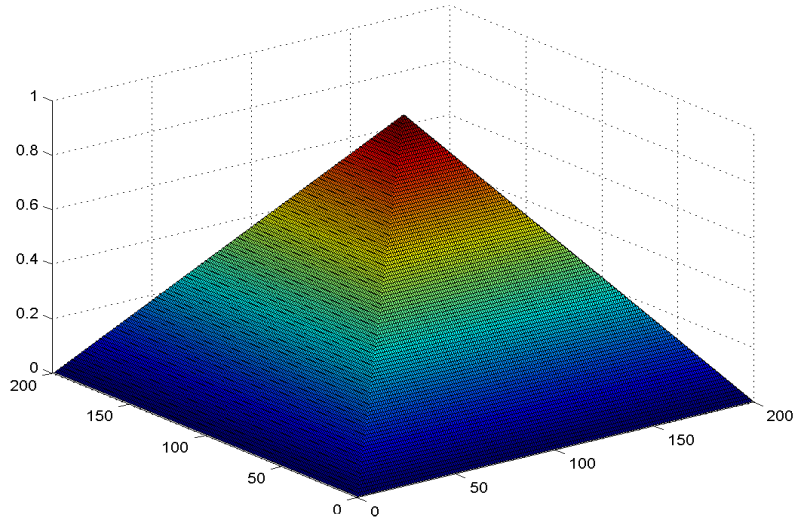
the interest of Policeman is completely opposite.

• *What are the optimal mixed strategies of Burglar and Policeman?*

♠ **Equivalently:** *Solve the matrix game*

$$\min_{u\geq 0,\ \sum_{j=1}^n u_j=1} \ \max_{v\geq 0,\ \sum_{i=1}^n v_i=1} \ \phi(u,v) := v^T A u$$

$$A_{ij} = w_i(1-\exp\{-\theta \mathrm{dist}(i,j)\})$$



Wealth on $n \times n$ square grid of houses

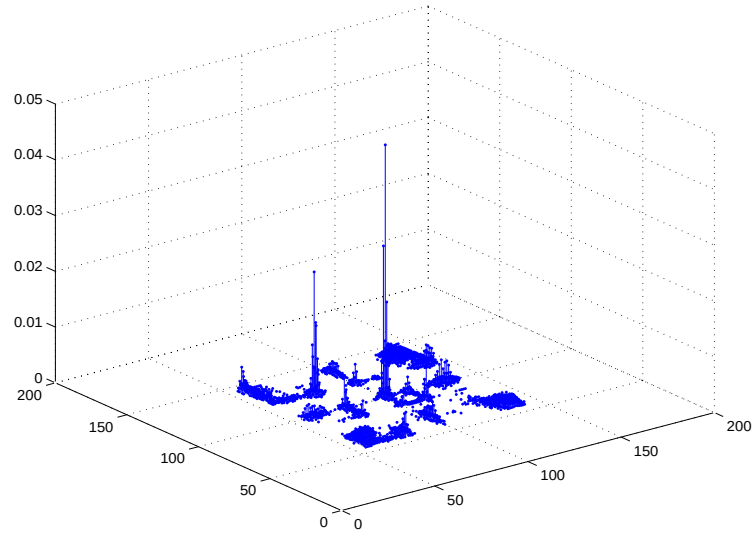
		$N = 1600$	$N = 6400$	$N = 14400$	$N = 40000$
IPM	Steps	21	21	—	—
	CPU, sec	120	6930	not tested	out of memory
	$\mathcal{E}$	6.0e-9	1.1e-8	—	—
MP	Steps	78	80	95	15†
	CPU, sec	6	31	191	5533†
	$\mathcal{E}$	1.0e-3	1.0e-3	1.0e-3	0.022†
Rand MP	Steps	10556	10408	9422	10216
	CPU, sec	264	796	1584	4931
	$\mathcal{E}$	1.0e-3	1.0e-3	1.0e-3	1.0e-3

Policeman vs. Burglar, N houses

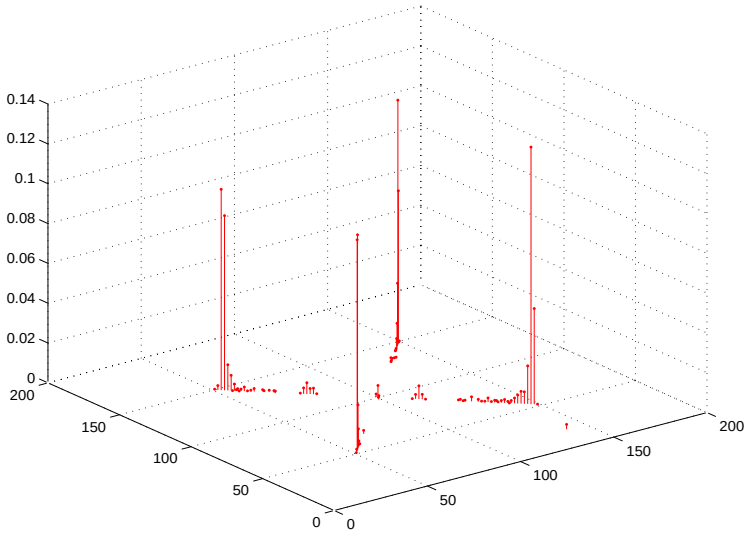
Target residual $\varepsilon_t \leq$ 1.e-3 IPM: `mosekopt`

†: *termination when reaching the CPU limit of 5,400 sec*

Platform: 2×2.67 GHz CPU with 8.0 GB RAM and 64-bit operating system

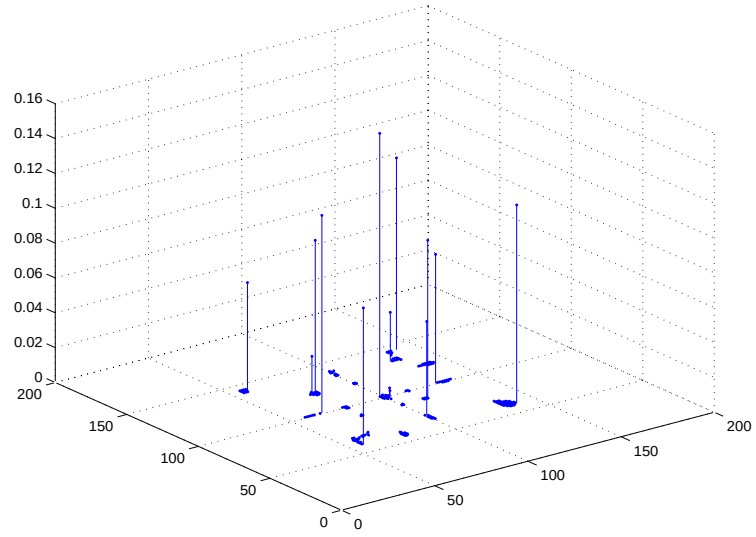


Policeman

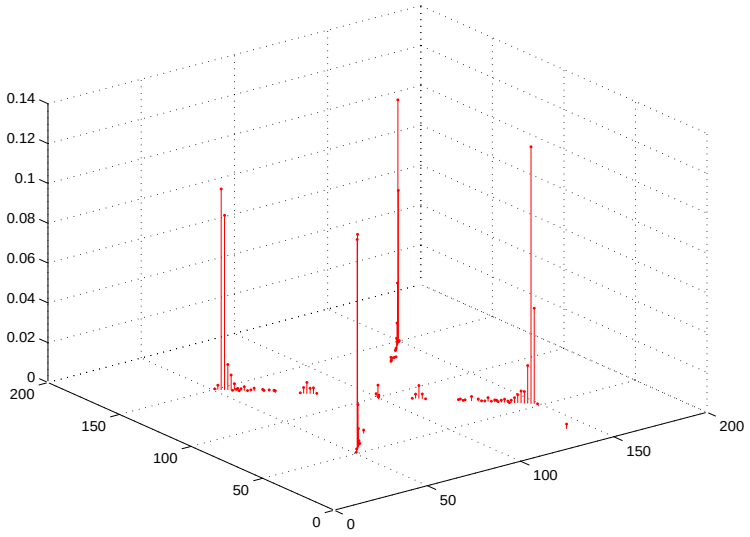


Burglar

♠ The resulting *highly sparse* near-optimal solution can be refined by further optimizing it *on its support* by an interior point method. This reduces inaccuracy from 0.0008 to 0.0005 in just $39'$.



Policeman, refined



Burglar, refined

200×200 grid of houses

Lecture IV.

Smooth Large-Scale Convex Minimization

- Nesterov's Optimal algorithm for Smooth and Composite Minimization
- Beyond the scope of proximal algorithms: Conditional Gradients
  - Linear Minimization Oracle
  - CndG algorithm
  - CndG algorithm for Norm-Regularized Smooth Convex Minimization

Nesterov's Fast Gradient method for Smooth and Composite Minimization

♣ Problem of interest: Composite minimization

$$\mathrm{Opt} = \min_{x \in X} \{\phi(x) = \Psi(x) + f(x)\}$$

- X: closed convex nonempty subset in Euclidean space E (X, E) *is equipped with proximal setup* $(\omega(\cdot), \|\cdot\|)$
- $\Psi : X \to \mathbb{R} \cup \{+\infty\}$: convex lower semicontinuous function which is finite on the relative interior of X
- $f : X \to \mathbb{R}$: 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

$$\min_{x \in X} \{\omega(x) + \langle h, x \rangle + \alpha \Psi(x)\} \qquad [\alpha \geq 0]$$

♡ **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 norm $\sum_{j=1}^n \|x^j\|_2$ on $E=\mathbb{R}^{k_1}\times ...\times \mathbb{R}^{k_n}$ (ℓ_1 *case*)
  - (b) nuclear norm on the space E of block diagonal matrices of a given block diagonal structure (*nuclear norm case*)
- $A(\cdot): E\to\mathbb{R}^m$: linear mapping
- X: either the unit $\|\cdot\|_E$-ball, or the entire E

♡ Main Assumption is satisfied, provided that the proximal setup in use is

- Euclidean setup (cases (a) and (b)), or
- ℓ_1/ℓ_2 setup (case (a)), or
- nuclear norm setup (case (b)).

Here *computing composite prox mapping*

$$\min_{x\in X}\{\omega(x)+\langle h,x\rangle+\alpha\Psi(x)\}\qquad[\alpha\geq 0]$$

takes $O(\dim E)$ *a.o. in the case of (a) and reduces to finding svd of a matrix from* E *in the case of (b).*

Nesterov's Fast Gradient algorithm for Composite Minimization

♣ **Problem:**

$$\mathrm{Opt} = \min_{x \in X \subset E} \{\phi(x) := \Psi(x) + f(x)\} \qquad (CP)$$

- Ψ, f: convex and

$$\forall x, y \in X : \|\nabla f(x) - \nabla f(y)\|_* \leq L_f \|x - y\|$$

♠ **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*

$$V_x(y) = \omega(y) - \omega(x) - \langle \omega'(x), y - x \rangle \ \ [x \in X^o, y \in X]$$

$$\mathrm{Opt} = \min_{x \in X \subset E} \{\phi(x) := \Psi(x) + f(x)\}$$

♣ **Algorithm:**

♠ **Initialization:** Set

$$A_0 = 0,\ y_0 = x_\omega = \mathrm{argmin}_X\, \omega,\ \psi_0(x) = V_{x_\omega}(x)$$

and select $y_0^+ \in X$ such that $\phi(y_0^+) \le \phi(y_0)$.

♠ **Step** $t = 0, 1, 2, ...$**:** Given

$$\psi_t(\cdot) = \omega(\cdot) + \alpha\Psi(\cdot) + \text{<affine form>} \quad [\alpha \ge 0],$$

$y_t^+ \in X$ and L_t, $0 < L_t \le L_f$,

• Compute $z_t = \mathrm{argmin}_{x \in X}\, \psi_t(x)$ (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

$$\widehat{x}_{t+1} = \mathrm{argmin}_{x \in X} \left\{ \langle \nabla f(x_{t+1}), x \rangle + \Psi(x) + \frac{1}{a_{t+1}} V_{z_t}(x) \right\}$$

(reduces to computing composite prox-mapping)

• Set

$$\begin{aligned} y_{t+1} &= \tau_t \widehat{x}_{t+1} + (1 - \tau_t) y_t^+ \\ \psi_{t+1}(x) &= \psi_t(x) \\ &\quad + a_{t+1} \left[f(x_{t+1}) + \langle \nabla f(x_{t+1}), x - x_{t+1} \rangle + \Psi(x) \right] \end{aligned}$$

Step t is completed; go to step $t + 1$.

♣ **Theorem** [Yu. Nesterov '83, '07] *Assume that* $\{L_t \in (0, L_f]\}$ *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*

$$\boxed{\phi(y_t^+) \leq A_t^{-1} V_{x_\omega}(x_*) \leq \frac{4L_f}{t^2} V_{x_\omega}(x_*), \ t = 1, 2, ...}$$

♠ **Illustration:** *As applied to a solvable LASSO problem*

$$x_* = \mathrm{argmin}_x \left\{ \phi(x) := \lambda \|x\|_E + \frac{1}{2}\|A(x) - b\|_2^2 \right\}$$

with $\|\cdot\|_E$ *either block* ℓ_1 *norm on* $E = \mathbb{R}^{k_1} \times ... \times \mathbb{R}^{k_n}$*, or nuclear norm on* $E = \mathbb{R}^{p \times q}$ *with* $n = \min[p, q]$*, the Fast Gradient method in* $t = 1, 2, ...$ *steps ensures*

$$\phi(y_t^+) \leq \mathrm{Opt} + \begin{cases} O(1) \dfrac{\|A\|^2_{\|\cdot\|_2,2}}{t^2} \|x_*\|_2^2, & \text{Euclidean proximal setup} \\ O(\ln(n)) \dfrac{\|A\|^2_{\|\cdot\|_E,2}}{t^2} \|x_*\|_E^2, & \ell_1/\ell_2 \text{ or nuclear norm proximal setup} \end{cases}$$

where $\|A\|_{\|\cdot\|,2} = \max\{\|A(x)\|_2 : \|x\| \leq 1\}$

♣ **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*

$$\mathrm{Opt} = \min_x \|Ax - b\|_2^2 \qquad (QP)$$

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,*
- *$\mathrm{Opt} = 0$, 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*

$$\|Ay - b\|_2^2 \geq O(1)\frac{L^2R^2}{t^2}$$

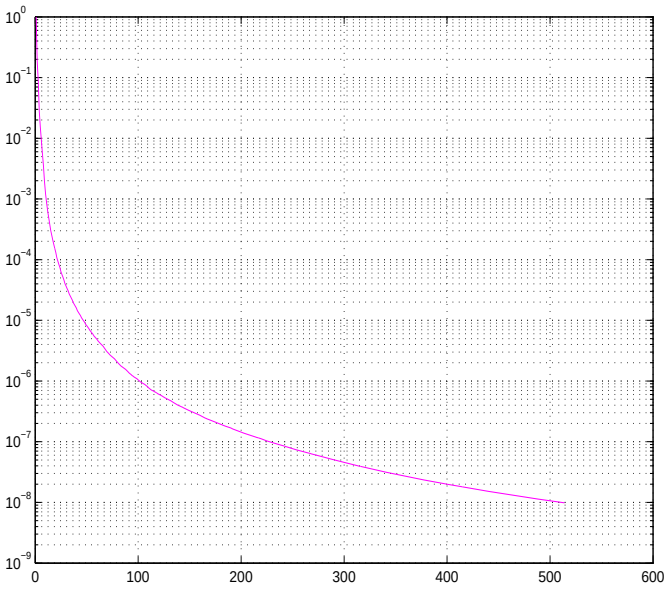
How it Works:
Fast Composite Minimization for LASSO

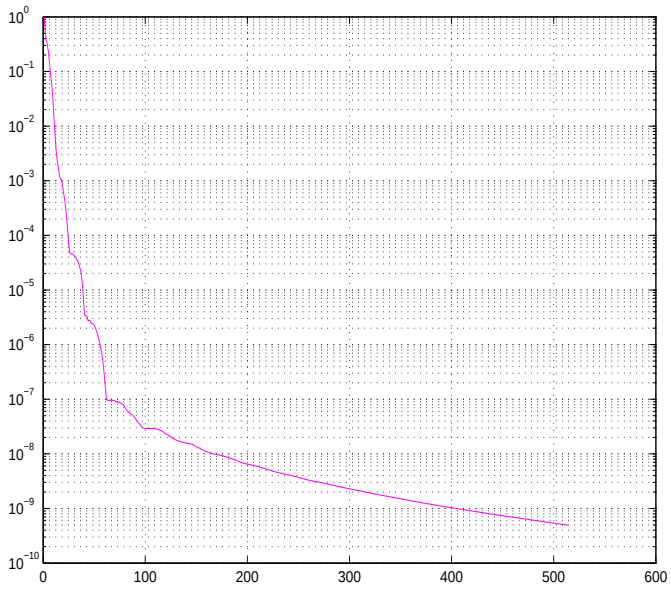
♣ Test problem:

$$\mathrm{Opt} = \min_x \left\{ \phi(x) := 0.01\|x\|_1 + \tfrac{1}{2}\|Ax - b\|_2^2 \right\}$$

with 4096×2048 randomly generated matrix A.

Method	Setup	Iterations	CPU, sec	Nonoptimality
IPM	—	11	103.1	<1.e-12
FGr	Euclidean	512	36.3	2.4e-6
FGr	ℓ_1	512	36.5	1.2e-7





Euclidean setup ℓ_1 setup

Progress in accuracy $\frac{\phi(y_t^+) - \mathrm{Opt}}{\phi(y_0^+) - \mathrm{Opt}}$ vs. t

Platform: 2×3.40 GHz CPU, 16.0 GB RAM, 64-bit Windows 7

Beyond the Scope of Proximal Algorithms: Conditional Gradients

$$\mathrm{Opt} = \min_{x \in X} f(x)$$

♣ **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 (in order to have a reasonable iteration count in the large scale case) *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 $\mathbb{R}^{n\times n}$ or a spectahedron 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*

$$\min_{x\in\mathbb{R}^{m\times n}}\left\{\lambda\cdot \mathrm{TV}(x)+\frac{1}{2}\|A(x)-b\|_2^2\right\},$$

where $x=[x_{ij}]\in\mathbb{R}^{m\times n}$ is an $(m\times n)$-pixel image, and $\mathrm{TV}(x)$ is the *Total Variation*:

$$\mathrm{TV}(x)=\sum_{i=1}^{m-1}\sum_{j=1}^{n}|x_{i+1,j}-x_{i,j}|+\sum_{i=1}^{m}\sum_{j=1}^{n-1}|x_{i,j+1}-x_{i,j}|$$

— 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*

$$x[\eta] \in \mathrm{Argmin}_{x \in X} \langle \eta, x \rangle$$

Indeed, the optimization program

$$\min_{x \in X} \langle \eta, x \rangle$$

is the "limiting case," as $\theta \to +0$, 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 $X = \{x \in \mathbb{R}^{m\times n} : \|x\|_{\rm nuc} \leq 1\}$. Here computing $x[\eta]$ reduces to finding the left and the right *leading* singular vectors of $\eta \in \mathbb{R}^{m\times n}$, 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 the full svd of η required when computing prox-mapping.

B: Spectahedron $X = \{x \in \mathbf{S}^n : x \geq 0, {\rm Tr}(x) = 1\}$. Here computing $x[\eta]$ reduces to finding the leading eigenvector of $-\eta$, i.e., to solving the problem

$$\min_{\|u\|_2=1} u^T\eta u.$$

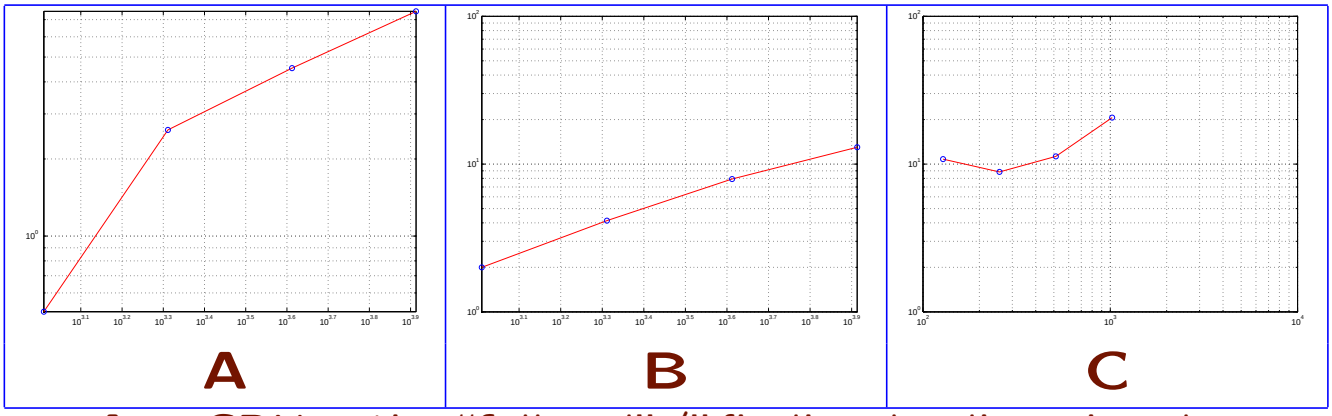
For large n, this is incomparably easier than the full eigenvalue decomposition of η required when computing prox-mapping.

C: Unit TV-ball $X = \{x \in \mathbf{M}_0^{m,n} : {\rm TV}(x) \leq 1\}$: *For $\eta \in \mathbf{M}_0^{m,n}$, a point $x[\eta] \in {\rm Argmin}_{x\in X} {\rm Tr}(\eta x^T)$ 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 $(i, j) \to (i + 1, j)$, $(i, j) \to (i, j + 1)$

♠ 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 $\mathbf{M}_0^{n,n}$ 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

$$\mathrm{Opt} = \min_{x\in X} f(x)$$

[• $X \subset E$: convex compact set • $f : X \to R$: convex]

(CM)

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

$$\begin{array}{rcl} x_1 & \in & X \\ x_{t+1} & \in & X : f(x_{t+1}) \le f\left(x_t + \frac{2}{t+1}(x_t^+ - x_t)\right), \\ & & x_t^+ = x[\nabla f(x_t)] \in \mathrm{Argmin}_{y\in X}\langle \nabla f(x), y\rangle \\ f_*^t & = & \max_{\tau\le t}\left[f(x_\tau) + \langle \nabla f(x_\tau), x_\tau^+ - x_\tau\rangle\right] \le \mathrm{Opt} \end{array}$$

♠ **Theorem:** *Let $f : X \to \mathbb{R}$ be convex and (κ, L)-smooth:*

$$\forall x, y \in X : \\ f(y) \le f(x) + \langle \nabla f(x), y - x\rangle + \frac{L}{\kappa}\|x - y\|_X^\kappa$$

[• $L < \infty$, $\kappa \in (1, 2]$: *parameters*
• $\|\cdot\|_X$: *norm with the unit ball* $\frac{1}{2}[X - X]$]

When solving (CP) by CndG, one has for $t = 2, 3, ...$

$$f(x_t) - \mathrm{Opt} \le f(x_t) - f_t^* \le \frac{2^{2\kappa}}{\kappa(3-\kappa)} \cdot \frac{L}{(t+1)^{\kappa-1}}$$

$$\forall x, y \in X:$$
$$f(y) \leq f(x) + \langle \nabla f(x), y - x \rangle + \frac{L}{\kappa}\|x - y\|^{\kappa} \qquad (!)$$

[• $L < \infty$, $\kappa \in (1, 2]$: parameters]

Note: When X is convex, a *sufficient* condition for (!) is Hölder continuity of $\nabla f(x)$:

$$\|\nabla f(x) - \nabla f(y)\|_* \leq L\|x - y\|^{\kappa - 1} \ \forall x, y \in X$$

For convex f and $\kappa = 2$, this condition is also *necessary* for (!).

Example: Minimization over a Box

♣ 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 $\mathbb{R}^n$ with $p = 1$ or $p = 2$ (in fact, with $1 \le p \le 2$), or
- the unit nuclear norm ball in $\mathbb{R}^{n\times n}$,

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:*

$$X = \{x \in \mathbb{R}^n : \|x\|_\infty \le 1\}.$$

Then for every $t \le n$, $L < \infty$, $\kappa \in (1, 2]$, 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

$$f(x_{\mathcal{B}}) - \min_X f \ge \frac{O(1)}{\mathsf{ln}(n)} \frac{L}{t^{\kappa-1}}$$

$\Rightarrow$ *When minimizing smooth convex functions, represented by a local oracle, over an n-dimensional box, t-step CndG cannot be accelerated by more than $O(\mathsf{ln}(n))$ factor*, provided $t \le n$.

- The result remains true when replacing n-dimensional box X with its matrix analogy

$$\{x \in \mathbb{R}^{n\times n} : \text{ spectral norm of } x \text{ is } \le 1\}$$

- When minimizing (κ, L)-smooth functions over n-dimensional $\|\cdot\|_p$-balls with $2 \le p \le \infty$, the rate-of-convergence advantages of proximal algorithms over CndG rapidly deteriorate as p grows and disappears (up to $O(\mathsf{ln}(n))$-factor) when p becomes as large as $O(\mathsf{ln}(n))$.

Proof of Theorem

$$(a) \quad f(y) \le f(x) + \langle \nabla f(x), y - x \rangle + \frac{L}{\kappa} \|y - x\|_X^\kappa$$

$$(b) \quad f(x_{t+1}) \le f(x_t + \gamma_t(x_t^+ - x_t)),$$

$$\gamma_t = \frac{2}{t+1}, \; x_t^+ \in \mathrm{Argmin}_{y \in X} \langle \nabla f(x_t), y \rangle$$

$$f_*^t := \underbrace{\max_{\tau \le t} \left[f(x_\tau) + \langle \nabla f(x_\tau), x_\tau^+ - x_\tau \rangle \right]}_{\le \min_X f}$$

$$? \Rightarrow ? \qquad f(x_t) - f_*^t \le \frac{2^{\kappa+1} L}{\kappa(3-\kappa)} \gamma_t^{\kappa-1} \quad (!_t), t \ge 2$$

Let

$$\epsilon_t = f(x_t) - f_*^t, \; e_t = \langle \nabla f(x_t), x_t - x_t^+ \rangle$$

- $f_*^t \ge f(x_t) + \langle \nabla f(x_t), x_t^+ - x_t \rangle \Rightarrow e_t \ge \epsilon_t$

We have

$$(c) \quad \|x_t - x_t^+\|_X \le 2$$

$$\begin{aligned}
\Rightarrow \quad f(x_{t+1}) &\le f(x_t + \gamma_t(x_t^+ - x_t)) \text{ [by } (b)\text{]} \\
&\le f(x_t) + \gamma_t \langle \nabla f(x_t), x_t^+ - x_t \rangle + \frac{L}{\kappa} [2\gamma_t]^\kappa \\
&\qquad \text{[by } (a), \; (c)\text{]} \\
&= f(x_t) - \gamma_t e_t + \frac{2^\kappa L}{\kappa} \gamma_t^\kappa \\
&\le f(x_t) - \gamma_t \epsilon_t + \frac{2^\kappa L}{\kappa} \gamma_t^\kappa \text{ [since } e_t \ge \epsilon_t\text{]}
\end{aligned}$$

$$\begin{aligned}
\Rightarrow \quad \epsilon_{t+1} = f(x_{t+1}) - f_*^{t+1} &\le f(x_{t+1}) - f_*^t \\
&\qquad \text{[since } f_*^{t+1} \ge f_*^t\text{]} \\
&\le \epsilon_t(1 - \gamma_t) + \frac{2^\kappa L}{\kappa} \gamma_t^\kappa
\end{aligned}$$

$$[0 \le] \quad \epsilon_{t+1} \le \epsilon_t(1-\gamma_t) + \frac{2^\kappa L}{\kappa}\gamma_t^\kappa \quad (*_t)$$

$$? \Rightarrow ? \quad \epsilon_t \le \frac{2^{\kappa+1}L}{\kappa(3-\kappa)}\gamma_t^{\kappa-1},\ t \ge 2 \quad [\gamma_t = \frac{2}{t+1}] \quad (!_t)$$

- By $(*_2)$, we have $\epsilon_2 \le \frac{2^\kappa L}{\kappa} \Rightarrow \epsilon_2 \le \frac{2^{\kappa+1}L}{\kappa(3-\kappa)}(2/3)^{\kappa-1}$ due to $1 < \kappa \le 2 \Rightarrow (!_2)$ *holds true.*
- Assuming $(!_t)$ true for some $t \ge 2$, we have

  $\epsilon_{t+1} \le \frac{2^{\kappa+1}L}{\kappa(3-\kappa)}\gamma_t^{\kappa-1}(1-\gamma_t) + \frac{2^\kappa L}{\kappa}\gamma_t^\kappa$ [by $(*_t)$ and $(!_t)$]

  $= \frac{2^{\kappa+1}L}{\kappa(3-\kappa)}\left[\gamma_t^{\kappa-1} - \frac{\kappa-1}{2}\gamma_t^\kappa\right]$

  $= \frac{2^{\kappa+1}L}{\kappa(3-\kappa)}2^{\kappa-1}\left[(t+1)^{1-\kappa} + (1-\kappa)(t+1)^{-\kappa}\right]$

  $\le \frac{2^{\kappa+1}L}{\kappa(3-\kappa)}2^{\kappa-1}(t+2)^{1-\kappa}$ [by convexity of $(t+1)^{1-\kappa}$]

  $= \frac{2^{\kappa+1}L}{\kappa(3-\kappa)}\gamma_{t+1}^{\kappa-1} \Rightarrow (!_{t+1})$ *holds true.*

Thus, $(!_t)$ holds true for all t, Q.E.D.

Conditional Gradient Algorithm for Norm-regularized Smooth Convex Minimization

♣ "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*

$$\mathrm{Opt} = \min_{x \in \mathbf{K}} \{\lambda \|x\| + f(x)\}$$

- $\mathbf{K}$: closed convex cone in Euclidean space E
- $\|\cdot\|$: norm on E
- $\lambda > 0$:penalty
- $f : \mathbf{K} \to \mathbb{R}$: convex function with Lipshitz continuous gradient:
$$\|\nabla f(x) - \nabla f(y)\|_* \leq L_f \|x - y\|, \ x, y \in \mathbf{K}$$

♠ **Main Assumption:** *We have at our disposal LMO oracle for* $(\|\cdot\|, K)$. *Given on input a linear form* $\langle \eta, \cdot \rangle$ *on* E, *the oracle returns*

$$x[\eta] \in \mathrm{Argmin}_x \{\langle \eta, x \rangle : x \in K, \|x\| \leq 1\}$$

Examples:

A. $E = \mathbb{R}^{m \times n}$, $\|\cdot\| = \|\cdot\|_{\mathrm{nuc}}$, $\mathbf{K} = E$

B. $E = \mathbf{S}^n$, $\|\cdot\| = \|\cdot\|_{\mathrm{nuc}}$, $\mathbf{K} = \mathbf{S}^n_+ = \{x \in E : x \succeq 0\}$

C. $E = \mathbf{M}_0^{m,n}$, $\|\cdot\| = \mathrm{TV}(\cdot)$, $\mathbf{K} = E$.

♣ We can reformulate the problem of interest as

$$\mathrm{Opt} = \min_{[x;r]\in\mathbf{K}^+} \{\phi(x,r) := \lambda r + f(x)\}$$
$$\mathbf{K}^+ = \{[x;r] \in E^+ := E \times \mathbb{R} : \|x\| \le r\}$$

♠ **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** $t = 1, 2, ...$ Given $y_t = [x_t; r_t] \in \mathbf{K}^+$,
  - compute $\nabla f(x_t)$
  - compute
    $$x_t^+ = x[\nabla f(x_t)] \in \mathrm{Argmin}_x \{\langle \nabla f(x_t), x\rangle : x \in \mathbf{K}, \|x\| \le 1\}$$
  - set $\Delta_t = \mathrm{Conv}\left\{y_t, 0, D^+[x_t^+; 1]\right\} \subset \mathbf{K}^+$ and find
    $$y_{t+1} \in \mathbf{K}^+ : \phi(y_{t+1}) \le \min_{y\in\Delta_t} \phi(y)$$

Step t is completed; pass to step $t+1$.

Note: One can set $y_{t+1} \in \mathrm{Argmin}_{y\in\Delta_t} \phi(y)$. 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).

$$\mathrm{Opt} = \min_{[x;r]\in \mathbf{K}^+} \{\phi(x,r) := \lambda r + f(x)\}$$
$$\mathbf{K}^+ = \{[x;r] \in E^+ := E \times \mathbb{R} : \|x\| \le r\}$$

♣ **Theorem:** *For the outlined algorithm,*

$$\phi(y_t) - \mathrm{Opt} \le \frac{8L_f D_*^2}{t+14},\ t = 2,3,...$$

♠ **Bundle Implementation:** We can set

$$y_{t+1} \in \mathrm{Argmin}_y \{\phi(y) : y \in \mathrm{Conv}\{0 \cup Y_t\}\} \qquad (*)$$

$Y_t \subset \mathbf{K}^+$: finite set containing $y_t = [x_t; r_t]$ and $D^+[x_t^+; 1]$, with

$$x_t^+ \in \mathrm{Argmin}_x \{\langle \nabla f(x_t), x\rangle : x \in \mathbf{K}, \|x\| \le 1\}$$

For example, we can comprise Y_t of $y_t, D^+[x_t^+; 1]$ and several of the previous iterates $y_1, ..., t_{t-1}$.

♡ 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, ∇f at a convex (or linear) combination $x = \sum \lambda_i x_i$ of points x_i *with already computed* Ax_i becomes cheap: $Ax = \sum_i \lambda_i (Ax_i)$.

$\Rightarrow$ *the FO oracle for* $(*)$ *is computationally cheap*

$$y_{t+1} \in \mathrm{Argmin}_y \{\phi(y) : y \in \mathrm{Conv}\{0 \cup Y_t\}\} \qquad (*)$$

$Y_t \subset \mathbf{K}^+$: finite set containing $y_t = [x_t; r_t]$ and $D^+[x_t^+; 1]$, with

$$x_t^+ \in \mathrm{Argmin}_x \{\langle \nabla f(x_t), x\rangle : x \in \mathbf{K}, \|x\| \le 1\}$$

- For example, with $f(x) = \frac{1}{2}\|Ax - b\|_2^2$, solving $(*)$ reduces to solving $k_t = \mathrm{Card}(Y_t)$-dimensional convex quadratic problem

$$\min_{\lambda \in \mathbb{R}^{k_t}} \left\{\tfrac{1}{2}\lambda^T Q_t \lambda + 2q_t^T \lambda : \lambda \ge 0, \textstyle\sum_j \lambda_j \le 1\right\}, \qquad (!)$$
$$Q_t = [x_i^T A^T A x_j]_{i,j}$$

where x_j, $1 \le j \le k_t$, 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

True image

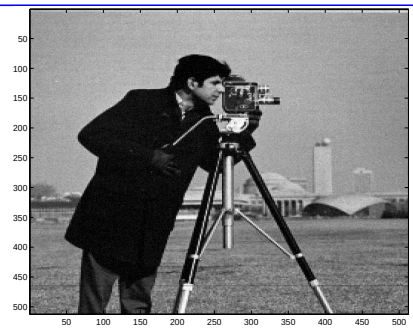
Blurred noisy image, 40% noise

Recovery

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

True image

Blurred noisy image, 40% noise



Recovery

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* $[\nabla f(x_t); \mathrm{TV}(\nabla f(x_t))]$ and removing (up to) 3 old points according to "first in — first out." *Adding* $[\nabla f(x_t); \mathrm{TV}(\nabla f(x_t))]$ *to the bundle dramatically accelerated the algorithm.*

How It Works:
Low Rank Matrix Completion

♠ Problem:

$$\text{Opt} = \min_{x \in \mathbb{R}^{n \times n}} \left\{ 0.1\|x\| + \|x - a\|_F^2 \right\}$$

[• $\|\cdot\|$: nuclear norm • $\|\cdot\|_F$: Frobenius norm • $a = \bar{x} + \xi$
Rank$(\bar{x}) \approx \sqrt{n}$, $\|\bar{x}\| \approx \sqrt{2n/\pi}$, $\|\xi\|_F \approx 0.1\|\bar{x}\|_F$ with i.i.d. Gaussian ξ_{ij}]

- 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 ⇒ *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!*

References

♠ The material of Lectures I, II is covered by [1,Chapter 7] and [2,Section 3.4].

♠ The material of Lecture III is covered by [2,Chapter 5], see also [3].

♠ The material of Lecture IV is covered by [1, Section 5.4.1] and [4].

1. A. Nemirovski, *Introduction to Linear Optimization*, Lecture Notes

http://www2.isye.gatech.edu/~nemirovs/OPTI_LectureNotes.pdf

2. A. Ben-Tal, A. Nemirovski, *Lectures on Modern Convex Optimization*, Lecture Notes

http://www2.isye.gatech.edu/~nemirovs/Lect_ModConvOpt.pdf

3. A. Nemirovski, *Mirror Descent for Large-Scale Deterministic and Stochastic Convex Optimization: Selected proofs*

http://www2.isye.gatech.edu/~nemirovs/Lecture3Proofs.pdf

4. Z. Harchaoui, A. Juditsky, A. Nemirovski, "Conditional Gradient Algorithms for Norm-Regularized Smooth Convex Optimization"

http://arxiv.org/pdf/1302.2325.pdf