Course:

Linear and Convex Optimization

ISyE 4803 D Spring 2015

Instructor: **Dr. Arkadi Nemirovski**

nemirovs@isye.gatech.edu, Groseclose 446

Office hours: Monday 10:00-12:00

Teaching Assistant: TBA

Lecture Notes, Transparencies, Assignments, Optional Projects: T-Square

Note: In addition to Lecture Notes *per se*, Transparencies can be viewed as self-contained (up to proofs) lecture notes.

Note: Proofs are non-obligatory (although highly recommended!): in this course, *you never will be asked to prove something.*

Rules of the Game

♣ Grade Components:

A. Obligatory "Pen and Paper" (p.-p.) assignments, point weight 100 per assignment, to be graded by TA

— usual exercises like *"find this and that"* or *"is it true that..."*

- *P.-p. assignments will be posted at T-Square and have due dates*

B. MidTerm Exam, total point weight 100

C. Final Exam, total point weight 100

D. Optional *bonus* "Model and Solve" (m.s.) assignments, each with its own points weight, to be graded by TA

— ask to model a story as optimization problem and process the model numerically

- *M.-s. assignments are/will be posted at T-Square, deadline for submissions 04/01/2014*

Software environment: `MATLAB` and `CVX`

`CVX` is an excellent user-friendly optimization solver working under `MATLAB`, free download at `http://cvxr.com/cvx/`

E. Optional *bonus* modeling and computational project, total weight 100 bonus points, to be graded by me

Project is posted at T-Square, deadline for submissions 04/20/2014

♣ Grading Formula:

Point Grade for the Course

$$= \min\Big[100,$$

$$0.05 \times \Big[\frac{\text{total \# of points earned for p.p. assignments}}{\text{total \# of p.p. assignments}}\Big]$$

$$+ \quad 0.45 \times \Big[\underbrace{\text{Grade in MidTerm}}_{G_{\mathsf{MT}}}\Big] + 0.50 \times \Big[\underbrace{\text{Grade in Final}}_{G_{\mathsf{F}}}\Big]$$

$$+ \quad 0.20 \times \frac{100 \times [\text{total \# of points earned for m.s. assignments}]}{\text{total point weight of m.s. assignments}}$$

$$+ \quad 0.50 \times \left[\begin{array}{c}\text{points earned in}\\ \text{Optional Project}\end{array}\right] \times \left\{\begin{array}{ll} 1, & \min[G_{\mathsf{MT}}, G_{\mathsf{F}}] \geq 50 \\ \frac{1}{2}, & \text{otherwise}\end{array}\right. \Big]$$

Note: This is the messiest formula you will see in class...

♣ *"There is nothing more practical than a good theory."* Our course is aimed at *methodology and theory* of Optimization rather than on acquiring ready-to-use practical skills

$\Rightarrow$ *Concentrate on What ? and not on What for ?* Understanding of "what for" will come later.

Please be patient!

♣ Questions are highly welcome. *Ask as many questions as you can, and then some!*

Main Notational Conventions

- By default, *all vectors are column vectors.*
- The space of all n-dimensional vectors is denoted $\mathbb{R}^n$; and the set of all $m \times n$ matrices is denoted $\mathbb{R}^{m \times n}$.
- Usually, "MATLAB notation" is used: a vector with coordinates $x_1, ..., x_n$ is written down as

$$x = [x_1; ...; x_n]$$

(pay attention to semicolon ";" !)

For example, $\begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix}$ is written as $[1; 2; 3]$.

- More generally, if $A_1, ..., A_m$ are matrices with the same number of columns, we write $[A_1; ...; A_m]$ to denote the matrix which is obtained when writing A_2 beneath A_1, A_3 beneath A_2, and so on.
- If $A_1, ..., A_m$ are matrices with the same number of rows, then $[A_1, ..., A_m]$ stands for the matrix which is obtained when writing A_2 to the right of A_1, A_3 to the right of A_2, and so on.

Examples:

- $A_1 = \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{bmatrix}, A_2 = \begin{bmatrix} 7 & 8 & 9 \end{bmatrix}$

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

- $A_1 = \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix}, A_2 = \begin{bmatrix} 7 \\ 8 \end{bmatrix}$

$$\Rightarrow \quad [A_1, A_2] = \begin{bmatrix} 1 & 2 & 7 \\ 4 & 5 & 8 \end{bmatrix}$$

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

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

Preface:

What Optimization is about?

What the course is about?

What Optimization is about

♣ To make decisions optimally is one of the basic desires of a human being

♠ **If** (and this is a big If indeed!)

A. *Our potential decisions can be quantified by values of a number of decision variables*

$$x_1, x_2, ..., x_n$$

,

B. *We can describe mathematically the set*

$$X = \{x = [x_1; ...; x_n]\}$$

of feasible decisions — those indeed implementable under circumstances,

C. *We can quantify the outcome of a decision* $x = [x_1; ...; x_n]$ *by a single objective — real-valued function* $f(x_1, ..., x_n)$

then we can select the optimal decision by solving *optimization problem*

$$\boxed{\begin{array}{l}\text{maximize}\\ \text{[minimize]}\end{array} \quad f(x) \text{ over } x = [x_1; ...; x_n] \text{ subject to } x \in X.}$$

Note: Whether the objective f should be maximized or minimized, it depends on what – profits or losses – it expresses.

♣ When applying optimization-oriented methods to a real-life situation, the *key* is to build an optimization model which

a: *is enough adequate* – models well enough the structure of potential decisions, relations between the decisions and outcomes, etc.,

b: *can be "fed" by necessary data* – we can identify numerical values of various parameters (demands, resources, performance characteristics of available devices and processes, etc.) treated as given quantities specifying "model's environment,"

c: *can be processed numerically in reasonable time.*

Note: These targets are somehow contradictory – the more adequate is the model, the more data it requires, and the more complicated it becomes as far as numerical processing is concerned...

a: *Model should be enough adequate*
b: *We can "feed" the model by meaningful data*
c: *Model should be amenable for accurate numerical processing taking reasonable time.*

♠ In our course we
- do *not* touch **a** — building an adequate model requires understanding the subject domain in question and goes beyond optimization *per se*
- *somehow* touch **b** by presenting the basics of *Robust Optimization* methodology – a popular optimization paradigm aimed at handling *data uncertainty* characteristic for optimization problems of real-life origin
- *focus* on **c**, via emphasis on what optimization can do well and what is problematic, and thus on what are "desirable" model structures allowing for reliable numerical processing.

Example: *Given time horizon of one year, we want to create a portfolio by distributing (at most) $ 1000 between 4 available assets.*

- A potential decision can be represented by values of 4 decision variables x_1, x_2, x_3, x_4, where x_j is the money to be invested in asset $\#j$, $j = 1, 2, 3, 4$
- We probably can define the set of feasible decisions as

$$X = \left\{ x = [x_1; x_2; x_3; x_4] : \begin{array}{l} x_1 + x_2 + x_3 + x_4 \le 1000 \\ \text{you cannot invest more than \$ 1000} \\ x_1 \ge 0, x_2 \ge 0, x_3 \ge 0, x_4 \ge 0 \\ \text{investments cannot be negative} \end{array} \right\}$$

- The simplest objective (to be maximized) is the expected value of the portfolio in a year from now:

$$f(x) = c_1 x_1 + c_2 x_2 + c_3 x_3 + c_4 x_4$$

where c_j are the expected yearly returns of the assets

$\Rightarrow$ *Optimal Portfolio Selection can be modeled by the optimization problem*

$$\max \left\{ c_1 x_1 + c_2 x_2 + c_3 x_3 + c_4 x_4 : \begin{array}{l} x_1 + x_2 + x_3 + x_4 \le 1000 \\ x_1 \ge 0, x_2 \ge 0, x_3 \ge 0, x_4 \ge 0 \end{array} \right\}$$

♠ **Quiz:** What are the optimal portfolio selections in the cases when

- $c_1 = 1.1, c_2 = 1.2, c_3 = 1.3, c_4 = 1.4$
- $c_1 = 1.1, c_2 = 1.2, c_3 = c_4 = 1.3$
- $c_1 = -0.1, c_2 = -0.2, c_3 = -0.3, c_4 = -0.4$

♠ **Quiz:** Does the model meet the requirements **a** - **c**? What is the most problematic requirement?

♠ Same as in our toy Portfolio Selection problem, the set X of feasible decisions is usually (not always!) given by a system of constraints

$$g_i(x) \begin{matrix} \geq \\ = \\ \leq \end{matrix} b_i, \ i = 1, ..., m$$

every one of them representing a specific requirement, like upper bound on a resource consumed, or a balance constraint, or a lower bound on a particular outcome.

Note: *The constraints in the system are always linked by "and" – a feasible decision x must satisfy every one of the constraints, and not some of them:*

$$X = \left\{ x = [x_1; ...; x_n] : g_i(x) \begin{matrix} \geq \\ = \\ \leq \end{matrix} b_i \text{ for all } i = 1, ..., m, \right\}$$

In terms of the objective and the constraints, an optimization problem reads

$$\begin{matrix} \min_x \\ [\max_x] \end{matrix} \left\{ f(x) : g_i(x) \begin{matrix} \geq \\ = \\ \leq \end{matrix} b_i \text{ for all } i = 1, ..., m \right\}$$

[Mathematical Programming format of an optimization problem]

$$\begin{array}{c}\min_x\\ [\max_x]\end{array}\left\{f(x):g_i(x)\begin{array}{c}\geq\\ =\\ \leq\end{array}b_i \text{ for all } i=1,...,m\right\}$$

♠ **Note:** We always can pose an optimization problem as a *maximization* one and make all its constraints $\leq$-*inequalities* [same as always can make the problem a minimization one and/or write down the constraints as $\geq$-inequalities].

Indeed,

- to maximize $f(x)$ is the same as to minimize $-f(x)$
- inequality constraint $g_i(x) \geq b_i$ is equivalent to $-g_i(x) \leq -b_i$, and equality constraint $g_i(x) = b_i$ can be expressed by a pair of opposite inequalities $g_i(x) \leq b_i$ and $-g_i(x) \leq -b_i$.

Example: Problem

$$\min_{x=[x_1;x_2]}\left\{f(x)=x_1-2x_2:\begin{array}{rcl}x_1&\geq&5\\ x_1+x_2&=&10\end{array}\right\}$$

is equivalent to

$$\max_{x=[x_1;x_2]}\left\{\widehat{f}(x)=-x_1+2x_2:\begin{array}{rcl}-x_1&\leq&-5\\ x_1+x_2&\leq&10\\ -x_1-x_2&\leq&-10\end{array}\right\}.$$

Example: Optimization problem

$$\min_{x=[x_1;x_2]} \left\{ \underbrace{x_1+x_2}_{f(x)} : \underbrace{x_1-x_2}_{g_1(x)} \le \underbrace{3}_{b_1} \text{ or } \underbrace{\sin(x_2)}_{g_2(x)} \le \underbrace{0.5}_{b_2} \right\}$$

is *not* in MP format: the requirements $g_1(x) \le b_1$ and $g_2(x) \le b_2$ are linked by "or," not by "and," and a feasible x is allowed to violate one of these requirements, provided it meets the other one.

- The MP form of the problem is

$$\min_{x=[x_1;x_2]} \left\{ \underbrace{x_1+x_2}_{f(x)} : \underbrace{\min[x_1-x_2-3, \sin(x_2)-0.5]}_{g(x):=\min[g_1(x)-b_1, g_2(x)-b_2]} \le 0 \right\}$$

♠ Indeed, to say that

$$g_1(x) \le b_1 \text{ or } g_2(x) \le b_2 \text{ or } \ldots \text{ or } g_m(x) \le b_m$$

is exactly the same as to say that

$$g(x) := \min[g_1(x)-b_1, g_2(x)-b_2, ..., g_m(x)-b_m] \le 0.$$

♡ In contrast, to say that

$$g_1(x) \le b_1 \text{ and } g_2(x) \le b_2 \text{ and } \ldots \text{ and } g_m(x) \le b_m$$

is exactly the same as to say that

$$g(x) := \max[g_1(x)-b_1, g_2(x)-b_2, ..., g_m(x)-b_m] \le 0.$$

Optimization: Challenges

♣ When people want to say that certain task is in fact intractable, they say *"it is like to find a needle in a haystack."*

• As far as computations are concerned, *optimization is exactly about finding a needle in a haystack*, with elaboration stemming from the fact that the haystack in typical applications is of dimension in the range of thousands and millions, rather than to be of modest dimension 3.

♠ Finding a needle in n-dimensional haystack can be posed as optimization problem as follows:

• The haystack – the feasible set of the problem – is the box $X = \{x \in \mathbb{R}^n : 0 \le x_j \le 1, 1 \le j \le n\}$

• The needle, which is a small domain in $\mathbb{R}^n$, is modeled by the objective $f(x)$: this is a function of $x \in \mathbb{R}^n$ which is 0 outside of the needle and is negative inside the needle.

$\Rightarrow$ *To find the needle means to find a feasible solution to the problem*

$$\min_{x=[x_1;...;x_n]} \Big\{ f(x) : 0 \le x_j \le 1, 1 \le j \le n \Big\}$$

where the objective is negative, to which end it suffices to find a good enough approximate solution to the problem.

$$\min_{x=[x_1;...;x_n]} \left\{ f(x) : 0 \le x_j \le 1,\ 1 \le j \le n \right\}$$

Our needle is buried in hay – we cannot see it from a distance. What we can do is to generate, one by one, "search points" x_1, x_2,... in the haystack and to check whether these points hit the needle.

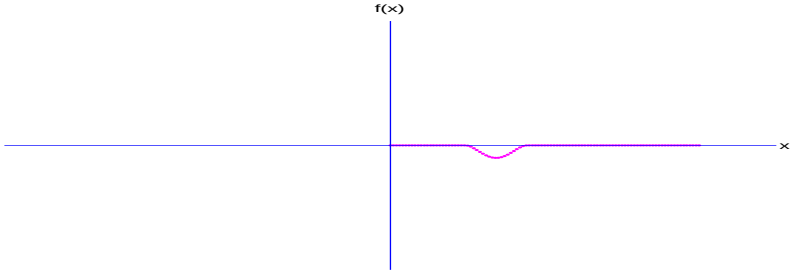
Equivalently: At search step t, we select a point x_t and call *an oracle* which provides us with *local* information on f at the point, e.g., reports the value $f(x_t)$ of f at the point (and perhaps also the derivatives of f).

♠ **Note:** *This is the standard model of a general-purpose optimization process: when solving optimization problem*

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

we learn the problem by calling "an oracle" – a subroutine reporting local information on f and g_i, most notably, values and gradients – along a sequence of search points x_t, $t = 1, 2, ...$ A "general purpose" solution algorithm is, essentially, the collection of rules for generating these search points.

♠ In our "needle in haystack" example, f is pretty simple – it is zero outside the needle and negative inside it:



1D haystack (the domain of f)
and the needle (the interval where f is negative)

• When querying local oracle outside of the needle, we get no information on its location

$\Rightarrow$ When looking for a needle in 1D stack, the best we can do is to scan for the needle along a grid with resolution of order of the needle length ℓ, or generate the search points at random from the uniform distribution. *Typical number of steps before the needle is found will be of order of* $1/\ell$.

♠ When looking for a needle in n-dimensional haystack, the situation is similar: the best we can do is to scan along a "dense enough" grid in the haystack or to look for the needle along a sequence of points drawn at random from the uniform distribution in the haystack. In both cases, *typical number of steps before the needle is found is of order of* $N = \frac{\mathrm{Vol}_n(\text{haystack})}{\mathrm{Vol}_n(\text{needle})}$

• Vol_n: n-dimensional volume.

♠ When modeling the haystack as n-dimensional unit box $\{x : 0 \leq x_j \leq 1, 1 \leq j \leq n\}$, and the needle — as a box with $n-1$ edges equal to 0.005 and one edge equal to 0.05, the typical number of steps before the needle is found is of order of

$$N = 2^n \cdot 10^{2n-1}$$

n	N	comment
1	20	*easy*
3	800,000	*proverbial case*
10	$1.024 \cdot 10^{22}$	*by far beyond reach*
36	$\approx 6.87 \cdot 10^{81}$	*more than # of atoms in Universe!*

Note: When minimizing within accuracy ϵ a function over n-dimensional domain X, we should reach the domain X_ϵ of ϵ-optimal solutions. With "needle in the haystack" approach, it would typically take

$$\frac{\mathrm{Vol}_n(X)}{\mathrm{Vol}_n(X_\epsilon)} \sim \left(\frac{1}{\sqrt{\epsilon}}\right)^n$$

steps – an astronomical number already for $\epsilon = 0.01$ and $n = 25$. *We would be unable to carry out that many steps even if the future of mankind were at stake!*

♠ **Question:** *How happens that we can solve problems with tens of thousands of variables within accuracies like* $\epsilon = 10^{-6}$ *or* $\epsilon = 10^{-10}$ *?*

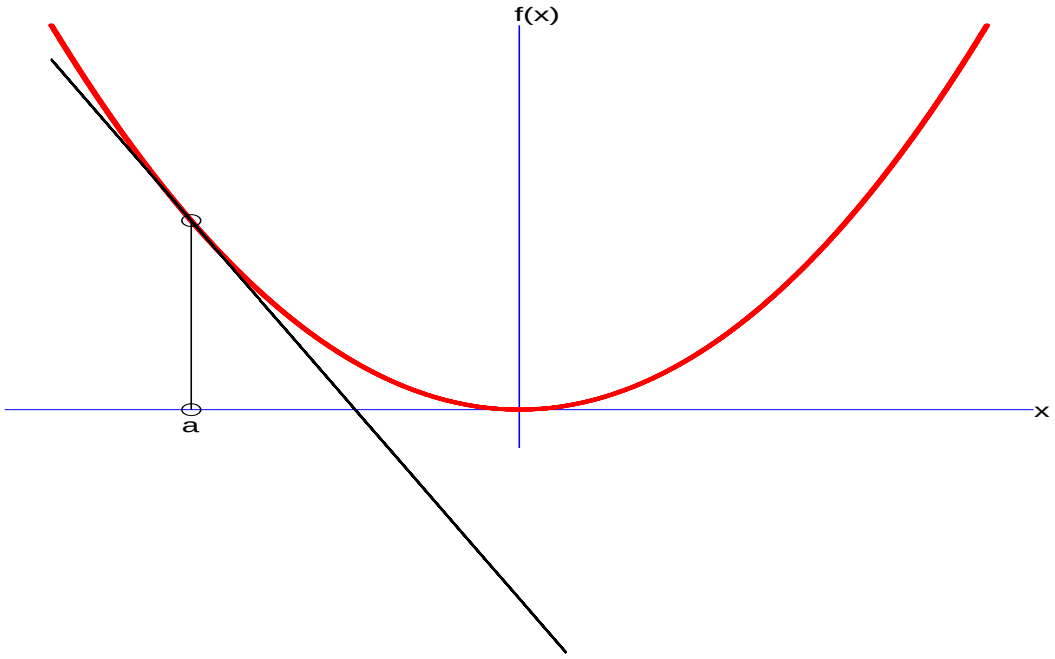
♡ **Answer:** The key is in utilizing *problem's structure.* With favorable structure, *already local information on objective and constraints conveys information on where the globally optimal solution is*.

♣ **Fact:** The standard "favorable structure" is *convexity* of the MP problem under consideration.

♠ **Convexity: First acquaintance.** *For starters*, consider the problem $\min_{x\in X} f(x)$ of minimizing a *differentiable* function f over a *simple* domain, specifically, n-dimensional box $X = \{x \in \mathbb{R}^n : -1 \le x_1, ..., x_n \le 1\}$.

• *For a differentiable f, convexity* can be defined as *the property of f to dominate its linearizations*:

$$\begin{array}{rcl} f(y) & \ge & f(x) + [\nabla f(x)]^T (y - x) \\ & := & f(x) + \sum_{i=1}^n \frac{\partial f(x)}{\partial x_i}(y_i - x_i) \text{ for all } x, y \end{array}$$

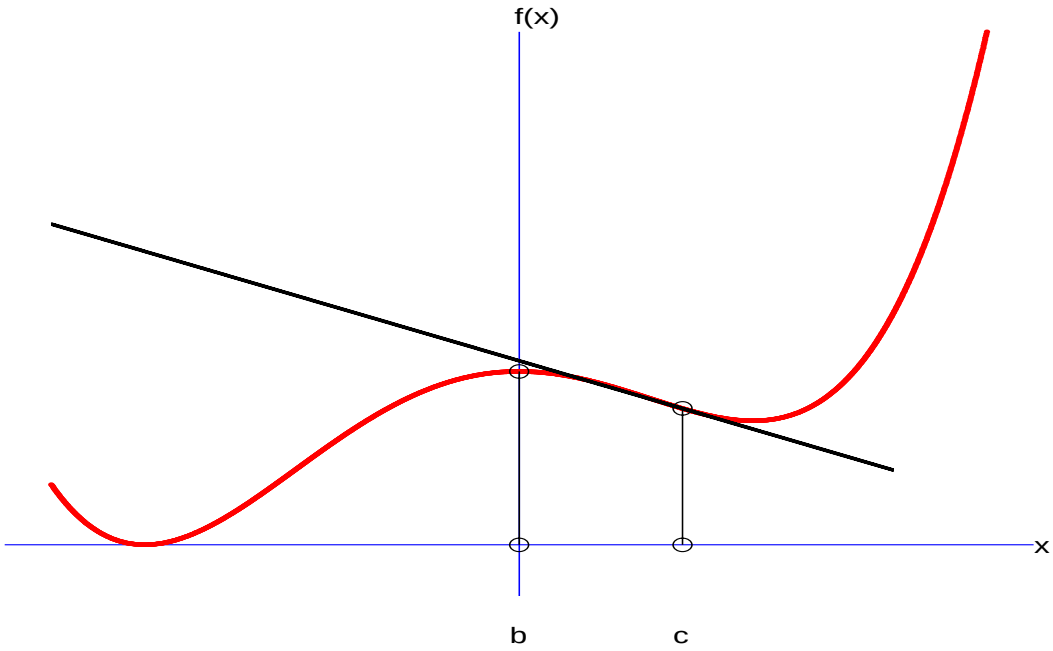


What we see: We have computed f and ∇f at a (feasible) solution a, and $\nabla f(a)$ turns out to be negative $\Rightarrow$ to the left of a, linearization of f is $> f(a) \Rightarrow$ *to the left of a, f itself is $> f(a)$*

$\Rightarrow$ *we can reduce the optimization domain by cutting off all points $< a$!*

• This "cut off" scheme can be extended to multi-dimensional convex (i.e., with convex objective and constraints) problems!

♠ **Note:** In the "cut off" scheme, convexity of f is crucial. For example, in the case of (nonconvex!) f al follows:



local information to the right of b reveals nothing about the location of the global minimum and does *not* allow for cutting off a massive set of candidate solutions

$\Rightarrow$ *In the non-convex case, to approximate well the globally optimal solutions, a kind of brute search is necessary.*

Utilizing problem's structure, we can make brute search more efficient than in the "needle in the haystack" case, but usually cannot eliminate the "curse of dimensionality" — *exponential explosion of the number of steps as problem's dimension grows*.

♣ **Course-related consequences:**

A. *Emphasis on Convex optimization – the "solvable case" in Mathematical Programming.* Convex optimization

— covers a wide and constantly extending range of applications in

- *Decision Making (Linear Optimization models without integrality constraints),*
- *Engineering (Signal Processing, Imaging, Machine Learning, High-dimensional Statistics, Structural Design, Synthesis of linear controllers,...)*

— is the major "working horse" when solving difficult non-convex optimization problems arising in Discrete Optimization

— in theory, and to some extent also in practice, *allows to find, in a computationally efficient fashion, high accuracy approximations to globally optimal solutions*. Among optimization algorithms, those of convex optimization are by far closer to the ideal *"You Press the Button, We Do the Rest"* than algorithms for solving non-convex problems.

♣ **Course-related consequences (continued):**

B. In our course, "Emphasis on Convex Optimization" will mean *primary focus on "well-structured" families of convex problems, specifically, conic ones.*

• Convex problems typically possess much more structure than postulated by plain convexity, and utilizing this "extra" structure in solution algorithms was and still is the key driving force in the dramatic progress in the area during the last two decades.

As a result of this progress, the performance of convex optimization techniques increased by factor like 10^6, with nearly equal contributions of hardware and of algorithmic improvements.

• Conic Programming is a far-reaching extension of Linear Programming. Linear Programming possesses an extremely rich and relatively simple structure which underlies fundamental theoretical developments (duality) and extremely efficient algorithms. It turns out that these theory and algorithms can be extended to an extremely wide variety of convex *nonlinear* problems captured by Conic Programming.

♣ **Course-related consequences (continued):**

C. Availability of software (e.g., CVX) of the type *"You Press the Button, We Do the Rest"* allows to switch from the traditional emphasis on *how* convex optimization algorithms work to *what* these algorithms can solve.

Cf.: *When car engines are reliable, a driver should not know much about "what is under the hood" and may focus on route planning and safe driving.*

⇒ In our course, algorithms will be presented at the "executive summary" level. Our emphasis will be on

- *basic theory of Convex Optimization*, most notably, *duality*
- *"calculus" of well-structured convex problems:*
  - how to recognize convexity?
  - how to convert a problem into a form well-suited for numerical processing?
  - what are the key factors affecting the performance of state-of-the-art algorithms?
- *instructive application examples*

Part I: Linear Optimization

- What can be expressed via LO?
- Geometry of polyhedral sets
- LO Duality
- Simplex Method

Linear Optimization Program

A *Linear Optimization problem*, or program (LO), called also *Linear Programming* problem/program, is the problem of optimizing a *linear function*

$$f(x) = \sum_{j=1}^{n} c_i x_i$$

of an n-dimensional decision vector x under *finitely many linear* equality and nonstrict inequality constraints.

Equivalently: *An LO program is a Mathematical Programming program*

$$\begin{matrix}\min_x \\ [\max_x]\end{matrix} \left\{ f(x) : g_i(x) \begin{matrix}\geq \\ = \\ \leq\end{matrix} b_i \text{ for all } i = 1, ..., m \right\}$$

where f and g_i are linear functions of x.

- For example, the MP problem

$$\min_x \left\{ x_1 : \left\{ \begin{array}{rcl} x_1 + x_2 & \leq & 20 \\ x_1 - x_2 & = & 5 \\ x_1, x_2 & \geq & 0 \end{array} \right\} \right. \qquad (1)$$

is an LO program, while the problem

$$\min_x \left\{ \exp\{x_1\} : \left\{ \begin{array}{rcl} x_1 + x_2 & \leq & 20 \\ x_1 - x_2 & = & 5 \\ x_1, x_2 & \geq & 0 \end{array} \right\} \right. \qquad (1')$$

is *not* an LO program, since the objective in (1′) is nonlinear.

- Similarly, the problem

$$\max_x \left\{ x_1 + x_2 : \left\{ \begin{array}{rcl} 2x_1 & \geq & 20 - x_2 \\ x_1 - x_2 & = & 5 \\ x_1 & \geq & 0 \\ x_2 & \leq & 0 \end{array} \right\} \right. \qquad (2)$$

is an LO program, while the problem

$$\max_x \left\{ x_1 + x_2 : \left\{ \begin{array}{rcl} \forall i \geq 2: & & \\ ix_1 & \geq & 20 - x_2, \\ x_1 - x_2 & = & 5 \\ x_1 & \geq & 0 \\ x_2 & \leq & 0 \end{array} \right\} \right. \qquad (2')$$

is *not* an LO program – it has infinitely many linear constraints.

However: Problems (1),(1′) are reducible to each other, and similarly for problems (2) and (2′)

♠ **Note:** Property of an MP problem to be or not to be an LO program is the property of a *representation* of the problem. We classify optimization problems according to *how they are presented*, and not according to *what they can be equivalent/reduced to*.

What is good in Linear Optimization?

♠ Linear Optimization is, historically, the first "chapter" of Mathematical Programming in general and Convex Optimization in particular. Discovered in late 1940's, *it still remains the most frequently used optimization methodology and computational toolbox*.

Reasons:

- In spite of simple structure, linear models cover reasonably well a wide spectrum of applications in Decision Making and Engineering
- It is relatively easy to "feed" linear model by data. To specify a linear function of 1000 variables, you need 1000 coefficients; to specify a quadratic function, you need 501,500 coefficients!
- *LO methodology was from the very beginning complemented by extremely powerful solution algorithm – Simplex method,* which make LO a working tool rather than a wishful thinking.

♠ *LO possesses deep, rich and instructive theory* which is the major source and prototype when developing theory and algorithms of more general and complicated classes of optimization problems. In particular,

• LO was the prototype when developing *Conic Programming* and *Interior Point Methods* which extended dramatically the practical grasp and computational power of Convex optimization

• LO theory underlies optimality conditions in Nonlinear optimization. These conditions serve as the major "driving force" when designing algorithms for *non-convex* continuous optimization.

♠ LO techniques are the major "working horse" in crucial for applications *Discrete and Combinatorial* optimization.

Illustration: Portfolio Selection

♣ I have \$1 to invest for one year. My options are

A. To put some of the money, x_0, to my savings account which *guarantees* 5% interest, so that in a year from now I shall get $1.05x_0$ in the bank.

B. To distribute the rest of the money between 9 available assets. Investing x_i into asset # i, in a year from now the value of my investment will be $[1+\rho_i]x_i$, where $[1+\rho_i]$ is the yearly return of asset i.

$\Rightarrow$ In a year from now, I shall own

$$u = 1.05x_0 + [1+\rho_1]x_1 + ... + [1+\rho_9]x_9$$

I would like to maximize u under natural constraints

$$x_0 \geq 0, x_1 \geq 0, ..., x_9 \geq 0;\ x_0 + x_1 + ... + x_9 = 1.$$

♠ **Difficulty:** The profits ρ_i are *uncertain* and *random*, and their joint probability distribution is only *partially* known. Specifically, from the historical data I know

- the ranges of the random profits ρ_1,...,ρ_9
- the expected profits $r_i = \mathbf{E}\{\rho_i\}$, $i = 1, ..., 9$

In addition, I know that

- the profits are *uncorrelated* across the assets:

$$\mathbf{E}\{(\rho_i - r_i)(\rho_j - r_j)\} = 0,\ 1 \leq i < j \leq 9.$$

How to proceed?

♠ The answer depends on my attitude to *risk.*

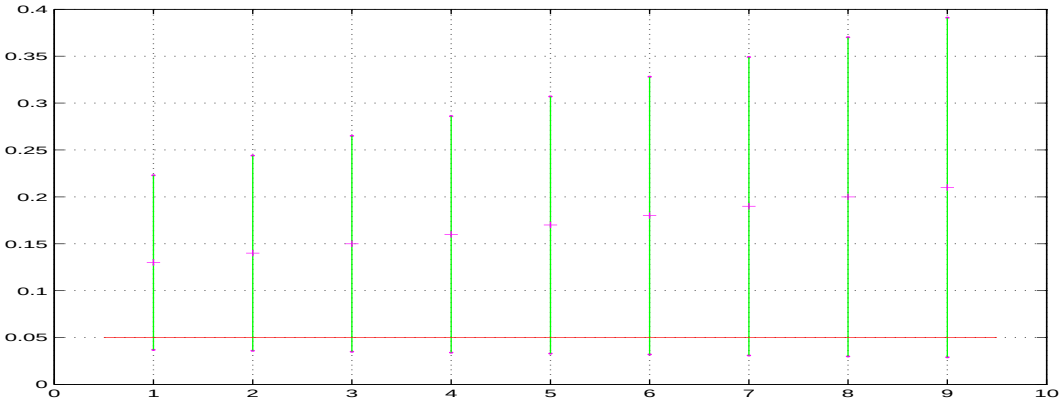
♠ Problem:

"maximize" $u = 1.05x_0 + [1+\rho_1]x_1 + ... + [1+\rho_9]x_9$

s.t. $x_0 \geq 0, x_1 \geq 0, ..., x_9 \geq 0$; $x_0 + x_1 + ... + x_9 = 1$

[ρ_i: uncorrelated profits with known expectations r_i and ranges]

♡ Available data:



- green vertical segments: ranges of profits ρ_i
- +: expected profits which are the midpoints of the ranges
- ———- : profit guaranteed by saving account

(?) What should we maximize?

- A *risk-neutral* person would maximize the expected profit

$\Rightarrow$ invest all the money in the most promising asset # 9

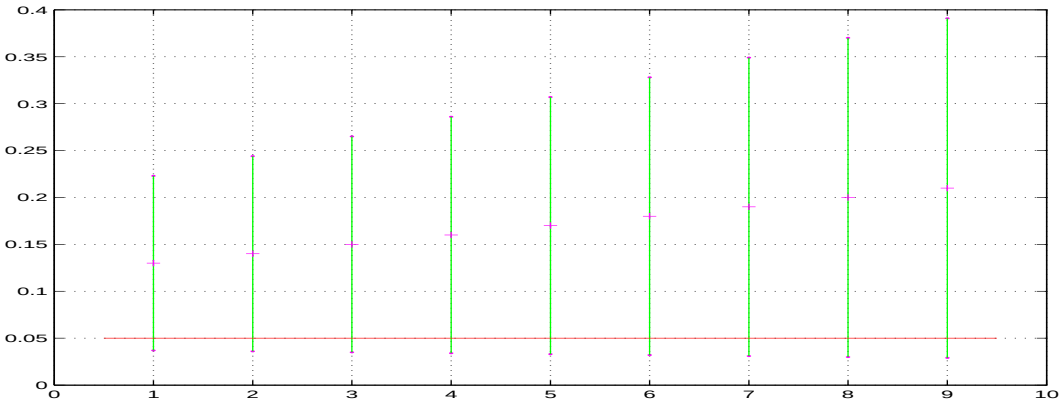
- A *risk-averse* person, like me, would maximize, for a reasonable risk level α, the *value-at-risk* α — the largest u such that

$$\text{Prob}\{1.05x_0 + [1+\rho_1]x_1 + ... + [1+\rho_9]x_9 < u\} \leq \alpha \quad (!)$$

Quiz: What is the value-at-risk 0.05 for r.v. ξ given by

value v	1	2	3	4	5	6	7
Prob$\{\xi = v\}$	0.01	0.03	0.04	0.02	0.30	0.25	0.35
Prob$\{\xi < v\}$	0.00	0.01	0.04	0.08	0.10	0.40	0.65
Prob$\{\xi \leq v\}$	0.01	0.04	0.08	0.10	0.40	0.65	1.00

Obstacle: I do *not* know what exactly is the distribution of profits! $\Rightarrow$ *It makes sense to insist on the validity of (!) for every distribution of returns compatible with my a priori information.*



- green vertical segments: ranges of profits ρ_i
- $+$: expected profits
- ———- : profit guaranteed by saving account

• I want to maximize value-at-risk 0.05, that is, to maximize u over x, u under the constraints

$x_0 \geq 0, ..., x_9 \geq 0, \sum_{i=0}^{9} x_i = 1$

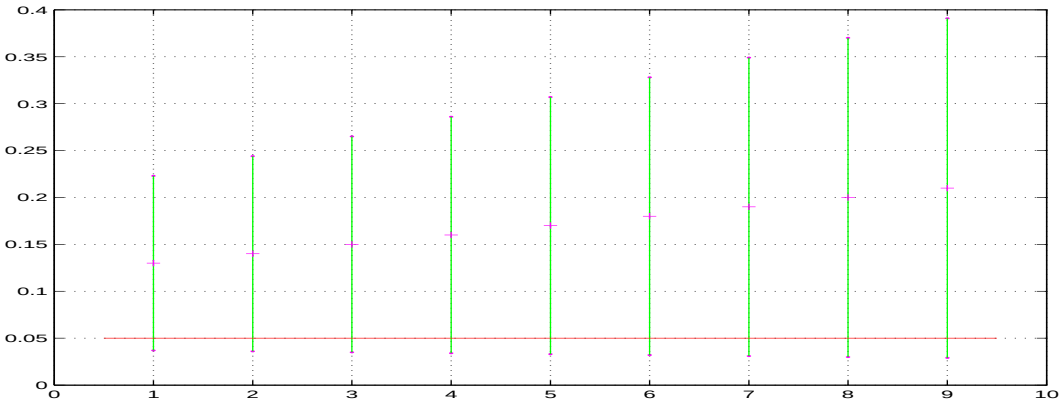
$\text{Prob}\{1.05x_0 + [1+\rho_1]x_1 + ... + [1+\rho_9]x_9 < u\} \leq 0.05$

for every distribution with *uncorrelated* ρ_i varying in given ranges and possessing given expectations

(?) *To invest or not to invest?*

♠ **Key observation:** The worst case profits of every one of the assets are less than the profit guaranteed by savings $\Rightarrow$ *If the largest, over all distributions in question, probability of a crisis – for all ρ_i simultaneously to take their worst-case values, is > 0.05, then I should not invest into assets!*

Indeed, let my family of distributions contain one with probability of the crisis > 0.05. *If this bad distribution is the true one* (why not?), with probability > 0.05 profit on (nonzero) investments in the assets will be less than 5% of these investments. With 5% profit from savings, *the overall profit will be less than 5% of my initial capital of $1.* In contrast, $1 in savings does yield 5% profit...



- green vertical segments: ranges of profits ρ_i
- $+$: expected profits
- ———- : profit guaranteed by saving account

♠ To understand how high could be the probability of a crisis, I use *educated guess* stating that the most dangerous distribution is the one where each ρ_i takes only extreme values in its range. Indeed, such a distribution yields the largest possible volatilities.

- With my data, the expectation r_i of ρ_i is the mid-point of profit's range

$\Rightarrow$ My educated guess says that ρ_i *takes every one of its two extreme values with probability* $1/2$*:*

$$\rho_i = r_i + \delta_i \xi_i$$

- ξ_i: half-length of the range of ρ_i;
- δ_i: takes values ± 1 with probabilities $1/2$.

♠ In terms of δ_i's, the problem of maximizing the probability of the crisis reads:

Given $n = 9$ *uncorrelated random variables* $\delta_1, ..., \delta_n$ *taking values* ± 1 *with probabilities* $1/2$*, how large could be the probability of the event* $\delta_1 = \delta_2 = ... = \delta_n = -1$*?*

This is a Linear Programming program!

Given $n = 9$ uncorrelated random variables $\delta_1, ..., \delta_n$ taking values ± 1 with probabilities $1/2$, how large could be the probability that $\delta_1 = \delta_2 = ... = \delta_n = -1$?

• Denoting by $\varepsilon = [\epsilon_1; ...; \epsilon_n]$ a collection of n ± 1's, we

— introduce $N = 2^n$ decision variables p_ε indexed by the collections; for a particular $\varepsilon = [\epsilon_1; ...; \epsilon_n]$, p_ε is the probability that $\{\delta_1 = \epsilon_1, ..., \delta_n = \epsilon_n\}$;

— subject the variables to linear constraints

$p_\varepsilon \geq 0$ for all ε $\sum_\varepsilon p_\varepsilon = 1$	probabilities are ≥ 0 and sum up to 1
$\sum_{\varepsilon:\epsilon_i=-1} p_\varepsilon = \frac{1}{2}$ for all i	$\forall i, \mathrm{Prob}\{\delta_i = -1\} = \frac{1}{2}$
$\sum_{\varepsilon:\epsilon_i=1} p_\varepsilon = \frac{1}{2}$ for all i	$\forall i, \mathrm{Prob}\{\delta_i = 1\} = \frac{1}{2}$
$\sum_{\omega=\pm 1,\ \omega'=\pm 1} \Big[\sum_{\varepsilon:\epsilon_i=\omega,\ \epsilon_j=\omega'} p_\varepsilon\Big]\omega\omega' = 0$ for all $i < j$	no correlations

— maximize under these constraints the probability $p_{[-1;...;-1]}$ of crisis.

Quiz: *Are all the constraints necessary?*

• LP solver finds the optimal value equal to 0.10

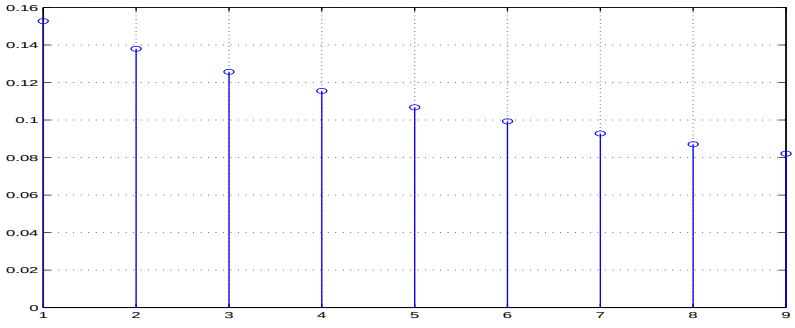
$\Rightarrow$ With my a priori information, probability of crisis can be > 0.05

$\Rightarrow$ *I should not invest in assets!*

♠ Absence of correlations often is interpreted as *absence of statistical dependency.* What would I do if the profits were indeed independent across the assets?

♠ The problem "*Given that the profits of assets are independent r.v.'s with given ranges and expectations, find the investment of $1 between savings and assets with the largest possible value-at-risk* α" is difficult – the value-at-risk is a difficult to optimize function of the investments, and on the top of it, we still have only partial knowledge of the distribution of profits.

• **However**, one can maximize a properly built "optimization friendly" *lower bound* of the value at risk, valid for all distributions with independent $\rho_1, ..., \rho_9$ meeting our priori information on ranges and expectations. With our data and $\alpha = 0.05$, maximizing the bound yields a nicely diversified portfolio



Near-optimal investment of $ 1 (no money in savings)

Lower bound on profit's value-at-risk 0.05: 0.0616

But: Under the bad distribution we have found, *the value-at-risk 0.05 of the profit of our nice portfolio is as low as 0.0335!*

♠ Given our data, we can specify two distributions:

• "bad" one, where the profits of assets are *uncorrelated* and take their extreme values with probabilities 1/2, and the probability of crisis is > 0.05;

• "good" one, where the profits of assets are *independent* and take their extreme values with probabilities 1/2.

♡ Known in advance to be the true ones, *these distributions lead to completely different investment policies*, as far as optimizing value-at-risk 0.05 is concerned.

Quiz: What is the probability of crisis for the good distribution?

Quiz: Assume that one of the above distribution indeed takes place, and our historical data are sampled from the true distribution, independently across time, for T years. How large should be T in order to infer from the historical data, with confidence 0.95, which one of the two candidate distributions is the true one?

- at least 1 year
- at least 2 years
- at least 4 years
- at least 8 years
- at least 16 years
- at least 32 years

Canonical and Standard formats of LO programs

♣ We can somehow "standardize" the formats in which LO programs are written.

• every linear equality/inequality can be equivalently rewritten in the form where the left hand side is a weighted sum $\sum_{j=1}^{n} a_j x_j$ of variables x_j with coefficients, and the right hand side is a real constant:

$$2x_1 \geq 20 - x_2 \Leftrightarrow 2x_1 + x_2 \geq 20$$

• the sign of a nonstrict linear inequality always can be made "$\leq$", since the inequality $\sum_j a_j x_j \geq b$ is equivalent to $\sum_j [-a_j] x_j \leq [-b]$:

$$2x_1 + x_2 \geq 20 \Leftrightarrow -2x_1 - x_2 \leq -20$$

• a linear equality constraint $\sum_j a_j x_j = b$ can be represented equivalently by the pair of opposite inequalities $\sum_j a_j x_j \leq b$, $\sum_j [-a_j] x_j \leq [-b]$:

$$2x_1 - x_2 = 5 \Leftrightarrow \begin{cases} 2x_1 - x_2 \leq 5 \\ -2x_1 + x_2 \leq -5 \end{cases}$$

• to minimize a linear function $\sum_j c_j x_j$ is exactly the same to maximize the linear function $\sum_j [-c_j] x_j$.

♣ *Every LO program is equivalent to an LO program in the canonical form*, where the objective should be maximized, and the constraints are "$\leq$" inequalities:

$$\text{Opt} = \max_x \left\{ \sum_{j=1}^n c_j x_j : \begin{array}{rcl} \sum_{j=1}^n a_{1j} x_j & \leq & b_1 \\ \sum_{j=1}^n a_{2j} x_j & \leq & b_2 \\ \vdots & \vdots & \vdots \\ \sum_{j=1}^n a_{mj} x_j & \leq & b_m \end{array} \right\}$$

["term-wise" notation]

Example:

$$\text{Opt} = \max_{x_1,x_2,x_3} \left\{ 2x_1 + 3x_2 - x_3 : \begin{array}{rcl} 3x_1 + 4x_2 + 5x_3 & \leq & 6 \\ 7x_1 + 8x_2 + 9x_3 & \leq & 10 \end{array} \right\}$$

$$\Leftrightarrow \quad \text{Opt} = \max_x \left\{ c^T x : a_i^T x \leq b_i,\ 1 \leq i \leq m \right\}$$

["constraint-wise" notation]

$$\Leftrightarrow \quad \text{Opt} = \max_x \left\{ c^T x : Ax \leq b \right\}$$

["matrix-vector" notation]

$c = [c_1; ...; c_n],\ b = [b_1; ...; b_m],\ a_i = [a_{i1}; ...; a_{in}]$

$$A = [a_1^T; a_2^T; ...; a_m^T] = \begin{bmatrix} a_{11} & a_{12} & ... & a_{1n} \\ a_{21} & a_{22} & ... & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1} & a_{m2} & ... & a_{mn} \end{bmatrix}$$

Recall: a^T is the transpose of vector/matrix a. For column vectors a, x of the same dimension n,

$$a^T x = \begin{bmatrix} a_1 & \cdots & a_n \end{bmatrix} \cdot \begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix} = a_1 x_1 + a_2 x_2 + ... + a_n x_n$$

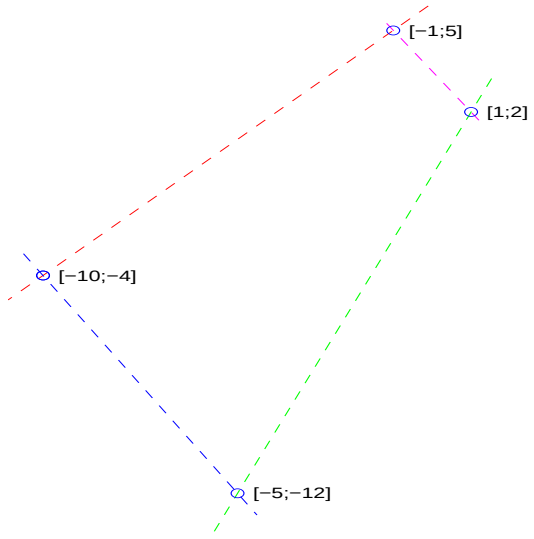
In Example: $m = 2,\ n = 2,$

$$c = \begin{bmatrix} 2 \\ 3 \\ -1 \end{bmatrix},\ b = \begin{bmatrix} 6 \\ 10 \end{bmatrix},\ A = \begin{bmatrix} a_1^T \\ a_2^T \end{bmatrix} = \begin{bmatrix} 3 & 4 & 5 \\ 7 & 8 & 9 \end{bmatrix}$$

♠ A set $X \subset \mathbb{R}^n$ given by $X = \{x : Ax \leq b\}$ – the solution set of a finite system of nonstrict linear inequalities $a_i^T x \leq b_i$, $1 \leq i \leq m$ in variables $x \in \mathbb{R}^n$ – is called *polyhedral set*, or *polyhedron*. An LO program in the canonical form is to maximize a linear objective over a polyhedral set, called the *feasible set* (or *feasible domain*) of the program.

♠ **Note:** The solution set of an arbitrary finite system of linear equalities and nonstrict inequalities in variables $x \in \mathbb{R}^n$ is a polyhedral set.

$$\max_x \left\{ x_2 : \left\{ \begin{array}{rcl} -x_1 + x_2 & \leq & 6 \\ 3x_1 + 2x_2 & \leq & 7 \\ 7x_1 - 3x_2 & \leq & 1 \\ -8x_1 - 5x_2 & \leq & 100 \end{array} \right\} \right\}$$



LO program and its feasible domain

♣ **Standard form of an LO program** is to maximize a linear function over the intersection of the *nonnegative orthant* $\mathbb{R}^n_+ = \{x \in \mathbb{R}^n : x \geq 0\}$ and the *feasible plane* $\{x : Ax = b\}$:

$$\mathrm{Opt} = \max_x \left\{ \sum_{j=1}^n c_j x_j : \begin{array}{r} \sum_{j=1}^n a_{1j} x_j = b_1 \\ \sum_{j=1}^n a_{2j} x_j = b_2 \\ \cdots\cdots\cdots\cdots\cdots\cdots \\ \sum_{j=1}^n a_{mj} x_j = b_m \\ x_j \geq 0,\ j = 1, ..., n \end{array} \right\}$$

["term-wise" notation]

Example:

$$\mathrm{Opt} = \max_{x_1, x_2, x_3} \left\{ 2x_1 + 3x_2 - x_3 : \begin{array}{rcl} 3x_1 + 4x_2 + 5x_3 & = & 6 \\ 7x_1 + 8x_2 + 9x_3 & = & 10 \\ x_1 \geq 0, x_2 \geq 0, x_3 \geq 0 & & \end{array} \right\}$$

$$\Leftrightarrow \quad \mathrm{Opt} = \max_x \left\{ c^T x : \begin{array}{l} a_i^T x = b_i,\ 1 \leq i \leq m \\ x_j \geq 0,\ 1 \leq j \leq n \end{array} \right\}$$

["constraint-wise" notation]

$$\Leftrightarrow \quad \mathrm{Opt} = \max_x \left\{ c^T x : Ax = b, x \geq 0 \right\}$$

["matrix-vector" notation]

$c = [c_1; ...; c_n]$, $b = [b_1; ...; b_m]$, $a_i = [a_{i1}; ...; a_{in}]$,

$$A = [a_1^T; a_2^T; ...; a_m^T] = \begin{bmatrix} a_{11} & a_{12} & ... & a_{1n} \\ a_{21} & a_{22} & ... & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1} & a_{m2} & ... & a_{mn} \end{bmatrix}$$

In the standard form LO program

- all variables are restricted to be nonnegative
- all "general-type" linear constraints are equalities.

♣ **Observation:** *The standard form of LO program is universal: every LO program is equivalent to an LO program in the standard form.*

Indeed, it suffices to convert to the standard form a canonical LO $\max_x\{c^Tx : Ax \leq b\}$. This can be done as follows:

• we introduce *slack variables*, one per inequality constraint, and rewrite the problem equivalently as

$$\max_{x,s}\{c^Tx : Ax+s = b,\ s \geq 0\}$$

• we further represent x as the difference of two new *nonnegative* vector variables $x = u - v$, thus arriving at the program

$$\max_{u,v,s}\left\{c^Tu - c^Tv : Au - Av + s = b,\ [u; v; s] \geq 0\right\}.$$

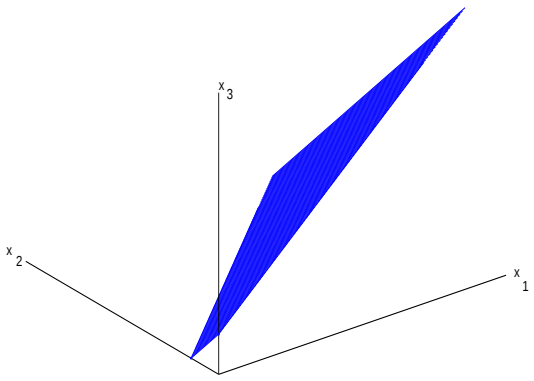
Illustration:

$$\begin{aligned}
& \text{Opt} = \max_{x} [2x_1 + 3x_2 - x_3] \\
\text{s.t.} \quad & \\
& 3x_1 + 4x_2 + 5x_3 \leq 6 \\
& 7x_1 + 8x_2 + 9x_3 \leq 10
\end{aligned}$$

$$\begin{aligned}
\Leftrightarrow \quad & \text{Opt} = \max_{x,s} [2x_1 + 3x_2 - x_3] \\
\text{s.t.} \quad & \\
& 3x_1 + 4x_2 + 5x_6 + s_1 = 6 \\
& 7x_1 + 8x_2 + 9x_3 + s_2 = 10 \\
& s_1 \geq 0, s_2 \geq 0
\end{aligned}$$

$$\begin{aligned}
\Leftrightarrow \quad & \text{Opt} = \max_{u,v,s} [2[u_1 - v_1] + 3[u_2 - v_2] - [u_3 - v_3]] \\
\text{s.t.} \quad & \\
& 3[u_1 - v_1] + 4[u_2 - v_2] + 5[u_3 - v_3] + s_1 = 6 \\
& 7[u_1 - v_1] + 8[u_2 - v_2] + 9[u_3 - v_3] + s_2 = 10 \\
& s_1 \geq 0, s_2 \geq 0, u_1 \geq 0, u_2 \geq 0, u_3 \geq 0, \\
& v_1 \geq 0, v_2 \geq 0, v_3 \geq 0,
\end{aligned}$$

$$\max_x \{-2x_1 + x_3 : -x_1 + x_2 + x_3 = 1,\, x \geq 0\}$$



Standard form LO program
and its feasible domain

LO Terminology

$$\mathrm{Opt} = \max_x \{c^T x : Ax \le b\} \qquad [A : m \times n] \qquad \text{(LO)}$$

• The variable vector $x = [x_1; ...; x_n]$ in (LO) is called the *decision vector* of the program; its entries x_j are called *decision variables*.

• The linear function to be maximized

$$c^T x = c_1 x_1 + ... + c_n x_n$$

is called the *objective function* (or *objective*) of the program, and the inequalities

$$\underbrace{a_{i1} x_1 + ... + a_{in} x_n}_{a_i^T x} \le b_i, \ i = 1, ..., m$$

are called the *constraints*.

• The *structure* of (LO) reduces to the *sizes* m (number of constraints) and n (number of variables). The *data* of (LO) is the collection of numerical values of the coefficients in the *cost vector*

$$c = [c_1; ...; c_n],$$

in the *right hand side vector*

$$b = [b_1; ...; b_m],$$

and in the *constraint matrix*

$$A = [a_{ij}]_{\substack{1 \le i \le m \\ 1 \le j \le n}}.$$

$$\mathrm{Opt} = \max_x \{c^T x : Ax \le b\} \qquad [A : m \times n] \qquad \text{(LO)}$$

• A *solution* to (LO) is an arbitrary value of the decision vector.

• A solution x is called *feasible* if it satisfies the constraints: $Ax \le b$.

The set of all feasible solutions is called the *feasible set* (or *feasible domain*) of the program.

• The program is called *feasible*, if the feasible set is nonempty, and is called *infeasible* otherwise.

Example: The vector $x = [2; 0; -0.1]$ is a solution to the LO program

$$\mathrm{Opt} = \max_{x_1,x_2,x_3} [2x_1 + 3x_2 - x_3]$$

s.t.

$$\begin{array}{rcl} 3x_1 + 4x_2 + 5x_3 & \le & 6 \\ 7x_1 + 8x_2 + 9x_3 & \le & 10 \end{array}$$

This solution is *infeasible*.

• $x = [-1; -1; -1]$ is a *feasible* solution to the same problem

$\Rightarrow$ Feasible solutions do exist

$\Rightarrow$ *The problem is feasible*

• Adding to the problem additional constraint

$$10x_1 + 12x_2 + 14x_3 \ge 17$$

we make the problem *infeasible*

Quiz: Why?

Optimal value of LO program

$$\mathrm{Opt} = \max_x \{c^T x : Ax \le b\} \qquad \text{(LO)}$$
$$[A : m \times n]$$

♠ Given a program (LO), there are three possibilities:

• *the program is infeasible*. In this case, $\mathrm{Opt} = -\infty$ by definition.

• the program is feasible, and the objective is *not* bounded from above on the feasible set, i.e., for every $a \in \mathbb{R}$ there exists a feasible solution x such that $c^T x > a$. In this case, the program is called *unbounded*, and $\mathrm{Opt} = +\infty$ by definition.

♡ The program which is not unbounded is called *bounded*; a program is bounded iff its objective is bounded from above on the feasible set (e.g., due to the fact that the latter is empty).

iff: if and only if.

• the program is feasible, and the objective is bounded from above on the feasible set: there exists a real a such that $c^T x \le a$ for all feasible solutions x. In this case, the optimal value Opt is the supremum, over the feasible solutions, of the values of the objective at a solution. Thus, $\mathrm{Opt} = 5$ means that

— there is no feasible solution with the value of the objective > 5

— for every $\epsilon > 0$, there is a feasible solution with the value of the objective $\ge 5 - \epsilon$

$$\mathrm{Opt} = \max_x \{c^T x : Ax \leq b\} \qquad [A : m \times n] \qquad \text{(LO)}$$

• a solution to the program is called *optimal*, if it is feasible, and the value of the objective at the solution equals to Opt. A program is called *solvable*, if it admits an optimal solution.

♠ **In the case of a minimization problem**

$$\mathrm{Opt} = \min_x \{c^T x : Ax \leq b\} \qquad [A : m \times n] \qquad \text{(LO)}$$

• the optimal value of an infeasible program is $+\infty$,

• *unboundedness* means that the objective to be minimized is *not* bounded *from below* on the feasible set. The optimal value in an unbounded problem is $-\infty$

• the optimal value of a *feasible and bounded* program is the *infinum* of values of the objective at feasible solutions to the program. Thus, *for a minimization problem*, $\mathrm{Opt} = 5$ means that

— there is no feasible solution with the value of the objective < 5

— for every $\epsilon > 0$, there is a feasible solution with the value of the objective $\leq 5 + \epsilon$

♣ The notions of feasibility, boundedness, solvability and optimality can be straightforwardly extended from LO programs to arbitrary MP ones. With this extension, a solvable problem definitely is feasible and bounded, while the inverse not necessarily is true, as is illustrated by the program

$$\mathrm{Opt} = \max_x \{-\exp\{-x\} : x \geq 0\},$$

$\mathrm{Opt} = 0$, but the optimal value is not achieved – there is no feasible solution where the objective is equal to 0! As a result, the program is unsolvable.

$\Rightarrow$ In general, the fact that an optimization program has a "legitimate" – real, and not $\pm\infty$ – optimal value, is *strictly weaker* that the fact that the program is solvable (i.e., has an optimal solution).

♠ In LO the situation is much better: we shall prove that *an LO program is solvable iff it is feasible and bounded*.

Quiz: What is the "status" (feasible/infeasible, bounded/unbounded, solvable/unsolvable) of LO programs below?

A:

$$\text{Opt} = \max_{x_1,x_2,x_3} [2x_1 + 3x_2 - x_3]$$

s.t.

$$\begin{array}{rcl} 3x_1 + 4x_2 + 5x_3 & \leq & 6 \\ 7x_1 + 8x_2 + 9x_3 & \leq & 10 \end{array}$$

feasible: bounded: solvable:

B:

$$\text{Opt} = \max_{x_1,x_2,x_3} [2x_1 + 3x_2 - x_3]$$

s.t.

$$\begin{array}{rcl} 3x_1 + 4x_2 + 5x_3 & \leq & 6 \\ 7x_1 + 8x_2 + 9x_3 & \leq & 10 \\ 10x_1 + 12x_2 + 14x_3 & \geq & 17 \end{array}$$

feasible: bounded: solvable:

C:

$$\text{Opt} = \max_{x_1,x_2,x_3} [2x_1 + 3x_2 - x_3]$$

s.t.

$$\begin{array}{rcl} 3x_1 + 4x_2 + 5x_3 & \leq & 6 \\ 7x_1 + 8x_2 + 9x_3 & \leq & 10 \end{array}$$

$$-1 \leq x_1 \leq 1, -1 \leq x_2 \leq 1, -1 \leq x_3 \leq 1$$

feasible: bounded: solvable:

Examples of LO Models

♣ **Diet Problem:** There are n types of products and m types of nutrition elements. A unit of product # j contains p_{ij} grams of nutrition element # i and costs c_j. The daily consumption of a nutrition element # i should be within given bounds $[\underline{b}_i, \overline{b}_i]$. Find the cheapest possible "diet" – mixture of products – which provides appropriate daily amounts of every one of the nutrition elements.

Denoting x_j the amount of j-th product in a diet, the LO model reads

$$\min_x \sum_{j=1}^n c_j x_j \quad \text{[cost to be minimized]}$$

subject to

$$\left.\begin{array}{l} \sum_{j=1}^n p_{ij} x_j \geq \underline{b}_i \\ \sum_{j=1}^n p_{ij} x_j \leq \overline{b}_i \\ 1 \leq i \leq m \end{array}\right\} \quad \left[\begin{array}{l} \text{upper \& lower bounds on} \\ \text{the contents of nutrition} \\ \text{elements in a diet} \end{array}\right]$$

$$x_j \geq 0, 1 \leq j \leq n \quad \left[\begin{array}{l} \text{you cannot put into a} \\ \text{diet a negative amount} \\ \text{of a product} \end{array}\right]$$

• Diet problem is routinely used in nourishment of poultry, livestock, etc. As about nourishment of humans, the model is of no much use since it ignores factors like food's taste, food diversity requirements, etc.

• Here is the optimal daily human diet as computed by the software at

`http://www.neos-guide.org/content/diet-problem-demo`

(when solving the problem, I allowed to use all 64 kinds of food offered by the code; the prices of 2010 are used):

Food	Serving	Cost
Raw Carrots	0.12 cups shredded	0.02
Peanut Butter	7.20 Tbsp	0.25
Popcorn, Air-Popped	4.82 Oz	0.19
Potatoes, Baked	1.77 cups	0.21
Skim Milk	2.17 C	0.28

Daily cost $ 0.96

♣ **Production planning:** A factory

- consumes R types of resources (electricity, raw materials of various kinds, various sorts of manpower, processing times at different devices, etc.)
- produces P types of products.

♠ There are n possible production processes, j-th of them can be used with "intensity" x_j (fraction of the planning period during which j-th process is used).

- Used at unit intensity, production process # j consumes A_{rj} units of resource r, $1 \leq r \leq R$, and yields C_{pj} units of product p, $1 \leq p \leq P$.
- The profit of selling a unit of product p is c_p.

♠ Given upper bounds $b_1, ..., b_R$ on the recourses available during the planning period, and lower bounds $d_1, ..., d_P$ on the amounts of products to be produced, find a production plan which maximizes the profit.

♠ Denoting by x_j the intensity of production process j, the LO model reads:

$$\max_x \sum_{j=1}^{n} \left(\sum_{p=1}^{P} c_p C_{pj}\right) x_j \quad \text{[profit to be maximized]}$$

subject to

$$\sum_{j=1}^{n} A_{rj} x_j \le b_r,\ 1 \le r \le R \quad \left[\begin{array}{l}\text{upper bounds on}\\ \text{resources should}\\ \text{be met}\end{array}\right]$$

$$\sum_{j=1}^{n} C_{pj} x_j \ge d_p,\ 1 \le p \le P \quad \left[\begin{array}{l}\text{lower bounds on}\\ \text{products should}\\ \text{be met}\end{array}\right]$$

$$\left.\begin{array}{l}\sum_{j=1}^{n} x_j \le 1\\ x_j \ge 0,\ 1 \le j \le n\end{array}\right\} \quad \left[\begin{array}{l}\text{total intensity should be} \le 1\\ \text{and intensities must be}\\ \text{nonnegative}\end{array}\right]$$

Implicit assumptions:

- all production can be sold
- there are no setup costs when switching between production processes
- the products are infinitely divisible

♣ **Inventory:** An inventory operates over time horizon of T days $1, ..., T$ and handles K types of products.

• Products share common warehouse with space C. Unit of product k takes space $c_k \geq 0$ and its day-long storage costs h_k.

• Inventory is replenished via ordering from a supplier; a replenishment order sent in the beginning of day t is executed immediately, and ordering a unit of product k costs $o_k \geq 0$.

• The inventory is affected by external demand of d_{tk} units of product k in day t. Backlog is allowed, and a day-long delay in supplying a unit of product k costs $p_k \geq 0$.

♠ *Given the initial amounts s_{0k}, $k = 1, ..., K$, of products in warehouse, all the (nonnegative) cost coefficients and the demands d_{tk}, we want to specify the replenishment orders v_{tk} (v_{tk} is the amount of product k which is ordered from the supplier at the beginning of day t) in such a way that at the end of day T there is no backlogged demand, and we want to meet this requirement at as small total inventory management cost as possible.*

Building the model

1. Let *state variable* s_{tk} be the amount of product k stored at warehouse at the end of day t. s_{tk} can be negative, meaning that at the end of day t the inventory owes the customers $|s_{tk}|$ units of product k.
Let also U be an upper bound on the total management cost. The problem reads:

$$\min_{U,v,s} \quad U$$

$$U \geq \sum_{\substack{1\leq k\leq K,\\ 1\leq t\leq T}} [o_k v_{tk} + \max[h_k s_{tk}, 0] + \max[-p_k s_{tk}, 0]]$$

[cost description]

$$s_{tk} = s_{t-1,k} + v_{tk} - d_{tk}, 1 \leq t \leq T, 1 \leq k \leq K$$

[state equations]

$$\sum_{k=1}^{K} \max[c_k s_{tk}, 0] \leq C,\ 1 \leq t \leq T$$

[space restriction should be met]

$$s_{Tk} \geq 0, 1 \leq k \leq K$$

[no backlogged demand at the end]

$$v_{tk} \geq 0, 1 \leq k \leq K, 1 \leq t \leq T$$

[no returns to the supplier are allowed]

Implicit assumption: replenishment orders are executed, and the demands are shipped to customers at the beginning of day t.

Quiz: Is the above an LO program?

♠ Our problem is *not* and LO program – it includes *nonlinear* constraints of the form

$$\sum_{k,t}[o_k v_{tk} + \max[h_k s_{tk}, 0] + \max[-p_k s_{tk}, 0]] \le U$$
$$\sum_k \max[c_k s_{tk}, 0] \le C,\ t = 1, ..., T$$

Let us *represented equivalently* "troublemaking" constraints by linear constraints.

• for every term $\max[h_k s_{tk}, 0]$, introduce a new decision variable x_{tk} – an *upper bound* on the term. The fact that x_{tk} upper-bounds the term can be represented by linear constraints

$$x_{tk} \ge h_k s_{tk} \text{ and } x_{tk} \ge 0$$

• similarly, for every term $\max[-p_k s_{tk}, 0]$, introduce a variable y_{tk} upper-bounding the term and say that it indeed is so:

$$y_{tk} \ge -p_k s_{tk} \text{ and } y_{tk} \ge 0$$

• similarly, for every term $\max[c_k s_{tk}, 0]$, introduce a variable z_{tk} upper-bounding the term and say that it indeed is so:

$$z_{tk} \ge c_k s_{tk} \text{ and } z_{tk} \ge 0$$

• *Rewrite the problem by replacing all troublemaking terms with their upper bounds and adding the "upper-bounding" constraints.*

♠ Applying the above construction to the Inventory problem, we end up with the following LO model:

$$
\begin{array}{l}
\min\limits_{U,v,s} \quad U \\
U \geq \sum\limits_{\substack{1\leq k\leq K,\\ 1\leq t\leq T}} [o_k v_{tk} + \max[h_k s_{tk}, 0] + \max[-p_k s_{tk}, 0]] \\
s_{tk} = s_{t-1,k} + v_{tk} - d_{tk}, 1 \leq t \leq T, 1 \leq k \leq K \\
\sum_{k=1}^{K} \max[c_k s_{tk}, 0] \leq C,\ 1 \leq t \leq T \\
s_{Tk} \geq 0, 1 \leq k \leq K \\
v_{tk} \geq 0, 1 \leq k \leq K, 1 \leq t \leq T
\end{array}
$$

$$\Downarrow$$

$$
\begin{array}{l}
\min\limits_{U,v,s,x,y,z} \quad U \\
U \geq \sum\limits_{k,t} [o_k v_{tk} + x_{tk} + y_{tk}] \\
x_{tk} \geq h_k s_{tk},\ x_{tk} \geq 0,\ 1 \leq k \leq K, 1 \leq t \leq T \\
y_{tk} \geq -p_k s_{tk},\ y_{tk} \geq 0,\ 1 \leq k \leq K, 1 \leq t \leq T \\
s_{tk} = s_{t-1,k} + v_{tk} - d_{tk}, 1 \leq t \leq T, 1 \leq k \leq K \\
\sum_{k=1}^{K} z_{tk} \leq C,\ 1 \leq t \leq T \\
z_{tk} \geq c_k s_{tk},\ z_{tk} \geq 0,\ 1 \leq k \leq K, 1 \leq t \leq T \\
s_{Tk} \geq 0, 1 \leq k \leq K \\
v_{tk} \geq 0,\ 1 \leq k \leq K, 1 \leq t \leq T
\end{array}
$$

- The original and the reformulated programs are *equivalent* in the sense that

a. solution to the first problem is feasible iff it can be extended, by properly selected added variables x, y, z, to a feasible solution of the second problem

b. the objective functions in both problems are identical to each other

⇒ *Programs have the same optimal value, and feasible/optimal solution to the second problem induces a feasible, resp., optimal solution, with the same value of the objective, to the first problem.*

Example 1: *How to eliminate red nonlinearity in the constraint* $....+\max[x_1-x_2, x_1+x_3]+....\leq 5$?

Answer: Rewrite the constraint as

$$....+z+....\leq 5$$

and

$$z\geq x_1-x_2 \text{ and } z\geq x_1+x_3$$

Example 2: *How to eliminate magenta nonlinearity in the constraint* $....+\min[x_1-x_2, x_1+x_3]+....\geq 5$?

Answer: Rewrite the constraint as

$$....+z+....\geq 5$$

and

$$z\leq x_1-x_2 \text{ and } z\leq x_1+x_3$$

Example 3: *How to eliminate red nonlinearity in the constraint* $....+\max[x_1-x_2, x_1+x_3]+....\geq 5$?

Answer: The above recipe does not work! The constraint is equivalent to

$$....+z+....\geq 5$$

and

$$z\leq x_1-x_2 \text{ or } z\leq x_1+x_3$$

but this is *not* a *system* of constraints!

♠ In order to eliminate a nonlinear term

$$\text{Term}=\begin{matrix}\max\\ \min\end{matrix}[\text{linear expression}, ...\text{linear expression}]$$

in a constraint

$$....+\text{Term}+....\begin{matrix}\leq\\ \geq\end{matrix}....$$

the type of the term should match the type of the inequality. Good cases are

- max-type term and $\leq$-type inequality
- min-type term and $\geq$-type inequality.

♠ **Note:** I spoke only about eliminating nonlinearities in *constraints*. This indeed is the only interesting case, since *in optimization, linearity of objective is "for free."* Indeed, by adding extra variable, you always can make your objective linear by converting the original objective into a new constraint:

$$\max_x f(x) \text{ s.t. a system of constraints on } x$$

is equivalent to

$$\max_{x,t} t \text{ s.t. } \left\{ \begin{array}{c} \text{the original system} \\ \text{of constraints on } x \\ \text{and the constraint} \\ f(x) \geq t \end{array} \right.$$

Example of LO Model in Engineering: Sparsity-oriented Signal Processing

♠ Traditional applications of LO primarily deal with various *Decision Making* problems: production planning, supply chain management, transportation, facility location, etc.

Recent years witness steady growth of applications of LO in Engineering. A nice example is ℓ_1 *minimization in Sparsity-oriented Signal Processing.*

♠ **Basic Signal Processing** problem is to recover *unknown* signal x_* (which is an n-dimensional vector) from its observation

$$y = A(x_*) + \xi$$

• $x \mapsto A(x) : \mathbb{R}^n \to \mathbb{R}^m$: *known* "signal-to-observation" transformation

• ξ: observation noise.

♣ In many applications, the signal-to-observation transformation is just *linear*:

$$A(x) = Ax \text{ for some known } m \times n \text{ matrix } A.$$

♠ Assume from now on that $A(\cdot)$ is linear

$\Rightarrow$ the recovery problem is just *to solve a system of linear equations*

$$Ax = b := Ax_*$$

given $m \times n$ matrix A and a noisy observation y of the "true" right hand side b.

♣ **Problem of interest:** *to solve a linear system*

$$Ax = b := Ax_*$$

given $m \times n$ matrix A and a noisy observation y of the "true" right hand side b.

♠ As of now, there are two typical settings of the problem:

• $m \geq n$ (typically, $m \gg n$) — we have (much) more observations than unknowns. This is the classical case studied in numerical Linear Algebra (where noise is non-random) and Statistics (where noise is random). Unless A is "pathological," the only difficulty here is the presence of noise. The challenge is to reproduce well the true signal while suppressing as much as possible the influence of noise.

• $m < n$ (and even $m \ll n$) – we have (much) less observations than unknowns.

Till recently, this case was thought of as completely meaningless. Indeed, as Linear Algebra says, *an underdetermined* (with more unknowns than equations) *system of linear equations either has no solutions at all, or has infinitely many solutions which can be arbitrarily far away from each other.*

$\Rightarrow$ *When $m < n$, the true signal cannot be recovered from observations even in the noiseless case!*

♠ **Remedy:** Add some information on the true signal.

♣ **Problem of interest:** *to solve a linear system*

$$Ax = b := Ax_*$$

given $m \times n$ matrix A and a noisy observation y of the "true" right hand side b in the case of $m \ll n$

♠ **Sparsity-oriented remedy:** *Reduce the problem to the one where the signal is sparse – has $s \ll n$ nonzero entries, and utilize sparsity in your recovery routine.*

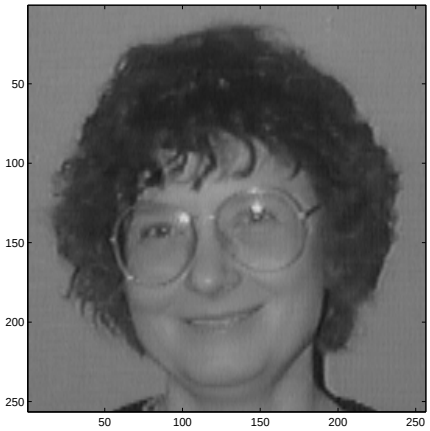
♠ **Fact:** *Many real-life signals x when presented by their coefficients in properly selected basis ("dictionary") B:*

$$x = Bu$$

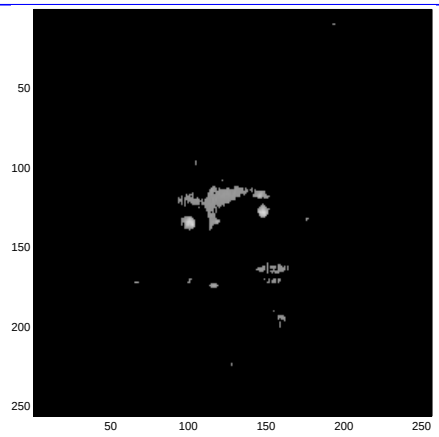
- columns of B: vectors of basis B
- u: coefficients of x in basis B

become sparse (or nearly so): u has just $s \ll n$ nonzero entries (or can be well approximated by vector with $s \ll n$ nonzero entries).

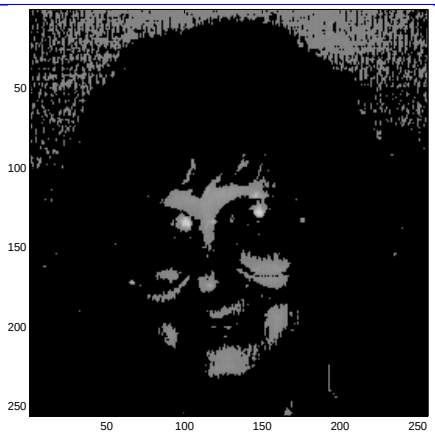
Illustration: The 256×256 image



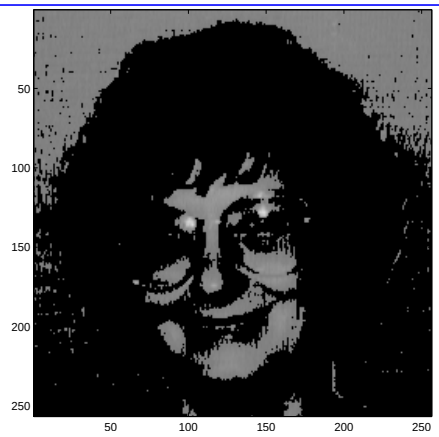
can be thought of as $256^2 = 65536$-dimensional vector (write down the intensities of pixels column by column). "As is," this vector is not sparse and cannot be approximated well by highly sparse vectors. This is what happens when we keep several leading (i.e., largest in magnitude) entries and zero out all other entries:



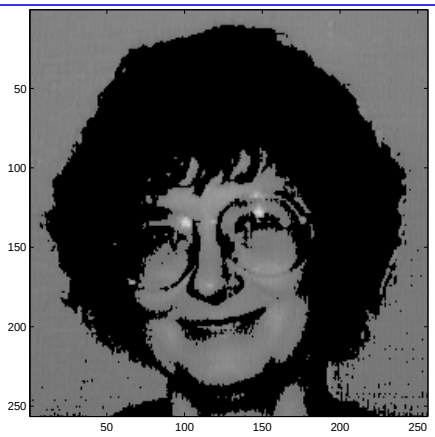
1% of leading entries kept



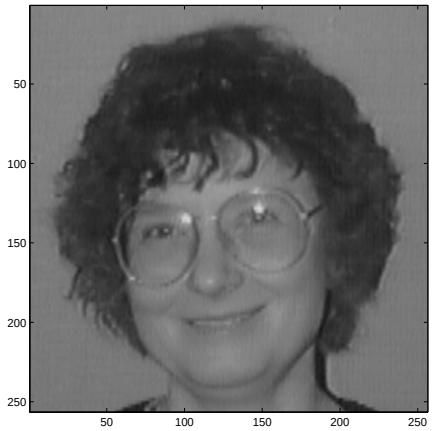
10% of leading entries kept



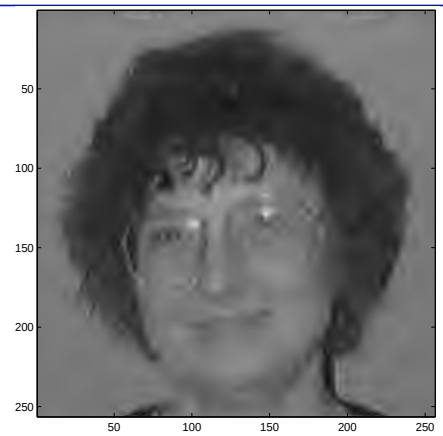
25% of leading entries kept



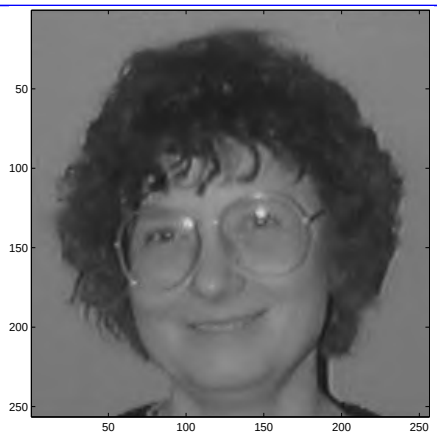
50% of leading entries kept



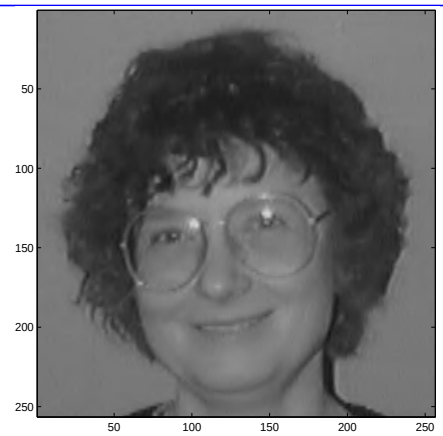
However, the image (same as other "non-pathological" images) is nearly sparse when represented in *wavelet* basis:



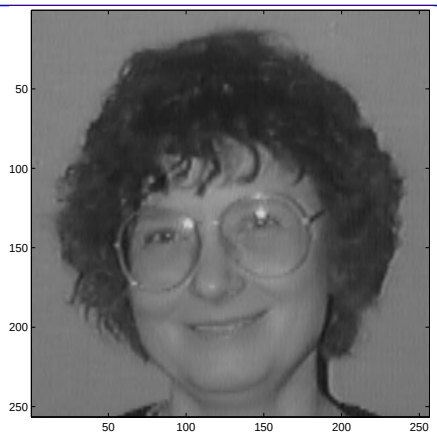
1% of leading wavelet coefficients kept



5% of leading wavelet coefficients kept



10% of leading wavelet coefficients kept



25% of leading wavelet coefficients kept

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

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

the situation reduces to the one when the signal to be recovered is just sparse.

Indeed, we can first recover *sparse* u_* from observations

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

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

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

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

(?) How to recover a *sparse* (or nearly so) n-dimensional signal x from $m \ll n$ observations

$$y = Ax_* + \xi \ ?$$

♠ To get an idea, consider the case when x_* is exactly sparse – has $s \ll n$ nonzero entries – and there is no observation noise:

$$y = Ax_*$$

• *If we knew the positions $i_1, ..., i_s$ of the nonzero entries in x_*,* we could recover x_* by solving the system *with just s unknowns:*

$$y = \left[A_{i_1}, ..., A_{i_s}\right] \cdot \left[x_{i_1}; ...; x_{i_s}\right]. \quad (!)$$

When $s \leq m$ (which, with $s \ll n$, still allows for $m \ll n$), we would get *over-determined* system of linear equations on the nonzero entries in x. Assuming A "non-pathologic," so that every $s \leq m$ columns of A are linearly independent, (!) has a unique solution which can be easily found.

But: *We never know in advance where the nonzeros in x are located!*

(?) How to recover a *sparse* (or nearly so) n-dimensional signal x from $m \ll n$ observations

$$y = Ax_* + \xi \ ?$$

♠ A straightforward way to account for the fact that we *never know where the nonzeros in x_* stand*, is to look for *the sparsest* solution to the system $y = Ax$. This amounts to solving the optimization problem

$$\min_x \mathrm{nnz}(x) \text{ s.t. } y = Ax \qquad (!)$$

- nnz(x): # of nonzero entries in x.

• It is easily seen that *if x_* is s-sparse and every $2s$ columns in A are linearly independent* (which is so when $2s \le m$, unless A is pathological), *then x_* is the unique optimal solution to (!)*, and thus our procedure recovers x_* *exactly*.

But: nnz(z) is a bad (nonconvex and discontinuous) function, so that (!) is a disastrously complicated combinatorial problem. Seemingly, the only way to solve (!) is to use brute force search where we test one by one all collections of potential locations of nonzero entries in a solution. Brute force is completely unrealistic: to recover s-sparse signal, it would require looking through *at least*

$$N = \binom{n}{s-1} = \frac{n!}{(s-1)!(n-s+1)!}$$

candidate solutions.

• with $s = 17, n = 128$, N is as large as $1.49 \cdot 10^{21}$

• with $s = 49, n = 1024$, N is as large as $3.94 \cdot 10^{84}$

(?) How to recover a *sparse* (or nearly so) n-dimensional signal x from $m \ll n$ observations

$$y = Ax_* + \xi \ ?$$

- Solving problem

$$\min_x \mathrm{nnz}(x) \text{ s.t. } y = Ax \qquad (!)$$

would yield the desired recovery, but (!) is heavily computationally intractable...

♠ **Partial remedy:** Replace the difficult to minimize objective $\mathrm{nnz}(\theta)$ with an "easy-to-minimize" objective, specifically, with $\|\theta\|_1 = \sum_i |\theta_i|$, thus arriving at ℓ_1-*recovery*

$$\widehat{x} = \mathrm{argmin}_x \{\textstyle\sum_i |x_i| : Ax = y := Ax_*\} \qquad (!!)$$

♠ **Observation:** (!!) is just an LO program!

Indeed,

- the constraints in (!!) are linear equalities.
- $|x_i| = \max[x_i, -x_i]$, so that the terms in the objective can be "linearized."

♠ The LO reformulation of (!!) is

$$\min_{x,z} \left\{\textstyle\sum_j z_j : Ax = y, z_j \geq x_j, z_j \geq -x_j \, \forall j \leq n\right\}.$$

• *In the noiseless case*, ℓ_1 recovery is given by

$$\widehat{x} = \mathrm{argmin}_x \{\textstyle\sum_i |x_i| : Ax = y := Ax_*\}$$

♠ When the observation y is noisy:

$$y = Ax_* + \xi$$

the constraint $Ax = y$ on a candidate recovery should be relaxed.

• *When we know an upper bound* δ on some norm $\|\xi\|$ of the noise ξ, a natural version of ℓ_1 recovery is

$$\widehat{x} \in \mathrm{Argmin}_x \{\textstyle\sum_i |x_i| : \|Ax - y\| \le \delta\} \qquad (*)$$

Note: When $\|\xi\| = \|\xi\|_\infty := \max_i |\xi_i|$ ("uniform norm"), $(*)$ reduces to the LO program

$$\min_{x,z} \left\{ \textstyle\sum_j z_j : \begin{array}{l} -z_j \le x_j \le z_j,\ 1 \le j \le n \\ y_i - \delta \le [Ax]_i \le y_i + \delta, 1 \le i \le m \end{array} \right\}$$

• *When the noise* ξ *is random with zero mean*, there are reasons to define ℓ_1 recovery by *Dantzig Selector:*

$$\widehat{x} \in \mathrm{Argmin}_x \{\textstyle\sum_i |x_i| : \|Q(Ax - y)\|_\infty \le \delta\}$$

with $M \times m$ *contrast matrix* Q and $\delta > 0$ chosen according to noise's structure and intensity. This again is reducible to LO program, specifically,

$$\min_{x,z} \left\{ \textstyle\sum_j z_j : \begin{array}{l} -z_j \le x_j \le z_j,\ 1 \le j \le n \\ -\delta \le [QAx - Qy]_i \le \delta, 1 \le i \le M \end{array} \right\}$$

• **Note:** In Dantzig Selector proper, $Q = A^T$.

(?) How to recover a *sparse* (or nearly so) n-dimensional signal x from $m \ll n$ observations

$$y = Ax_* + \xi \ ?$$

(!) Use ℓ_1 minimization

$$\widehat{x} \in \mathrm{Argmin}_x \{\textstyle\sum_i |x_i| : \|Ax - y\| \leq \delta\}$$

♣ Compressed Sensing theory shows that *under appropriate assumptions on A, in a meaningful range of sizes m, n and sparsities s, ℓ_1-minimization recovers the unknown signal x_**

— *exactly,* when x_* is s-sparse and there is no observation noise,

— within inaccuracy $\leq C(A)[\delta_n + \delta_s]$

- δ_n: magnitude of noise
- δ_s: deviation of x_* from its best s-sparse approximation

♠ **Bad news:** "Appropriate assumptions on A" are *difficult to verify*

Partial remedy: there are conservative *verifiable* sufficient conditions for "appropriate assumptions."

♠ **Good news:** *For A drawn at random from natural distributions, "appropriate assumptions" are satisfied with overwhelming probability.*

• E.g., when entries in $m \times n$ matrix A are, independently of each other, sampled from Gaussian distribution, the resulting matrix, *with probability approaching 1 as m, n grow*, ensures the validity of ℓ_1 recovery of sparse signals with as many as

$$s = O(1)\frac{m}{\ln(n/m)}$$

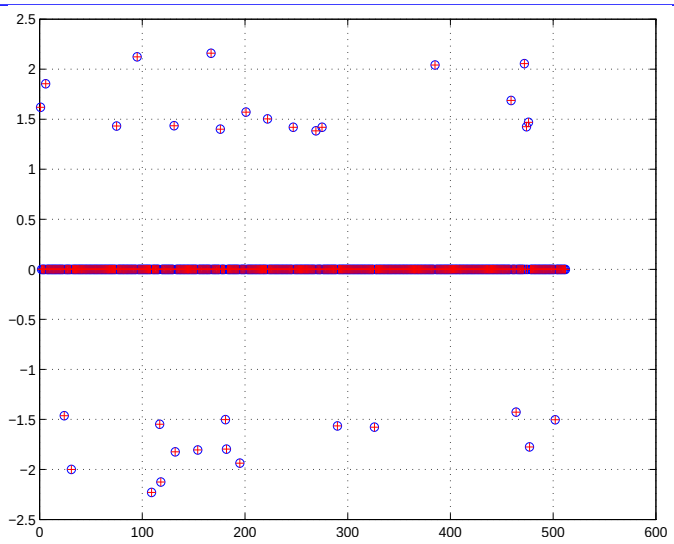
nonzero entries.

♠ **More good news:** In many applications (Imaging, Radars, Magnetic Resonance Tomography,...), signal acquisition via randomly generated matrices A makes perfect sense and results in significant acceleration of the acquisition process.

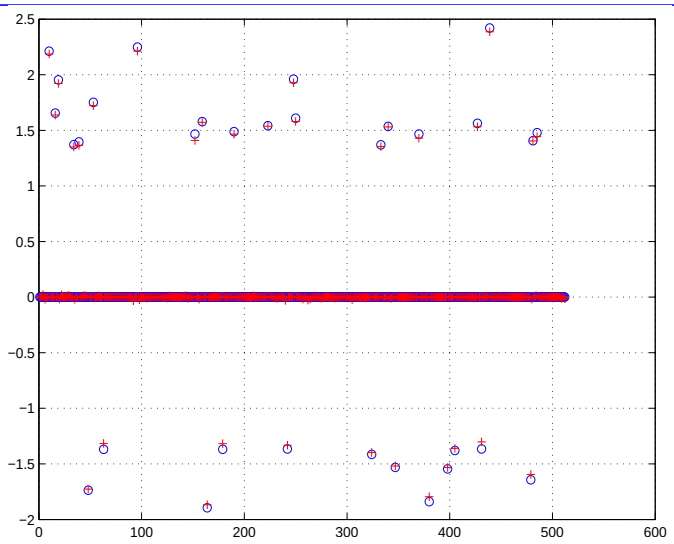
In these applications, signals of interest are sparse in properly selected bases

$\Rightarrow$ *With accelerated acquisition, no information is lost!*

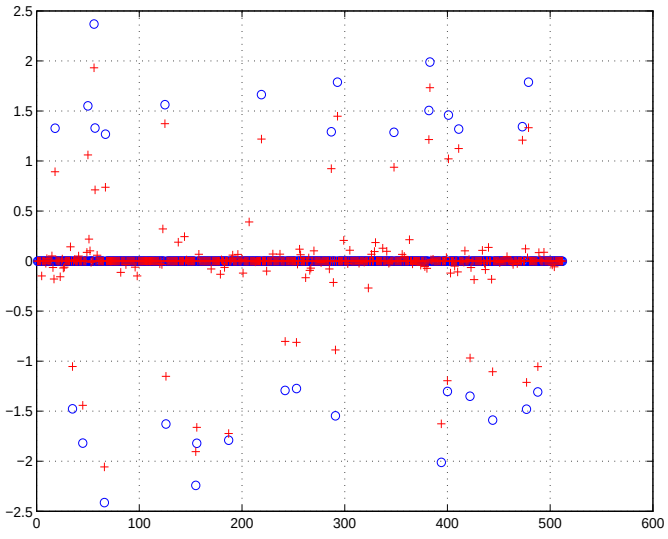
How it works:
Sparse recovery via Dantzig Selector



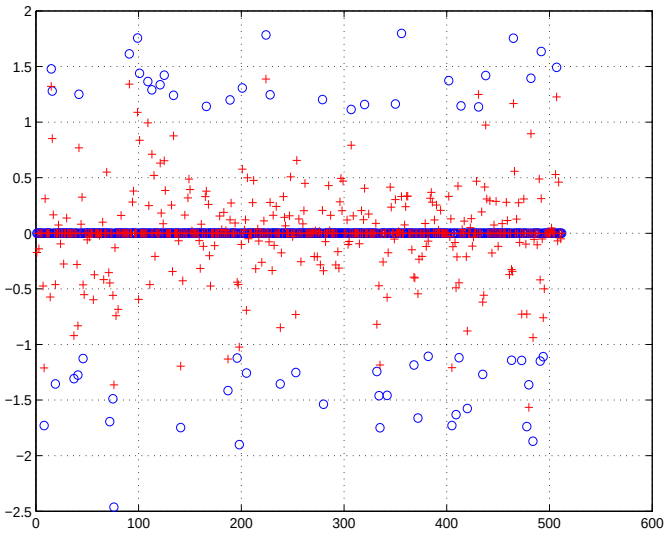
32 nonzero entries
No noise
Recovery error: $7.8 \cdot 10^{-9}$



32 nonzero entries
Noise's StD 0.01
Recovery error: 0.064



32 nonzero entries
Noise's StD 0.10
Recovery error: 0.66



64 nonzero entries
No noise
Recovery error: 1.47

o: signal $+$: recovery
256×512 Gaussian sensing matrix A

What Can Be Reduced to LO?

♣ We have seen several examples of optimization programs which can be reduced to LO, *although in its original "maiden" form the program is not an LO one.* Typical "maiden form" of a MP problem is

$$\text{(MP)}: \quad \max_{x \in X \subset \mathbb{R}^n} f(x)$$
$$X = \{x \in \mathbb{R}^n : g_i(x) \le 0,\ 1 \le i \le m\}$$

In LO,

- The objective is linear: $c_1 x_1 + ... + c_n x_n$
- The constraints are affine:

$$g_i(x) = a_{i1} x_1 + ... + a_{in} x_n - b_i$$

♠ **Recall:** Every MP program is equivalent to a program with linear objective.

Indeed, adding slack variable t, we can rewrite (MP) equivalently as

$$\max_{y=[x;t] \in Y} c^T y := t,$$
$$Y = \{[x; t] : g_i(x) \le 0, t - f(x) \le 0\}$$

⇒ *we can assume from the very beginning that the objective in* (MP) *is linear:* $f(x) = c^T x$.

$$\text{(MP)}: \quad \max_{x \in X \subset \mathbb{R}^n} c^T x$$
$$X = \{x \in \mathbb{R}^n : g_i(x) \le 0,\ 1 \le i \le m\}$$

♣ **Definition:** A polyhedral representation (p.r.) of a set $X \subset \mathbb{R}^n$ is a representation of X of the form:

$$X = \{x : \exists w : Px + Qw \le r\},$$

that is, a representation of X as the a projection onto the space of x-variables of the polyhedral set

$$X^+ = \{[x; w] : Px + Qw \le r\}$$

in the space of x, w-variables.

♠ **Observation:** *Given a p.r. of the feasible set X of* (MP)*, we can pose* (MP) *as the LO program*

$$\max_{[x;w]} \Big\{c^T x : Px + Qw \le r\Big\}.$$

♣ A polyhedral representation (p.r.) of a set $X \subset \mathbb{R}^n$ is a representation of X of the form:

$$X = \{x : \exists w : Px + Qw \leq r\}$$

♠ Examples of polyhedral representations:

- The set $X = \{x \in \mathbb{R}^n : \sum_i |x_i| \leq 1\}$ admits the p.r.

$$X = \left\{ x \in \mathbb{R}^n : \exists w \in \mathbb{R}^n : \begin{array}{r} -w_i \leq x_i \leq w_i, \\ 1 \leq i \leq n, \\ \sum_i w_i \leq 1 \end{array} \right\}.$$

- The set

$$X = \left\{ x \in \mathbb{R}^6 : \max[x_1, x_2, x_3] + 2\max[x_4, x_5, x_6] \leq x_1 - x_6 + 5 \right\}$$

admits the p.r.

$$X = \left\{ x \in \mathbb{R}^6 : \exists w \in \mathbb{R}^2 : \begin{array}{l} x_1 \leq w_1, x_2 \leq w_1, x_3 \leq w_1 \\ x_4 \leq w_2, x_5 \leq w_2, x_6 \leq w_2 \\ w_1 + 2w_2 \leq x_1 - x_6 + 5 \end{array} \right\}.$$

Whether a Polyhedrally Represented Set is Polyhedral?

♣ **Question:** Let X be given by a p.r.:

$$X = \{x \in \mathbb{R}^n : \exists w : Px + Qw \leq r\},$$

that is, as the *projection* of the solution set

$$Y = \{[x; w] : Px + Qw \leq r\} \qquad (*)$$

of a finite system of linear inequalities in variables x, w onto the space of x-variables.

Is it true that X is polyhedral, i.e., X is a solution set of finite system of linear inequalities *in variables x only?*

Fact: *Every polyhedrally representable set is polyhedral.*

Proof is given by the *Fourier — Motzkin elimination scheme* which demonstrates that the projection of the set $(*)$ onto the space of x-variables is a polyhedral set.

$$Y = \{[x; w] : Px + Qw \leq r\},\ w = [w_1; ...; w_m] \qquad (*)$$

Elimination step: eliminating a *single* slack variable. Given set $(*)$, assume that $m > 0$, and let Y^+ be the projection of Y on the space of variables $x, w_1, ..., w_{m-1}$:

$$Y^+ = \{[x; w_1; ...; w_{m-1}] : \exists w_m : Px + Qw \leq r\} \qquad (!)$$

We want to prove that Y^+ is polyhedral.

To get the idea, let us look at numerical example:

$$Y = \left\{ [x_1; x_2; w_1] : \begin{array}{rrrcrl} 2x_1 & -x_2 & +w_1 & \leq & 9 & (a) \\ x_1 & +6x_2 & -w_1 & \leq & 5 & (b) \\ 3x_1 & +x_2 & & \leq & 6 & (c) \\ x_1 & -7x_2 & -w_1 & \leq & -8 & (d) \\ 5x_1 & -6x_2 & +w_1 & \leq & 1 & (e) \end{array} \right\}$$

Question: What the constraints say about w_1?

- Constraints (a), (e) where the coefficients at w_1 are positive *upper-bound* w_1 in terms of the remaining variables. They read
$\begin{cases} w_1 \leq 9 - 2x_1 + x_2 \\ w_1 \leq 1 - 5x_1 + 6x_2 \end{cases}$
- Constraints (b), (d) where the coefficients at w_1 are negative *lower-bound* w_1 in terms of the remaining variables. They read
$\begin{cases} w_1 \geq -5 + x_1 + 6x_2 \\ w_1 \geq 8 + x_1 - 7x_2 \end{cases}$
- Constraint (c) where the coefficient at w_1 is zero "says nothing" about w_1. It reads $3x_1 + x_2 \leq 6$

Question: *When $[x_1; x_2]$ belongs to the projection of Y onto the x-space?*

$\Leftrightarrow$ *When $[x_1; x_2]$ can be augmented by properly selected w_1 to satisfy the constraints defining Y?*

Answer: *This is the case iff $[x_1; x_2]$ satisfies the black inequality and there is "enough room" for w_1 between the red upper bounds and green lower bounds on the variable.*

(!) The latter takes place iff every green lower bound is $\leq$ every red upper bound..

$$Y = \left\{ [x_1; x_2; w_1] : \begin{array}{rrrcrl} 2x_1 & -x_2 & +w_1 & \le & 9 & (a) \\ x_1 & +6x_2 & -w_1 & \le & 5 & (b) \\ 3x_1 & +x_2 & & \le & 6 & (c) \\ x_1 & -7x_2 & -w_1 & \le & -8 & (d) \\ 5x_1 & -6x_2 & +w_1 & \le & 1 & (e) \end{array} \right\}$$

is equivalent to

$$\left\{ \begin{array}{l} w_1 \le 9 - 2x_1 + x_2 \\ w_1 \le 1 - 5x_1 + 6x_2 \end{array} \right. , \quad \left\{ \begin{array}{l} w_1 \ge -5 + x_1 + 6x_2 \\ w_1 \ge 8 + x_1 - 7x_2 \end{array} \right. ,$$

$$3x_1 + x_2 \le 6$$

• In order for $x = [x_1; x_2]$ to be in the projection of Y onto the x-space, x should satisfy the black inequality and make every green lower bound on w_1 to be $\le$ every red upper bound on w_1.

$\Rightarrow$ The projection of Y on the x-space is given by the system of linear inequalities

$$\begin{array}{rcl} 3x_1 + x_2 & \le & 6, \\ -5 + x_1 + 6x_2 & \le & 9 - 2x_1 + x_2, \\ -5 + x_1 + 6x_2 & \le & 1 - 5x_1 + 6x_2, \\ 8 + x_1 - 7x_2 & \le & 9 - 2x_1 + x_2, \\ 8 + x_1 - 7x_2 & \le & 1 - 5x_1 + 6x_2 \end{array}$$

$$Y = \{[x; w] : Px + Qw \leq r\},\ w = [w_1; ...; w_m] \qquad (*)$$

Elimination step: eliminating a *single* slack variable. Given set $(*)$, assume that $m > 0$, and let Y^+ be the projection of Y on the space of variables $x, w_1, ..., w_{m-1}$:

$$Y^+ = \{[x; w_1; ...; w_{m-1}] : \exists w_m : Px + Qw \leq r\} \qquad (!)$$

To prove that Y^+ is polyhedral, we use exactly the same approach as in Example:

- We split the inequalities $p_i^T x + q_i^T w \leq r_i,\ 1 \leq i \leq I$ defining Y into three groups:
  - black – the coefficient at w_m is 0
  - red – the coefficient at w_m is > 0
  - green – the coefficient at w_m is < 0

Then

$$\begin{aligned} Y = \{[x; w] \in \mathbb{R}^{n+m} :\ & \\ & a_i^T x + b_i^T [w_1; ...; w_{m-1}] \leq c_i,\ \ i \text{ is black} \\ & w_m \leq a_i^T x + b_i^T [w_1; ...; w_{m-1}] + c_i,\ \ i \text{ is red} \\ & w_m \geq a_i^T x + b_i^T [w_1; ...; w_{m-1}] + c_i,\ \ i \text{ is green}\} \end{aligned}$$

$\Rightarrow$

$$\begin{aligned} Y^+ = \{&[x; w_1; ...; w_{m-1}] : \\ & a_i^T x + b_i^T [w_1; ...; w_{m-1}] \leq c_i,\ \ i \text{ is black} \\ & a_\mu^T x + b_\mu^T [w_1; ...; w_{m-1}] + c_\mu \\ & \qquad \geq a_\nu^T x + b_\nu^T [w_1; ...; w_{m-1}] + c_\nu \\ & \qquad\qquad \text{whenever } \mu \text{ is red} \\ & \qquad\qquad \text{and } \nu \text{ is green}\} \end{aligned}$$

$\Rightarrow$ Y^+ is polyhedral.

• We have seen that the projection

$$Y^+ = \{[x; w_1; ...; w_{m-1}] : \exists w_m : [x; w_1; ...; w_m] \in Y\}$$

of the polyhedral set $Y = \{[x, w] : Px + Qw \leq r\}$ is polyhedral. Iterating the process, we conclude that the set $X = \{x : \exists w : [x, w] \in Y\}$ is polyhedral, Q.E.D.

♣ Given an LO program

$$\text{Opt} = \max_x \left\{c^T x : Ax \leq b\right\}, \qquad (!)$$

observe that the set of values of the objective at feasible solutions can be represented as

$$\begin{aligned} T &= \{t \in \mathbb{R} : \exists x : Ax \leq b, c^T x - t = 0\} \\ &= \{t \in \mathbb{R} : \exists x : Ax \leq b, c^T x \leq t, c^T x \geq t\} \end{aligned}$$

that is, T *is polyhedrally representable*. By Theorem, T is polyhedral, that is, T can be represented by a finite system of linear inequalities *in variable* t *only.* It immediately follows that *if* T *is nonempty and is bounded from above,* T *has the largest element*. Thus, we have proved

Corollary. *A feasible and bounded LO program admits an optimal solution and thus is solvable.*

$$\begin{aligned} T &= \{t \in \mathbb{R} : \exists x : Ax \leq b, c^T x - t = 0\} \\ &= \{t \in \mathbb{R} : \exists x : Ax \leq b, c^T x \leq t, c^T x \geq t\} \end{aligned}$$

♣ Fourier-Motzkin Elimination Scheme suggests a finite algorithm for solving an LO program, where we

• first, apply the scheme to get a representation of T by a finite system S of linear inequalities in variable t,

• second, analyze S to find out whether the solution set is nonempty and bounded from above, and when it is the case, to find out the optimal value Opt $\in T$ of the program,

• third, use the Fourier-Motzkin elimination scheme in the backward fashion to find x such that $Ax \leq b$ and $c^T x =$ Opt, thus recovering an optimal solution to the problem of interest.

Bad news: The resulting algorithm is completely impractical, since the number of inequalities we should handle at a step usually rapidly grows with the step number and can become astronomically large when eliminating just tens of variables.

Polyhedrally Representable Functions

♣ **Definition:** *The domain* $\mathrm{Dom} f$ *of a function* $f(x)$ *is the set of all points* x *where the value of* f *is well defined.*

• In general, description of $\mathrm{Dom} f$ is a part of the description of f.

• When $f(x)$ is given by analytical expression, $\mathrm{Dom} f$ *by default* is the set of all values of x where the expression makes sense.

For example, *by default*

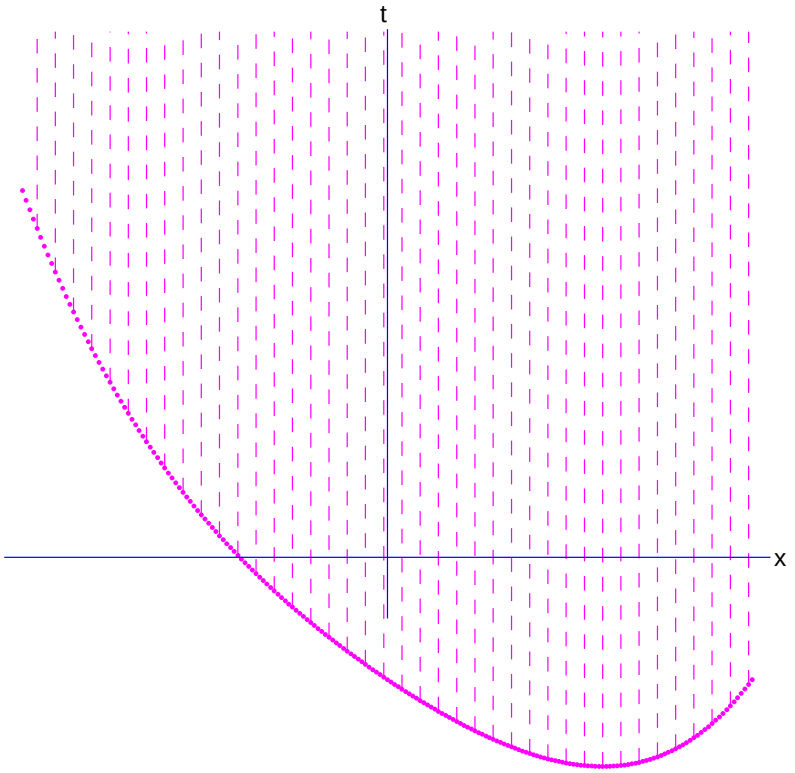
• $f(x) = \sqrt{x} \Rightarrow \mathrm{Dom} f = \{x \in \mathbb{R} : x \geq 0\}$ (nonnegative ray)

• $f(x) = \sin(x) \Rightarrow \mathrm{Dom} f = \mathbb{R}$ (real axis)

• $f(x_1, ..., x_n) = \sqrt{x_1^2 + ... + x_n^2} \Rightarrow \mathrm{Dom} f = \mathbb{R}^n$ (n-dimensional space)

♣ **Definition:** *Let f be a real-valued function with* $\mathrm{Dom} f \subset \mathbb{R}^n$*. The epigraph of f is the set*

$$\mathrm{Epi}\{f\} = \{[x; t] \in \mathbb{R}^n \times \mathbb{R} : x \in \mathrm{Dom} f, t \geq f(x)\}.$$



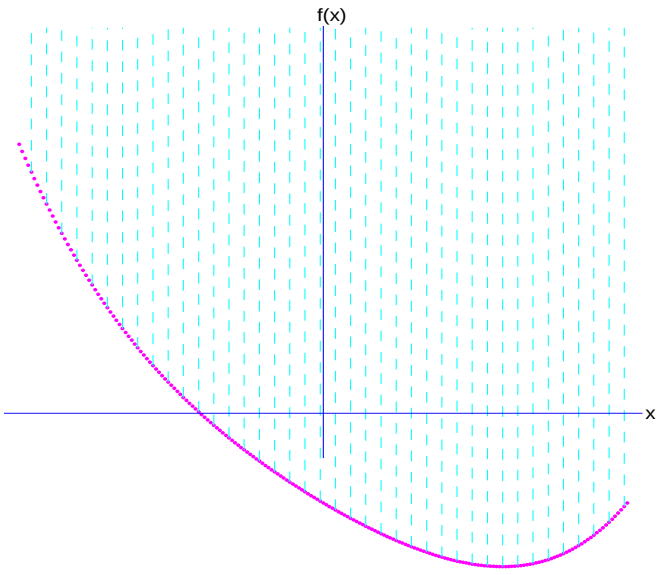
- bold magenta curve: *graph* of f – set of pairs $[x; t{=}f(x)]$
- magenta domain: *epigraph* of f – set of pairs $[x; t{\geq}f(x)]$

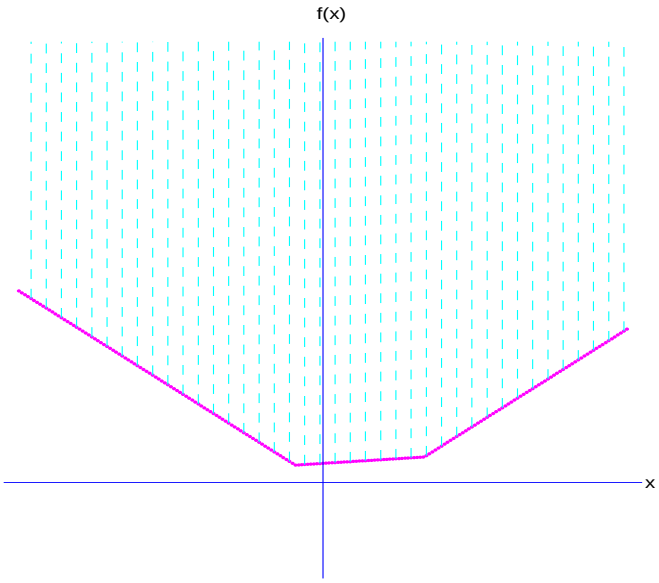
Epigraph of real-valued function $f(\cdot)$ with domain $\mathrm{Dom} f \subset \mathbb{R}^n$ is the set

$$\mathrm{Epi}\{f\} = \{[x;t] \in \mathbb{R}^n \times \mathbb{R} : x \in \mathrm{Dom} f, t \geq f(x)\}.$$

Definition. A polyhedral representation of $\mathrm{Epi}\{f\}$ is called a *polyhedral representation of* f. Function f is called *polyhedrally representable*, if it admits a polyhedral representation.

Quiz: Among the functions below, which are polyhedrally representable?





♠ **Observation:** *A Lebesque set*

$$\{x \in \mathrm{Dom} f : f(x) \leq a\}$$

of a polyhedrally representable function is polyhedral, with a p.r. readily given by a p.r. of $\mathrm{Epi}\{f\}$*:*

$$\mathrm{Epi}\{f\} = \{[x;t] : \exists w : Px + tp + Qw \leq r\} \Rightarrow$$
$$\left\{x : \begin{array}{c} x \in \mathrm{Dom} f \\ f(x) \leq a \end{array}\right\} = \{x : \exists w : Px + ap + Qw \leq r\}.$$

Examples:

• The function $f(x) = \max\limits_{1\le i\le I}[\alpha_i^T x + \beta_i]$ is polyhedrally representable:

$$\mathrm{Epi}\{f\} = \{[x;t] : \alpha_i^T x + \beta_i - t \le 0,\ 1 \le i \le I\}.$$

• **Extension:** Let $D = \{x : Ax \le b\}$ be a polyhedral set in $\mathbb{R}^n$. A function f with the domain D given in D as $f(x) = \max\limits_{1\le i\le I}[\alpha_i^T x + \beta_i]$ is polyhedrally representable:

$$\mathrm{Epi}\{f\} = \{[x;t] : x \in D, t \ge \max_{1\le i\le I} \alpha_i^T x + \beta_i\} =$$
$$\{[x;t] : Ax \le b,\ \alpha_i^T x - t + \beta_i \le 0,\ 1 \le i \le I\}.$$

In fact, *every polyhedrally representable function f is of the form stated in Extension.*

Calculus of Polyhedral Representations

♣ In principle, speaking about polyhedral representations of sets and functions, we could restrict ourselves with representations which do not exploit slack variables, specifically,

- *for sets* — with representations of the form

$$X = \{x \in \mathbb{R}^n : Ax \leq b\};$$

- *for functions* — with representations of the form

$$\mathrm{Epi}\{f\} = \{[x;t] : Ax \leq b, t \geq \max_{1 \leq i \leq I} \alpha_i^T x + \beta_i\}$$

♠ However, "general" – involving slack variables – polyhedral representations of sets and functions are much more flexible and can be much more "compact" that the straightforward – without slack variables – representations.

Examples:

• The function $f(x) = \|x\|_1 := \sum_{i=1}^n |x_i| : \mathbb{R}^n \to \mathbb{R}$ admits the p.r.

$$\mathrm{Epi}\{f\} = \left\{ [x;t] : \exists w \in \mathbb{R}^n : \begin{array}{r} -w_i \le x_i \le w_i, \\ 1 \le i \le n \\ \sum_i w_i \le t \end{array} \right\}$$

which requires n slack variables and $2n+1$ linear inequality constraints. In contrast to this, the straightforward — without slack variables — representation of f

$$\mathrm{Epi}\{f\} = \left\{ [x;t] : \begin{array}{l} \sum_{i=1}^n \epsilon_i x_i \le t \\ \forall (\epsilon_1 = \pm 1, ..., \epsilon_n = \pm 1) \end{array} \right\}$$

requires 2^n inequality constraints.

• The set $X_n = \{x \in \mathbb{R}^n : \sum_{i=1}^n \max[x_i, 0] \le 1\}$ admits the p.r.

$$X_n = \{x \in \mathbb{R}^n : \exists w : 0 \le w,\, x_i \le w_i \,\forall i, \sum_i w_i \le 1\}$$

which requires n slack variables and $2n+1$ inequality constraints.

Quiz: How to represent X_3 by linear constraints *in variables* x_1, x_2, x_3 *only*?

$$X_n = \{x \in \mathbb{R}^n : \sum_{i=1}^n \max[x_i, 0] \leq 1\}$$

Fact: Every straightforward — without slack variables — p.r. of X_n requires at least $2^n - 1$ linear constraints. A representation of X_n by $2^n - 1$ constraints in x-variables is

$$X_n = \{x \in \mathbb{R}^n : \sum_{i \in I} x_i \leq 1, \, \emptyset \neq I \subset \{1, ..., n\}\}$$

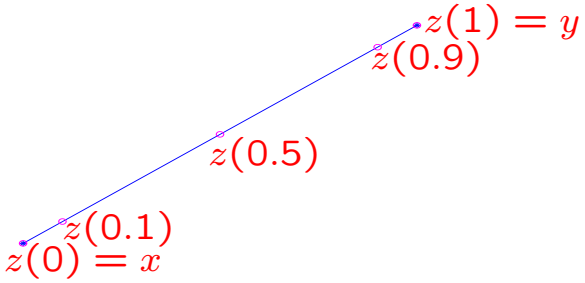
♣ Polyhedral representations admit a kind of simple and "fully algorithmic" calculus which, essentially, demonstrates that all *convexity-preserving* operations with polyhedral sets produce polyhedral results, and a p.r. of the result is readily given by p.r.'s of the operands.

♠ **Role of Convexity:** A set $X \subset \mathbb{R}^n$ is called *convex*, if whenever two points x, y belong to X, the entire segment $[x, y]$ linking these points belongs to X.

• Segment $[x, y]$ with endpoints x, y is built of the points

$$z(\lambda) = x + \lambda[y - x] = (1 - \lambda)x + \lambda y, \ 0 \leq \lambda \leq 1$$

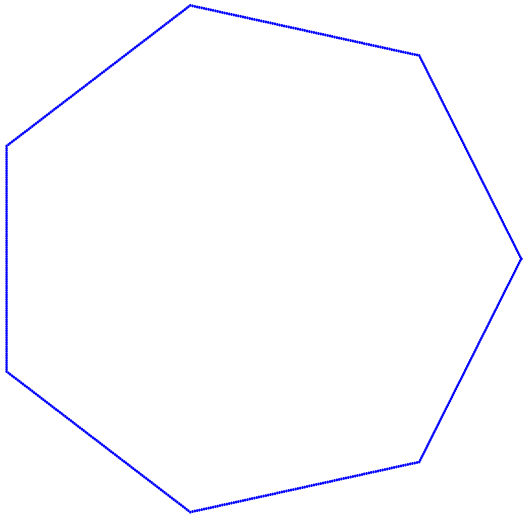
we can reach when shifting x by a fraction $\lambda \in [0, 1]$ of the vector $y - x$:



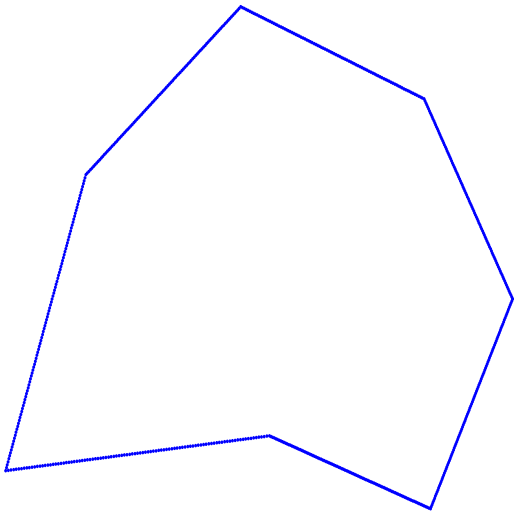
$\Rightarrow$ Analytically, convexity of a set $X \subset \mathbb{R}^n$ means that

$$\forall (x, y \in X, \lambda \in [0, 1]) : \\ x + \lambda(y - x) = (1 - \lambda)x + \lambda y \in X \ .$$

Quiz: Here are two closed contours in 2D plane:



contour A



contour B

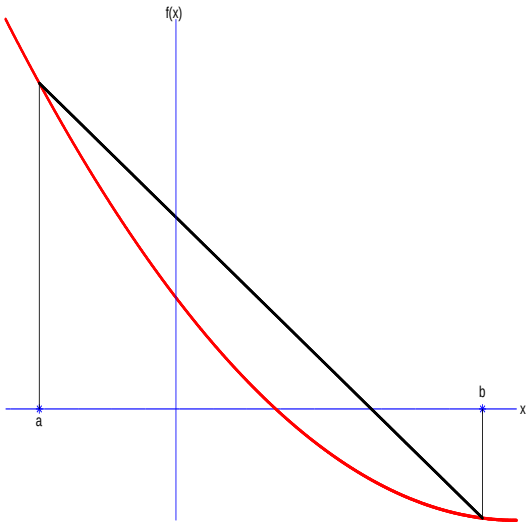
Fill the table

Set	convex [Y/N]
contour A	
domain inside A, A included	
domain inside A, A excluded	
what is outside of A, A included	
contour B	
domain inside B, B included	
domain inside B, B excluded	
what is outside of B, B included	

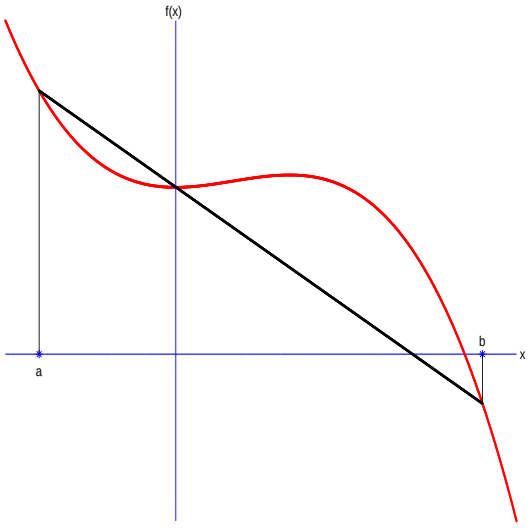
♠ A function $f : \mathrm{Dom} f \to \mathbb{R}$ is called *convex*, if its epigraph $\mathrm{Epi}\{f\}$ is a convex set, or, equivalently, if the domain $\mathrm{Dom} f$ of f is a convex set, and

$$x, y \in \mathrm{Dom} f, \lambda \in [0,1]$$
$$\Rightarrow f((1-\lambda)x + \lambda y) \leq (1-\lambda) f(x) + \lambda f(y).$$

• **Geometrically:** Convexity of f means that $\mathrm{Dom} f$ is convex, and for every pair $a \in \mathrm{Dom} f$, $b \in \mathrm{Dom} f$, *the restriction of f on the segment $[a, b]$ is dominated by the secant – the linear function on $[a, b]$ with the same values at the endpoints as those of f*



convex function



nonconvex function

♠ *Function f is called concave, if $-f$ is convex.*

Fact: *A polyhedral set $X = \{x : Ax \leq b\}$ is convex. In particular, a polyhedrally representable function is convex.*

Indeed,

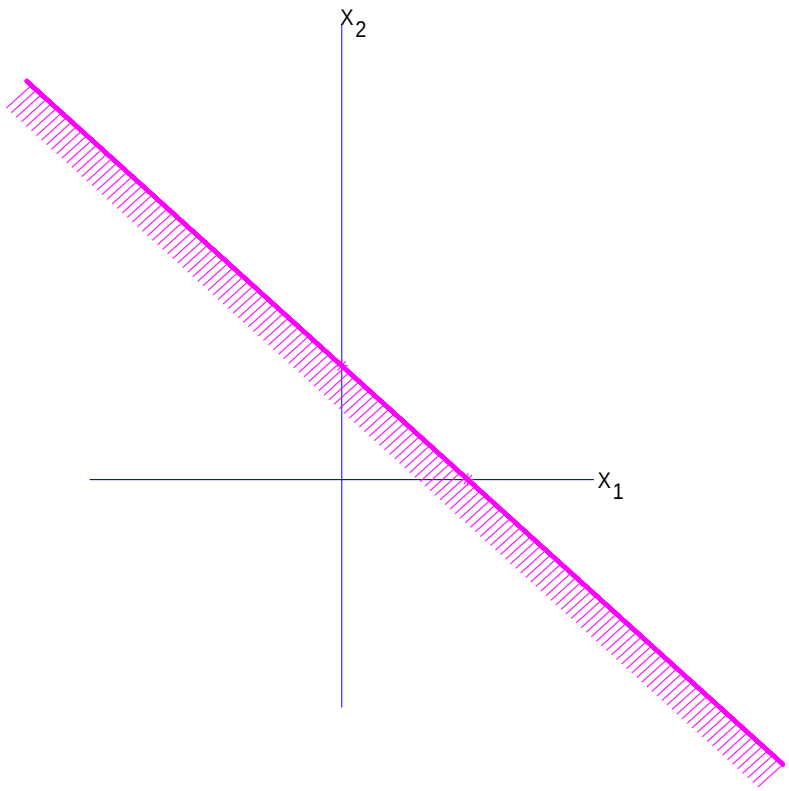
$$\begin{array}{rl} & Ax \leq b, Ay \leq b, \lambda \geq 0, 1-\lambda \geq 0 \\ \Rightarrow & \begin{array}{rcl} A(1-\lambda)x & \leq & (1-\lambda)b \\ A\lambda y & \leq & \lambda b \end{array} \\ \Rightarrow & A[(1-\lambda)x + \lambda y] \leq b \end{array}$$

Consequences:

- lack of convexity makes impossible polyhedral representation of a set/function,
- consequently, operations with functions/sets allowed by "calculus of polyhedral representability" we intend to develop should be convexity-preserving operations.

Calculus of Polyhedral Sets

♠ **Raw materials:** $X = \{x \in \mathbb{R}^n : a^T x \leq b\}$ (when $a \neq 0$, or, which is the same, when the set is nonempty and differs from the entire space, such a set is called *half-space*)



Half-plane $a^T x := [2; 4]^T [x_1; x_2] \leq 1$

• boundary line is given by equality $[2; 4]^T [x_1; x_2] \leq 1$

• vector $a = [2; 4]$ is the outward normal to the boundary line of the half-plane

Quiz: Where the boundary of the half-plane intersects the coordinate axes?

♠ Calculus rules:

S.1. Taking finite intersections: *If the sets $X_i \subset \mathbb{R}^n$, $1 \leq i \leq k$, are polyhedral, so is their intersection, and a p.r. of the intersection is readily given by p.r.'s of the operands.*

Indeed, if

$$X_i = \{x \in \mathbb{R}^n : \exists w^i : P_i x + Q_i w^i \leq r_i\},\ i = 1, ..., k,$$

then

$$\bigcap_{i=1}^{k} X_i = \left\{ x : \exists w = [w^1; ...; w^k] : \begin{array}{r} P_i x + Q_i w^i \leq r_i, \\ 1 \leq i \leq k \end{array} \right\},$$

which is a polyhedral representation of $\bigcap\limits_i X_i$.

Warning: *Taking the union of sets does not preserve convexity*

$\Rightarrow$ *The union of several polyhedral sets is, in general, non-polyhedral* (and even non-convex)

S.2. Taking direct products. Given k sets $X_i \subset \mathbb{R}^{n_i}$, their *direct product* $X_1 \times ... \times X_k$ is the set in $\mathbb{R}^{n_1+...+n_k}$ comprised of all block-vectors $x = [x^1; ...; x^k]$ with blocks x^i belonging to X_i, $i = 1, ..., k$.

Example: The direct product of k segments $[-1, 1]$ on the axis is the unit k-dimensional box $\{x \in \mathbb{R}^k : -1 \leq x_i \leq 1, i = 1, ..., k\}$.

If the sets $X_i \subset \mathbb{R}^{n_i}$, $1 \leq i \leq k$, are polyhedral, so is their direct product, and a p.r. of the product is readily given by p.r.'s of the operands.

Indeed, if

$$X_i = \{x^i \in \mathbb{R}^{n_i} : \exists w^i : P_i x^i + Q_i w^i \leq r_i\}, \; i = 1, ..., k,$$

then

$$\begin{array}{l} X_1 \times ... \times X_k \\ = \Big\{x = [x^1; ...; x^k] : \exists w = [w^1; ...; w^k] : \\ \qquad\qquad\qquad\qquad P_i x^i + Q_i w^i \leq r_i, \\ \qquad\qquad\qquad\qquad 1 \leq i \leq k \Big\}. \end{array}$$

S.3. Taking affine image. *If $X \subset \mathbb{R}^n$ is a polyhedral set and $y = Ax + b : \mathbb{R}^n \to \mathbb{R}^m$ is an affine mapping, then the set $Y = AX + b := \{y = Ax + b : x \in X\} \subset \mathbb{R}^m$ is polyhedral, with p.r. readily given by the mapping and a p.r. of X.*

Indeed, if $X = \{x : \exists w : Px + Qw \leq r\}$, then

$$\begin{aligned} Y &= \{y : \exists [x; w] : Px + Qw \leq r, y = Ax + b\} \\ &= \left\{ y : \exists [x; w] : \begin{array}{l} Px + Qw \leq r, \\ y - Ax \leq b, Ax - y \leq -b \end{array} \right\} \end{aligned}$$

Since Y admits a p.r., Y is polyhedral.

S.4. Taking inverse affine image. *If $X \subset \mathbb{R}^n$ is polyhedral, and $x = Ay + b : \mathbb{R}^m \to \mathbb{R}^n$ is an affine mapping, then the set $Y = \{y \in \mathbb{R}^m : Ay + b \in X\} \subset \mathbb{R}^m$ is polyhedral, with p.r. readily given by the mapping and a p.r. of X.*

Indeed, if $X = \{x : \exists w : Px + Qw \leq r\}$, then

$$\begin{aligned} Y = & \{y : \exists w : P[Ay + b] + Qw \leq r\} \\ & = \{y : \exists w : [PA]y + Qw \leq r - Pb\}. \end{aligned}$$

S.5. Taking arithmetic sum: *If the sets $X_i \subset \mathbb{R}^n$, $1 \le i \le k$, are polyhedral, so is their arithmetic sum $X_1 + ... + X_k := \{x = x_1 + ... + x_k : x_i \in X_i, 1 \le i \le k\}$, and a p.r. of the sum is readily given by p.r.'s of the operands.*

Indeed, the arithmetic sum of $X_1, ..., X_k$ is the image of $X_1 \times ... \times X_k$ under the linear mapping $[x^1; ...; x^k] \mapsto x^1 + ... + x^k$, and both operations preserve polyhedrality. Here is an explicit p.r. for the sum: if $X_i = \{x : \exists w^i : P_i x + Q_i w^i \le r_i\}$, $1 \le i \le k$, then

$$\begin{aligned} & X_1 + ... + X_k \\ &= \left\{ x : \exists x^1, ..., x^k, w^1, ..., w^k : \begin{array}{r} P_i x^i + Q_i w^i \le r_i, \\ 1 \le i \le k, \\ x = \sum_{i=1}^k x^i \end{array} \right\}, \end{aligned}$$

and it remains to replace the vector equality in the right hand side by a system of two opposite vector inequalities.

Calculus of Polyhedrally Representable Functions

♣ **Preliminaries:** Arithmetics of partially defined functions.

• a scalar function f of n variables is specified by indicating its *domain* $\mathrm{Dom} f$– the set where the function is well defined, and by the description of f as a real-valued function in the domain.

When speaking about convex functions f, *it is very convenient to think of* f *as of a function defined everywhere on* $\mathbb{R}^n$ *and taking real values in* $\mathrm{Dom} f$ *and the value* $+\infty$ *outside of* $\mathrm{Dom} f$.

With this convention, f *becomes an everywhere defined function on* $\mathbb{R}^n$ *taking values in* $\mathbb{R} \cup \{+\infty\}$, *and* $\mathrm{Dom} f$ *becomes the set where* f *takes real values.*

♠ In order to allow for basic operations with partially defined functions, like their addition or comparison, we augment our convention with the following agreements on the arithmetics of the "extended real axis" $\mathbb{R} \cup \{+\infty\}$:

• *Addition*: for a real a, $a + (+\infty) = (+\infty) + (+\infty) = +\infty$.

• *Multiplication by a nonnegative real λ:* $\lambda \cdot (+\infty) = +\infty$ when $\lambda > 0$, and $0 \cdot (+\infty) = 0$.

• *Comparison:* for a real a, $a < +\infty$ (and thus $a \leq +\infty$ as well), and of course $+\infty \leq +\infty$.

Note: Our arithmetic is incomplete — operations like $(+\infty) - (+\infty)$ and $(-1) \cdot (+\infty)$ remain undefined.

♠ **Raw materials:** $f(x) = a^T x + b$ (*affine functions*)

$$\mathrm{Epi}\{a^T x + b\} = \{[x; t] : a^T x + b - t \leq 0\}$$

♠ **Calculus rules:**

F.1. Taking linear combinations with positive coefficients. *If $f_i : \mathbb{R}^n \to \mathbb{R} \cup \{+\infty\}$ are p.r.f.'s and $\lambda_i > 0$, $1 \leq i \leq k$, then $f(x) = \sum_{i=1}^k \lambda_i f_i(x)$ is a p.r.f., with a p.r. readily given by those of the operands.*

Indeed, if

$$\begin{aligned}&\{[x; t] : t \geq f_i(x)\}\\ &= \{[x; t] : \exists w^i : P_i x + t p_i + Q_i w^i \leq r_i\}, 1 \leq i \leq k,\end{aligned}$$

then

$$\begin{aligned}&\{[x; t] : t \geq \textstyle\sum_{i=1}^k \lambda_i f_i(x)\}\\ &= \left\{[x; t] : \exists t_1, ..., t_k : \begin{array}{l} t_i \geq f_i(x), 1 \leq i \leq k, \\ \sum_i \lambda_i t_i \leq t \end{array}\right\}\\ &= \left\{[x; t] : \exists t_1, ..., t_k, w^1, ..., w^k : \right.\\ &\qquad\qquad \left.\begin{array}{rl} P_i x + t_i p_i + Q_i w^i \leq r_i, & \\ & 1 \leq i \leq k, \\ \sum_i \lambda_i t_i \leq t & \end{array}\right\}.\end{aligned}$$

F.2. Direct summation. *If $f_i : \mathbb{R}^{n_i} \to \mathbb{R} \cup \{+\infty\}$, $1 \le i \le k$, are p.r.f.'s, then so is their direct sum*

$$f([x^1; ...; x^k]) = \sum_{i=1}^{k} f_i(x^i) : \mathbb{R}^{n_1+...+n_k} \to \mathbb{R} \cup \{+\infty\}$$

and a p.r. for this function is readily given by p.r.'s of the operands.

Indeed, if

$$\begin{aligned}&\{[x^i; t] : t \ge f_i(x^i)\}\\ &= \{[x^i; t] : \exists w^i : P_i x^i + t p_i + Q_i w^i \le r_i\},\ 1 \le i \le k,\end{aligned}$$

then

$$\begin{aligned}&\{[x^1; ...; x^k; t] : t \ge \textstyle\sum_{i=1}^{k} f_i(x^i)\}\\ &= \left\{[x^1; ...; x^k; t] : \exists t_1, ..., t_k : \begin{array}{r} t_i \ge f_i(x^k), \\ 1 \le i \le k, \\ \sum_i t_i \le t \end{array}\right\}\\ &= \left\{[x^1; ...; x^k; t] : \exists t_1, ..., t_k, w^1, ..., w^k : \right.\\ &\qquad\qquad \left.\begin{array}{r} P_i x^i + t_i p_i + Q_i w^i \le r_i, \\ 1 \le i \le k, \\ \sum_i \lambda_i t_i \le t \end{array}\right\}.\end{aligned}$$

F.3. Taking maximum. *If $f_i : \mathbb{R}^n \to \mathbb{R} \cup \{+\infty\}$ are p.r.f.'s, so is their maximum $f(x) = \max[f_1(x), ..., f_k(x)]$, with a p.r. readily given by those of the operands.*

Indeed, if

$$\begin{aligned}&\{[x;t] : t \geq f_i(x)\}\\ &= \{[x;t] : \exists w^i : P_i x + t p_i + Q_i w^i \leq r_i\},\ 1 \leq i \leq k,\end{aligned}$$

then

$$\begin{aligned}&\{[x;t] : t \geq \max_i f_i(x)\}\\ &= \left\{[x;t] : \exists w^1, ..., w^k : \begin{array}{r} P_i x + t p_i + Q_i w^i \leq r_i, \\ 1 \leq i \leq k \end{array}\right\}.\end{aligned}$$

F.4. Affine substitution of argument. *If a function $f(x) : \mathbb{R}^n \to \mathbb{R} \cup \{+\infty\}$ is a p.r.f. and $x = Ay + b : \mathbb{R}^m \to \mathbb{R}^n$ is an affine mapping, then the function $g(y) = f(Ay + b) : \mathbb{R}^m \to \mathbb{R} \cup \{+\infty\}$ is a p.r.f., with a p.r. readily given by the mapping and a p.r. of f.*

Indeed, if

$$\begin{array}{l}\{[x;t] : t \geq f(x)\}\\ = \{[x;t] : \exists w : Px + tp + Qw \leq r\},\end{array}$$

then

$$\begin{array}{l}\{[y;t] : t \geq f(Ay + b)\}\\ = \{[y;t] : \exists w : P[Ay + b] + tp + Qw \leq r\}\\ = \{[y;t] : \exists w : [PA]y + tp + Qw \leq r - Pb\}.\end{array}$$

F.5. Theorem on superposition. *Let*

- $f_i(x) : \mathbb{R}^n \to \mathbb{R} \cup \{+\infty\}$ *be p.r.f.'s, and let*
- $F(y) : \mathbb{R}^m \to \mathbb{R} \cup \{+\infty\}$ *be a p.r.f. which is nondecreasing w.r.t. every one of the variables* $y_1, ..., y_m$. *Then the superposition*

$$g(x) = \begin{cases} F(f_1(x), ..., f_m(x)), & f_i(x) < +\infty \, \forall i \\ +\infty, & \text{otherwise} \end{cases}$$

of F *and* $f_1, ..., f_m$ *is a p.r.f., with a p.r. readily given by those of* f_i *and* F.

Indeed, let

$$\begin{array}{l} \{[x;t] : t \geq f_i(x)\} \\ = \{[x;t] : \exists w^i : P_i x + tp + Q_i w^i \leq r_i\}, \\ \{[y;t] : t \geq F(y)\} \\ = \{[y;t] : \exists w : Py + tp + Qw \leq r\}. \end{array}$$

Then

$$\begin{array}{l} \{[x;t] : t \geq g(x)\} \\ \underbrace{=}_{(*)} \left\{ [x;t] : \exists y_1, ..., y_m : \begin{array}{r} y_i \geq f_i(x), \\ 1 \leq i \leq m, \\ F(y_1, ..., y_m) \leq t \end{array} \right\} \\ = \left\{ [x;t] : \exists y, w^1, ..., w^m, w : \right. \\ \qquad \left. \begin{array}{r} P_i x + y_i p_i + Q_i w^i \leq r_i, \\ 1 \leq i \leq m, \\ Py + tp + Qw \leq r \end{array} \right\}, \end{array}$$

where $(*)$ is due to the monotonicity of F.

Note: if some of f_i, say, $f_1, ..., f_k$, are affine, then the Superposition Theorem remains valid when we require the monotonicity of F w.r.t. the variables $y_{k+1}, ..., y_m$ only; a p.r. of the superposition in this case reads

$$
\begin{aligned}
&\{[x;t] : t \geq g(x)\} \\
&= \left\{ [x;t] : \exists y_{k+1} ..., y_m : \right. \\
&\quad \left. \begin{array}{l} y_i \geq f_i(x),\ k+1 \leq i \leq m, \\ F(f_1(x), ..., f_k(x), y_{k+1}, ..., y_m) \leq t \end{array} \right\} \\
&= \left\{ [x;t] : \exists y_1, ..., y_m, w^{k+1}, ..., w^m, w : \right. \\
&\qquad \left. \begin{array}{r} y_i = f_i(x),\ 1 \leq i \leq k, \\ P_i x + y_i p_i + Q_i w^i \leq r_i, \\ k+1 \leq i \leq m, \\ Py + tp + Qw \leq r \end{array} \right\},
\end{aligned}
$$

and the linear equalities $y_i = f_i(x)$, $1 \leq i \leq k$, can be replaced by pairs of opposite linear inequalities.

Fast Polyhedral Approximation of the Second Order Cone

♠ **Fact:** The canonical polyhedral representation $X = \{x \in \mathbb{R}^n : Ax \leq b\}$ of the projection

$$X = \{x : \exists w : Px + Qw \leq r\}$$

of a polyhedral set $X^+ = \{[x; w] : Px + Qw \leq r\}$ given by a moderate number of linear inequalities in variables x, w can require a huge number of linear inequalities in variables x.

Question: Can we use this phenomenon in order to *approximate* to high accuracy a non-polyhedral set $X \subset \mathbb{R}^n$ by projecting onto $\mathbb{R}^n$ a higher-dimensional *polyhedral and simple* (given by a moderate number of linear inequalities) set X^+ ?

Theorem: *For every n and every ϵ, $0 < \epsilon < 1/2$, one can point out a polyhedral set $\mathbf{L}^+$ given by an explicit system of homogeneous linear inequalities in variables $x \in \mathbb{R}^n$, $t \in \mathbb{R}$, $w \in \mathbb{R}^k$:*

$$\mathbf{L}^+ = \{[x; t; w] : Px + tp + Qw \leq 0\} \qquad (!)$$

such that

- *the number of inequalities in the system ($\approx 0.7n\ln(1/\epsilon)$) and the dimension of the slack vector w ($\approx 2n\ln(1/\epsilon)$) do not exceed $O(1)n\ln(1/\epsilon)$*
- *the projection*

$$\mathbf{L} = \{[x; t] : \exists w : Px + tp + Qw \leq 0\}$$

of $\mathbf{L}^+$ on the space of x, t-variables is in-between the Second Order Cone and $(1+\epsilon)$-extension of this cone:

$$\begin{array}{l}\mathbf{L}^{n+1} := \{[x; t] \in \mathbb{R}^{n+1} : \|x\|_2 \leq t\} \subset \mathbf{L} \\ \subset \mathbf{L}_\epsilon^{n+1} := \{[x; t] \in \mathbb{R}^{n+1} : \|x\|_2 \leq (1+\epsilon)t\}.\end{array}$$

In particular, we have

$$\begin{array}{c}B_n^1 \subset \{x : \exists w : Px + p + Qw \leq 0\} \subset B_n^{1+\epsilon} \\ B_n^r = \{x \in \mathbb{R}^n : \|x\|_2 \leq r\}\end{array}$$

Note: When $\epsilon = 1.e\text{-}17$, a usual computer does not distinguish between $r = 1$ and $r = 1 + \epsilon$. Thus, *for all practical purposes*, the n-dimensional Euclidean ball admits polyhedral representation with $\approx 79n$ slack variables and $\approx 28n$ linear inequality constraints.

Note: A straightforward representation $X = \{x : Ax \leq b\}$ of a polyhedral set X satisfying

$$B_n^1 \subset X \subset B_n^{1+\epsilon}$$

requires at least $N = O(1)\epsilon^{-\frac{n-1}{2}}$ linear inequalities. With $n = 100$, $\epsilon = 0.01$, we get

$$N \geq 3.0\text{e}85 \approx 300,000 \times [\#\text{ of atoms in universe}]$$

With "fast polyhedral approximation" of B_n^1, a 0.01-approximation of B_{100} requires just 325 linear inequalities on 100 original and 922 slack variables.

♣ With fast polyhedral approximation of the cone $\mathbf{L}^{n+1} = \{[x;t] \in \mathbb{R}^{n+1} : \|x\|_2 \le t\}$, *Conic Quadratic* Optimization programs

$$\max_x \left\{ c^T x : \|A_i x - b_i\|_2 \le c_i^T x + d_i,\ 1 \le i \le m \right\} \quad \text{(CQI)}$$

"for all practical purposes" become LO programs. Note that numerous highly nonlinear optimization problems, like

$$\text{minimize } c^T x \text{ subject to}$$

$$Ax = b$$

$$x \ge 0$$

$$\left(\sum_{i=1}^{8} |x_i|^3 \right)^{1/3} \le x_2^{1/7} x_3^{2/7} x_4^{3/7} + 2 x_1^{1/5} x_5^{2/5} x_6^{1/5}$$

$$5x_2 \ge \frac{1}{x_1^{1/2} x_2^2} + \frac{2}{x_2^{1/3} x_3^3 x_4^{5/8}}$$

$$\begin{bmatrix} x_2 & x_1 & & \\ x_1 & x_4 & x_3 & \\ & x_3 & x_6 & x_3 \\ & & x_3 & x_8 \end{bmatrix} \succeq 5I$$

$$\exp\{x_1\} + 2\exp\{2x_2 - x_3 + 4x_4\} + 3\exp\{x_5 + x_6 + x_7 + x_8\} \le 12$$

can be *in a systematic fashion* converted to/rapidly approximated by problems of the form (CQI) and thus "for all practical purposes" are just LO programs.

Building Fast Polyhedral Approximation

♣ **Goal:** To *nearly* represent by linear inequalities the set

$$\mathbf{L}^{n+1} = \{[x_1; ...; x_n; t] : \sqrt{x_1^2 + ... + x_n^2} \leq t$$

that is, to find a polyhedrally represented set

$$\widehat{\mathbf{L}} = \{[x = [x_1; ...; x_n; t] : \exists w : Px + tp + Qw \leq 0\}$$

such that

$$\mathbf{L}^{n+1} \subset \widehat{\mathbf{L}} \subset \mathbf{L}_\epsilon^{n+1},$$

$$\mathbf{L}_\epsilon^{n+1} = \{[x_1; ...; x_n; t] : \sqrt{x_1^2 + ... + x_n^2} \leq (1+\epsilon)t\}$$

- $\epsilon > 0$: given tolerance.

♠ **Observation:** *It suffices to solve our problem when* $n = 2$.

Reason: Inequality $\sqrt{x_1^2 + ... + x_n^2} \leq t$ can be represented by a system of similar inequalities with 3 variables in each.

Example: To represent the set

$$\mathbf{L}^6 = \{[x;t] \in \mathbb{R}^6 : \sqrt{x_1^2 + x_2^2 + ... + x_5^2} \leq t\},$$

by a system of constraints of the form $\sqrt{p^2 + q^2} \leq r$, we

♠ add to x, t variable w_1 and write down the system

$$\sqrt{x_4^2 + x_5^2} \leq w_1,\ \sqrt{x_1^2 + x_2^2 + x_3^2 + w_1^2} \leq t$$

• *the system does represent* $\mathbf{L}^6$ – the projection of its solution set on the space of x, t-variables is *exactly* $\mathbf{L}^6$

• the "sizes" (# of variables involved) of the constraints in the system are ≤ 5, while the size of the constraint in the original description of $\mathbf{L}^6$ was 6.

♠ add to x, t, w_1 variable w_2 and write down the system

$$\sqrt{x_4^2 + x_5^2} \leq w_1,\ \sqrt{x_3^2 + w_1^2} \leq w_2\ \sqrt{x_1^2 + x_2^2 + w_2^2} \leq t$$

This system still represents $\mathbf{L}^6$, and the maximal size of its constraints is 4.

♠ add to x, t, w_1, w_2 variable w_3 and write down the system

$$\sqrt{x_4^2 + x_5^2} \leq w_1,\ \sqrt{x_3^2 + w_1^2} \leq w_2,\ \sqrt{x_2^2 + w_2^2} \leq w_3,\ \sqrt{x_1^2 + w_3^2} \leq t$$

This system represents $\mathbf{L}^6$, and all its constraints are of the form $\sqrt{p^2 + q^2} \leq r$. We are done.

Note: The above recipe clearly extends from the 6-dimensional case to the general one. Representing $\mathbf{L}^{n+1}$ via constraints of the form $\sqrt{p^2 + q^2} \leq r$ requires $n - 2$ slack variables and $n - 1$ constraints.

Quiz: The number of steps in the above construction is $n-2$. Find an alternative which represents $\mathbf{L}^n$ by $n-1$ constraints of the form $\sqrt{p^2+q^2} \leq t$ and requires $n-2$ slack variables, but takes at most $\mathrm{Ceil}(\log_2(n))-1$ steps.

♠ **Conclusion:** In order to find a tight polyhedral approximation of

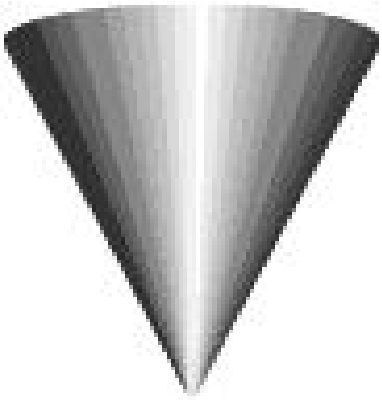
$$\mathbf{L}^{n+1} = \{[x_1; ...; x_n; t] : \sqrt{x_1^2 + ... + x_n^2} \leq t\} \quad ,$$

we can

- represent the constraint $\sqrt{x_1^2 + ... + x_n^2} \leq t$ by a system of inequalities of the form $\sqrt{p^2+q^2} \leq r$
- to replace every one of the resulting constraints by its tight polyhedral approximation.

Note: We should account for "accumulation of errors." This is an easy task...

Fast polyhedral approximation of $\mathbf{L}^3 = \{[p;q;r] : \sqrt{p^2+q^2} \leq r\}$



"Ice-cream" cone $\mathbf{L}^3$

♠ Given variables p, q, r, we choose a positive integer K, and consider $K+1$ points $P_1, ..., P_{K+1}$ on the 2D plane as follows.

• The first points $P_1 = [u_1; v_1]$ satisfies

$$u_1 \geq |p|,\ v_1 \geq |q|$$

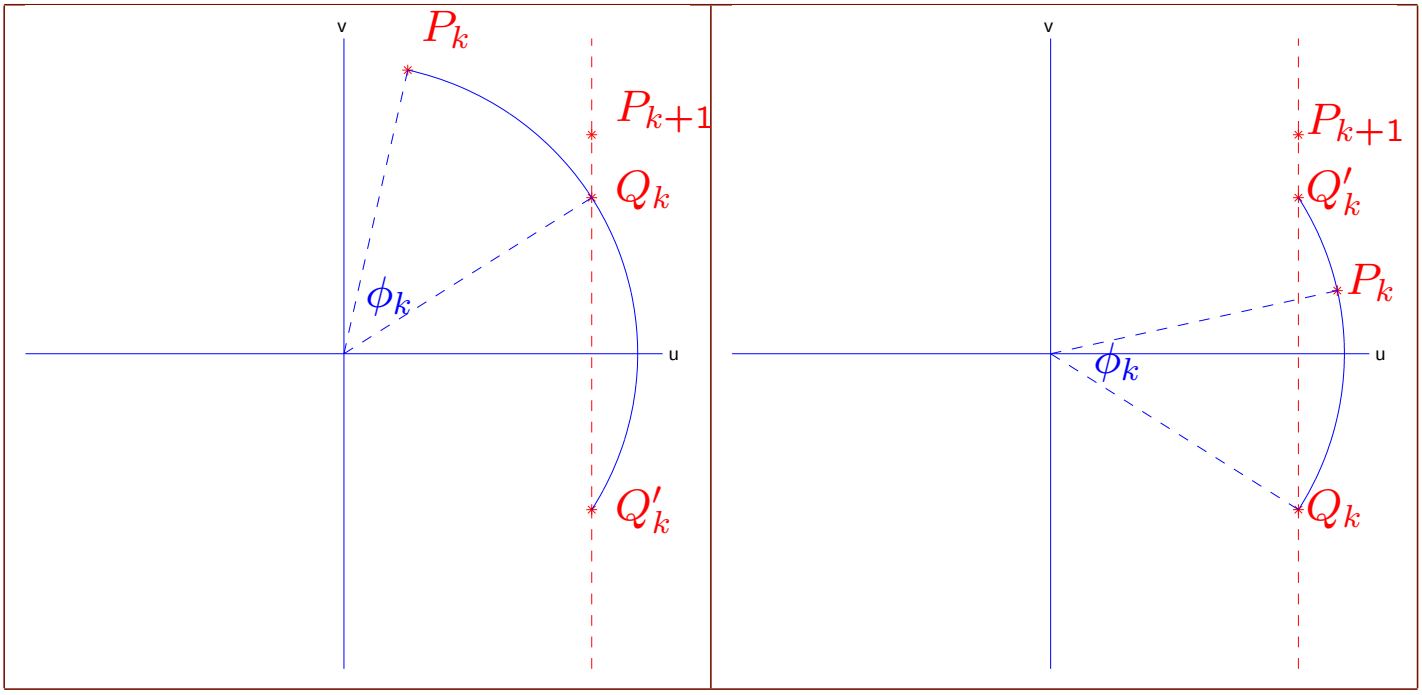
which can be represented by a system of 4 linear constraints in variables p, q, u_1, v_1.

• The relation between $P_k = [u_k; v_k]$ and $P_{k+1} = [u_{k+1}; v_{k+1}]$ is as follows.

  • we rotate P_k clockwise by the angle $\phi_k = \frac{\pi}{2^{k+1}}$, thus getting a point Q_k.

  • we reflect Q_k w.r.t. the u-axis, thus getting point Q'_k.

  • we impose on $P_{k+1} = [u_{k+1}; v_{k+1}]$ the restriction to belong to the vertical line passing through Q_k and Q'_k and to be not lower than Q_k and Q'_k.



♠ **Note:** Relations between $P_k = [u_k; v_k]$ and $P_{k+1} = [u_{k+1}; v_{k+1}]$ amount to a system of linear constraints

$$u_{k+1} = \cos(\phi_k)u_k + \sin(\phi_k)v_k$$

right hand side: u-coordinate of Q_k and Q'_k

$$v_{k+1} \geq -\sin(\phi_k)u_k + \cos(\phi_k)v_k$$

right hand side: v-coordinate of Q_k

$$v_{k+1} \geq \sin(\phi_k)u_k - \cos(\phi_k)v_k$$

right hand side: v-coordinate of Q'_k

in variables $u_k, v_k, u_{k+1}, v_{k+1}$.

♠ Let us write down all built so far constraints on original and slack variables

$$\begin{array}{|rcl|}\hline u_1 & \geq & p\\ u_1 & \geq & -p\\ v_1 & \geq & q\\ v_2 & \geq & -q\\ \hline u_{k+1} & = & \cos(\phi_k)u_k+\sin(\phi_k)v_k\\ v_{k+1} & \geq & -\sin(\phi_k)u_k+\cos(\phi_k)v_k\\ v_{k+1} & \geq & \sin(\phi_k)u_k-\cos(\phi_k)v_k\\ & & k=1,...,K\\ \hline\end{array}$$

and augment this system by the requirement for P_{K+1} to be close to the segment $[0,r]$ of the u-axis:

$$0\leq u_{K+1}\leq r,\ 0\leq v_{K+1}\leq \tan(\phi_K)\cdot r$$

Observation 1: *When p,q,r can be augmented by properly selected u's and v's to satisfy the above constraints, we have*

$$\sqrt{p^2+q^2}\leq r\sqrt{1+\tan^2(\phi_K)}$$

Indeed, by the above constraints on p,q,r and the slack variables, the points $P_k=[u_k;v_k]$ satisfy

$$\|[p;q]\|_2\leq\|P_1\|_2\leq...\leq\|P_{K+1}\|_2=\sqrt{u_{K+1}^2+v_{K+1}^2}\leq r\sqrt{1+\tan^2(\phi_K)}.$$

$$
\begin{array}{|rcl|}
\hline
u_1 & \geq & p \\
u_1 & \geq & -p \\
v_1 & \geq & q \\
v_2 & \geq & -q \\
\hline
u_{k+1} & = & \cos(\phi_k)u_k + \sin(\phi_k)v_k \\
v_{k+1} & \geq & -\sin(\phi_k)u_k + \cos(\phi_k)v_k \\
v_{k+1} & \geq & \sin(\phi_k)u_k - \cos(\phi_k)v_k \\
 & | & k = 1, ..., K \\
\hline
\multicolumn{3}{|c|}{0 \leq u_{K+1} \leq r,\ 0 \leq v_{K+1} \leq \tan(\phi_K) \cdot r} \\
\hline
\end{array}
$$

Observation 2: *When $\sqrt{p^2+q^2} \leq r$, p, q, r indeed can be augmented by u's and v's to satisfy our constraints.*

This combines with Observation 1 to imply that *the projection of the polyhedral set given by our constraints onto the space of p, q, r variables is in-between the $\mathbf{L}^3$ and $\mathbf{L}^3_{\delta_K}$, with*

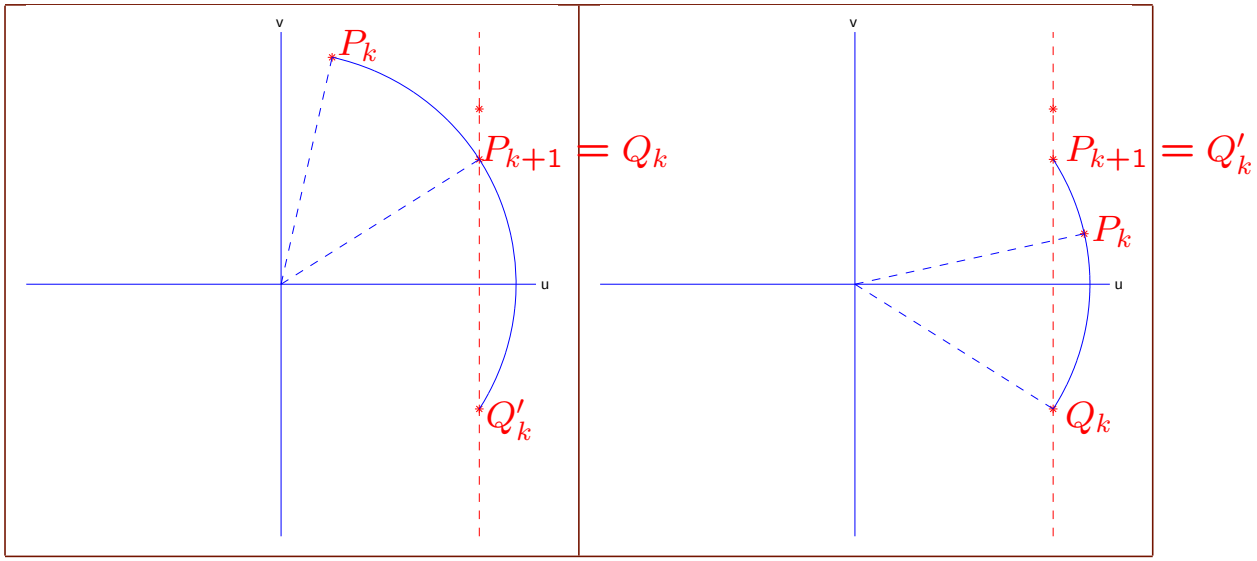
$$
\begin{aligned}
\delta_K &= \sqrt{1 + \tan^2(\phi_K)} - 1 \\
&= \sqrt{1 + \tan^2\left(\frac{\pi}{2^{K+1}}\right)} - 1 \leq \frac{\pi^2}{2^{2K+2}}.
\end{aligned}
$$

$\Rightarrow$ *To make $\delta_K \leq \epsilon$, we need just $O(1)\ln(1/\epsilon)$ slack variables and linear constraints!*

$$
\begin{array}{|l|}
\hline
u_1 \geq p, u_1 \geq -p \\
v_1 \geq q, v_2 \geq -q \\
\hline
\left.\begin{array}{l} u_{k+1} = \cos(\phi_k)u_k + \sin(\phi_k)v_k \\ v_{k+1} \geq -\sin(\phi_k)u_k + \cos(\phi_k)v_k \\ v_{k+1} \geq \sin(\phi_k)u_k - \cos(\phi_k)v_k \end{array}\right\} k = 1, ..., K \\
\hline
0 \leq u_{K+1} \leq r,\ 0 \leq v_{K+1} \leq \tan(\phi_K) \cdot r \\
\hline
\end{array} \qquad (*)
$$

Observation 2: *When $\sqrt{p^2+q^2} \leq r$, p, q, r indeed can be augmented by u's and v's to satisfy $(*)$.*

♠ To justify Observation 2, let us augment p, q with u's and v's which "rigidly" satisfy the magenta constraints, specifically, let us set $u_1 = |p|$, $v_1 = |q|$, and let P_{k+1} be the "highest" of the points Q_k, Q'_k:



Then

$$r \geq \sqrt{p^2+q^2} = \|[p;q]\|_2 = \|P_1\|_2 = ... = \|P_{K+1}\|_2$$

and the angle between P_{k+1} and the nonnegative ray of the u-axis does not exceed $\phi_k = \frac{\pi}{2^{k+1}}$.

$\Rightarrow$ $P_{K+1} = [u_{K+1}, v_{K+1}]$ indeed satisfies

$$0 \leq u_{K+1} \leq r \text{ and } 0 \leq v_{K+1} \leq \tan(\phi_K) \cdot r.$$

♡ To justify the claim on the angles, observe that with our "rigid" construction of $P_1, ..., P_{K+1}$,

• P_1 lives in the first quadrant, and P_2 is obtained from P_1 by rotating clockwise by the angle $\phi_1 = \pi/4$ (and, perhaps, reflecting the result w.r.t. the u-axis to bring it to the first quadrant).

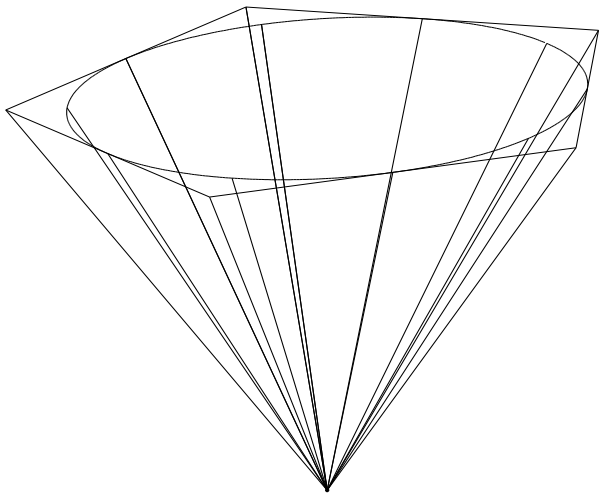
After rotation, the angle between the point and the u-axis does not exceed $\pi/4$, and reflection, if any, keeps this angle intact

$\Rightarrow$ *P_2 lives in the first quadrant and makes angle at most $\phi_1 = \pi/4$ with the u-axis*

$\Rightarrow$ P_3, which is obtained from P_2 by rotating clockwise by the angle $\phi_2 = \pi/8$ (and, perhaps, reflecting the result w.r.t. u-axis to bring it to the first quadrant), *lives in the first quadrant and makes the angle at most $\phi_2 = \pi/8$ with the u-axis*

$\Rightarrow$ .....$\Rightarrow$ *P_{K+1} lives in the first quadrant and makes angle at most $\phi_K = \frac{\pi}{2^{K+1}}$ with the u-axis.*

♣ The simplest way to build a polyhedral approximation of the Lorentz cone is to take the tangent planes along a "fine" finite grid of generators and to use, as the approximation, the resulting polyhedral cone:



This approach is a complete failure: the number of tangent planes required to get an 0.5-approximation of $\mathbf{L}^m$ is at least

$$N = \sqrt{2\pi(m-2)}\exp\{m/6\},$$

which is $> 429,481,377$ for $m = 100$.

♣ With our approach, we approximate $\mathbf{L}^m$ by a *projection of a higher-dimensional polyhedron*. When projecting an N-dimensional polyhedron onto a plane of dimension $<< N$, the number of facets may grow up exponentially, so that a low-dimensional projection of a "simple" high-dimensional polyhedron may have astronomically many facets. With our approach, we build a family of polyhedral cones $P^{m,k} \subset \mathbb{R}^{O(mk)}$ given by just $O(mk)$ linear inequalities, while their projections $\widehat{P}^{m,k}$ on $\mathbb{R}^m$ have enough facets to approximate $\mathbf{L}^m$ within accuracy $\exp\{-O(k)\}$:

- $P^{3,3} \subset \mathbb{R}^{10}$ is given by 12 inequalities.

$\widehat{P}^{3,3}$ approximates $\mathbf{L}^3$ within accuracy 5.e-3
(as good as the 16-facet circumscribed cone)

- $P^{3,6} \subset \mathbb{R}^{13}$ is given by 18 linear inequalities.

$\widehat{P}^{3,6}$ approximates $\mathbf{L}^3$ within accuracy 3.e-4
(as good as the 127-facet circumscribed cone)

- $P^{3,12} \subset \mathbb{R}^{19}$ is given by 30 linear inequalities.

$\widehat{P}^{3,12}$ approximates $\mathbf{L}^3$ within accuracy 7.e-8
(as good as the 8,192-facet circumscribed cone)

- $P^{3,24} \subset \mathbb{R}^{31}$ is given by 54 linear inequalities.

$\widehat{P}^{3,24}$ approximates $\mathbf{L}^3$ within accuracy 4.e-15
(as good as the 34,200,933-facet circumscribed cone)

♠ Polyhedral approximation of $\mathbf{L}^m$ is basically the same as polyhedral approximation of m-dimensional Euclidean ball

$$\mathbf{B}_m = \{x \in \mathbb{R}^m : \|x\|_2 \leq 1\}.$$

There is a less sophisticated way to approximate Euclidean balls by projections of polyhedral sets:

Theorem [Lindenstrauss-Johnson]: *For two positive integers N, n with $N \geq 10n$, random n-dimensional projection of N-dimensional unit box – the set*

$$B = \{x \in \mathbb{R}^n : \exists y \in \mathbb{R}^N : x = Ay,\ -1 \leq y_1, ..., y_N \leq 1\}$$

[A: drawn at random from Gaussian distribution]

with probability approaching one as N, n grow, is in-between two n-dimensional Euclidean balls with the ratio of radii $(1 + O(\sqrt{n/N}))$.

This result has tremendous theoretical implications. However,

— no *individual* matrices A yielding "nearly round" B are known (pity! these matrices would be ideally suited for Compressed Sensing)

Note: *Our fast polyhedral approximation is explicit!*

— to make B an ϵ-approximation of B_n, you need $N = O(1/\epsilon^2)n$

Note: With fast polyhedral approximation, you need much smaller N: $N = O(\ln(1/\epsilon))n$

♠ **Open question:** With fast polyhedral approximation, *centrally symmetric* ball $\mathbf{B}_n$ is ϵ-approximated by the projection of a *highly asymmetric* polyhedron of dimension $N = O(\ln(1/\epsilon))n$ given by $M = O(N)$ linear inequalities. *Is it possible to make this higher-dimensional polyhedron centrally symmetric*, preserving the type of dependence of N, M on n and ϵ?

Geometry of a Polyhedral Set

♣ An LO program $\max\limits_{x\in\mathbb{R}^n}\left\{c^Tx : Ax\leq b\right\}$ is the problem of maximizing a linear objective over a *polyhedral set* $X=\{x\in\mathbb{R}^n : Ax\leq b\}$ – the solution set of a *finite* system of *nonstrict linear* inequalities

$\Rightarrow$ *Understanding geometry of polyhedral sets is the key to LO theory and algorithms.*

♣ Our ultimate goal is to understand the following fundamental

Theorem. *A nonempty polyhedral set*

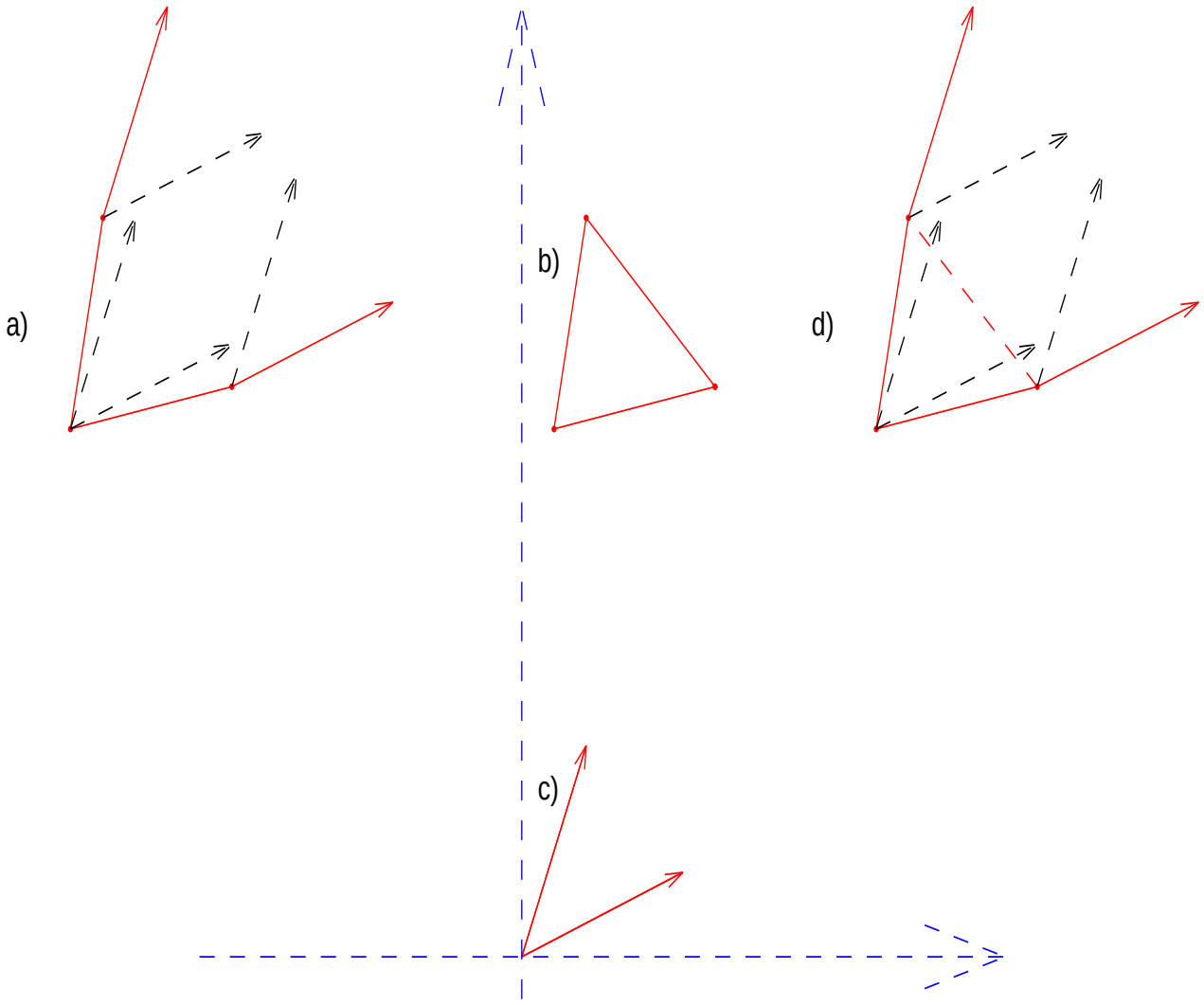
$$X=\{x\in\mathbb{R}^n : Ax\leq b\}$$

admits a representation of the form

$$X=\left\{x=\sum_{i=1}^{M}\lambda_i v_i+\sum_{j=1}^{N}\mu_j r_j : \begin{array}{l}\lambda_i\geq 0\,\forall i\\ \sum\limits_{i=1}^{M}\lambda_i=1\\ \mu_j\geq 0\,\forall j\end{array}\right\} \qquad (!)$$

where $v_i\in\mathbb{R}^n$, $1\leq i\leq M$ *and* $r_j\in\mathbb{R}^n$, $1\leq j\leq N$ *are properly chosen "generators."*

Vice versa, every set X *representable in the form of* (!) *is polyhedral.*



a): a polyhedral set

b): $\{\sum_{i=1}^{3} \lambda_i v_i : \lambda_i \geq 0, \sum_{i=1}^{3} \lambda_i = 1\}$

c): $\{\sum_{j=1}^{2} \mu_j r_j : \mu_j \geq 0\}$

d): The set a) is the sum of sets b) and c)

Note: shown are the boundaries of the sets.

$$\emptyset \neq X = \{x \in \mathbb{R}^n : Ax \leq b\}$$
$$\Updownarrow$$
$$X = \left\{ x = \sum_{i=1}^{M} \lambda_i v_i + \sum_{j=1}^{N} \mu_j r_j : \begin{array}{l} \lambda_i \geq 0 \, \forall i \\ \sum_{i=1}^{M} \lambda_i = 1 \\ \mu_j \geq 0 \, \forall j \end{array} \right\} \quad (!)$$

♠ $X = \{x \in \mathbb{R}^n : Ax \leq b\}$ is an "outer" description of a polyhedral set X: it says what should be cut off $\mathbb{R}^n$ to get X.

♠ (!) is an "inner" description of a polyhedral set X: it explains how can we get all points of X, starting with two finite sets of vectors in $\mathbb{R}^n$.

♡ Taken together, these two descriptions offer a powerful "toolbox" for investigating polyhedral sets. For example,

• To see that the intersection of two polyhedral subsets X, Y in $\mathbb{R}^n$ is polyhedral, we can use their outer descriptions:

$$X = \{x : Ax \leq b\}, Y = \{x : Bx \leq c\}$$
$$\Rightarrow X \cap Y = \{x : Ax \leq b, Bx \leq c\} \quad .$$

$$\emptyset \neq X = \{x \in \mathbb{R}^n : Ax \leq b\}$$
$$\Updownarrow$$
$$X = \left\{ \sum_{i=1}^{M} \lambda_i v_i + \sum_{j=1}^{N} \mu_j r_j : \begin{array}{l} \lambda_i \geq 0 \, \forall i \\ \sum_{i=1}^{M} \lambda_i = 1 \\ \mu_j \geq 0 \, \forall j \end{array} \right\} \text{ (!)}$$

• To see that the image $Y = \{y = Px + p : x \in X\}$ of a polyhedral set $X \subset \mathbb{R}^n$ under an affine mapping $x \mapsto Px + p : \mathbb{R}^n \to \mathbb{R}^m$, we can use the inner descriptions:

$$X \text{ is given by (!)}$$
$$\Rightarrow Y = \left\{ \sum_{i=1}^{M} \lambda_i (Pv_i + p) + \sum_{j=1}^{N} \mu_j P r_j : \begin{array}{l} \lambda_i \geq 0 \, \forall i \\ \sum_{i=1}^{M} \lambda_i = 1 \\ \mu_j \geq 0 \, \forall j \end{array} \right\}$$

Preliminaries: Linear Subspaces

♣ **Definition:** *A linear subspace in $\mathbb{R}^n$ is a nonempty subset L of $\mathbb{R}^n$ which is closed w.r.t. taking linear combinations of its elements:*

$$x_i \in L, \lambda_i \in \mathbb{R}, 1 \le i \le I \Rightarrow \sum_{i=1}^{I} \lambda_i x_i \in L$$

♣ **Examples:**

- $L = \mathbb{R}^n$
- $L = \{0\}$
- $L = \{x \in \mathbb{R}^n : x_1 = 0\}$
- $L = \{x \in \mathbb{R}^n : Ax = 0\}$
- Given a set $X \subset \mathbb{R}^n$, let $\mathrm{Lin}(X)$ be set of all finite linear combinations of vectors from X. This set – *the linear span of X* – is a linear subspace which contains X, and this is the intersection of all linear subspaces containing X.

Convention: A sum of vectors from $\mathbb{R}^n$ with empty set of terms is well defined and is the zero vector. In particular, $\mathrm{Lin}(\emptyset) = \{0\}$.

♠ **Note:** The last two examples are "universal:" *Every linear subspace L in $\mathbb{R}^n$ can be represented as $L = \mathrm{Lin}(X)$ for a properly chosen finite set $X \subset \mathbb{R}^n$, same as can be represented as $L = \{x : Ax = 0\}$ for a properly chosen matrix A.*

Quiz: Consider the set

$$L = \{[x_1; x_2; x_3] \in \mathbb{R}^3 : x_1 + x_2 + x_3 = 0\}$$

- *Is L a linear subspace? If "yes," how to represent L as a linear span of finitely many vectors?*

Quiz: Consider the set

$$L = \{[x_1; x_2; x_3] \in \mathbb{R}^3 : x_1 + x_2 + x_3 = 0\}$$

• *Is L a linear subspace? If "yes," how to represent L as a linear span of finitely many vectors?*

• L is a linear subspace (as the solution set of a system of homogeneous linear equations).

• Vectors from L are exactly the vectors in $\mathbb{R}^3$ with $x_3 = -(x_1 + x_2)$ $\Rightarrow$ *Vectors from L are exactly vectors of the form*

$$[x_1; x_2; -(x_1 + x_2)] = x_1[1; 0; -1] + x_2[0; 1; -1]$$

$\Rightarrow L = \mathsf{Lin}\{[1; 0; -1], [0; 1; -1]\}$.

♣ **Bases and dimension of a linear subspace**

♣ Let L be a linear subspace in $\mathbb{R}^n$.

♠ For properly chosen $x_1, ..., x_m$, we have

$$L = \mathrm{Lin}(\{x_1, ..., x_m\}) = \left\{\sum_{i=1}^m \lambda_i x_i\right\};$$

whenever this is the case, we say that $x_1, ..., x_m$ *linearly span* L.

Quiz: Let $L = \{[x_1; x_2; x_3] : x_1 + x_2 + x_3 = 0\}$.

- *Is it true that L is spanned by the two vectors*
  $[1; 0; -1]$, $[0; 1; -1]$ *?*
- *Is it true that L is spanned by the three vectors*
  $[1; 0; -1]$, $[0; 1; -1]$; $[-1; 0; 1]$ *?*
- *Is it true that L is spanned by the 3 vectors*
  $[1; 0; 0]$, $[0; 1; 0]$, $[0; 0; 1]$ *?*

Quiz: Let $L = \{[x_1; x_2; x_3] : x_1 + x_2 + x_3 = 0\}$.

- *Is it true that L is spanned by the two vectors*
  $[1; 0; -1], [0; 1; -1]$ *?*

Yes; we have seen in the previous Quiz that $L = \text{Lin}\{[1; 0; -1], [0; 1; -1]\}$.

- *Is it true that L is spanned by the three vectors*
  $[1; 0; -1], [0; 1; -1]; [-1; 0; 1]$ *?*

Yes. All three vectors belong to L, so that linear span of the vectors cannot be larger than L. And since the linear span of already the first two vectors is the entire L, the span of all three vectors cannot be smaller than L.

- *Is it true that L is spanned by the 3 vectors*
  $[1; 0; 0], [0; 1; 0], [0; 0; 1]$ *?*

No. If linear span of vectors belongs to L, all these vectors should belong to L, which is not the case.
In fact, $\text{Lin}\{[1; 0; 0], [0; 1; 0], [0; 0; 1]\} = \mathbb{R}^3$:

$$[x_1; x_2; x_3] = x_1 \cdot [1; 0; 0] + x_2 \cdot [0; 1; 0] + x_3 \cdot [0; 0; 1].$$

♠ Vectors $x_1, ..., x_m \in \mathbb{R}^n$ are called *linearly independent*, if *every nontrivial* (not all coefficients are zeros) *linear combination of* $x_1, ..., x_m$ *is a nonzero vector*.

Equivalently: $x_1, ..., x_m$ *are linearly independent, if the coefficients in a linear combination*

$$x = \sum_{i=1}^{m} \lambda_i x_i$$

are uniquely defined by the value x *of this combination.*

Quiz:

- *Are the two vectors* $[1; 0; -1]$, $[0; 1; -1]$ *linearly independent?*
- *Are the three vectors* $[1; 0; -1]$, $[0; 1; -1]$*;* $[-1; 0; 1]$ *linearly independent?*
- *Are the 3 vectors* $[1; 0; 0]$, $[0; 1; 0]$, $[0; 0; 1]$ *linearly independent ?*

Quiz:

- *Are the two vectors* $[1;0;-1]$, $[0;1;-1]$ *linearly independent?*

Yes:

$$\lambda_1[1;0;-1]+\lambda_2[0;1;-1]=0 \Leftrightarrow [\lambda_1;\lambda_2;-\lambda_1-\lambda_2]=0$$
$$\Leftrightarrow \lambda_1=\lambda_2=0$$

- *Are the three vectors* $[1;0;-1]$, $[0;1;-1]$*;* $[-1;0;1]$ *linearly independent?*

No. As we know, the third of the vectors is linear combination of the first two:

$$[-1;0;1]=\underbrace{1}_{\lambda_1}\cdot[1;0;1]+\underbrace{1}_{\lambda_2}\cdot[0;1;-1]$$

whence

$$\underbrace{1}_{\lambda_1}\cdot[1;0;1]+\underbrace{1}_{\lambda_2}\cdot[0;1;-1]+\underbrace{(-1)}_{\lambda_3}[1;0;-1]=0,$$

while not all of λ_i's are zero.

- *Are the 3 vectors* $[1;0;0]$, $[0;1;0]$, $[0;0;1]$ *linearly independent ?*

Yes.:

$$\lambda_1[1;0;0]+\lambda_2[0;1;0]+\lambda_3[0;0;1]=0 \Leftrightarrow [\lambda_1;\lambda_2;\lambda_3]=0$$
$$\Leftrightarrow \lambda_1=\lambda_2=\lambda_3=0$$

♠ Vectors $x_1, ..., x_m$ from L are called a linear *basis* of L, if they are linearly independent and linearly span L.

Equivalently: *$x_1, ..., x_m$ from a linear subspace L form a linear basis of L, if every $x \in L$ is a linear combination of $x_1, ..., x_m$ and the coefficients of this linear combination are uniquely defined by x.*

Example: By the above Quizzes, the two vectors

$$[1; 0; -1], [0; 1; -1]$$

form a linear basis of the linear space

$$L = \{[x_1; x_2; x_3] : x_1 + x_2 + x_3 = 0\}.$$

♣ Let L be a linear subspace in $\mathbb{R}^n$.

Facts:

♡ *L admits bases, and all these bases have the same cardinality, called the dimension* $\dim L$ *of* L

♡ Let $x_1, ..., x_m$ be a collection of vectors from L. The following properties of $x_1, ..., x_m$ are equivalent to each other:

- *Vectors $x_1, ..., x_m$ form a maximal w.r.t. inclusion linearly independent set in L* (i.e., they are linearly independent, but extending the collection by a vector *from L* always yields a linearly dependent collection)
- *Vectors $x_1, ..., x_m$ are linearly independent and $m = \dim L$*
- *Vectors $x_1, ..., x_m$ form a minimal w.r.t. inclusion set which linearly spans L* (i.e., $x_1, ..., x_m$ linearly span L, and this property is lost when eliminating from the collection one of its members)
- *Vectors $x_1, ..., x_m$ linearly span L and $m = \dim L$*
- *$x_1, ..., x_m$ form a basis in L*

In addition,

- *Every collection of linearly independent vectors from L can be extended to a basis of L*
- *From every collection of vectors from L which linearly spans L one can extract a basis of L.*

♠ **Examples**:

• $\dim\{0\} = 0$, and the only basis of $\{0\}$ is the empty collection.

• $\dim \mathbb{R}^n = n$. When $n > 0$, there are infinitely many bases in $\mathbb{R}^n$, e.g., one comprised of *standard basic orths* $e_i = [0; ...; 0; 1; 0; ...; 0]$ ("1" in i-th position), $1 \leq i \leq n$.

• $L = \{x \in \mathbb{R}^n : x_1 = 0\} \Rightarrow \dim L = n-1\}$. An example of a basis in L is $e_2, e_3, ..., e_n$.

Facts:

♡ *The smaller is linear subspace, the less is dimension: if $L \subset L'$ are linear subspaces in $\mathbb{R}^n$, then $\dim L \leq \dim L'$, with equality taking place iff $L = L'$.*

⇒ *Whenever L is a linear subspace in $\mathbb{R}^n$, we have $\{0\} \subset L \subset \mathbb{R}^n$, whence $0 \leq \dim L \leq n$*

♡ *In every representation of a linear subspace as*

$$L = \{x \in \mathbb{R}^n : Ax = 0\},$$

the number of rows in A is at least $n - \dim L$.

This number is equal to $n - \dim L$ iff the rows of A are linearly independent.

Quiz: *What is the dimension of the linear subspace*

$$L = \{x \in \mathbb{R}^3 : x_1 + x_2 + x_3 = 0\} \text{ ?}$$

Quiz: *What is the dimension of the linear subspace*

$$L = \{x \in \mathbb{R}^3 : x_1 + x_2 + x_3 = 0\}$$

?

Answer: The dimension is 2.

- *First explanation:* We have seen that the two vectors $[1; 0; -1]$, $[0; 1; -1]$ form a basis in L
- *Second explanation:* L is cut off $\mathbb{R}^3$ by a system of homogeneous linear equations (single nontrivial – not all coefficients are zero – equation $x_1 + x_2 + x_3 = 0$). The number of linearly independent equations in the system is $1 \Rightarrow \dim L = 3 - 1 = 2$.

"Calculus" of linear subspaces

♡ [taking intersection] *When L_1, L_2 are linear subspaces in $\mathbb{R}^n$, so is the set $L_1 \cap L_2$.*

Extension: The intersection $\bigcap_{\alpha\in\mathcal{A}} L_\alpha$ of an arbitrary family $\{L_\alpha\}_{\alpha\in\mathcal{A}}$ of linear subspaces of $\mathbb{R}^n$ is a linear subspace.

♡ [summation] *When L_1, L_2 are linear subspaces in $\mathbb{R}^n$, so is their arithmetic sum*

$$L_1 + L_2 = \{x = u + v : u \in L_1, v \in L_2\}.$$

Note "dimension formula:"

$$\dim L_1 + \dim L_2 = \dim (L_1 + L_2) + \dim (L_1 \cap L_2)$$

♡ [taking orthogonal complement] When L is a linear subspace in $\mathbb{R}^n$, so is its *orthogonal complement*

$$L^\perp = \{y \in \mathbb{R}^n : y^T x = 0 \, \forall x \in L\}.$$

Note:

- $(L^\perp)^\perp = L$
- $L + L^\perp = \mathbb{R}^n$, $L \cap L^\perp = \{0\} \Rightarrow \dim L + \dim L^\perp = n$
- $L = \{x : Ax = 0\}$ if and only if the (transposes of) the rows in A linearly span $L^\perp$
- $x \in \mathbb{R}^n \Rightarrow \exists!(x_1 \in L, x_2 \in L^\perp) : x = x_1 + x_2$, and for these x_1, x_2 one has $x^T x = x_1^T x_1 + x_2^T x_2$.

♡ [taking direct product] *When $L_1 \subset \mathbb{R}^{n_1}$ and $L_2 \subset \mathbb{R}^{n_2}$ are linear subspaces, the direct product (or direct sum) of L_1 and L_2 – the set*

$$L_1 \times L_2 := \{[x_1; x_2] \in \mathbb{R}^{n_1+n_2} : x_1 \in L_1, x_2 \in L_2\}$$

is a linear subspace in $\mathbb{R}^{n_1+n_2}$, and

$$\dim(L_1 \times L_2) = \dim L_1 + \dim L_2.$$

♡ [taking image under linear mapping] *When L is a linear subspace in $\mathbb{R}^n$ and $x \mapsto Px : \mathbb{R}^n \to \mathbb{R}^m$ is a linear mapping, the image*

$$PL = \{y = Px : x \in L\}$$

of L under the mapping is a linear subspace in $\mathbb{R}^m$.

♡ [taking inverse image under linear mapping] *When L is a linear subspace in $\mathbb{R}^n$ and $x \mapsto Px : \mathbb{R}^m \to \mathbb{R}^n$ is a linear mapping, the inverse image*

$$P^{-1}(L) = \{y : Py \in L\}$$

of L under the mapping is a linear subspace in $\mathbb{R}^m$.

Quiz: Consider the set in $\mathbb{R}^4$

$$Y = \{y : \exists x : \begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 0 & 0 & 0 \\ -1 & 0 & 0 & 1 & 1 & 0 \\ 0 & -1 & 0 & -1 & 0 & 1 \\ 0 & 0 & -1 & 0 & -1 & -1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \end{bmatrix}\}$$

A. Is Y a polyhedral set?

B. Is Y a linear subspace?

C. If Y is a linear subspace, then point out

— a set of vectors spanning Y

— $\dim Y$

— a basis in Y

— a representation of Y as a solution set of homogeneous system of linear equations

— the orthogonal complement to Y

Quiz: Consider the set in $\mathbb{R}^4$

$$Y = \{y : \exists x : \begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \end{bmatrix} = \overbrace{\begin{bmatrix} 1 & 1 & 1 & 0 & 0 & 0 \\ -1 & 0 & 0 & 1 & 1 & 0 \\ 0 & -1 & 0 & -1 & 0 & 1 \\ 0 & 0 & -1 & 0 & -1 & -1 \end{bmatrix}}^{A} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \end{bmatrix}\}$$

A. Is Y a polyhedral set? *Yes, it is given by polyhedral representation*

B. Is Y a linear subspace? *Yes, as the image of $\mathbb{R}^6$ under linear mapping*

C. If Y is a linear subspace, then point out

— a set of vectors spanning Y *For example, all 6 columns of A*

— $\dim Y$ $\dim Y = 3$:

• The dimension could be at most 4. The sum of entries in every column of A is 0 $\Rightarrow$ the sum of entries in every $y \in Y$ is 0 $\Rightarrow$ $\dim Y \leq 3$.

But: Y contains 3 linearly independent vectors (e.g., the first 3 columns of A) $\Rightarrow$ $\dim Y \geq 3$

— a basis in Y *For example, the first three columns in A*

— a representation of Y as a solution set of homogeneous system of linear equations

$$Y = \{y \in \mathbb{R}^4 : y_1 + y_2 + y_3 + y_4 = 0\} \qquad (*)$$

— $Y^\perp$ is the line

$$\mathbb{R} \cdot [1; 1; 1; 1] = \{[y_1; y_2; y_3; y_4] : y_1 = y_2 = y_3 = y_4\}$$

spanned by the vector of coefficients in $(*)$

Preliminaries: Affine Subspaces

♣ **Definition:** *An affine subspace* (or *affine plane*, or simply *plane*) *in* $\mathbb{R}^n$ *is a nonempty subset* M *of* $\mathbb{R}^n$ *which can be obtained from a linear subspace* $L \subset \mathbb{R}^n$ *by a shift:*

$$M = a + L = \{x = a + y : y \in L\} \qquad (*)$$

Note: In a representation $(*)$,

- L us uniquely defined by M:

$$L = M - M = \{x = u - v : u, v \in M\}.$$

L is called the linear subspace which is *parallel* to M;

- a can be chosen as an arbitrary element of M, and only as an element from M.

♠ **Equivalently:** *An affine subspace in* $\mathbb{R}^n$ *is a nonempty subset* M *of* $\mathbb{R}^n$ *which is closed with respect to taking affine combinations* (linear combinations with coefficients summing up to 1) *of its elements:*

$$\left\{x_i \in M, \lambda_i \in \mathbb{R}, \sum_{i=1}^{I} \lambda_i = 1\right\} \Rightarrow \sum_{i=1}^{I} \lambda_i x_i \in M$$

♣ **Examples:**

• $M = \mathbb{R}^n$. The parallel linear subspace is $\mathbb{R}^n$

• $M = \{a\}$ (singleton). The parallel linear subspace is $\{0\}$

• $M = \{a + \lambda \underbrace{[b-a]}_{\neq 0} : \lambda \in \mathbb{R}\} = \{(1-\lambda)a + \lambda b : \lambda \in \mathbb{R}\}$

– (straight) *line* passing through two distinct points $a, b \in \mathbb{R}^n$.

The parallel linear subspace is the linear span $\mathbb{R}[b-a]$ of $b-a$.

Fact: *A nonempty subset $M \subset \mathbb{R}^n$ is an affine subspace if and only if with any pair of distinct points a, b from M, M contains the entire line*

$$\ell = \{(1-\lambda)a + \lambda b : \lambda \in \mathbb{R}\}$$

spanned by a, b.

Examples of affine subspaces (continued):

- $\emptyset \neq M = \{x \in \mathbb{R}^n : Ax = b\}$.

  The parallel linear subspace is $\{x : Ax = 0\}$.

- Given a *nonempty* set $X \subset \mathbb{R}^n$, let $\mathrm{Aff}(X)$ be the set of all finite *affine* combinations of vectors from X. This set – *the affine span* (or *affine hull*) *of* X – is an affine subspace, contains X, and is the intersection of all affine subspaces containing X.

  The parallel linear subspace is $\mathrm{Lin}(X - a)$, where a is an arbitrary point from X.

♠ **Note:** The last two examples are "universal:" *Every affine subspace M in $\mathbb{R}^n$ can be represented as $M = \mathrm{Aff}(X)$ for a properly chosen finite and nonempty set $X \subset \mathbb{R}^n$, same as can be represented as $M = \{x : Ax = b\}$ for a properly chosen matrix A and vector B such that the system $Ax = b$ is solvable.*

♣ **Affine bases and dimension.** Let M be an affine subspace in $\mathbb{R}^n$, and L be the parallel linear subspace.

♠ *By definition*, the *affine dimension* (or simply *dimension*) $\dim M$ of M is the (linear) dimension $\dim L$ of the linear subspace L to which M is parallel.

♠ We say that vectors $x_0, x_1..., x_m$, $m \geq 0$, from M

• *are affinely independent*, if no nontrivial (not all coefficients are zeros) linear combination of these vectors *with zero sum of coefficients* is the zero vector

Equivalently: $x_0, ..., x_m$ are affinely independent if and only if the coefficients in an *affine* combination $x = \sum_{i=0}^{m} \lambda_i x_i$ are uniquely defined by the value x of this combination.

• *affinely span* M, if

$$M = \mathrm{Aff}(\{x_0, ..., x_m\}) = \left\{\sum_{i=0}^m \lambda_i x_i : \sum_{i=0}^m \lambda_i = 1\right\}$$

• *form an affine basis in* M, if $x_0, ..., x_m$ are affinely independent and affinely span M.

Equivalently: *$x_0, x_1, ..., x_m$ from an affine subspace M form an affine basis in M, if every $x \in M$ is an affine combination of $x_0, x_1, ..., x_m$ and the coefficients of this affine combination are uniquely defined by x.*

♠ **Facts:** Let M be an affine subspace in $\mathbb{R}^n$, L be the parallel linear subspace, and let $x_0, x_1, ..., x_m$ be a collection of vectors from M. Then

♡ *The collection $x_0, x_1, ..., x_m$ is an affine basis in M if and only if $x_0 \in M$ and the vectors $x_1 - x_0, x_2 - x_0, ..., x_m - x_0$ form a (linear) basis in L*

♡ The following properties of the collection $x_0, ..., x_m$ are equivalent to each other:

- *Vectors $x_0, x_1, ..., x_m$ form a maximal w.r.t. inclusion set affinely independent set in M* (i.e., they are affinely independent, but extending the collection by a vector *from* M always yields an affinely dependent collection)
- *Vectors $x_0, x_1, ..., x_m$ are affinely independent and* $m = \dim M$
- *Vectors $x_0, x_1, ..., x_m$ form a minimal w.r.t. inclusion collection which affinely spans M* (i.e., $x_0, x_1, ..., x_m$ affinely span M, and this property is lost when eliminating from the collection one of its members)
- *Vectors $x_0, x_1, ..., x_m$ affinely span M and* $m = \dim L$
- *$x_0, x_1, ..., x_m$ form an affine basis in M*

In addition,

- *Every collection of affinely independent vectors from M can be extended to an affine basis of M*
- *From every collection of vectors from M which affinely spans M one can extract an affine basis of M.*

Examples:

• $\dim\{a\} = 0$, and the only affine basis of $\{a\}$ is $x_0 = a$.

• $\dim \mathbb{R}^n = n$. When $n > 0$, there are infinitely many affine bases in $\mathbb{R}^n$, e.g., one comprised of the zero vector and the n standard basic orths.

• $M = \{x \in \mathbb{R}^n : x_1 = 1\} \Rightarrow \dim M = n - 1$. An example of an affine basis in M is

$$e_1, e_1 + e_2, e_1 + e_3, ..., e_1 + e_n.$$

Extension: M is an affine subspace in $\mathbb{R}^n$ of dimension $n - 1$ iff M can be represented as

$$M = \{x \in \mathbb{R}^n : a^T x = b\}$$

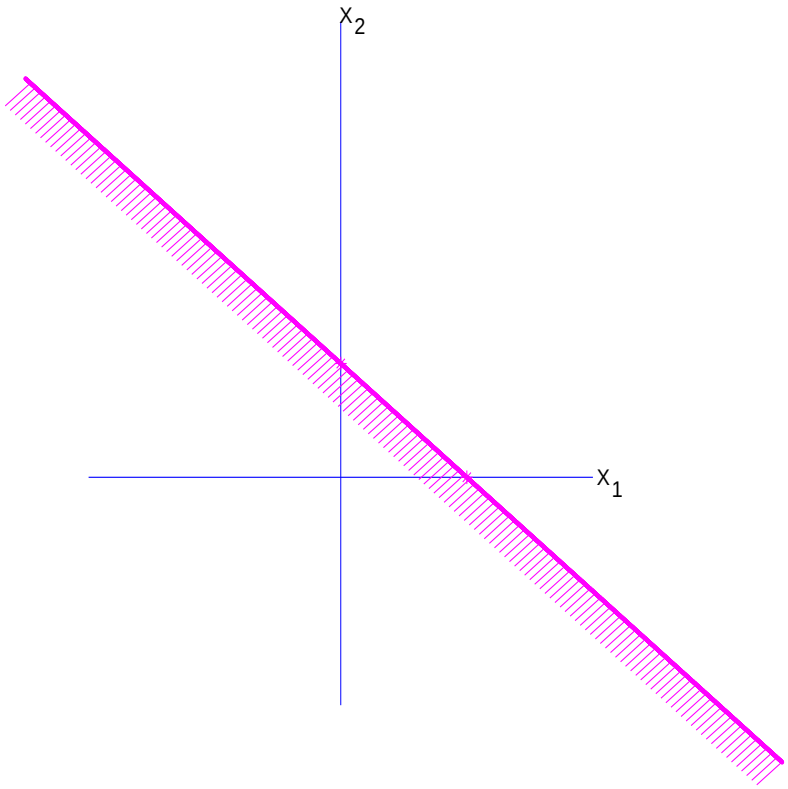
with $a \neq 0$. Such a set is called *hyperplane*.

♠ **Note:** A hyperplane $M = \{x : a^T x = b\}$ $(a \neq 0)$ splits $\mathbb{R}^n$ into two *half-spaces*

$$\Pi_+ = \{x : a^T x \geq b\}, \Pi_- = \{x : a^T x \leq b\}$$

and is the common boundary of these half-spaces.

• A polyhedral set is the intersection of a finite (perhaps empty) family of half-spaces.



Hyperplane in 2D (just line!) $x_1 + 2x_2 = 1$ (bold line)

- dashed half-space (half-plane):
$$\Pi_- = \{[x_1; x_2] : x_1 + 2x_2 \leq 1\}$$
- Complement to dashed half-space:
$$\Pi_+ = \{[x_1; x_2] : x_1 + 2x_2 \geq 1\}$$
- $a = [1; 2]$ is the outward normal to the boundary of Π_- and the inward normal to the boundary of Π_+

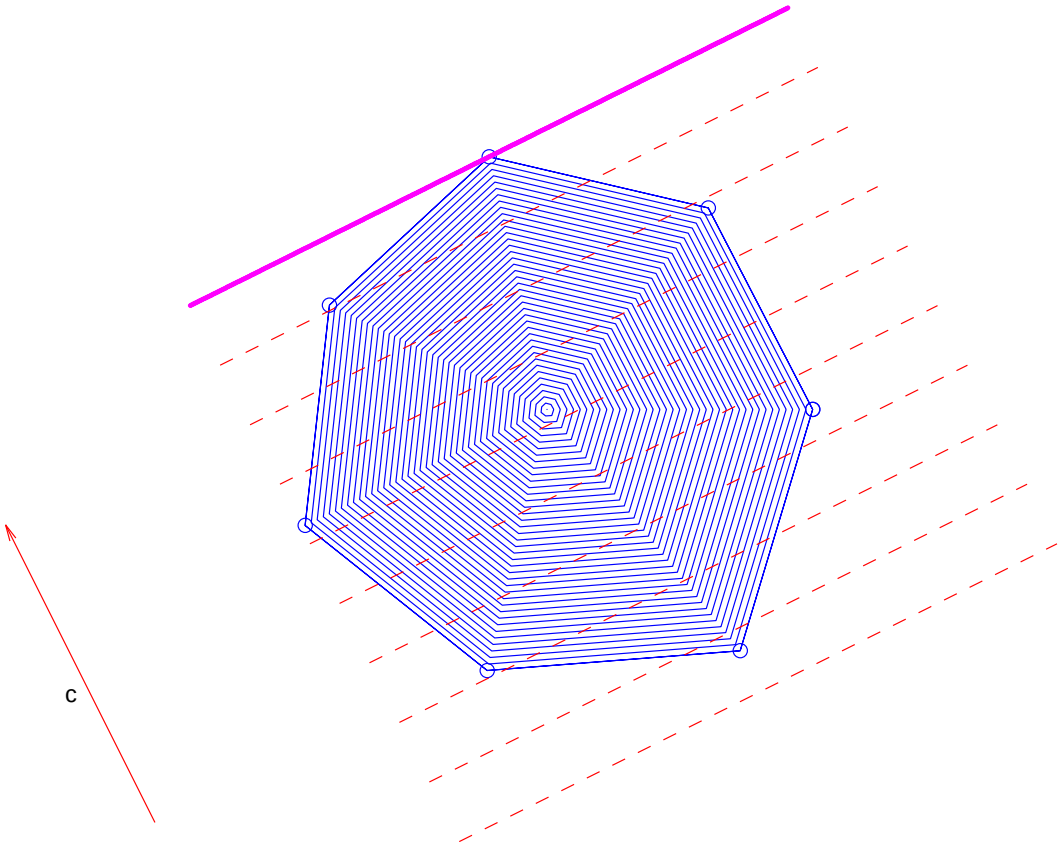
♠ Let $c \in \mathbb{R}^n$ be a nonzero vector. Consider the family of hyperplanes

$$\Pi_t = \{x \in \mathbb{R}^n : c^T x = t\},\ -\infty < t < \infty$$

• Hyperplanes of the family are parallel to each other: when $t \neq t'$, Π_t does not intersect $\Pi_{t'}$

• Hyperplanes of the family are shifts of the linear subspace $\{x : c^T x = 0\}$ (orthogonal complement to the line $\mathbb{R} \cdot c$ linearly spanned by c)

• *In a LO program* $\max_x\{c^T x : Ax \leq b\}$ *we want to find the largest* t *for which the hyperplane* $\{x : c^T x = t\}$ *intersects the feasible set of the problem.* The intersection of this "extreme" hyperplane with the feasible set is the set of optimal solutions:



Blue domain: feasible set.

Dashed lines: hyperplanes Π_t for various values of t

Bold line: the "extreme" hyperplane yielding optimal solution

♠ **Facts:**

♡ *The less is affine subspace, the smaller is dimension:* $M \subset M'$ are affine subspaces in $\mathbb{R}^n \Rightarrow \dim M \leq \dim M'$, with equality taking place iff $M = M'$.

$\Rightarrow$ Whenever M is an affine subspace in $\mathbb{R}^n$, we have $0 \leq \dim M \leq n$

♡ In every representation of an affine subspace as

$$M = \{x \in \mathbb{R}^n : Ax = b\},$$

the number of rows in A is *at least* $n - \dim M$. This number is equal to $n - \dim M$ iff the rows of A are linearly independent.

Quiz: Consider the set in $\mathbb{R}^4$

$$Y = \{y : \exists x : \begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \end{bmatrix} = \begin{bmatrix} 1 \\ 2 \\ 3 \\ 4 \end{bmatrix} + \begin{bmatrix} 1 & 1 & 1 & 0 & 0 & 0 \\ -1 & 0 & 0 & 1 & 1 & 0 \\ 0 & -1 & 0 & -1 & 0 & 1 \\ 0 & 0 & -1 & 0 & -1 & -1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \end{bmatrix} \}$$

A. Is Y a polyhedral set?

B. Is Y an affine subspace?

C. If Y is an affine subspace, then point out

— *a* shift vector a and the parallel linear subspace L

— a set of vectors affinely spanning Y

— $\dim Y$

— an affine basis in Y

— a representation of Y as a solution set of a system of linear equations

D. Is Y a linear subspace?

E. Is Y a hyperplane?

Quiz: Consider the set in $\mathbb{R}^4$

$$Y = \{y : \exists x : \begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \end{bmatrix} = \begin{bmatrix} 1 \\ 2 \\ 3 \\ 4 \end{bmatrix} + \overbrace{\begin{bmatrix} 1 & 1 & 1 & 0 & 0 & 0 \\ -1 & 0 & 0 & 1 & 1 & 0 \\ 0 & -1 & 0 & -1 & 0 & 1 \\ 0 & 0 & -1 & 0 & -1 & -1 \end{bmatrix}}^{A} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \end{bmatrix}\}$$

A. Is Y a polyhedral set? – *Yes – Y is given by polyhedral representation*

B. Is Y an affine subspace? – Yes – Y is the image of $\mathbb{R}^6$ under affine mapping

C. If Y is an affine subspace, then point out

— *a* shift vector and the parallel linear subspace L – *For example*,

$$a = \begin{bmatrix} 1 \\ 2 \\ 3 \\ 4 \end{bmatrix}, \quad \begin{array}{rcl} L & = & \text{Linear span of columns } \mathrm{Col}_j[A] \text{ of } A \\ & = & \{y \in \mathbb{R}^4 : y_1 + y_2 + y_3 + y_4 = 0\} \end{array}$$

— a set of vectors affinely spanning Y – *For example,* $\{x_0 = a, x_j = a + \mathrm{Col}_j[A], 1 \le j \le 6\}$

— $\dim Y - \dim Y = \dim L = 3$ ($\dim L$ was found in the previous quiz)

— an affine basis in Y – *For example,* $\{x_0 = a, x_j = a + u_j, j = 1, 2, 3\}$, *where* u_j, $j = 1, 2, 3$, *is a linear basis in* L (we have computed one in the previous quiz). *A sample affine basis in* Y *is*

$$\begin{bmatrix} 1 \\ 2 \\ 3 \\ 4 \end{bmatrix}, \begin{bmatrix} 2 \\ 1 \\ 3 \\ 4 \end{bmatrix}, \begin{bmatrix} 2 \\ 2 \\ 2 \\ 4 \end{bmatrix}, \begin{bmatrix} 2 \\ 2 \\ 3 \\ 3 \end{bmatrix}$$

... point out

— a representation of Y as a solution set of a system of linear equations – *A representation is*

$$Y = \{[y_1; y_2; y_3; y_4] : y_1 + y_2 + y_3 + y_4 = 10\},$$

since $Y = [1; 2; 3; 4] + L$ and

$$L = \{y \in \mathbb{R}^4 : y_1 + y_2 + y_3 + y_4 = 0\}.$$

D. Is Y a linear subspace? – No, Y does not contain the origin

E. Is Y a hyperplane? – Yes

"Calculus" of affine subspaces

♡ [taking intersection] *When M_1, M_2 are affine subspaces in $\mathbb{R}^n$ and $M_1 \cap M_2 \neq \emptyset$, the set $M_1 \cap M_2$ is an affine subspace as well.*

Extension: *If nonempty, the intersection $\bigcap\limits_{\alpha\in\mathcal{A}} M_\alpha$ of an arbitrary family $\{M_\alpha\}_{\alpha\in\mathcal{A}}$ of affine subspaces in $\mathbb{R}^n$ is an affine subspace.*

The parallel linear subspace is $\bigcap_{\alpha\in\mathcal{A}} L_\alpha$, where L_α are the linear subspaces parallel to M_α.

♡ [summation] *When M_1, M_2 are affine subspaces in $\mathbb{R}^n$, so is their arithmetic sum*

$$M_1 + M_2 = \{x = u + v : u \in M_1, v \in M_2\}.$$

The linear subspace parallel to $M_1 + M_2$ is $L_1 + L_2$, where the linear subspaces L_i are parallel to M_i, $i = 1, 2$

♡ [taking direct product] *When $M_1 \subset \mathbb{R}^{n_1}$ and $M_2 \subset \mathbb{R}^{n_2}$, the direct product (or direct sum) of M_1 and M_2 – the set*

$$M_1 \times M_2 := \{[x_1; x_2] \in \mathbb{R}^{n_1+n_2} : x_1 \in M_1, x_2 \in M_2\}$$

is an affine subspace in $\mathbb{R}^{n_1+n_2}$.

The parallel linear subspace is $L_1 \times L_2$, where linear subspaces $L_i \subset \mathbb{R}^{n_i}$ are parallel to M_i, $i = 1, 2$.

♡ [taking image under affine mapping] *When M is an affine subspace in $\mathbb{R}^n$ and $x \mapsto Px + p : \mathbb{R}^n \to \mathbb{R}^m$ is an affine mapping, the image*

$$PM + p = \{y = Px + p : x \in M\}$$

of M under the mapping is an affine subspace in $\mathbb{R}^m$.

The parallel linear subspace is

$$PL = \{y = Px : x \in L\},$$

where L is the linear subspace parallel to M.

♡ [taking inverse image under affine mapping] *When M is a linear subspace in $\mathbb{R}^n$, $x \mapsto Px + p : \mathbb{R}^m \to \mathbb{R}^n$ is an affine mapping and the inverse image*

$$Y = \{y : Py + p \in M\}$$

of M under the mapping is nonempty, Y is an affine subspace in $\mathbb{R}^m$.

The parallel linear subspace is

$$P^{-1}(L) = \{y : Py \in L\},$$

where L is the linear subspace parallel to M.

Convex Sets and Functions

♣ Definitions:

♠ A *set* $X \subset \mathbb{R}^n$ is called *convex*, if along with every two points x, y it contains the entire segment linking the points:

$$x, y \in X, \lambda \in [0,1] \Rightarrow (1-\lambda)x + \lambda y \in X.$$

♡ **Equivalently:** $X \in \mathbb{R}^n$ is convex, if X is closed w.r.t. taking all *convex combinations* of its elements (i.e., *linear combinations with nonnegative coefficients summing up to 1*):

$$\forall k \geq 1 : x_1, ..., x_k \in X, \lambda_1 \geq 0, ..., \lambda_k \geq 0, \textstyle\sum_{i=1}^k \lambda_i = 1 \\ \Rightarrow \sum_{i=1}^k \lambda_i x_i \in X$$

Indeed, computing k-term convex combination reduces to computing $k-1$ two-term ones:

$$\begin{array}{rcl} \frac{1}{10}x_1 + \frac{2}{10}x_2 + \frac{3}{10}x_3 + \frac{4}{10}x_4 & = & \frac{1}{10}x_1 + \frac{9}{10}\left[\frac{2}{9}x_2 + \frac{3}{9}x_3 + \frac{4}{9}x_4\right] \\ & = & \frac{1}{10}x_1 + \frac{9}{10}\left[\frac{2}{9}x_2 + \frac{7}{9}\left[\frac{3}{7}x_3 + \frac{4}{7}x_4\right]\right] \end{array}$$

Assuming that all 2-term convex combinations of points from X belong to X, the magenta representation of the red expression shows that the value of the red expression belongs to X.

Example of a convex set: *A polyhedral set $X = \{x \in \mathbb{R}^n : Ax \leq b\}$ is convex. In particular, linear and affine subspaces are convex sets.*

♠ A *function* $f(x) : \mathbb{R}^n \to \mathbb{R} \cup \{+\infty\}$ is called *convex*, if its *epigraph*

$$\mathrm{Epi}\{f\} = \{[x;\tau] : \tau \geq f(x)\}$$

is convex.

♡ **Equivalently:** f is convex, if

$$\begin{array}{c} x, y \in \mathbb{R}^n, \lambda \in [0,1] \\ \Rightarrow f((1-\lambda)x + \lambda y) \leq (1-\lambda)f(x) + \lambda f(y) \end{array}$$

♡ **Equivalently:** f is convex, if f satisfies the *Jensen's Inequality:*

$$\begin{array}{c} \forall k \geq 1 : x_1, ..., x_k \in \mathbb{R}^n, \lambda_1 \geq 0, ..., \lambda_k \geq 0, \sum_{i=1}^k \lambda_i = 1 \\ \Rightarrow f\left(\sum_{i=1}^k \lambda_i x_i\right) \leq \sum_{i=1}^k \lambda_i f(x_i) \end{array}$$

Example: A piecewise linear function

$$f(x) = \begin{cases} \max\limits_{i \leq I}[a_i^T x + b_i], & Px \leq p \\ +\infty, & \text{otherwise} \end{cases}$$

is convex.

♠ **Convex hull:** For a nonempty set $X \subset \mathbb{R}^n$, its *convex hull* is the set comprised of all convex combinations of elements of X:

$$\boxed{\mathrm{Conv}(X) = \left\{ x = \sum_{i=1}^{m} \lambda_i x_i : \begin{array}{c} x_i \in X,\, 1 \le i \le m \in \mathbf{N} \\ \lambda_i \ge 0\, \forall i, \sum_i \lambda_i = 1 \end{array} \right\}}$$

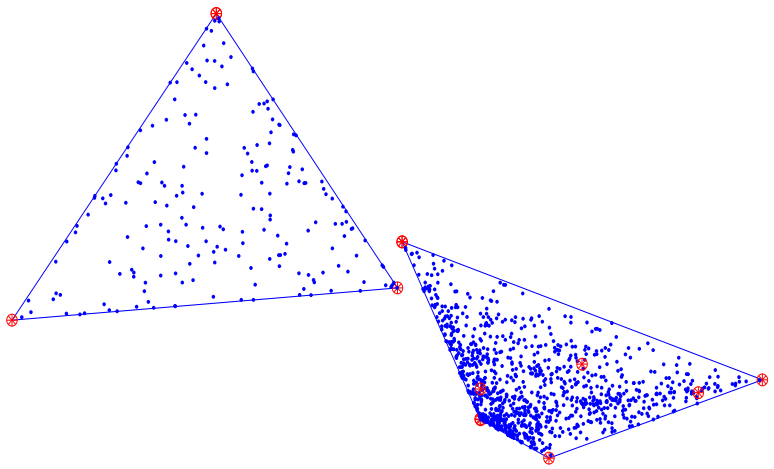
By definition, $\mathrm{Conv}(\emptyset) = \emptyset$.

Fact: *The convex hull of X is convex, contains X and is the intersection of all convex sets containing X and thus is the smallest, w.r.t. inclusion, convex set containing X.*

Note: a convex combination is an affine one, and an affine combination is a linear one, whence

$$\boxed{\begin{array}{c} X \subset \mathbb{R}^n \Rightarrow \mathrm{Conv}(X) \subset \mathrm{Lin}(X) \\ \emptyset \neq X \subset \mathbb{R}^n \Rightarrow \mathrm{Conv}(X) \subset \mathrm{Aff}(X) \subset \mathrm{Lin}(X) \end{array}}$$

Example: Convex hulls of a 3- and an 8-point sets (red dots) on the 2D plane:



♣ **Dimension of a *nonempty* set $X \in \mathbb{R}^n$:**

♡ When X is a linear subspace, $\dim X$ is the linear dimension of X (the cardinality of (any) linear basis in X)

♡ When X is an *affine subspace*, $\dim X$ is the *linear dimension of the linear subspace parallel to* X (that is, *the cardinality of* (any) *affine basis of* X *minus 1*)

♡ When X is an arbitrary nonempty subset of $\mathbb{R}^n$, $\dim X$ *is the dimension of the affine hull* $\mathrm{Aff}(X)$ of X.

Note: Some sets X are in the scope of more than one of these three definitions. For these sets, all applicable definitions result in the same value of $\dim X$.

Calculus of Convex Sets

♠ [taking intersection] If X_1, X_2 are convex sets in $\mathbb{R}^n$, so is their intersection $X_1 \cap X_2$. In fact, *the intersection*

$$\bigcap_{\alpha \in \mathcal{A}} X_\alpha$$

of a whatever family of convex subsets in $\mathbb{R}^n$ is convex.

Warning: *The union of convex sets is, in general, non-convex!*

♠ [taking arithmetic sum] *If X_1, X_2 are convex sets in $\mathbb{R}^n$, so is the set*

$$X_1 + X_2 = \{x = x_1 + x_2 : x_1 \in X_1, x_2 \in X_2\}.$$

♠ [taking affine image]: *If X is a convex set in $\mathbb{R}^n$, A is an $m \times n$ matrix, and $b \in \mathbb{R}^m$, then the set*

$$AX + b := \{Ax + b : x \in X\} \subset \mathbb{R}^m$$

(the image of X under the affine mapping $x \mapsto Ax + b : \mathbb{R}^n \to \mathbb{R}^m$) *is a convex set in $\mathbb{R}^m$.*

♠ [taking inverse affine image] *If X is a convex set in $\mathbb{R}^n$, A is an $n \times k$ matrix, and $b \in \mathbb{R}^m$, then the set*

$$\{y \in \mathbb{R}^k : Ay + b \in X\}$$

(the inverse image of X under the affine mapping $y \mapsto Ay + b : \mathbb{R}^k \to \mathbb{R}^n$) *is a convex set in $\mathbb{R}^k$.*

♠ [taking direct product] *If the sets $X_i \subset \mathbb{R}^{n_i}$, $1 \le i \le k$, are convex, so is their direct product*

$$X_1 \times ... \times X_k := \{[x^1; ...; x^k] : x^i \in X_i, 1 \le i \le k\} \subset \mathbb{R}^{n_1 + ... + n_k}.$$

Calculus of Convex Functions

♠ [taking linear combinations with positive coefficients] *If functions* $f_i : \mathbb{R}^n \to \mathbb{R} \cup \{+\infty\}$ *are convex and* $\lambda_i > 0$, $1 \le i \le k$, *then the function*

$$f(x) = \sum_{i=1}^k \lambda_i f_i(x)$$

is convex.

♠ [direct summation] *If functions* $f_i : \mathbb{R}^{n_i} \to \mathbb{R} \cup \{+\infty\}$, $1 \le i \le k$, *are convex, so is their direct sum*

$$f([x^1; ...; x^k]) = \sum_{i=1}^k f_i(x^i) : \mathbb{R}^{n_1+...+n_k} \to \mathbb{R} \cup \{+\infty\}$$

♠ [taking supremum] *The supremum* $f(x) = \sup\limits_{\alpha \in \mathcal{A}} f_\alpha(x)$ *of a whatever (nonempty) family* $\{f_\alpha\}_{\alpha \in \mathcal{A}}$ *of convex functions is convex.*

♠ [affine substitution of argument] *If a function* $f(x) : \mathbb{R}^n \to \mathbb{R} \cup \{+\infty\}$ *is convex and* $x = Ay + b : \mathbb{R}^m \to \mathbb{R}^n$ *is an affine mapping, then the function*

$$g(y) = f(Ay + b) : \mathbb{R}^m \to \mathbb{R} \cup \{+\infty\}$$

is convex.

♠ [projective transformation] *If a function* $f(x) : \mathbb{R}^n \to \mathbb{R} \cup \{+\infty\}$ *is convex, so is its projective transformation*

$$g(y = [x; \alpha]) = \begin{cases} \alpha f(x/\alpha), & \alpha > 0 \text{ and } x/\alpha \in \mathrm{Dom} f \\ +\infty, & \text{otherwise} \end{cases}$$

♠ [partial minimization] *If a function*

$$f([u;v]) : \mathbb{R}^{n_u} \times \mathbb{R}^{n_v} \to \mathbb{R} \cup \{+\infty\}$$

is convex, then the function

$$g(u) = \inf_v f(u,v) : \mathbb{R}^{n_v} \to \mathbb{R} \cup \{+\infty\} \cup \{-\infty\}$$

is convex on every convex set on which g does not take the value $-\infty$.

♠ **Theorem on superposition:** *Let*

$$f_i(x) : \mathbb{R}^n \to \mathbb{R} \cup \{+\infty\}$$

be convex functions, and let

$$F(y) : \mathbb{R}^m \to \mathbb{R} \cup \{+\infty\}$$

be a convex function which is nondecreasing w.r.t. every one of the variables $y_1, ..., y_m$*. Then the superposition*

$$g(x) = \begin{cases} F(f_1(x), ..., f_m(x)), & f_i(x) < +\infty, 1 \le i \le m \\ +\infty, & \text{otherwise} \end{cases}$$

of F *and* $f_1, ..., f_m$ *is convex.*

Note: *Monotonicity of the outer function* F *is essential!*

For example,

• $f(x) = \exp\{x\}$ and $F(y) = -y$ are convex functions, but $g(x) = F(f(x)) = -\exp\{x\}$ is nonconvex!

Note: If some of f_i's, say, $f_1, ..., f_k$, are affine, then the conclusion of Theorem on superposition remains valid when we require the monotonicity of F w.r.t. $y_{k+1}, ..., y_m$ only.

Cones

♣ **Definition:** *A set $X \subset \mathbb{R}^n$ is called a cone, if X is nonempty, convex and is homogeneous*, that is,

$$x \in X, \lambda \geq 0 \Rightarrow \lambda x \in X$$

Geometrically: A cone is a convex set comprised of *rays* emanating from the origin.

Equivalently: *A set $X \subset \mathbb{R}^n$ is a cone, if X is nonempty and is closed w.r.t. addition of its elements and multiplication of its elements by nonnegative reals:*

$$x, y \in X, \lambda, \mu \geq 0 \Rightarrow \lambda x + \mu y \in X$$

Equivalently: *A set $X \subset \mathbb{R}^n$ is a cone, if X is nonempty and is closed w.r.t. taking conic combinations of its elements* (that is, *linear combinations with nonnegative coefficients*):

$$\forall m : x_i \in X, \lambda_i \geq 0, 1 \leq i \leq m \Rightarrow \sum_{i=1}^{m} \lambda_i x_i \in X.$$

Examples:

• Every linear subspace in $\mathbb{R}^n$ (i.e., every solution set of a *homogeneous* system of linear equations with n variables) is a cone

• The solution set $X = \{x \in \mathbb{R}^n : Ax \leq 0\}$ of a *homogeneous* system of linear inequalities is a cone. Such a cone is called *polyhedral*.

Quiz: By our definition, a polyhedral cone is the solution set of *homogeneous* system $Ax \leq 0$ of linear inequalities.
Is it exactly the same as to say that X is a cone *and* X is polyhedral?

Quiz: By our definition, a polyhedral cone is the solution set of *homogeneous* system $Ax \leq 0$ of linear inequalities.

Is it exactly the same as to say that X is a cone *and* X is polyhedral?

Yes! If a polyhedral set $X = \{x : Ax \leq b\}$ is a cone, then $X = \{x : Ax \leq 0\}$, that is, X is a polyhedral cone.

Indeed, when $X = \{x : Ax \leq b\}$ is a cone, then $0 \in X$ and therefore $b \geq 0$, so that the polyhedral cone

$$\bar{X} = \{x : Ax \leq 0\}$$

is contained in X. On the other hand, for every $x \in X$, we have $A[tx] \leq b$ for all $t \geq 0$ (since X is a cone)

$\Rightarrow$ $t[Ax] \leq b$ for all positive t,

$\Rightarrow$ $Ax \leq 0$,

so that X is contained in $\bar{X}$.

♣ **Conic hull:** *For a nonempty set* $X \subset \mathbb{R}^n$*, its conic hull* $\mathrm{Cone}\,(X)$ *is defined as the set of all conic combinations of elements of* X*:*

$$X \neq \emptyset$$
$$\Rightarrow \mathrm{Cone}\,(X) = \left\{ x = \sum_i \lambda_i x_i : \begin{array}{l} \lambda_i \geq 0, 1 \leq i \leq m \in \mathbf{N} \\ x_i \in X, 1 \leq i \leq m \end{array} \right\}$$

By definition, $\mathrm{Cone}\,(\emptyset) = \{0\}$.

Fact: $\mathrm{Cone}\,(X)$ *is a cone, contains* X *and is the intersection of all cones containing* X*, and thus is the smallest, w.r.t. inclusion, cone containing* X.

Example: *The conic hull of the set* $X = \{e_1, ..., e_n\}$ *of all basic orths in* $\mathbb{R}^n$ *is the nonnegative orthant* $\mathbb{R}^n_+ = \{x \in \mathbb{R}^n : x \geq 0\}$.

Calculus of Cones

♠ [taking intersection] *If X_1, X_2 are cones in $\mathbb{R}^n$, so is their intersection $X_1 \cap X_2$.*

In fact, *the intersection $\bigcap_{\alpha\in\mathcal{A}} X_\alpha$ of a whatever family $\{X_\alpha\}_{\alpha\in\mathcal{A}}$ of cones in $\mathbb{R}^n$ is a cone.*

♠ [taking arithmetic sum] *If X_1, X_2 are cones in $\mathbb{R}^n$, so is the set $X_1 + X_2 = \{x = x_1 + x_2 : x_1 \in X_1, x_2 \in X_2\}$*

♠ [taking linear image] *If X is a cone in $\mathbb{R}^n$ and A is an $m \times n$ matrix, then the set*

$$AX := \{Ax : x \in X\} \subset \mathbb{R}^m$$

(the image of X under the linear mapping $x \mapsto Ax : \mathbb{R}^n \to \mathbb{R}^m$) *is a cone in $\mathbb{R}^m$.*

♠ [taking inverse linear image] *If X is a cone in $\mathbb{R}^n$ and A is an $n \times k$ matrix, then the set*

$$\{y \in \mathbb{R}^k : Ay \in X\}$$

(the inverse image of X under the linear mapping $y \mapsto Ay : \mathbb{R}^k \to \mathbb{R}^n$) *is a cone in $\mathbb{R}^k$.*

♠ [taking direct products] *If $X_i \subset \mathbb{R}^{n_i}$ are cones, $1 \le i \le k$, so is the direct product*

$$X_1 \times ... \times X_k := \{[x^1; ...; x^k] : x^i \in X_i, 1 \le i \le k\} \subset \mathbb{R}^{n_1+...+n_k}.$$

♠ [passing to the dual cone] *If X is a cone in $\mathbb{R}^n$, so is its dual cone defined as*

$$X_* = \{y \in \mathbb{R}^n : y^T x \geq 0 \ \forall x \in X\}.$$

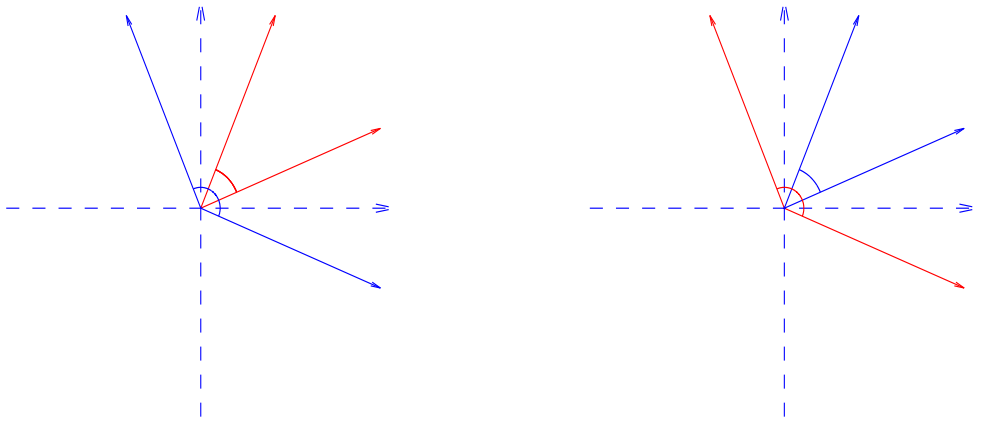
Examples:

• The cone dual to a linear subspace L is the orthogonal complement $L^\perp$ of L

• The cone dual to the nonnegative orthant $\mathbb{R}^n_+$ is the nonnegative orthant itself:

$$(\mathbb{R}^n_+) := \{y \in \mathbb{R}^n : y^T x \geq 0 \, \forall x \geq 0\} = \{y \in \mathbb{R}^n : y \geq 0\}.$$

• 2D cones bounded by blue rays are dual to cones bounded by red rays:



Quiz: Let $a_1, ..., a_m \in \mathbb{R}^n$, and let

$$K = \text{Cone}\{a_1, ..., a_m\} := \left\{\sum_{i=1}^m \lambda_i a_i : \lambda \geq 0\right\}$$

be the conic hull of $a_1, ..., a_m$.

- Is K a polyhedral cone?
- What is the cone K_* dual to K?

Quiz: Let $a_1, ..., a_m \in \mathbb{R}^n$, and let

$$K = \text{Cone}\{a_1, ..., a_m\} := \left\{\sum_{i=1}^m \lambda_i a_i : \lambda \geq 0\right\}$$

be the conic hull of $a_1, ..., a_m$.

• Is K a polyhedral cone? – Yes! K admits immediate polyhedral representation:

$$K = \{x : \exists \lambda : x = \sum_{i=1}^m \lambda_i a_i \ \& \ \lambda \geq 0\}$$

$\Rightarrow$ K is polyhedral $\Rightarrow$ K is a polyhedral cone

• What is the cone K_* dual to K? – This is the cone

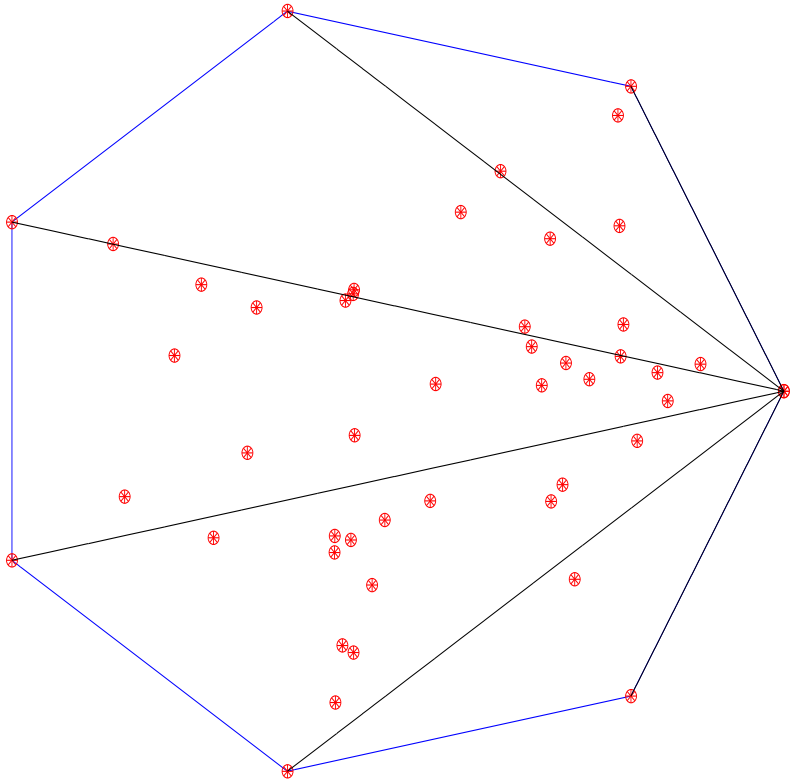
$$K_* = \{x \in \mathbb{R}^n : a_i^T x \geq 0, \ i = 1, ..., m\}$$

Indeed, a vector x has nonnegative inner products with all conic combinations of $a_1, ..., a_m$ iff x has nonnegative inner products with every one of the vectors $a_1, ..., a_m$.

Useful Fact: Caratheodory Theorem

Theorem. *Let $x_1, ..., x_N \in \mathbb{R}^n$ and $m = \dim\{x_1, ..., x_N\}$. Then every point x which is a convex combination of $x_1, ..., x_N$ can be represented as a convex combination of at most $m+1$ of the points $x_1, ..., x_N$.*

Illustration:



What we see: The 2D polygon X bounded by the blue contour is the convex hull of the set of red points. The dimension of the polygon is $m = 2$.

• Splitting the polygon into triangles, we see that *every point from X is a convex combination of $3 = m + 1$ of the red points* (and even of a triple of red points which form vertices of X).

Quiz:

- In the nature, there are 26 "pure" types of tea, denoted A, B,..., Z; all other types are mixtures of these "pure" types. In the market, 111 blends of pure types, rather than the pure types of tea themselves, are sold.
- John prefers a specific blend of tea which is not sold in the market; from experience, he found that in order to get this blend, he can buy 93 of the 111 market blends and mix them in certain proportion.
- An OR student pointed out that to get his favorite blend, John could mix appropriately just 27 properly selected market blends. Another OR student found that just 26 of market blends are enough.
- John does not believe the students, since no one of them asked what exactly is his favorite blend. Is John right?

Quiz:

• In the nature, there are 26 "pure" types of tea, denoted A, B,..., Z. In the market, 111 blends of these types are sold.

• John knows that his favorite blend can be obtained by mixing in appropriate proportion 93 of the 111 market blends. Is it true that the same blend can be obtained by mixing

- 27 market blends?
- 26 market blends?

Both answers are true. Let us speak about *unit weight* portions of tea blends. Then

• a blend can be identified with 26-dimensional vector

$$x = [x_A; ...; x_Z]$$

where $x_?$ is the weight of pure tea ? in the unit weight portion of the blend. The 26 entries in x are nonnegative and sum up to 1;

• denoting the marked blends by $x^1, ..., x^{111}$ and the favorite blend of John by $\bar{x}$, we know that

$$\bar{x} = \sum_{i=1}^{111} \lambda_i x^i$$

with nonnegative coefficients λ_i. Comparing the weights of both sides, we conclude that $\sum_{i=1}^{111} \lambda_i = 1$

$\Rightarrow$ $\bar{x}$ is a convex combination of $x^1, ..., x^{111}$

$\Rightarrow$ [by Caratheodory and due to $\dim x^i = 26$] $\bar{x}$ *is a convex combination of just* $26 + 1 = 27$ *of the market blends*, thus the first student is right.

• The vectors $x^1, ..., x^{111}$ have unit sums of entries thus belong to the hyperplane

$$M = \{[x_A; ...; x_Z] : x_A + ... + x_Z = 1\}$$

which has dimension 25

$\Rightarrow$ The dimension of the set $\{x^1, x^2, ..., x^{111}\}$ is at most $m = 25$

$\Rightarrow$ *By Caratheodory, $\bar{x}$ is a convex combination of just $m + 1 = 26$ vectors from $\{x^1, ..., x^{111}\}$*, thus the second student also is right.

Proof of Caratheodory Theorem

• Let $M = \mathrm{Aff}\{x_1, ..., x_N\}$, so that $\dim M = m$. By shifting M (which does not affect the statement we intend to prove) we can make M a m-dimensional linear subspace in $\mathbb{R}^n$. Representing points from the linear subspace M by their m-dimensional vectors of coordinates in a basis of M, we can identify M and $\mathbb{R}^m$, and this identification does not affect the statement we intend to prove. Thus, assume w.l.o.g. that $m = n$.

• Let $x = \sum_{i=1}^{N} \mu_i x_i$ be a representation of x as a convex combination of $x_1, ..., x_N$ *with as small number of nonzero coefficients as possible.* Reordering $x_1, ..., x_N$ and omitting terms with zero coefficients, assume w.l.o.g. that $x = \sum_{i=1}^{M} \mu_i x_i$, so that $\mu_i > 0,\ 1 \le i \le M$, and $\sum_{i=1}^{M} \mu_i = 1$. It suffices to show that $M \le n+1$. Let, on the contrary, $M > n+1$.

• Consider the system of linear equations in variables $\delta_1, ..., \delta_M$:

$$\sum_{i=1}^{M} \delta_i x_i = 0;\ \sum_{i=1}^{M} \delta_i = 0$$

This is a homogeneous system of $n+1$ linear equations in $M > n+1$ variables, and thus it has a nontrivial solution $\bar{\delta}_1, ..., \bar{\delta}_M$. Setting $\mu_i(t) = \mu_i + t\bar{\delta}_i$, we have

$$\forall t : x = \sum_{i=1}^{M} \mu_i(t) x_i, \sum_{i=1}^{M} \mu_i(t) = 1.$$

• Since $\bar{\delta}$ is nontrivial and $\sum_i \bar{\delta}_i = 0$, the set $I = \{i : \delta_i < 0\}$ is nonempty. Let $\bar{t} = \min\limits_{i \in I} \mu_i / |\delta_i|$. Then all $\mu_i(\bar{t})$ are ≥ 0, at least one of $\mu_i(\bar{t})$ is zero, and

$$x = \sum_{i=1}^{M} \mu_i(\bar{t}) x_i,\ \sum_{i=1}^{M} \mu_i(\bar{t}) = 1.$$

We get a representation of x as a convex combination of x_i with *less than* M nonzero coefficients, which is impossible. □

Useful Fact: Helley Theorem

Theorem. *Let $A_1, ..., A_N$ be convex sets in $\mathbb{R}^n$ which belong to an affine subspace M of dimension m. Assume that every $m+1$ sets of the collection have a point in common. Then all N sets have a point in common.*

Illustration:

- If in a system of 2013 segments on the real axis, every 2 segments intersect, all 2013 segments have a point in common (easy!)
- If in a system of 2013 triangles on the 2D plane, every triple of triangles have a point in common, all 2013 triangles have a point in common (???)

Quiz: The daily functioning of a plant is described by the linear constraints

$$\begin{array}{lrcl} (a) & Ax & \leq & f \in \mathbb{R}^{10} \\ (b) & Bx & \geq & d \in \mathbb{R}^{2013} \\ (c) & Cx & \leq & c \in \mathbb{R}^{2000} \end{array} \qquad (!)$$

- x: decision vector
- $f \in \mathbb{R}^{10}_{+}$: vector of resources
- d: vector of demands
- There are N demand scenarios d^i. In the evening of day $t-1$, the manager knows that the demand of day t will be one of the N scenarios, but he does *not* know which one. The manager should arrange a vector of resources f for the next day, at a price $c_\ell \geq 0$ per unit of resource f_ℓ, in order to make the next day production problem feasible.
- It is known that every one of the demand scenarios can be "served" by \$1 purchase of resources.

(?) *How much should the manager invest in resources to make the next day problem feasible when*

- $N = 1$ • $N = 2$ • $N = 10$ • $N = 11$
- $N = 12$ • $N = 2013$?

$(a): Ax \le f \in \mathbb{R}^{10}$; $(b): Bx \ge d \in \mathbb{R}^{2013}$; $(c): Cx \le c \in \mathbb{R}^{2000}$

Quiz answer: With N scenarios, \$ $\min[N, 11]$ is enough!

Indeed, the vector of resources $f \in \mathbb{R}^{10}_+$ appears only in the constraints (a)

$\Rightarrow$ surplus of resources makes no harm

$\Rightarrow$ with N scenarios d^i, \$ N in resources is enough: every d^i can be "served" by \$ 1 purchase of appropriate resource vector $f^i \ge 0$, thus it suffices to buy the vector $f^1 + ... + f^N$ which costs \$ N and is $\ge f^i$ for every $i = 1, ..., n$.

To see than \$ 11 is enough, let F_i be the set of all resource vectors f which cost at most \$11 and allow to "serve" demand $d^i \in D$.

A. $F_i \in \mathbb{R}^{10}$ is convex (and even polyhedral): it admits polyhedral representation

$F_i = \{f \in \mathbb{R}^{10} : \exists x : Cx \le c, Bx \ge d^i, Ax \le f, f \ge 0, \sum_{\ell=1}^{10} c_\ell f_\ell \le 11\}$

B. Every 11 sets $F_{i_1}, ..., F_{i_{11}}$ of the family $F_1, ..., F_i$ have a point in common. Indeed, scenario d^{i_s} can be "served" by \$ 1 vector $f^s \ge 0$

$\Rightarrow$ every one of the scenarios $d^{i_1}, ..., d^{i_11}$ can be served by the \$ 11 vector of resources $f = f^1 + ... + f^{11}$

$\Rightarrow$ f belongs to every one of $F_{i_1}, ..., F_{i_{11}}$

• By Helley, **A** and **B** imply that all the sets $F_1, ..., F_N$ have a point f in common. f costs at most \$ 11 (the description of F_i) and allows to "serve" *every one* of the demands d^1,...,d^N.

Proof of Helley Theorem.

• Same as in the proof of Caratheodory Theorem, we can assume w.l.o.g. that $m = n$.

• We need the following fact:

Theorem [Radon] *Let $x_1, ..., x_N$ be points in $\mathbb{R}^n$. If $N \geq n + 2$, we can split the index set $\{1, ..., N\}$ into two nonempty non-overlapping subsets I, J such that*

$$\text{Conv}\{x_i : i \in I\} \cap \text{Conv}\{x_i : i \in J\} \neq \emptyset.$$

From Radon to Helley: Let us prove Helley's theorem by indiction in N. There is nothing to prove when $N \leq n + 1$. Thus, assume that $N \geq n + 2$ and that the statement holds true for all collections of $N - 1$ sets, and let us prove that the statement holds true for N-element collections of sets as well.

Proof of Helley Theorem (continued)

• Given $A_1, ..., A_N$, we define the N sets

$$B_i = A_1 \cap A_2 \cap ... \cap A_{i-1} \cap A_{i+1} \cap ... \cap A_N.$$

By inductive hypothesis, all B_i are nonempty. Choosing a point $x_i \in B_i$, we get $N \geq n+2$ points x_i, $1 \leq i \leq N$.

• By Radon Theorem, after appropriate reordering of the sets $A_1, ..., A_N$, we can assume that for certain k, $\text{Conv}\{x_1, ..., x_k\} \cap \text{Conv}\{x_{k+1}, ..., x_N\} \neq \emptyset$. We claim that *if* $b \in \text{Conv}\{x_1, ..., x_k\} \cap \text{Conv}\{x_{k+1}, ..., x_N\}$, *then* b *belongs to all* A_i, which would complete the inductive step.

To support our claim, note that

— when $i \leq k$, $x_i \in B_i \subset A_j$ for all $j = k+1, ..., N$, that is, $i \leq k \Rightarrow x_i \in \cap_{j=k+1}^{N} A_j$. Since the latter set is convex and b is a convex combination of $x_1, ..., x_k$, we get $b \in \cap_{j=k+1}^{N} A_j$.

— when $i > k$, $x_i \in B_i \subset A_j$ for all $1 \leq j \leq k$, that is, $i \geq k \Rightarrow x_i \in \cap_{j=1}^{k} A_j$. Similarly to the above, it follows that $b \in \cap_{j=1}^{k} A_j$.

Thus, our claim is correct.

Proof of Radon Theorem

Let $x_1, ..., x_N \in \mathbb{R}^n$ and $N \geq n+2$. We want to prove that we can split the set of indexes $\{1, ..., N\}$ into non-overlapping nonempty sets I, J such that $\mathrm{Conv}\{x_i : i \in I\} \cap \mathrm{Conv}\{x_i : i \in J\} \neq \emptyset$.

Indeed, consider the system of $n+1 < N$ homogeneous linear equations in $n+1$ variables $\delta_1, ..., \delta_N$:

$$\sum_{i=1}^{N} \delta_i x_i = 0, \ \sum_{i=1}^{N} \delta_i = 0. \qquad (*)$$

This system has a nontrivial solution $\bar{\delta}$. Let us set $I = \{i : \bar{\delta}_i > 0\}$, $J = \{i : \bar{\delta}_i \leq 0\}$. Since $\bar{\delta} \neq 0$ and $\sum_{i=1}^{N} \bar{\delta}_i = 0$, both I, J are nonempty, do not intersect and $\mu := \sum_{i \in I} \bar{\delta}_i = \sum_{i \in J} [-\bar{\delta}_i] > 0$. $(*)$ implies that

$$\underbrace{\sum_{i \in I} \frac{\bar{\delta}_i}{\mu} x_i}_{\in \mathrm{Conv}\{x_i : i \in I\}} = \underbrace{\sum_{i \in J} \frac{[-\bar{\delta}_i]}{\mu} x_i}_{\in \mathrm{Conv}\{x_i : i \in J\}} \qquad \square$$

Useful Fact: Homogeneous Farkas Lemma

♣ **Question:** When a *homogeneous* linear inequality

$$a^T x \geq 0 \qquad (*)$$

is a consequence of a system of *homogeneous* linear inequalities

$$a_i^T x \geq 0,\ i = 1, ..., m \qquad (!)$$

i.e., when $(*)$ is satisfied at every solution to $(!)$?

Observation: If a is a conic combination of $a_1, ..., a_m$:

$$\exists \lambda_i \geq 0 : a = \sum_i \lambda_i a_i, \qquad (+)$$

then $(*)$ is a consequence of $(!)$.

Indeed, $(+)$ implies that

$$a^T x = \sum_i \lambda_i a_i^T x \ \forall x,$$

and thus for every x with $a_i^T x \geq 0 \, \forall i$ one has $a^T x \geq 0$.

♣ **Homogeneous Farkas Lemma:** $(*)$ *is a consequence of* $(!)$ *if and only if* a *is a conic combination of* $a_1, ..., a_m$.

♣ **Equivalently:** *Given vectors* $a_1, ..., a_m \in \mathbb{R}^n$*, let*

$$K = \mathrm{Cone}\,\{a_1, ..., a_m\} = \{\textstyle\sum_i \lambda_i a_i : \lambda \geq 0\}$$

be the conic hull of the vectors. Given a vector a*,*

- *it is easy to certify that* $a \in \mathrm{Cone}\,\{a_1, ..., a_m\}$*: a certificate is a collection of weights* $\lambda_i \geq 0$ *such that* $\sum_i \lambda_i a_i = a$*;*
- *it is easy to certify that* $a \notin \mathrm{Cone}\,\{a_1, ..., a_m\}$*: a certificate is a vector* d *such that* $a_i^T d \geq 0\ \forall i$ *and* $a^T d < 0$*.*

Proof of HFL: All we need to prove is that *If a is not a conic combination of $a_1, ..., a_m$, then there exists d such that $a^T d < 0$ and $a_i^T d \geq 0$, $i = 1, ..., m$.*

Fact: As we know from one of the quizzes, the cone $K = \text{Cone}\{a_1, ..., a_m\}$ is a polyhedral set and thus is a polyhedral cone

$\Rightarrow$ K can be represented as

$$K = \{x : d_j^T x \geq 0,\ 1 \leq j \leq M\}$$

- $a_i \in K \Rightarrow d_j^T a_i \geq 0$ for all $i = 1, ..., m$, $j = 1, ..., M$
- $a \notin K \Rightarrow d_{j_*}^T a < 0$ for some $j_* \leq M$.

$\Rightarrow$ Setting $d = d_{j_*}$, we get $a_i^T d \geq 0$ for all $i \leq m$, and $a^T d < 0$, Q.E.D.

Corollary: *Let $a_1, ..., a_m \in \mathbb{R}^n$ and $K = \text{Cone}\{a_1, ..., a_m\}$, and let $K_* = \{x \in \mathbb{R}^n : x^T u \geq 0 \forall u \in K\}$ be the dual cone. Then K itself is the cone dual to K_*:*

$$(K_*)_* := \{u : u^T x \geq 0 \ \forall u \in K_*\}$$
$$= K := \{\textstyle\sum_i \lambda_i a_i : \lambda_i \geq 0\}.$$

Proof.

♠ If K is a cone, then, by definition of K_*, every vector from K has nonnegative inner products with all vectors from K_* and thus $K \subset (K_*)_*$ for every cone K.

♠ To prove the opposite inclusion $(K_*)_* \subset K$ in the case when

$$K = \text{Cone}\{a_1, ..., a_m\},$$

recall that

$$K_* = \{d : d^T a_i \geq 0, \ 1 \leq i \leq m\}.$$

Now let $a \in (K_*)_*$, and let us verify that $a \in K$. Assuming this is *not* the case, by HFL there exists d such that $a^T d < 0$ and $a_i^T d \geq 0 \forall i \Rightarrow d \in K_*$, that is, $a \notin (K_*)_*$, which is a contradiction.

Corollary: *For every polyhedral cone K, the dual cone K_**

- *is polyhedral and can be represented as*

Cone (*finite set*)

- *satisfies $(K_*)_* = K$*

Indeed, let $K = \{x : d_i^T x \geq 0, 1 \leq i \leq M\}$ be a polyhedral cone. The dual cone K_* is, by definition, comprised of all vectors a which have nonnegative inner products with all vectors $x \in K$, i.e., with all x's satisfying $d_i^T x \geq 0$, $i = 1, ..., M$. By HFL, these vectors a are exactly conic combinations of $d_1, ..., d_M$, that is, $K_* = \text{Cone}\,\{d_1, ..., d_M\}$, and we know that the cone of this form is polyhedral. Besides this, by one of the quizzes

$$(K_*)_* = (\text{Cone}\,\{d_1, ..., d_M\})_* = \{x : d_i^T x \geq 0,\ i = 1, ..., M\},$$

$$\Rightarrow\ (K_*)_* = K$$

Corollary: *Every polyhedral cone K can be represented as $K = \text{Cone}\,\{$finite set$\}$.*

Indeed, by previous Corollary

- every polyhedral cone K is the dual of another polyhedral cone (specifically, K_*)
- the dual of a polyhedral cone is of the form Cone (finite set).

Quiz: $X = \{x \in \mathbb{R}^n : Ax \leq b\}$ is a polyhedral set. Let us set

$$X^+ = \{[x;t] \in \mathbb{R}^{n+1} : t \geq 0 \ \& \ Ax \leq tb\}$$

A: Is X a polyhedral cone? How to recover $X \subset \mathbb{R}^n$ given $X^+ \subset \mathbb{R}^{n+1}$?

B: By description of X_+, the t-coordinate of every point from X^+ is nonnegative. Is it true that when a point from X^+ differs from the origin, its t-coordinate is positive?

C: Now let X be *nonempty* and bounded, and let $[x;t]$ be a nonzero point from X^+. Is it true that $t > 0$?

D: Let X be nonempty and bounded. Is it true that X is the convex hull of a finite set?

Quiz: $X = \{x \in \mathbb{R}^n : Ax \leq b\}$ is a polyhedral set. Let us set

$$X^+ = \{[x;t] \in \mathbb{R}^{n+1} : t \geq 0 \ \& \ Ax \leq tb\}$$

A: Is X a polyhedral cone? How to recover $X \subset \mathbb{R}^n$ given $X^+ \subset \mathbb{R}^{n+1}$? – Yes, X^+ is a polyhedral cone - it is given by a finite system of nonstrict homogeneous linear inequalities. Besides,

$$X = \{x : [x;1] \in X^+\}$$

Geometrically: X is the cross-section of X^+ by the hyperplane $\{t = 1\}$.

$\Rightarrow$ X^+ contains points with $t = 1$ iff $X \neq \emptyset$.

B: By description of X_+, the t-coordinate of every point from X^+ is nonnegative. Is it true that when a point from X^+ differs from the origin, its t-coordinate is positive? – Not necessarily. For example, when

$$X = \{x \in \mathbb{R} : x \leq 0\},$$

we have

$$\begin{array}{rcl} X^+ & = & \{[x;t] \in \mathbb{R}^2 : t \geq 0, x \leq t \cdot 0 = 0\} \\ & = & \{[x;t] \in \mathbb{R}^2 : x \leq 0, t \geq 0\}, \end{array}$$

$\Rightarrow$ X^+ has plenty nonzero points with zero t-coordinate.

$$X = \{x : Ax \leq b\}, \quad X^+ = \{[x;t] : t \geq 0, Ax \leq tb\}$$
$$X = \{x : [x;1] \in X^+\}.$$

C: Let X be nonempty and bounded, and let $[x;t]$ be a nonzero point from X^+. Is it true that $t > 0$? – Yes!. Indeed, let $0 \neq [x;t] \in X^+$; we should verify that $t > 0$. Assuming the opposite, our point is $[x;0]$ with $x \neq 0$ and $Ax \leq 0$. Since X is nonempty, there exists $\bar{x}$ with $A\bar{x} \leq b$

$\Rightarrow A[\bar{x} + sx] \leq b$ for all $s \geq 0$

$\Rightarrow$ The ray $\{\bar{x} + sx : s \geq 0\}$ belongs to X. Since $x \neq 0$, this ray is unbounded, which is a desired contradiction – X is bounded and therefore cannot contain unbounded set!

$$X = \{x : Ax \le b\}, \quad X^+ = \{[x;t] : t \ge 0, Ax \le tb\}$$
$$X = \{x : [x;1] \in X^+\}$$

If $X \neq \emptyset$ and X is bounded, every *nonzero* $[x;t] \in X^+$ has $t > 0$

D: Let X be nonempty and bounded. Is it true that X is the convex hull of a finite set? – Yes!

- X^+ is a polyhedral cone and as such is of the form Cone $\{a_1, ..., a_m\}$.
- Since $X^+ \neq \{0\}$, not all a_i are zero vectors. Removing from the collection $a_1, ..., a_m$ zero vectors, if any, we do not affect the conic hull of the collection

$\Rightarrow$ We can assume that $X^+ =$ Cone $\{a_1, ...a_m\}$ with nonzero $a_i = [x_i; t_i]$.

- Since $0 \neq [x_i; t_i] \in X^+$, we have by above $t_i > 0$

$\Rightarrow$ the vectors $\bar{a}_i = [\bar{x}_i = x_i/t_i; 1]$ are well defined, and clearly $X^+ =$ Cone $\{a_1, ..., a_m\} =$ Cone $\{\bar{a}_1, ..., \bar{a}_m\}$.

- Since $\bar{a}_i = [\bar{x}_i; 1]$ belong to X^+, we have $\bar{x}_i \in X$, $1 \le i \le m$
- Now let $x \in X$, so that $[x; 1] \in X^+ =$ Cone $\{\bar{a}_1, ..., \bar{a}_m\}$

$\Rightarrow [x; 1] = \sum_{i=1}^m \lambda_i [\bar{x}_i; 1]$ for some $\lambda_i \ge 0 \Rightarrow \sum_i \lambda_i = 1$

$\Rightarrow$ every $x \in X$ is a convex combination of $\bar{x}_i \in X$, $i = 1, ..., m \Rightarrow X =$ Conv$\{\bar{x}_1, ..., \bar{x}_m\}$

Understanding Structure of a Polyhedral Set

♣ **Situation:** We consider a polyhedral set

$$X = \{x \in \mathbb{R}^n : Ax \leq b\}, A = \begin{bmatrix} a_1^T \\ \cdots \\ a_m^T \end{bmatrix} \in \mathbb{R}^{m \times n} \qquad (X)$$

$$\Leftrightarrow X = \left\{x \in \mathbb{R}^n : a_i^T x \leq b_i,\ i \in \mathcal{I} = \{1, ..., m\}\right\}.$$

Standing assumption: $X \neq \emptyset$.

♠ **Faces of X.** Let us pick a subset $I \subset \mathcal{I}$ and replace in (X) the inequality constraints $a_i^T x \leq b_i$, $i \in I$ with their equality versions $a_i^T x = b_i$, $i \in I$. The resulting set

$$X_I = \{x \in \mathbb{R}^n : a_i^T x \leq b_i, i \in \mathcal{I} \backslash I,\ a_i^T x = b_i,\ i \in I\}$$

if nonempty, is called a *face* of X.

Examples:

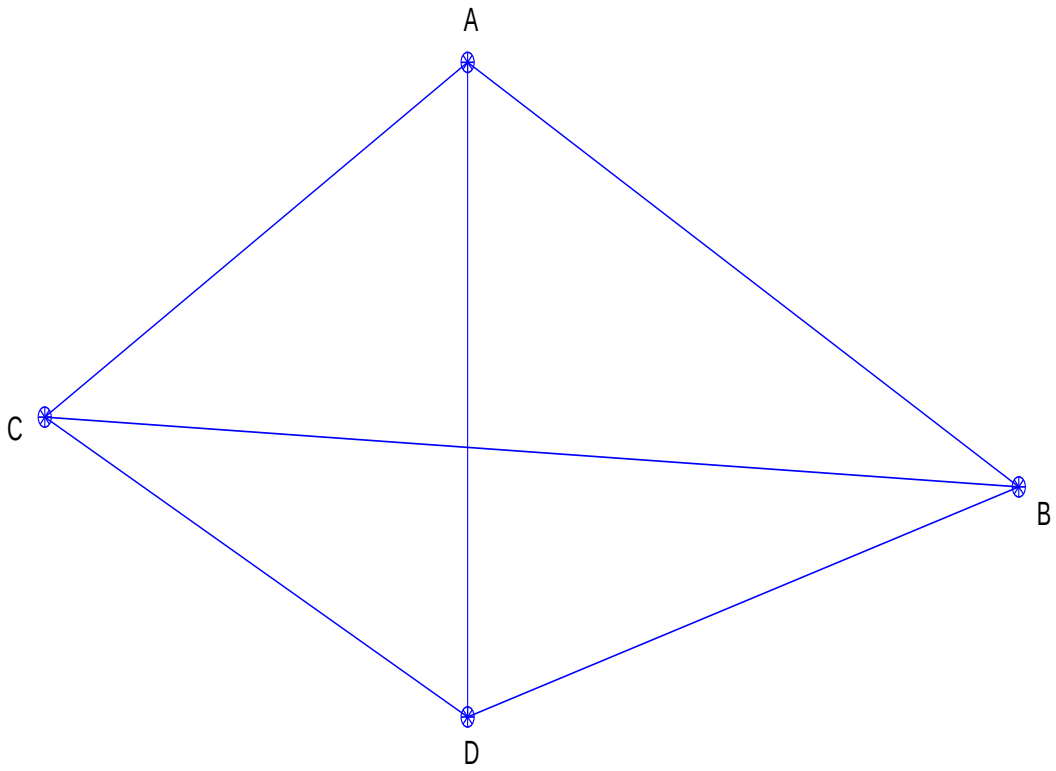
- X is a face of itself: $X = X_\emptyset$.
- Let $\Delta_n = \mathrm{Conv}\{0, e_1, ..., e_n\} = \{x \in \mathbb{R}^n : x \geq 0, \sum_{i=1}^n x_i \leq 1\}$

$\Rightarrow \mathcal{I} = \{1, ..., n+1\}$ with $a_i^T x := -x_1 \leq 0 =: b_i$, $1 \leq i \leq n$, and $a_{n+1}^T x := \sum_i x_i \leq 1 =: b_{n+1}$

Every subset $I \subset \mathcal{I}$ *different from* $\mathcal{I}$ defines a face. For example, $I = \{1, n+1\}$ defines the face

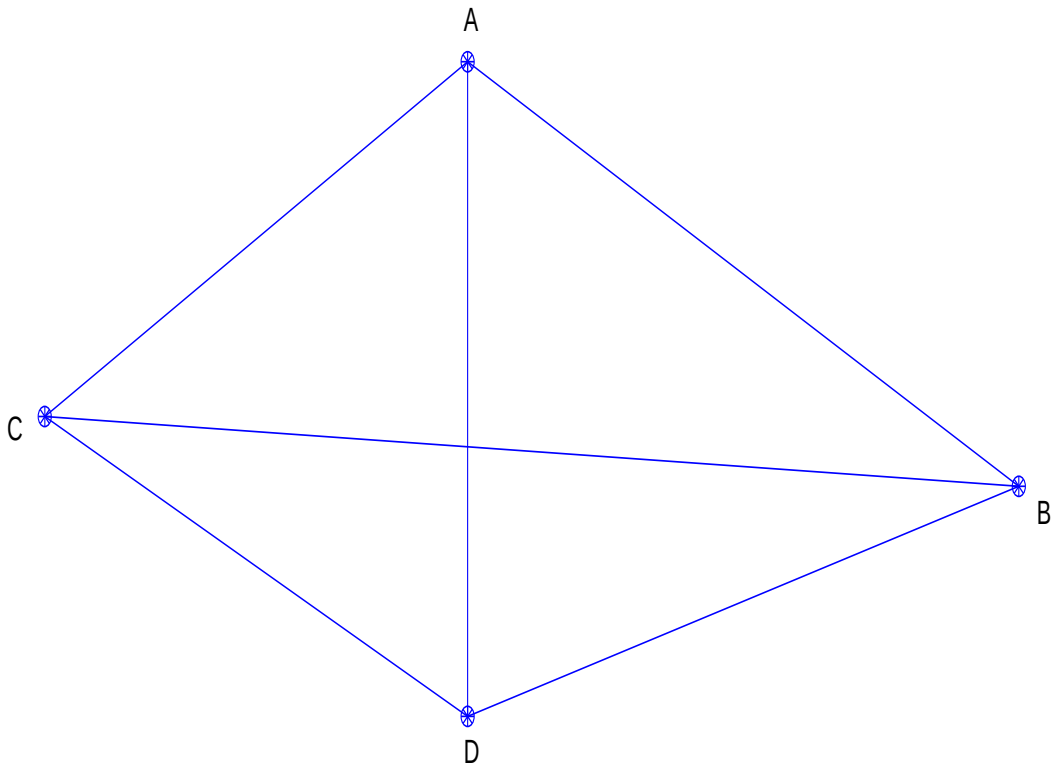
$$\{x \in \mathbb{R}^n : x_1 = 0, x_i \geq 0, \sum_i x_i = 1\}.$$

Quiz: What are the faces of 3D simplex Δ_3 – the convex hull of A,B,C,D ?



- 3-dimensional faces are
- 2-dimensional faces are
- 1-dimensional faces are
- 0-dimensional faces are

Quiz: What are the faces of 3D simplex Δ_3 – the convex hull of A,B,C,D ?



- 3-dimensional faces are the entire simplex
- 2-dimensional faces are 4 triangles ΔABC, ΔBCD, ΔCDA, ΔDAB
- 1-dimensional faces are 6 segments $[A, B]$, $[A, C]$, $[A, D]$, $[B, C]$, $[B, D]$, $[C, D]$
- 0-dimensional faces are 4 points $\{A\}$, $\{B\}$, $\{C\}$, $\{D\}$

Quiz: How many faces has

- 1D box (segment) $\{x \in \mathbb{R} : 0 \leq x \leq 1\}$?
- 2D box (square) $\{x \in \mathbb{R}^2 : 0 \leq x_1, x_2 \leq 1\}$?
- 3D box (cube) $\{x \in \mathbb{R}^3 : 0 \leq x_1, x_2, x_3 \leq 1\}$?
- n-dimensional box $\{x \in \mathbb{R}^n : 0 \leq x_1, ..., x_n \leq 1\}$?

Quiz: How many faces has n-dimensional box

$$\{x \in \mathbb{R}^n : 0 \leq x_1, ..., x_n \leq 1\} \ ?$$

Answer: 3^n.

Explanation: We can list all faces as follows:

— select $k \in \{0, 1, ..., n\}$ pairs $0 \leq x_i \leq 1$ of inequalities defining the box ($\begin{pmatrix} n \\ k \end{pmatrix}$ options)

— make one of the inequalities in every one of the selected pairs equality (2^k options).

$\Rightarrow$ The total number of faces is

$$\Sigma_{k=0}^{n} \begin{pmatrix} n \\ k \end{pmatrix} 2^k = (1+2)^n.$$

$$X = \left\{ x \in \mathbb{R}^n : a_i^T x \leq b_i, \ \ i \in \mathcal{I} = \{1, ..., m\} \right\}$$

Facts:

- *A face*

$$\emptyset \neq X_I = \{ x \in \mathbb{R}^n : a_i^T x \leq b_i, i \in \mathcal{I}, a_i^T x = b_i, i \in I \}$$

of X is a nonempty polyhedral set

- *A face of a face of X can be represented as a face of X*
- *if X_I and $X_{I'}$ are faces of X and their intersection is nonempty, this intersection is a face of X:*

$$\emptyset \neq X_I \cap X_{I'} \Rightarrow X_I \cap X_{I'} = X_{I \cup I'}.$$

♣ A face X_I is called *proper*, if $X_I \neq X$.

Fact: *A face X_I of X is proper if and only if*

$$\dim X_I < \dim X \qquad .$$

Proof:

One direction is evident. Now assume that X_I is a proper face, and let us prove that $\dim X_I < \dim X$.

• Since $X_I \neq X$, there exists $i_* \in I$ such that $a_{i_*}^T x \not\equiv b_{i_*}$ on X and thus on $M = \mathrm{Aff}(X)$.

$\Rightarrow$ The set $M_+ = \{x \in M : a_{i_*}^T x = b_{i_*}\}$ contains X_I (and thus is an affine subspace containing $\mathrm{Aff}(X_I)$), and is $\subsetneq M$.

$\Rightarrow$ $\mathrm{Aff}(X_I) \subsetneq M$, whence

$$\dim X_I = \dim \mathrm{Aff}(X_I) < \dim M = \dim X. \qquad \square$$

Extreme Points of a Polyhedral Set

$$X = \left\{ x \in \mathbb{R}^n : a_i^T x \leq b_i,\ i \in \mathcal{I} = \{1, ..., m\} \right\}$$

Definition. *A point $v \in X$ is called an extreme point, or a vertex of X, if it can be represented as a face of X:*

$$\begin{array}{l} \exists I \subset \mathcal{I}: \\ X_I := \{x \in \mathbb{R}^n : a_i^T x \leq b_i,\ i \in \mathcal{I}, a_i^T x = b_i,\ i \in I\} = \{v\}. \end{array} \qquad (*)$$

Geometric characterization of extreme points: *A point $v \in X$ is a vertex of X iff v is not the midpoint of a nontrivial segment contained in X:*

$$v \pm h \in X \Rightarrow h = 0. \qquad (!)$$

Proof:

$$v \pm h \in X \Rightarrow h = 0. \qquad (!)$$

$$\exists I \subset \mathcal{I}: \quad X_I := \{x \in \mathbb{R}^n : a_i^T x \le b_i,\ i \in \mathcal{I}, a_i^T x = b_i,\ i \in I\} = \{v\}. \qquad (*)$$

• Let v be a vertex, so that $(*)$ takes place for certain I, and let h be such that $v \pm h \in X$; we should prove that $h = 0$. We have $\forall i \in I$:

$$\{b_i \ge a_i^T(v-h) = b_i - a_i^T h \ \&\ b_i \ge a_i^T(v+h) = b_i + a_i^T h\}$$
$$\Rightarrow a_i^T h = 0 \Rightarrow a_i^T[v \pm h] = b_i.$$

Thus, $v \pm h \in X_I = \{v\}$, whence $h = 0$. We have proved that $(*)$ implies $(!)$.

• Let us prove that $(!)$ implies $(*)$. Indeed, let $v \in X$ be such that $(!)$ takes place; we should prove that $(*)$ holds true for certain I. Let

$$I = \{i \in \mathcal{I} : a_i^T v = b_i\},$$

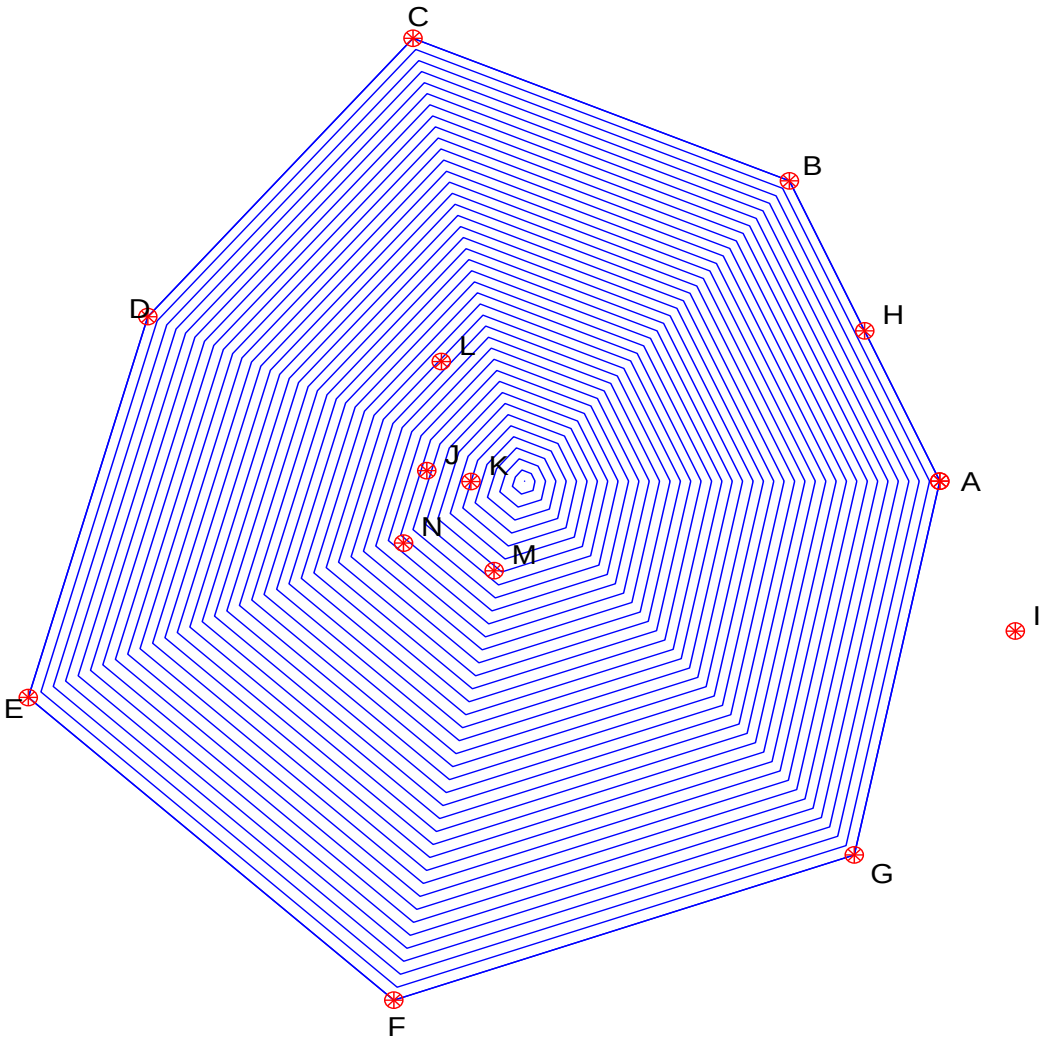
so that $v \in X_I$. It suffices to prove that $X_I = \{v\}$. Let, on the opposite, $\exists 0 \ne e : v + e \in X_I$. Then $a_i^T(v+e) = b_i = a_i^T v$ for all $i \in I$, that is, $a_i^T e = 0\, \forall i \in I$, that is,

$$a_i^T(v \pm te) = b_i\ \forall (i \in I, t > 0).$$

When $i \in \mathcal{I} \backslash I$, we have $a_i^T v < b_i$ and thus $a_i^T(v \pm te) \le b_i$ provided $t > 0$ is small enough.

$\Rightarrow$ *There exists* $\bar{t} > 0$*:* $a_i^T(v \pm \bar{t}e) \le b_i\, \forall i \in \mathcal{I}$*, that is* $v \pm \bar{t}e \in X$, which is a desired contradiction. $\square$

Quiz: Who is who? Which of the depicted points are extreme points of the blue polyhedral set?



Algebraic characterization of extreme points: *A point* $v \in X = \{x \in \mathbb{R}^n : a_i^T x \leq b_i, i \in \mathcal{I} = \{1, ..., m\}\}$ *is a vertex of* X *iff among the inequalities* $a_i^T x \leq b_i$ *which are active at* v (i.e., $a_i^T v = b_i$) *there are* n *with linearly independent* a_i*:*

$$\text{Rank}\{a_i : i \in I_v\} = n, \; I_v = \{i \in \mathcal{I} : a_i^T v = b_i\} \qquad (!)$$

Proof:

$$v \in X = \{x \in \mathbb{R}^n : a_i u^T x \le b_i, i \in \mathcal{I}\}$$
$$I_v = \{i \in \mathcal{I} : a_i^T v = b_i\} \qquad (!)$$

We should prove that $v \in X$ is an extreme point of X iff among the vectors a_i, $i \in I_v$, there are n linearly independent.

• Let v be a vertex of X; we should prove that among the vectors a_i, $i \in I_v$ there are n linearly independent. Assuming that this is not the case, the linear system $a_i^T e = 0$, $i \in I_v$ in variables e has a *nonzero* solution e. We have

$$i \in I_v \Rightarrow a_i^T[v \pm te] = b_i \, \forall t,$$
$$i \in \mathcal{I} \backslash I_v \Rightarrow a_i^T v < b_i$$
$$\Rightarrow a_i^T[v \pm te] \le b_i \text{ for all small enough } t > 0$$

whence $\exists \bar{t} > 0 : a_i^T[v \pm \bar{t}e] \le b_i \, \forall i \in \mathcal{I}$, that is $v \pm \bar{t}e \in X$, which is impossible due to $\bar{t}e \neq 0$. □

• Now assume that among the vectors a_i, $i \in I_v$, there are n linearly independent. We should prove that v is a vertex of X, that is, that the relation $v \pm h \in X$ implies $h = 0$. Indeed, when $v \pm h \in X$, we should have

$$\forall i \in I_v : b_i \ge a_i^T[v_i \pm h] = b_i \pm a_i^T h$$
$$\Rightarrow a_i^T h = 0 \; \forall i \in I_v.$$

Thus, $h \in \mathbb{R}^n$ is orthogonal to n linearly independent vectors from $\mathbb{R}^n$, whence $h = 0$. □

♠ **Fact:** *The set* $\mathrm{Ext}(X)$ *of extreme points of a polyhedral set is finite.*

Indeed, there could be no more extreme points than faces.

♠ **Observation:** *If* X *is a polyhedral set and* X_I *is a face of* X*, then* $\mathrm{Ext}(X_I) \subset \mathrm{Ext}(X)$*.*

Indeed, extreme points are singleton faces, and a face of a face of X can be represented as a face of X itself.

♡ **Note:** Geometric characterization of extreme points allows to define this notion for every convex set X: *A point $x \in X$ is called extreme, if $x \pm h \in X \Rightarrow h = 0$*

♡ **Fact:** *Let X be convex and $x \in X$. Then $x \in \mathrm{Ext}(X)$ iff in every representation*

$$x = \sum_{i=1}^{m} \lambda_i x_i$$

of x as a convex combination of points $x_i \in X$ with positive coefficients one has

$$x_1 = x_2 = ... = x_m = x$$

♡ **Fact:** *For a convex set X and $x \in X$, $x \in \mathrm{Ext}(X)$ iff the set $X \backslash \{x\}$ is convex.*

♡ **Fact:** *For every $X \subset \mathbb{R}^n$*

$$\mathrm{Ext}(\mathrm{Conv}(X)) \subset X$$

Quiz: What are extreme points of the set

$$\Delta^{=}_{n,k} = \{x \in \mathbb{R}^n : 0 \le x_i \le 1, \sum_i x_i = k\}$$

$$[k \in \mathbf{N}, 0 \le k \le n]$$?

Quiz: What are extreme points of the set

$$\Delta^{=}_{n,k} = \{x \in \mathbb{R}^n : 0 \le x_i \le 1, \sum_i x_i = k\}$$

$$[k \in \mathbf{N}, 0 \le k \le n] \qquad ?$$

Description: These are exactly Boolean (i.e., with entries 0 and 1) vectors from $\Delta_{n,k}$, i.e., Boolean n-dimensional vectors with exactly k entries equal to 1. In particular, the extreme points of the standard "flat" simplex $\Delta^{=}_{n,1} = \{x \in \mathbb{R}^n : x \ge 0, , \sum_i x_i = 1\}$ are exactly the n standard basic orths.

Indeed,

• If x is a Boolean vector from $\Delta^{=}_{n,k}$, then the set of active at x bounds $0 \le x_i \le 1$ is of cardinality n, and the corresponding vectors of coefficients are linearly independent

$\Rightarrow x \in \mathsf{Ext}(\Delta^{=}_{n,k})$

• If $x \in \mathsf{Ext}(\Delta^{=}_{n,k})$, then among the active at x constraints defining $\Delta_{n,k}$ there should be n linearly independent. One of these active constraints is the equality

$$\sum_i x_i = k \qquad ,$$

and the remaining $n-1$ should be among the bounds $0 \le x_i \le 1$, implying that $n-1$ entries in x are Boolean. Since the sum of all entries (k) is integer, the remaining entry also is integer, and since it is in $[0,1]$, it is Boolean as well.

Quiz: What are extreme points of the set

$$\Delta_{n,k} = \{x \in \mathbb{R}^n : 0 \le x_i \le 1, \sum_i x_i \le k\}$$

$$[k \in \mathbf{N}, 0 \le k \le n] \qquad ?$$

Quiz: What are extreme points of the set

$$\Delta_{n,k} = \{x \in \mathbb{R}^n : 0 \le x_i \le 1, \sum_i x_i \le k\}$$

$$[k \in \mathbf{N}, 0 \le k \le n] \qquad ?$$

Description: These are exactly Boolean (i.e., with entries 0 and 1) vectors from $\Delta_{n,k}$, i.e., Boolean n-dimensional vectors with at most k entries equal to 1. In particular,

— the extreme points of $\Delta_{n,n} = \{x \in \mathbb{R}^n : 0 \le x_i \le 1 \, \forall i\}$ (unit box) are all 2^n n-dimensional Boolean vectors;

— the extreme points of the "full-dimensional" simplex $\Delta_{n,1} = \{x \in \mathbb{R}^n : x \ge 0, \sum_i x_i \le 1\}$ are the n basic orths and the origin.

Indeed,

• If x is a Boolean vector from $\Delta_{n,k}$, then the set of active at x bounds $0 \le x_i \le 1$ is of cardinality n, and the corresponding vectors of coefficients are linearly independent

$\Rightarrow x \in \mathrm{Ext}(\Delta_{n,k})$

• If $x \in \mathrm{Ext}(\Delta_{n,k})$, then among the active at x constraints defining $\Delta_{n,k}$ there should be n linearly independent. There are two options:

— all n active constraints are among the bounds $0 \le x_i \le 1 \Rightarrow x$ is Boolean

— one of the active constraints is "sum of all entries is $\le k$" (that is, $\sum_i x_i = k$), and remaining $n-1$ are among the bounds $0 \le x_i \le 1$. We have seen that in this case x is Boolean.

Example: *An $n \times n$ matrix A is called double stochastic, if the entries are nonnegative and all the row and the column sums are equal to 1. The set of double-stochastic matrices is a polyhedral set in $\mathbb{R}^{n \times n}$:*

$$\Pi_n = \left\{ x = [x_{ij}]_{i,j} \in \mathbb{R}^{n \times n} : \begin{array}{l} x_{ij} \geq 0 \, \forall i, j \\ \sum_{i=1}^{n} x_{ij} = 1 \, \forall j \\ \sum_{j=1}^{n} x_{ij} = 1 \, \forall i \end{array} \right\}$$

What are the extreme points of Π_n ?

Quiz: Which of the matrices below are doubly stochastic?

$$\begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad \begin{bmatrix} 1/3 & 1/3 & 1/3 \\ 1/3 & 1/3 & 1/3 \\ 1/3 & 1/3 & 1/3 \end{bmatrix} \quad \begin{bmatrix} 2/3 & -1/3 & 2/3 \\ -1/3 & 2/3 & 2/3 \\ 2/3 & 2/3 & -1/3 \end{bmatrix}$$

An $n \times n$ matrix A is called double stochastic, if the entries are nonnegative and all the row and the column sums are equal to 1. The set of double-stochastic matrices is a polyhedral set in $\mathbb{R}^{n \times n}$:

$$\Pi_n = \left\{ x = [x_{ij}]_{i,j} \in \mathbb{R}^{n \times n} : \begin{array}{l} x_{ij} \geq 0 \, \forall i, j \\ \sum_{i=1}^{n} x_{ij} = 1 \, \forall j \\ \sum_{j=1}^{n} x_{ij} = 1 \, \forall i \end{array} \right\}$$

What are the extreme points of Π_n ?

♠ **Birkhoff's Theorem** *The vertices of Π_n are exactly the $n \times n$ permutation matrices (exactly one nonzero entry, equal to 1, in every row and every column).*

Proof:

• A permutation matrix P can be viewed as 0/1 vector of dimension n^2 and as such is an extreme point of the box

$$\{[x_{ij}] : 0 \leq x_{ij} \leq 1\}$$

which contains Π_n. Therefore P is an extreme point of Π_n since by geometric characterization of extreme points, *an extreme point x of a convex set is an extreme point of every smaller convex set to which x belongs.*

Birkhoff's Theorem *The vertices of the set*

$$\Pi_n = \left\{ x = [x_{ij}]_{i,j} \in \mathbb{R}^{n\times n} : \begin{array}{l} x_{ij} \geq 0 \, \forall i,j \\ \sum_{i=1}^n x_{ij} = 1 \, \forall j \\ \sum_{j=1}^n x_{ij} = 1 \, \forall i \end{array} \right\}$$

of double-stochastic $n \times n$ matrices are exactly the $n \times n$ permutation matrices (exactly one nonzero entry, equal to 1, in every row and every column).

• We can drop in the description of Π_n (any) one of linear equations, since if *all but one* among the $2n$ row and column sums of an $n \times n$ matrix are equal to 1, *all* $2n$ row and column sums are equal to 1.

$\Rightarrow$ We lose nothing when assuming that Π_n is given by n^2 bounds $0 \leq x_{i,j}$ and $2n-1$ linear equations.

• Let P be an extreme point of Π_n; we want to prove that Π_n is a permutation matrix. By algebraic characterization of extreme points, *at least* $n^2 - (2n-1) = (n-1)^2 > (n-2)n$ *entries in P should be zeros.*

$\Rightarrow$ *P has a column with at least $n-1$ zero entries*

$\Rightarrow$ $\exists i_*, j_* : P_{i_* j_*} = 1$

$\Rightarrow$ *P belongs to the face $\{x \in \Pi_n : x_{i_*,j_*} = 1\}$ of Π_n* (which we get when converting the bounds $x_{i,j_*} \geq 0$, $i \neq i_*$, and $x_{i_*,j} \geq 0$, $j \neq j_*$, into equalities)

$\Rightarrow$ *Extreme point P of Π_n belongs to the face $\{P \in \Pi_n : P_{i_* j_*} = 1\}$ of Π_n and thus is an extreme point of the face*

$\Rightarrow$ *the matrix obtained from P by eliminating i_*-th row and j_*-th column is an extreme point in the set Π_{n-1}.* Iterating the reasoning, we conclude that P is a permutation matrix.

Recessive Directions and Recessive Cone

$$X = \{x \in \mathbb{R}^n : Ax \leq b\} \text{ is nonempty}$$

♣ **Definition.** A vector $d \in \mathbb{R}^n$ is called a *recessive direction of* X, if X contains a ray directed by d:

$$\exists \bar{x} \in X : \bar{x} + td \in X \ \forall t \geq 0.$$

♠ **Observation:** *d is a recessive direction of X iff $Ad \leq 0$.*

♠ **Corollary:** *Recessive directions of X form a polyhedral cone, namely, the cone $\mathrm{Rec}(X) = \{d : Ad \leq 0\}$, called the recessive cone of X.*
Whenever $x \in X$ and $d \in \mathrm{Rec}(X)$, one has $x + td \in X$ for all $t \geq 0$. In particular,

$$X + \mathrm{Rec}(X) = X.$$

♠ **Observation:** *The larger is a polyhedral set, the larger is its recessive cone:*

$$X \subset Y \text{ are polyhedral} \Rightarrow \mathrm{Rec}(X) \subset \mathrm{Rec}(Y).$$

Quiz: *what are the recessive cones of the following polyhedral sets X:*

- $X = \{x \in \mathbb{R}^2 : 0 \leq x_1, x_2 \leq 1\}$
- $X = \mathbb{R}^n_+ = \{x \in \mathbb{R}^n : x \geq 0\}$
- $X = \{x \in \mathbb{R}^n : x_1 = 0\}$
- $X = \left\{ x \in \mathbb{R}^3 : \begin{array}{r} -1 \leq x_1 + x_2 \leq 1, \\ -1 \leq x_2 + x_3 \leq 1, \\ -1 \leq x_3 + x_1 \leq 1 \end{array} \right\}$
- $X = \left\{ x \in \mathbb{R}^4 : \begin{array}{r} -1 \leq x_1 + x_2 \leq 1, \\ -1 \leq x_2 + x_3 \leq 1, \\ -1 \leq x_3 + x_4 \leq 1, \\ -1 \leq x_4 + x_1 \leq 1 \end{array} \right\}$

Quiz: *what are the recessive cones of the following polyhedral sets X:*

- $X = \{x \in \mathbb{R}^2 : 0 \le x_1, x_2 \le 1\}$ $\mathrm{Rec}(X) = \{0\}$
- $X = \mathbb{R}^n_+ = \{x \in \mathbb{R}^n : x \ge 0\}$ $\mathrm{Rec}(X) = R^+_n$
- $X = \{x \in \mathbb{R}^n : x_1 = 0\}$

$\mathrm{Rec}(X) = X$, *as for every linear subspace or cone!*

- $X = \left\{ x \in \mathbb{R}^3 : \begin{array}{r} -1 \le x_1 + x_2 \le 1, \\ -1 \le x_2 + x_3 \le 1, \\ -1 \le x_3 + x_1 \le 1 \end{array} \right\}$

$\mathrm{Rec}(X) = \{0\}$ (X *is bounded!*)

- $X = \left\{ x \in \mathbb{R}^4 : \begin{array}{r} -1 \le x_1 + x_2 \le 1, \\ -1 \le x_2 + x_3 \le 1, \\ -1 \le x_3 + x_4 \le 1, \\ -1 \le x_4 + x_1 \le 1 \end{array} \right\}$

$\mathrm{Rec}(X) = \mathbb{R} \cdot [1; -1; 1; -1]$

Recessive Subspace of a Polyhedral Set

$$X = \{x \in \mathbb{R}^n : Ax \leq b\} \text{ is nonempty}$$

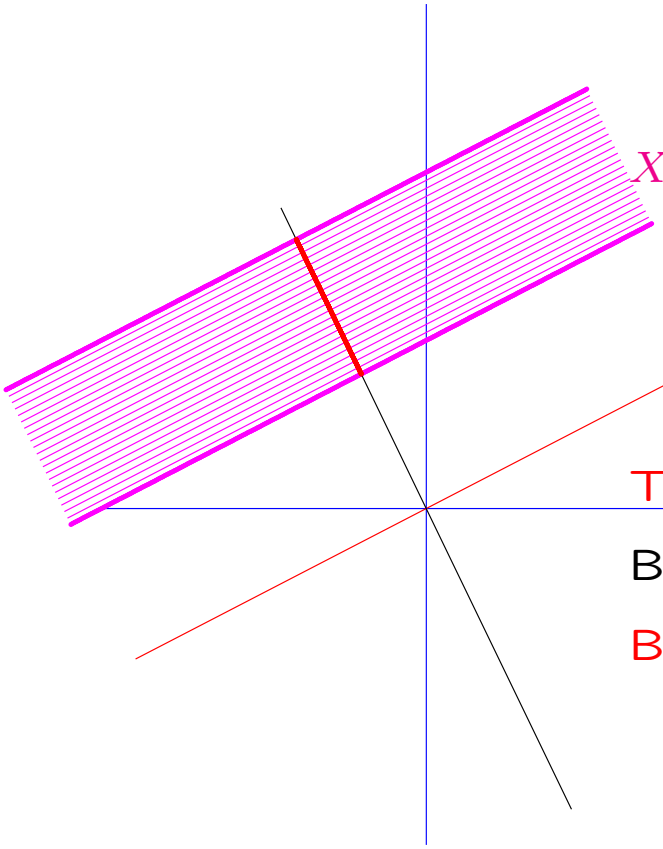
♠ **Observation:** *Directions d of lines contained in X are exactly the vectors from the recessive subspace*

$$L = \mathrm{Ker}A := \{d : Ad = 0\} = \mathrm{Rec}(X) \cap [-\mathrm{Rec}(X)]$$

of X, and $X = X + \mathrm{Ker}A$. In particular,

$$X = \bar{X} + \mathrm{Ker}A,$$
$$\bar{X} = X \cap [\mathrm{Ker}A]^\perp = \{x \in \mathbb{R}^n : Ax \leq b, x \in [\mathrm{Ker}A]^\perp\}$$

Note: *$\bar{X}$ is polyhedral and does not contain lines.*



$$X = \left\{x : \begin{array}{rrcr} x_1 & -2x_2 & \leq & -1 \\ -x_1 & +2x_2 & \leq & 2 \end{array}\right\}$$

Thin red line: $L = \mathrm{Ker}A = \mathbb{R} \cdot [2; 1]$

Black line: $L^\perp$

Bold red segment: $\bar{X} = X \cap L^\perp$

Pointed Polyhedral Cones & Extreme Rays

$$K = \{x : Ax \leq 0\}$$

♣ **Definition.** *Polyhedral cone K is called pointed, if it does not contain lines.* **Equivalently:** *K is pointed iff $K \cap \{-K\} = \{0\}$.*

Equivalently: *K is pointed iff $\mathrm{Ker}A = \{0\}$.*

♣ **Definition** *An extreme ray of K is a face of K which is a nontrivial ray (i.e., the set $\mathbb{R}_+ d = \{td : t \geq 0\}$ associated with a nonzero vector d, called a generator of the ray).*

♠ **Geometric characterization:** *A vector $d \in K$ is a generator of an extreme ray of K* (in short: d is an extreme direction of K) *iff d is nonzero and whenever d is a sum of two vectors from K, both vectors are nonnegative multiples of d:*

$$d = d_1 + d_2,\ d_1, d_2 \in K \Rightarrow \exists t_1 \geq 0, t_2 \geq 0 : d_1 = t_1 d, d_2 = t_2 d.$$

Example: $K = \mathbb{R}^n_+$. This cone is pointed, and its extreme directions are positive multiples of basic orths e_i. There are n extreme rays — nonnegative rays of coordinate axes $R_i = \mathbb{R}_+ \cdot e_i$, $1 \leq i \leq n$.

♠ **Observation:** d is an extreme direction of K iff some (and then – all) positive multiples of d are extreme directions of K.

$$\begin{aligned} K &= \{x \in \mathbb{R}^n : Ax \leq 0\} \\ &= \left\{x \in \mathbb{R}^n : a_i^T x \leq 0,\ i \in \mathcal{I} = \{1, ..., m\}\right\} \end{aligned}$$

♠ **Algebraic characterization of extreme directions:** *A vector $d \in K$ is an extreme direction of K iff d is nonzero and among the homogeneous inequalities $a_i^T x \leq 0$ which are active at d* (i.e., are satisfied at d as equalities) *there are $n - 1$ inequalities with linearly independent a_i's.*

Proof:

Let $0 \neq d \in K$, $I = \{i \in \mathcal{I} : a_i^T d = 0\}$.

• Let the set $\{a_i : i \in I\}$ contain $n-1$ linearly independent vectors, say, $a_1, ..., a_{n-1}$. Let us prove that then d is an extreme direction of K. Indeed, the set

$$L = \{x : a_i^T x = 0,\ 1 \leq i \leq n-1\} \supset K_I$$

is a one-dimensional linear subspace in $\mathbb{R}^n$. Since $0 \neq d \in L$, we have $L = \mathbb{R}d$. Since $d \in K$, the ray $\mathbb{R}_+ d$ is contained in K_I: $\mathbb{R}_+ d \subset K_I$. Since $K_I \subset L$, all vectors from K_I are real multiples of d. Since K is pointed, no negative multiples of d belong to K_I.

$\Rightarrow K_I = R_+ d$ and $d \neq 0$, i.e., K_I is an extreme ray of K, and d is a generator of this ray. $\square$

Proof (continued)

• Let d be an extreme direction of K, that is, $\mathbb{R}_+ d$ is a face of K, and let us prove that the set $\{a_i : i \in I\}$ contains $n-1$ linearly independent vectors.

Assuming the opposite, the solution set L of the homogeneous system of linear equations

$$a_i^T x = 0,\ i \in I$$

is of dimension ≥ 2 and thus contains a vector h which is *not* proportional to d. When $i \notin I$, we have $a_i^T d < 0$ and thus $a_i^T(d+th) \leq 0$ when $|t|$ is small enough.

$\Rightarrow \exists \bar{t} > 0 : |t| \leq \bar{t} \Rightarrow a_i^T[d+th] \leq 0\ \forall i \in \mathcal{I}$

$\Rightarrow$ *the face K_I of K* (which is the smallest face of K containing d)*contains two non-proportional nonzero vectors $d, d+\bar{t}h$, and thus is strictly larger than $\mathbb{R}_+ d$.*

$\Rightarrow \mathbb{R}_+ d$ is *not* a face of K, which is a desired contradiction. $\square$

What We Should Know After the Snow

♣ Consider a *nonempty* polyhedral set

$$X = \{x \in \mathbb{R}^n : a_i^T x \leq b_i,\ 1 \leq i \leq m\}$$

A. *Faces of X are nonempty sets which we can get by converting some of inequalities specifying X into equalities:*

$$X_I = \{x : a_i^T x = b_i,\ i \in I, a_i^T x \leq b_i, i \notin I\}$$

B. *Extreme points* (a.k.a. *vertices*) *of X are faces which are singletons*.

B.1. *Geometric characterization: A point $v \in X$ is a vertex iff it is not a midpoint of a nontrivial segment in X:*

$$v \pm h \in X \Rightarrow h = 0$$

B.2. *Algebraic characterization: A point $v \in X$ is a vertex iff among the constraints $a_i^T x \leq b_i$ which are active at v* (i.e., are satisfied at v as equalities) *there are n linearly independent* (i.e., with linearly independent a_i's).

$\Rightarrow$ *The set* $\mathrm{Ext}(X)$ *of extreme points of X is finite.*

• **Example:** When $k \leq n$ is a positive integer, the extreme points of the set

$$X = \{x \in \mathbb{R}^n : 0 \leq x_i \leq 1 \forall i, \textstyle\sum_i x_i = k\}$$

are exactly the 0/1 vectors from X (i.e., vectors with k entries equal to 1 and remaining entries equal to 0).

C. Fact: *Every nonempty and bounded polyhedral set X is the convex hull of a finite set.*

We shall see that *the finite set in question can be taken as the set* $\mathrm{Ext}(X)$ *of extreme points of X*.

♣ Consider a polyhedral *cone*

$$K = \{x \in \mathbb{R}^n : a_i^T x \leq 0,\ 1 \leq i \leq m\}$$

D. *Extreme rays of K are faces K_I of K which are one-dimensional rays:*

$$K_I = \{tr : t \geq 0\} \text{ with } r \neq 0.$$

Generators of extreme rays (a.k.a. *extreme directions* of K) *are nonzero vectors from the extreme rays.* All generators of a particular extreme ray K_I are positive multiples of each other, and for such a generator r, $K_I = \{tr : t \geq 0\}$.

D.1. *Geometric characterization: A direction $r \in K$ is extreme iff $r \neq 0$ and in any representation of r as the sum $r = d_1 + d_2$ of two vectors from K both terms d_1, d_2 are nonnegative multiples of r.*

D.2. *Algebraic characterization: A direction $r \in K$ is extreme iff $r \neq 0$ and among the constraints $a_i^T x \leq 0$ which are active at r there are $n-1$ linearly independent* (i.e., with linearly independent a_i's).

$\Rightarrow$ *The number of extreme rays of a polyhedral cone is finite.*

• **Example:** The extreme rays of $\mathbb{R}^n_+$ are the nonnegative rays of the coordinate axes. Extreme directions of $\mathbb{R}^n$ are vectors with one coordinate positive and the remaining coordinates equal to 0.

E. Fact: *Every polyhedral cone is the conic hull of a finite set:*

$$K = \text{Cone}\,\{r_1, ..., r_N\} = \{x = \textstyle\sum_i \lambda_i r_i : \lambda \geq 0\}$$

.

We shall see that *if K does not contain lines and is nontrivial* (i.e., $K \neq \{0\}$) *one can take, as r_i, generators of extreme rays of K.*

$$K = \{x \in \mathbb{R}^n : Ax \leq 0\}.$$

Observations:

- *If K possesses extreme rays, then K is nontrivial ($K \neq \{0\}$) and pointed ($K \cap [-K] = \{0\}$).*

In fact, the inverse is also true.

- *The set of extreme rays of K is finite.*

Indeed, there are no more extreme rays than faces.

Base of a Cone

$$K = \{x \in \mathbb{R}^n : Ax \leq 0\}.$$

♣ **Definition.** *A set B of the form*

$$B = \{x \in K : f^T x = 1\} \qquad (*)$$

is called a base of K, if it is nonempty and intersects with every (nontrivial) ray in K:

$$\forall 0 \neq d \in K \ \exists! t \geq 0 : td \in B.$$

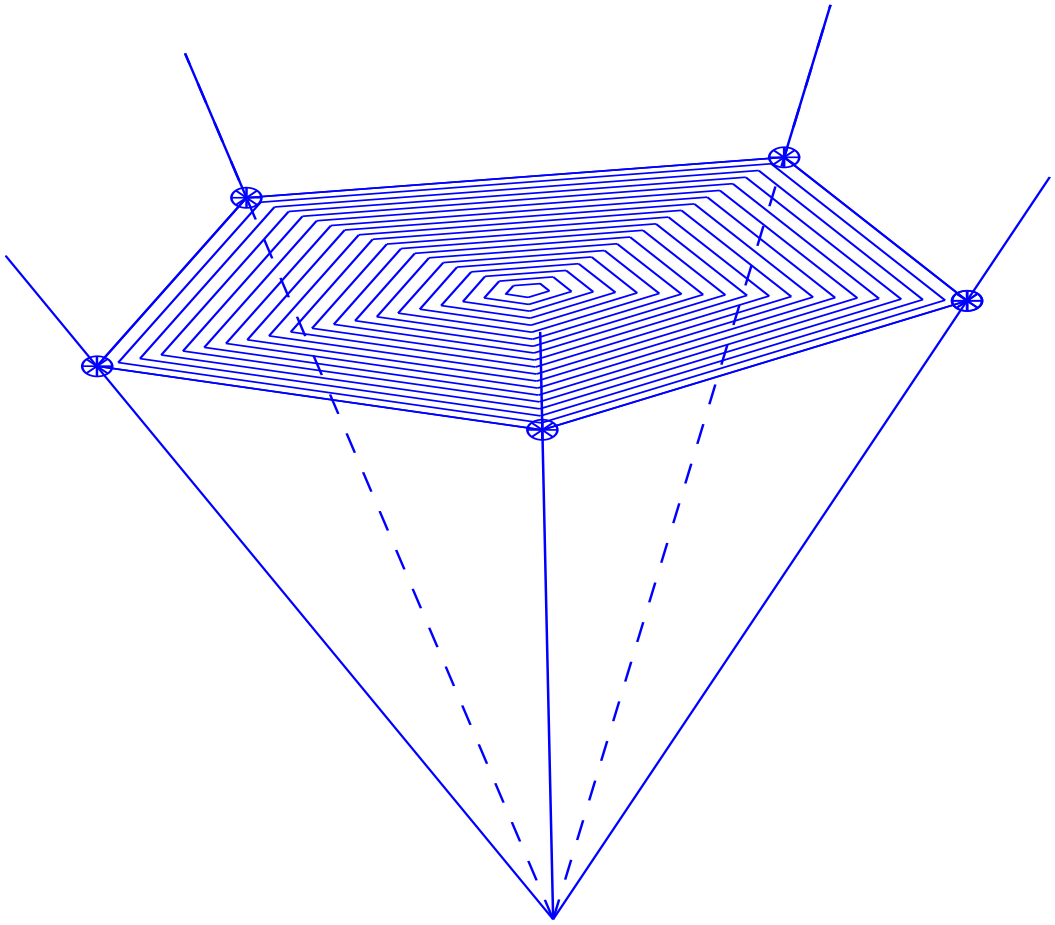
Example: The set

$$\Delta_n = \{x \in \mathbb{R}^n_+ : \sum_{i=1}^n x_i = 1\}$$

is a base of $\mathbb{R}^n_+$.

♠ **Observation:** *Set $(*)$ is a base of K iff $K \neq \{0\}$ and f makes strictly positive inner products with all nonzero vectors from K:*

$$0 \neq x \in K \Rightarrow f^T x > 0.$$



3D cone K and its base B (pentagon)

Note: *extreme rays of K are generated by extreme points of B*

♠ Facts:

- *K possesses a base iff $K \neq \{0\}$ and K is pointed.*
- *K possesses a base B iff K possesses extreme rays, and there is one-to-one correspondence between extreme rays of K and extreme points of B: extreme directions of K are exactly positive multiples of extreme points of B.*
- *The recessive cone of a base B of K is trivial:* $\mathrm{Rec}(B) = \{0\}$.

Towards the Main Theorem: First Step

Theorem. *Let*

$$X = \{x \in \mathbb{R}^n : Ax \leq b\}$$

be a nonempty polyhedral set which does not contain lines. Then

(i) *The set* $V = \mathrm{Ext}\{X\}$ *of extreme points of* X *is nonempty and finite:*

$$V = \{v_1, ..., v_N\}$$

(ii) *The set* R *of extreme rays of the recessive cone* $\mathrm{Rec}(X)$ *of* X *is finite:*

$$R = \{r_1, ..., r_M\}$$

(iii) *One has*

$$X = \mathrm{Conv}(V) + \mathrm{Cone}\,(R)$$
$$= \left\{ x = \sum_{i=1}^N \lambda_i v_i + \sum_{j=1}^M \mu_j r_j : \begin{array}{l} \lambda_i \geq 0 \, \forall i \\ \sum_{i=1}^N \lambda_i = 1 \\ \mu_j \geq 0 \, \forall j \end{array} \right\}$$

Main Lemma: *Let*

$$X = \{x \in \mathbb{R}^n : a_i^T x \leq b_i,\ 1 \leq i \leq m\}$$

be a nonempty polyhedral set which does not contain lines. Then the set $V = \mathrm{Ext}\{X\}$ *of extreme points of* X *is nonempty and finite, and*

$$X = \mathrm{Conv}(V) + \mathrm{Rec}(X).$$

Note: We already know that $\mathrm{Ext}(X)$ *is finite.*

Proof: Induction in $m = \dim X$.

Base $m = 0$ is evident: here

$$X = \{a\},\ V = \{a\},\ \mathrm{Rec}(X) = \{0\}.$$

Inductive step $m \Rightarrow m + 1$**:** Let the statement be true for polyhedral sets of dimension $\leq m$, and let $\dim X = m + 1$, $\mathcal{M} = \mathrm{Aff}(X)$, L be the linear subspace parallel to $\mathcal{M}$.

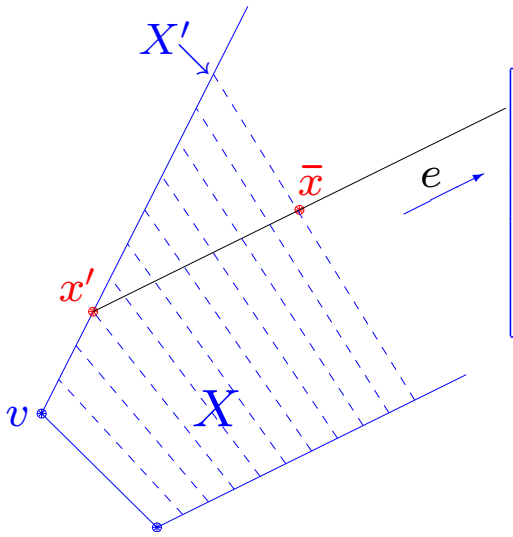
- Take a point $\bar{x} \in X$ and a nonzero direction $e \in L$ (it exists, since $\dim M = \dim L = m + 1 > 0$).
- Since X does not contain lines, either e, or $-e$, or both are *not* recessive directions of X. Swapping, if necessary, e and $-e$, assume that $-e \notin \mathrm{Rec}(X)$.

♠ **Case A: e, in contrast to $-e$, is a recessive direction of X.**

• Let us move from $\bar{x}$ along the direction $-e$.

— since $e \in L$, we all the time will stay in $\mathrm{Aff}(X)$

— since $-e$ is not a recessive direction of X, eventually we will be about to leave X. *When it happens, our position x' will belong to a proper face X' of X:*



$v \in \mathrm{Ext}(X') \subset \mathrm{Ext}(X)$
$x' \in v + \mathrm{Rec}(X') \subset v + \mathrm{Rec}(X)$
$\Rightarrow$ for some $\lambda > 0$
$\bar{x} = \underbrace{v}_{\in \mathrm{Ext}(X)} + \underbrace{[x' - v] + \lambda e}_{\in \mathrm{Rec}(X)}$

• Dimension of a proper face X' of X is less than $\dim X$

$\Rightarrow$ We can apply inductive hypothesis to X' and x' to conclude that $\mathrm{Ext}(X') \neq \emptyset$, whence $\mathrm{Ext}(X) \supset \mathrm{Ext}(X')$ also is nonempty. Besides this,

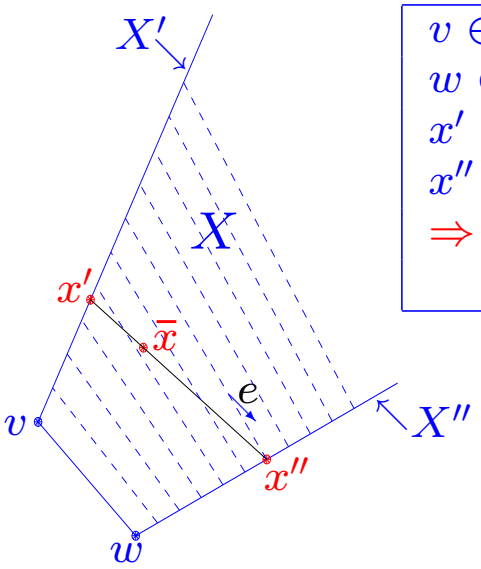
$$x' \in \mathrm{Conv}(\mathrm{Ext}(X')) + \mathrm{Rec}(X') \subset \mathrm{Conv}(\mathrm{Ext}(X)) + \mathrm{Rec}(X)$$

$\Rightarrow$ For some $\lambda \geq 0$,

$$\begin{array}{rcl} \bar{x} & = & x' + \overbrace{\lambda e}^{\in \mathrm{Rec}(X)} \\ & \in & [\mathrm{Conv}(\mathrm{Ext}(X)) + \mathrm{Rec}(X)] + \mathrm{Rec}(X) \\ & = & \mathrm{Conv}(\mathrm{Ext}(X)) + \mathrm{Rec}(X). \end{array}$$

♠ **Case B: e, same as $-e$, is not a recessive direction of X.**

- As in Case A, we move from $\bar{x}$ along the direction $-e$ until hitting a proper face X' of X at a point x'.
- *Since e is not a recessive direction of X,* when moving from $\bar{x}$ along the direction e, we eventually hit a proper face X'' of X at a point x''.



$$
\begin{aligned}
&v \in \mathrm{Ext}(X') \subset \mathrm{Ext}(X)\\
&w \in \mathrm{Ext}(X'') \subset \mathrm{Ext}(X)\\
&x' \in v + \mathrm{Rec}(X') \subset v + \mathrm{Rec}(X)\\
&x'' \in w + \mathrm{Rec}(X'') \subset w + \mathrm{Rec}(X)\\
&\Rightarrow \bar{x} \in \mathrm{Conv}\{x', x''\} \subset \mathrm{Conv}\{v, w\} + \mathrm{Rec}(X)\\
&\qquad\quad \subset \mathrm{Conv}(\mathrm{Ext}(X)) + \mathrm{Rec}(X)
\end{aligned}
$$

- Same as above, $\mathrm{Ext}(X) \subset \mathrm{Ext}(X') \neq \emptyset$,

$$x' \in \mathrm{Conv}(\mathrm{Ext}(X)) + \mathrm{Rec}(X)$$

and

$$x'' \in \mathrm{Conv}(\mathrm{Ext}(X)) + \mathrm{Rec}(X)$$

- Since $\bar{x}$ is a convex combination of x', x'' and $\mathrm{Conv}(\mathrm{Ext}(X)) + \mathrm{Rec}(X)$ is a convex set, we get

$$\bar{x} \in \mathrm{Conv}(\mathrm{Ext}(X)) + \mathrm{Rec}(X)$$

♠ **Summary:** We have proved that $\mathrm{Ext}(X)$ is nonempty and finite, and that every point $\bar{x} \in X$ belongs to

$$\mathrm{Conv}(\mathrm{Ext}(X)) + \mathrm{Rec}(X),$$

that is,

$$X \subset \mathrm{Conv}(\mathrm{Ext}(X)) + \mathrm{Rec}(X).$$

Since $\mathrm{Conv}(\mathrm{Ext}(X)) \subset X$ and $X + \mathrm{Rec}(X) = X$, we have also

$$X \supset \mathrm{Conv}(\mathrm{Ext}(X)) + \mathrm{Rec}(X)$$

$$\Rightarrow X = \mathrm{Conv}(\mathrm{Ext}(X)) + \mathrm{Rec}(X).$$

Induction is complete. □

Important observation: Our reasoning is *constructive:* it gives rise to an *algorithm* which, given a description $X = \{x : Ax \leq b\}$ of a polyhedral set, not containing lines, in $\mathbb{R}^n$ and a point $x \in X$, builds a representation of x as a convex combination of extreme points of X plus a recessive direction of X. "As it is" the algorithm induced by the reasoning is *not* efficient: the number of arithmetic operations needed to find a desired representation is, in general, *not polynomial* in the sizes m, n of A. The algorithm, however, can be converted to an efficient algorithm, where the number of a.o. is polynomial in m, n.

Corollaries of Main Lemma:

A. *Let X be a nonempty polyhedral set which does not contain lines. If X is has a trivial recessive cone, then*

$$X = \mathrm{Conv}(\mathrm{Ext}(X)).$$

B. *If K is a nontrivial pointed polyhedral cone, then the set of extreme rays of K is nonempty and finite, and if $r_1, ..., r_M$ are generators of the extreme rays of K, then*

$$K = \mathrm{Cone}\,\{r_1, ..., r_M\}.$$

Proof of B: Let B be a base of K, so that B is a nonempty polyhedral set with $\mathrm{Rec}(B) = \{0\}$. By **A**, $\mathrm{Ext}(B)$ is nonempty, finite and $B = \mathrm{Conv}(\mathrm{Ext}(B))$, whence $K = \mathrm{Cone}\,(\mathrm{Ext}(B))$ (since every nontrivial ray in K intersects B). It remains to note that a ray in K is extreme iff its intersection with B is an extreme point of B.

♠ Augmenting Main Lemma with Corollary **B**, we get the Theorem.

♣ We have seen that if X is a nonempty polyhedral set not containing lines, then X admits a representation

$$X = \mathrm{Conv}(V) + \mathrm{Cone}\,\{R\} \qquad (*)$$

where

• $V = V_*$ is the nonempty finite set of all extreme points of X;

• $R = R_*$ is a finite set comprised of generators of the extreme rays of $\mathrm{Rec}(X)$ (this set can be empty).

♠ It is easily seen that this representation is "minimal:" *Whenever X is represented in the form of $(*)$ with finite sets V, R,*

— V contains all extreme points of X

— R contains generators of all extreme rays of $\mathrm{Rec}(X)$.

Structure of a Polyhedral Set

Main Theorem (i) *Every nonempty polyhedral set $X \subset \mathbb{R}^n$ can be represented as*

$$X = \mathrm{Conv}(V) + \mathrm{Cone}\,(R) \qquad (*)$$

where $V \subset \mathbb{R}^n$ is a nonempty finite set, and $R \subset \mathbb{R}^n$ is a finite set.

(ii) *Vice versa, if a set X given by representation $(*)$ with a nonempty finite set V and finite set R, X is a nonempty polyhedral set.*

Proof. (i): We know that (i) holds true when X does not contain lines. We know also that *every* nonempty polyhedral set X can be represented as

$$X = \widehat{X} + L,\ L = \mathrm{Lin}\{f_1, ..., f_k\},$$

where $\widehat{X}$ is a nonempty polyhedral set which does not contain lines. In particular,

$$\begin{aligned} \widehat{X} &= \mathrm{Conv}\{v_1, ..., v_N\} + \mathrm{Cone}\,\{r_1, ..., r_M\} \\ \Rightarrow X &= \mathrm{Conv}\{v_1, ..., v_N\} \\ &\quad + \mathrm{Cone}\,\{r_1, ..., r_M, f_1, -f_1, ..., f_K, -f_K\} \end{aligned}$$

Note: *In every representation $(*)$ of X,* $\mathrm{Cone}\,(R) = \mathrm{Rec}(X)$.

(ii): Let

$$X = \mathrm{Conv}\{v_1, ..., v_N\} + \mathrm{Cone}\,\{r_1, .., r_M\} \subset \mathbb{R}^n \quad [N \geq 1, M \geq 0] \qquad (*)$$

To prove that X is a polyhedral set, note that $(*)$ induces a polyhedral representation of X:

$$X = \{x : \exists \lambda, \mu : \left\{ \begin{array}{l} x = \sum_i \lambda_i v_i + \sum_j \mu_j r_j \\ \lambda \geq 0, \sum_i \lambda_i = 1 \\ \mu \geq 0 \end{array} \right. \}$$

and every polyhedrally representable set is polyhedral.

□

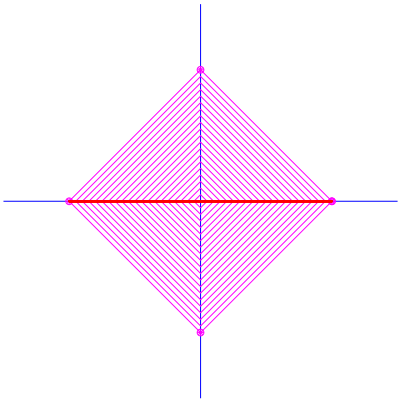
Quiz: Let X be a nonempty polyhedral set, and Y be its image under affine mapping $x \mapsto y = Ax + b$.
What are the relations between extreme points of X and Y ?

- *Is it always true that the image $y = Ax + b$ of an extreme point x of X is an extreme point of Y ?*
- *Is it always true that if y is an extreme point of Y and X does not contain lines, then $y = Ax + b$ for some extreme point x of X ?*
- *Is it always true that if y is an extreme point of Y, then $y = Ax + b$ for some $x \in \mathsf{Ext}(X)$?*

Quiz: Let X be a nonempty polyhedral set, and Y be its image under affine mapping $x \mapsto y = Ax + b$.

• *Is it always true that the image $y = Ax + b$ of $x \in \mathrm{Ext}(X)$ is an extreme point of Y?*

No!



• *Is it always true that if y is an extreme point of Y and X does not contain lines, then $y = Ax + b$ for some $x \in \mathrm{Ext}(X)$?*

Yes! Indeed, X does not contain lines

$\Rightarrow$ $X = \mathrm{Conv}\{v_1, ..., v_N\} + \mathrm{Cone}\{r_1, ..., r_N\}$ with $v_1, ..., v_N$ being extreme points of X

$\Rightarrow$ $Y = \{\mathrm{Conv}\{Av_1 + b, ..., Av_N + b\} + \mathrm{Cone}\{Ar_1, ..., Ar_M\}$

— if Y contains lines, it has no extreme points, and the claim is true by trivial reasons.

— if Y does not contain lines, then $Y = \mathrm{Conv}(\mathrm{Ext}(Y)) + \mathrm{Rec}(Y)$, and in this representation, $\mathrm{Ext}(Y)$ is the smallest finite set V such that $Y = \mathrm{Conv}(V) + \mathrm{Rec}(Y)$

$\Rightarrow$ $\{Av_1 + b, ..., Av_N + b\}$ *contains* $\mathrm{Ext}(Y)$, as claimed.

• *Is it always true that if y is an extreme point of Y, then $y = Ax + b$ for some extreme point x of X?*

No! The 2D vertical strip $X = \{[x_1 : x_2] : -1 \le x_1 \le 1\}$ contains lines and thus has no extreme points. However, the projection Y of X onto the x_1-axis is the segment $[-1, 1]$ which has extreme points.

Immediate Corollaries

Corollary I. *A nonempty polyhedral set X possesses extreme points iff X does not contain lines. In addition, the set of extreme points of X is finite.*

Indeed, if X does not contain lines, X has extreme points and their number is finite by Main Lemma. When X contains lines, *every* point of X belongs to a line contained in X, and thus X has no extreme points.

Corollary II. (i) *A nonempty polyhedral set X is bounded iff its recessive cone is trivial:* $\mathrm{Rec}(X) = \{0\}$, *and in this case X is the convex hull of the (nonempty and finite) set of its extreme points:*

$$\emptyset \neq \mathrm{Ext}(X) \text{ is finite and } X = \mathrm{Conv}(\mathrm{Ext}(X)).$$

(ii) *The convex hull of a nonempty finite set V is a bounded polyhedral set, and* $\mathrm{Ext}(\mathrm{Conv}(X)) \subset V$.

Corollary III. (i) *A cone K is polyhedral iff it is the conic hull of a finite set:*

$$K = \{x \in \mathbb{R}^n : Bx \leq 0\}$$
$$\Leftrightarrow \exists R = \{r_1, ..., r_M\} \subset \mathbb{R}^n : K = \mathrm{Cone}\,(R)$$

Note: this we already knew.

(ii) *When K is a nontrivial and pointed polyhedral cone, one can take as R the set of generators of the extreme rays of K.*

Proof of Corollary II:

(i): If $\mathrm{Rec}(X) = \{0\}$, then X does not contain lines and therefore $\emptyset \neq \mathrm{Ext}(X)$ is finite and

$$\begin{aligned} X &= \mathrm{Conv}(\mathrm{Ext}(X)) + \mathrm{Rec}(X) \\ &= \mathrm{Conv}(\mathrm{Ext}(X)) + \{0\} \qquad (*) \\ &= \mathrm{Conv}(\mathrm{Ext}(X)), \end{aligned}$$

and thus X is bounded as the convex hull of a finite set.

Vice versa, if X is bounded, then X clearly does not contain nontrivial rays and thus $\mathrm{Rec}(X) = \{0\}$.

(ii): By Main Theorem (ii),

$$X := \mathrm{Conv}(\{v_1, ..., v_m\})$$

is a polyhedral set, and this set clearly is bounded. Besides this, $X = \mathrm{Conv}(V)$ always implies that $\mathrm{Ext}(X) \subset V$. $\square$

Application examples:

- *Every vector x from the set*

$$\{x \in \mathbb{R}^n : 0 \le x_i \le 1,\ 1 \le i \le n, \sum_{i=1}^{n} x_i \le k\}$$

(k is an integer) is a convex combination of Boolean vectors from this set.

- *Every double-stochastic matrix is a convex combination of permutation matrices.*

Indeed, both sets clearly are bounded $\Rightarrow$ they are convex hulls of their extreme points.
Besides, we know that the extreme points of the first set are exactly Boolean vectors from this set, and the extreme points of the second set are exactly $n \times n$ permutation matrices.

Applications in LO

♣ **Theorem.** *Consider a LO program*

$$\mathrm{Opt} = \max_x \left\{ c^T x : Ax \le b \right\},$$

and let the feasible set $X = \{x : Ax \le b\}$ *be nonempty and thus representable as*

$$X = \mathrm{Conv}\{v_1, ..., v_N\} + \mathrm{Cone}\{r_1, ..., r_M\} \qquad (*)$$
$$[N \ge 1, M \ge 0]$$

Then

(i) *The program is solvable iff* c *has nonpositive inner products with all* r_j, $1 \le j \le M$.

(ii) *If* X *does not contain lines and the program is bounded, then among its optimal solutions there are extreme points of* X.

Indeed, by $(*)$ we have

$$\mathrm{Opt} = \sup_{\lambda,\mu} \left\{ \sum_i \lambda_i c^T v_i + \sum_j \mu_j c^T r_j : \begin{array}{l} \lambda_i \ge 0 \\ \sum_i \lambda_i = 1 \\ \mu_j \ge 0 \end{array} \right\}$$

$\Rightarrow$ Opt *is finite iff* $c^T r_j \le 0$ *for all* j, *in which case*

$$\mathrm{Opt} = \max_i c^T v_i,$$

i.e., the best (with the largest $c^T v_i$*) of the points* $v_1, ..., v_N$ *is an optimal solution.*

It remains to note that when X does not contain lines, we can set $\{v_i\}_{i=1}^N = \mathrm{Ext}(X)$.

Application to Knapsack problem. *A knapsack can store k items. You have $n \geq k$ items, j-th of value $c_j \geq 0$. How to select items to be placed into the knapsack in order to get the most valuable selection?*

Solution: Assuming for a moment that we can put to the knapsack fractions of items, let x_j be the fraction of item j we put to the knapsack. The most valuable selection then is given by an optimal solution to the LO program

$$\max_x \left\{ \sum_j c_j x_j : 0 \leq x_j \leq 1, \sum_j x_j \leq k \right\}$$

The feasible set is nonempty, polyhedral and bounded, and all extreme points are Boolean vectors from this set

$\Rightarrow$ There is a Boolean optimal solution.

In fact, the optimal solution is evident: we should put to the knapsack k most valuable of the items.

Application to Assignment problem. *There are n jobs and n workers. Every job takes one man-hour. The profit of assigning worker i with job j is c_{ij}. How to assign workers with jobs in such a way that every worker gets exactly one job, every job is carried out by exactly one worker, and the total profit of the assignment is as large as possible?*

Solution: Assuming for a moment that a worker can distribute his time between several jobs and denoting x_{ij} the fraction of activity of worker i spent on job j, we get a *relaxed* problem

$$\max_x \left\{ \sum_{i,j} c_{ij} x_{ij} : x_{ij} \geq 0, \sum_j x_{ij} = 1 \, \forall i, \sum_i x_{ij} = 1 \, \forall j \right\}$$

The feasible set is polyhedral, nonempty and bounded
$\Rightarrow$ Program is solvable, and among the optimal solutions there are extreme points of the set of double stochastic matrices, i.e., permutation matrices
$\Rightarrow$ Relaxation is exact!

Systems of Linear Inequalities and Duality

♣ We still do not know how to answer some most basic questions about polyhedral sets, e.g.:

♠ *How to recognize that a polyhedral set*

$$X = \{x \in \mathbb{R}^n : Ax \leq b\}$$

is/is not empty?

♠ *How to recognize that a polyhedral set*

$$X = \{x \in \mathbb{R}^n : Ax \leq b\}$$

is/is not bounded?

♠ *How to recognize that two polyhedral sets*

$$X = \{x \in \mathbb{R}^n : Ax \leq b\} \text{ and } X' = \{x : A'x \leq b'\}$$

are/are not distinct?

♠ *How to recognize that a given LO program is feasible/bounded/solvable?*

♠ *.....*

Our current goal is to find answers to these and similar questions, and these answers come from *Linear Programming Duality Theorem* which is the second (or even the first ?) main theoretical result in LO.

Theorem on Alternative

♣ Consider a system of m strict and nonstrict linear inequalities in variables $x \in \mathbb{R}^n$:

$$a_i^T x \begin{cases} < b_i, & i \in I \\ \leq b_i, & i \in \overline{I} \end{cases} \qquad (\mathcal{S})$$

- $a_i \in \mathbb{R}^n, b_i \in \mathbb{R}$, $1 \leq i \leq m$,
- $I \subset \{1, ..., m\}$, $\overline{I} = \{1, ..., m\} \backslash I$.

Note: $(\mathcal{S})$ is a universal form of a finite system of linear inequalities in n variables.

♣ **Main questions on $(\mathcal{S})$ [operational form]:**

- *How to find a solution to the system if one exists?*
- *How to find out that $(\mathcal{S})$ is infeasible?*

♠ **Main questions on $(\mathcal{S})$ [descriptive form]:**

- *How to certify that $(\mathcal{S})$ is solvable?*
- *How to certify that $(\mathcal{S})$ is infeasible?*

$$a_i^T x \begin{cases} < b_i, & i \in I \\ \leq b_i, & i \in \bar{I} \end{cases} \qquad (\mathcal{S})$$

♠ The simplest certificate for solvability of $(\mathcal{S})$ is *a* solution: plug a candidate certificate into the system and check that the inequalities are satisfied.

Example: The vector $\bar{x} = [10; 10; 10]$ is a solvability certificate for the system

$$\begin{array}{rrrcr} -x_1 & -x_2 & -x_3 & < & -29 \\ x_1 & +x_2 & & \leq & 20 \\ & x_2 & +x_3 & \leq & 20 \\ x_1 & & +x_3 & \leq & 20 \end{array}$$

– when plugging it into the system, we get valid numerical inequalities.

But: *How to certify that $(\mathcal{S})$ has no solution?* E.g., how to certify that the system

$$\begin{array}{rrrcr} -x_1 & -x_2 & -x_3 & < & -30 \\ x_1 & +x_2 & & \leq & 20 \\ & x_2 & +x_3 & \leq & 20 \\ x_1 & & +x_3 & \leq & 20 \end{array}$$

has no solutions???

$$a_i^T x \begin{cases} < b_i, & i \in I \\ \leq b_i, & i \in \overline{I} \end{cases} \qquad (\mathcal{S})$$

♣ *How to certify that* $(\mathcal{S})$ *has no solutions?*

♠ **A recipe:** *Take a weighted sum, with nonnegative weights, of the inequalities from the system, thus getting strict or non-strict scalar linear inequality which, due its origin, is a consequence of the system – it must be satisfied at every solution to* $(\mathcal{S})$*. If the resulting inequality has no solutions at all, then* $(\mathcal{S})$ *is unsolvable.*

Example: To certify that the system

$2\times$	$-x_1$	$-x_2$	$-x_3$	$<$	-30
$1\times$	x_1	$+x_2$		$\leq$	20
$1\times$		x_2	$+x_3$	$\leq$	20
$1\times$	x_1		$+x_3$	$\leq$	20

has no solutions, take the weighted sum of the inequalities with the weights marked in red, thus arriving at the inequality

$$0 \cdot x_1 + 0 \cdot x_2 + 0 \cdot x_3 < 0.$$

This is a contradictory inequality which is a consequence of the system

$\Rightarrow$ *weights* $\lambda = [2; 1; 1; 1]$ *certify insolvability.*

$$a_i^T x \begin{cases} < b_i, & i \in I \\ \le b_i, & i \in \bar{I} \end{cases} \qquad (\mathcal{S})$$

A recipe for certifying insolvability:

- *Assign inequalities of* $(\mathcal{S})$ *with weights* $\lambda_i \ge 0$ *and sum them up, thus arriving at the inequality*

$$[\textstyle\sum_{i=1}^m \lambda_i a_i]^T x \;?\; \sum_{i=1}^m \lambda_i b_i$$

$$\left[\begin{array}{l} ? = \text{"}<\text{"} \text{ when } \sum_{i\in I}\lambda_i > 0 \\ ? = \text{"}\le\text{"} \text{ when } \sum_{i\in I}\lambda_i = 0 \end{array}\right] \qquad (!)$$

- *If* $(!)$ *has no solutions,* $(\mathcal{S})$ *is insolvable.*

♠ **Observation:** *Inequality* $(!)$ *has no solution iff* $\sum_{i=1}^m \lambda_i a_i = 0$ *and, in addition,*

- $\sum_{i=1}^m \lambda_i b_i \le 0$ *when* $\sum_{i\in I}\lambda_i > 0$
- $\sum_{i=1}^m \lambda_i b_i < 0$ *when* $\sum_{i\in I}\lambda_i = 0$

♣ We have arrived at

Proposition: *Given system* $(\mathcal{S})$*, let us associate with it two systems of linear inequalities in variables* $\lambda_1, ..., \lambda_m$*:*

$$(\mathrm{I}): \begin{cases} \lambda_i \ge 0 \,\forall i \\ \sum_{i=1}^m \lambda_i a_i = 0 \\ \sum_{i=1}^m \lambda_i b_i \le 0 \\ \sum_{i\in I}\lambda_i > 0 \end{cases}, \qquad (\mathrm{II}): \begin{cases} \lambda_i \ge 0 \,\forall i \\ \sum_{i=1}^m \lambda_i a_i = 0 \\ \sum_{i=1}^m \lambda_i b_i < 0 \\ \lambda_i = 0, i \in I \end{cases}$$

If at least one of the systems (I), (II) *has a solution, then* $(\mathcal{S})$ *has no solutions.*

General Theorem on Alternative: *Consider, along with system of linear inequalities*

$$a_i^T x \begin{cases} < b_i, & i \in I \\ \le b_i, & i \in \overline{I} \end{cases} \qquad (\mathcal{S})$$

in variables $x \in \mathbb{R}^n$, two systems of linear inequalities in variables $\lambda \in \mathbb{R}^m$:

$$\text{(I)}: \begin{cases} \lambda_i \ge 0 \, \forall i \\ \sum_{i=1}^m \lambda_i a_i = 0 \\ \sum_{i=1}^m \lambda_i b_i \le 0 \\ \sum_{i \in I} \lambda_i > 0 \end{cases}, \quad \text{(II)}: \begin{cases} \lambda_i \ge 0 \, \forall i \\ \sum_{i=1}^m \lambda_i a_i = 0 \\ \sum_{i=1}^m \lambda_i b_i < 0 \\ \lambda_i = 0, i \in I \end{cases}$$

System $(\mathcal{S})$ has no solutions if and only if at least one of the systems (I), (II) *has a solution.*

Remark: Strict inequalities in $(\mathcal{S})$ in fact do not participate in (II). As a result, (II) *has a solution iff the "nonstrict" subsystem*

$$a_i^T x \le b_i, \; i \in \overline{I} \qquad (\mathcal{S}')$$

of $(\mathcal{S})$ has no solutions.

Remark: GTA says that a finite system of linear inequalities has no solutions if and only if (one of two) other systems of linear inequalities has a solution. Such a solution can be considered as a certificate of insolvability of $(\mathcal{S})$: $(\mathcal{S})$ is insolvable if and only if such an insolvability certificate exists.

Proof of GTA

$$a_i^T x \begin{cases} < b_i, & i \in I \\ \le b_i, & i \in \overline{I} \end{cases} \qquad (\mathcal{S})$$

$$\text{(I)}: \begin{cases} \lambda_i \ge 0 \, \forall i \\ \sum_{i=1}^m \lambda_i a_i = 0 \\ \sum_{i=1}^m \lambda_i b_i \le 0 \\ \sum_{i \in I} \lambda_i > 0 \end{cases}, \qquad \text{(II)}: \begin{cases} \lambda_i \ge 0 \, \forall i \\ \sum_{i=1}^m \lambda_i a_i = 0 \\ \sum_{i=1}^m \lambda_i b_i < 0 \\ \lambda_i = 0, i \in I \end{cases}$$

• In one direction: "If (I) or (II) has a solution, then $(\mathcal{S})$ has no solutions" the statement is already proved.

• Now assume that $(\mathcal{S})$ has no solutions, and let us prove that one of the systems (I), (II) has a solution. Consider the system of homogeneous linear inequalities in variables x, t, ϵ:

$$\begin{array}{rrcl} a_i^T x & -b_i t & \le & 0, i \in I \\ a_i^T x & -b_i t & \le & 0, \, i \in \overline{I} \\ & -t & \le & 0 \\ & -\epsilon & < & 0 \end{array}$$

We claim that *this system has no solutions.* Indeed, assuming that the system has a solution $\overline{x}, \overline{t}, \overline{\epsilon}$, we have $\overline{\epsilon} > 0$, whence

$$\overline{t} > 0 \ \& \ a_i^T \overline{x} < b_i \overline{t}, i \in I \ \& \ a_i^T \overline{x} \le b_i \overline{t} \le 0, \, i \in \overline{I},$$

$\Rightarrow x = \overline{x}/\overline{t}$ is well defined and solves *un*solvable system $(\mathcal{S})$, which is impossible.

Proof of GTA (continued)

Situation: System

$$\begin{array}{rrrcl} a_i^T x & -b_i t & +\epsilon & \leq & 0, i \in I \\ a_i^T x & -b_i t & & \leq & 0,\ i \in \overline{I} \\ & -t & +\epsilon & \leq & 0 \\ & & -\epsilon & < & 0 \end{array}$$

has no solutions, or, equivalently, the homogeneous linear inequality

$$-\epsilon \geq 0$$

is a consequence of the system of homogeneous linear inequalities

$$\begin{array}{rrrcl} -a_i^T x & +b_i t & -\epsilon & \geq & 0,\ i \in I \\ -a_i^T x & +b_i t & & \geq & 0\, i \in \overline{I} \\ & t & -\epsilon & \geq & 0 \end{array}$$

in variables x, t, ϵ. By Homogeneous Farkas Lemma, there exist $\mu_i \geq 0$, $1 \leq i \leq m$, $\mu \geq 0$ such that

$$\sum_{i=1}^m \mu_i a_i = 0 \ \& \ \sum_{i=1}^m \mu_i b_i + \mu = 0 \ \& \ \sum_{i \in I} \mu_i + \mu = 1$$

When $\mu > 0$, setting $\lambda_i = \mu_i/\mu$, we get

$$\lambda \geq 0, \textstyle\sum_{i=1}^m \lambda_i a_i = 0,\ \sum_{i=1}^m \lambda_i b_i = -1,$$

$\Rightarrow$ when $\sum_{i \in I} \lambda_i > 0$, λ solves (I), otherwise λ solves (II).

When $\mu = 0$, setting $\lambda_i = \mu_i$, we get

$$\lambda \geq 0, \textstyle\sum_i \lambda_i a_i = 0,\ \sum_i \lambda_i b_i = 0,\ \sum_{i \in I} \lambda_i = 1,$$

and λ solves (I). $\square$

♣ GTA is equivalent to the following

Principle: *A finite system of linear inequalities has no solution iff one can get, as a legitimate* (i.e., compatible with the common rules of operating with inequalities) *weighted sum of inequalities from the system, a contradictory inequality, i.e., either inequality* $0^T x \leq -1$*, or the inequality* $0^T x < 0$*.*

The advantage of this Principle is that it does not require converting the system into the standard form. For example, to see that the system of linear constraints

$$\begin{array}{rcl} x_1 + 2x_2 & < & 5 \\ 2x_1 + 3x_2 & \geq & 3 \\ 3x_1 + 4x_2 & = & 1 \end{array}$$

has no solutions, it suffices to take the weighted sum of these constraints with the weights $-1, 2, -1$, thus arriving at the contradictory inequality

$$0 \cdot x_1 + 0 \cdot x_2 > 0$$

♣ Specifying the system in question and applying GTA, we can obtain various particular cases of GTA, e.g., as follows:

Inhomogeneous Farkas Lemma: *A nonstrict linear inequality*

$$a^T x \le \alpha \qquad (!)$$

is a consequence of a solvable system of nonstrict linear inequalities

$$a_i^T x \le b_i,\ 1 \le i \le m \qquad (S)$$

if and only if $(!)$ *can be obtained by taking weighted sum, with nonnegative coefficients, of the inequalities from the system and the identically true inequality* $0^T x \le 1$*:* (S) implies $(!)$ iff there exist nonnegative weights $\lambda_0, \lambda_1, ..., \lambda_m$ such that

$$\lambda_0 \cdot [1 - 0^T x] + \sum_{i=1}^m \lambda_i [b_i - a_i^T x] \equiv \alpha - a^T x,$$

or, which is the same, iff there exist nonnegative $\lambda_0, \lambda_1, ..., \lambda_m$ such that

$$\sum_{i=1}^m \lambda_i a_i = a,\ \sum_{=1}^m \lambda_i b_i + \lambda_0 = \alpha$$

or, which again is the same, *iff there exist nonnegative* $\lambda_1, ..., \lambda_m$ *such that*

$$\sum_{i=1}^m \lambda_i a_i = a,\ \sum_{i=1}^m \lambda_i b_i \le \alpha.$$

Proof of Inhomogeneous Farkas Lemma

$$a^T x \le \alpha \qquad (!)$$

• If (!) can be obtained as a weighted sum, with nonnegative coefficients, of the inequalities from (S) and the inequality $0^T x \le 1$, then (!) clearly is a corollary of (S) independently of whether (S) is or is not solvable.

Now let (S) be solvable and (!) be a consequence of (S); we want to prove that (!) is a combination, with nonnegative weights, of the constraints from (S) and the constraint $0^T x \le 1$. Since (!) is a consequence of (S), the system

$$-a^T x < -\alpha,\ a_i^T x \le b_i\ 1 \le i \le m \qquad (M)$$

has no solutions, whence, by GTA, a legitimate weighted sum of the inequalities from the system is contradictory, that is, there exist $\mu \ge 0$, $\lambda_i \ge 0$:

$$-\mu a + \textstyle\sum_{i=1}^m \lambda_i a_i = 0,\ 0\ ??\ \sum_{i=1}^m \lambda_i b_i - \mu\alpha$$
$$?? = \begin{cases} " \ge ", & \mu > 0 \\ " > ", & \mu = 0 \end{cases} \qquad (!!)$$

Proof of Inhomogeneous Farkas Lemma (continued)

Situation: the system

$$-a^T x < -\alpha,\ a_i^T x \le b_i\ 1 \le i \le m \qquad (M)$$

has no solutions, whence there exist $\mu \ge 0, \lambda \ge 0$ such that

$$-\mu a + \sum_{i=1}^m \lambda_i a_i = 0,\ 0\ ??\ \sum_{i=1}^m \lambda_i b_i - \mu\alpha$$
$$?? = \begin{cases} " \ge ", & \mu > 0 \\ " > ", & \mu = 0 \end{cases} \qquad (!!)$$

Claim: $\mu > 0$. Indeed, otherwise the inequality $-a^T x < -\alpha$ does not participate in the weighted sum of the constraints from (M) which is a contradictory inequality

$\Rightarrow$ (S) can be led to a contradiction by taking weighted sum of the constraints

$\Rightarrow$ (S) is infeasible, which is a contradiction with the premise of Inhomogeneous Farkas Lemma.

- When $\mu > 0$, setting $\lambda_i = \mu_i/\mu$, we get from (!!)

$$\sum_i \lambda_i a_i = a\ \&\ \sum_{i=1}^m \lambda_i b_i - \alpha \le 0. \qquad \square$$

Why GTA is a deep fact?

♣ Consider the system of four linear inequalities in variables u, v:

$$-1 \le u \le 1,\ -1 \le v \le 1$$

and let us derive its consequence as follows:

$$\begin{array}{rl} & -1 \le u \le 1,\ -1 \le v \le 1 \\ \Rightarrow & u^2 \le 1, v^2 \le 1 \\ \Rightarrow & u^2 + v^2 \le 2 \\ \Rightarrow & u + v = 1 \cdot u + 1 \cdot v \le \sqrt{1^2 + 1^2}\sqrt{u^2 + v^2} \\ \Rightarrow & u + v \le \sqrt{2}\sqrt{2} = 2 \end{array}$$

This derivation is of a nonstrict *linear* inequality which is a consequence of a system of a solvable system of nonstrict linear inequalities is "highly nonlinear." A statement which says that *every* derivation of this type can be replaced by just taking weighted sum of the original inequalities and the trivial inequality $0^T x \le 1$ is a deep statement indeed!

♣ For *every* system $\mathcal{S}$ of inequalities, linear or nonlinear alike, taking weighted sums of inequalities of the system and trivial – identically true – inequalities always results in a consequence of $\mathcal{S}$

$\Rightarrow$ *In one direction, GTA always is true.*

However the other direction in GTA heavily exploits the fact that the inequalities of the original system and a consequence we are looking for are *linear.* Already for quadratic inequalities, the statement similar to GTA fails to be true. For example, the quadratic inequality

$$x^2 \leq 1 \qquad (!)$$

is a consequence of the system of linear (and thus quadratic) inequalities

$$-1 \leq x \leq 1 \qquad (*)$$

Nevertheless, (!) can*not* be represented as a weighted sum of the inequalities from $(*)$ and identically true linear and quadratic inequalities, like

$$0 \cdot x \leq 1, x^2 \geq 0, x^2 - 2x + 1 \geq 0, ...$$

Answering Questions

♣ *How to certify that a polyhedral set*

$$X = \{x \in \mathbb{R}^n : Ax \leq b\}$$

is empty/nonempty?

♠ A certificate for X to be *nonempty* is a solution $\bar{x}$ to the system $Ax \leq b$.

♠ A certificate for X to be *empty* is a solution $\bar{\lambda}$ to the system $\lambda \geq 0, A^T\lambda = 0, b^T\lambda < 0$.

• In both cases, X possesses the property in question iff it can be certified as explained above ("the certification scheme is complete").

Note: *All certification schemes to follow are complete!*

Examples:

• The vector $x = [1; ...; 1] \in \mathbb{R}^n$ certifies that the polyhedral set

$$X = \{x \in \mathbb{R}^n : x_1 + x_2 \leq 2, x_2 + x_3 \leq 2, ..., x_n + x_1 \leq 2, \\ -x_1 - ... - x_n \leq -n\}$$

is nonempty.

• The vector $\lambda = [1; 1; ...; 1; 2] \in \mathbb{R}^{n+1} \geq 0$ certifies that the polyhedral set

$$X = \{x \in \mathbb{R}^n : x_1 + x_2 \leq 2, x_2 + x_3 \leq 2, ..., x_n + x_1 \leq 2, \\ -x_1 - ... - x_n \leq -n - 0.01\}$$

is empty. Indeed, summing up the $n + 1$ constraints defining $X = \{x : Ax \leq b\}$ with weights λ_i, we get the contradictory inequality

$$\underbrace{0 \equiv 2\,(x_1 + ... + x_n) - 2\,[x_1 + ... + x_n]}_{[A^T\lambda]^T x \equiv 0}$$

$$\leq \underbrace{2n - 2(n + 0.01) = -0.02}_{b^T\lambda = -0.02 < 0}$$

♣ *How to certify that a linear inequality $c^T x \leq d$ is violated somewhere on a polyhedral set*

$$X = \{x \in \mathbb{R}^n : Ax \leq b\},$$

that is, the inequality is not a consequence of the system $Ax \leq b$?

A certificate is $\bar{x}$ such that $A\bar{x} \leq b$ and $c^T \bar{x} > d$.

♣ *How to certify that a linear inequality $c^T x \leq d$ is satisfied everywhere on a polyhedral set*

$$X = \{x \in \mathbb{R}^n : Ax \leq b\},$$

that is, the inequality is a consequence of the system $Ax \leq b$?

♠ The situation in question arises in two cases:

A. X is empty, the target inequality is an arbitrary one

B. X is nonempty, the target inequality is a consequence of the system $Ax \leq b$

Consequently, to certify the fact in question means

— either to certify that X is empty, the certificate being λ such that $\lambda \geq 0, A^T\lambda = 0, b^T\lambda < 0$,

— or to certify that X is nonempty by pointing out a solution $\bar{x}$ to the system $Ax \leq b$ *and* to certify the fact that $c^T x \leq d$ is a consequence of the solvable system $Ax \leq b$ by pointing out a λ which satisfies the system $\lambda \geq 0, A^T\lambda = c, b^T\lambda \leq d$ (we have used Inhomogeneous Farkas Lemma).

Note: In the second case, we can omit the necessity to certify that $X \neq \emptyset$, since the existence of λ satisfying $\lambda \geq 0, A^T\lambda = c, b^T\lambda \leq d$ *always is sufficient* for $c^T x \leq d$ to be a consequence of $Ax \leq b$.

Example:

- To certify that the linear inequality

$$c^T x := x_1 + ... + x_n \leq d := n - 0.01$$

is violated somewhere on the polyhedral set

$$\begin{array}{rcl} X & = & \{x \in \mathbb{R}^n : x_1 + x_2 \leq 2, x_2 + x_3 \leq 2, ..., x_n + x_1 \leq 2\} \\ & = & \{x : Ax \leq b\} \end{array}$$

it suffices to note that $x = [1; ...; 1] \in X$ and $n = c^T x > d = n - 0.01$

- To certify that the linear inequality

$$c^T x := x_1 + ... + x_n \leq d := n$$

is satisfied everywhere on the above X, it suffices to note that when taking weighted sum of inequalities defining X, the weights being $1/2$, we get the target inequality.

Equivalently: for $\lambda = [1/2; ...; 1/2] \in \mathbb{R}^n$ it holds $\lambda \geq 0, A^T \lambda = [1; ...; 1] = c, b^T \lambda = n \leq d$

♣ *How to certify that a polyhedral set*

$$X = \{x \in \mathbb{R}^n : Ax \leq b\}$$

does not contain a polyhedral set

$$Y = \{x \in \mathbb{R}^n : Cx \leq d\}?$$

• A certificate is a point $\bar{x}$ such that $C\bar{x} \leq d$ (i.e., $\bar{x} \in Y$) and $\bar{x}$ does *not* solve the system $Ax \leq b$ (i.e., $\bar{x} \notin X$).

♣ *How to certify that a polyhedral set*

$$X = \{x \in \mathbb{R}^n : Ax \leq b\}$$

contains a polyhedral set

$$Y = \{x \in \mathbb{R}^n : Cx \leq d\}?$$

This situation arises in two cases:

— $Y = \emptyset$, X is arbitrary. To certify that this is the case, it suffices to point out λ such that $\lambda \geq 0, C^T\lambda = 0, d^T\lambda < 0$

— Y is nonempty and every one of the m linear inequalities $a_i^T x \leq b_i$ defining X is satisfied everywhere on Y. To certify that this is the case, it suffices to point out $\bar{x}$, $\lambda^1, ..., \lambda^m$ such that

$$C^T\bar{x} \leq d \ \& \ \lambda^i \geq 0, C^T\lambda_i = a_i, d^T\lambda^i \leq b_i, \ 1 \leq i \leq m.$$

Note: Same as above, we can omit the necessity to point out $\bar{x}$.

Examples.

- To certify that the set

$$Y = \{x \in \mathbb{R}^3 : -2 \le x_1 + x_2 \le 2, -2 \le x_2 + x_3 \le 2, \\ -2 \le x_3 + x_1 \le 2\}$$

is *not* contained in the box

$$X = \{x \in \mathbb{R}^3 : |x_i| \le 2, 1 \le i \le 3\},$$

it suffices to note that the vector $\bar{x} = [3; -1; -1]$ belongs to Y and does not belong to X.

- To certify that the above Y is contained in

$$X' = \{x \in \mathbb{R}^3 : |x_i| \le 3, 1 \le i \le 3\}$$

note that summing up the 6 inequalities

$$x_1 + x_2 \le 2, -x_1 - x_2 \le 2, x_2 + x_3 \le 2, -x_2 - x_3 \le 2, \\ x_3 + x_1 \le 2, -x_3 - x_1 \le 2$$

defining Y with the nonnegative weights

$$\lambda_1 = 1, \lambda_2 = 0, \lambda_3 = 0, \lambda_4 = 1, \lambda_5 = 1, \lambda_6 = 0$$

we get

$$[x_1 + x_2] + [-x_2 - x_3] + [x_3 + x_1] \le 6 \Rightarrow x_1 \le 3$$

— with the nonnegative weights

$$\lambda_1 = 0, \lambda_2 = 1, \lambda_3 = 1, \lambda_4 = 0, \lambda_5 = 0, \lambda_6 = 1$$

we get

$$[-x_1 - x_2] + [x_2 + x_3] + [-x_3 - x_1] \le 6 \Rightarrow -x_1 \le 3$$

The inequalities $-3 \le x_2, x_3 \le 3$ can be obtained similarly $\Rightarrow Y \subset X'$.

♣ *How to certify that a polyhedral set*

$$X = \{x \in \mathbb{R}^n : Ax \leq b\}$$

is bounded/unbounded?

• X is bounded iff for properly chosen R it holds

$$X \subset X_R = \{x : |x_i| \leq R,\ 1 \leq i \leq n\}$$

To certify this means

— either to certify that X is empty, the certificate being λ: $\lambda \geq 0, A^T\lambda = 0, b^T\lambda < 0$,

— or to point out vectors R and vectors $\lambda^i_\pm$ such that $\lambda^i_\pm \geq 0, A^T\lambda^i_\pm = \pm e_i, b^T\lambda^i_\pm \leq R$ for all i. Since R can be chosen arbitrary large, the latter amounts to pointing out vectors $\lambda^i_\pm$ such that $\lambda^i_\pm \geq 0$, $A^T\lambda^i_\pm = \pm e_i$, $i = 1, ..., n$.

• X is *un*bounded iff X is nonempty and the recessive cone $\mathrm{Rec}(X) = \{x : Ax \leq 0\}$ is nontrivial. To certify that this is the case, it suffices to point out $\bar{x}$ satisfying $A\bar{x} \leq b$ and $\bar{d}$ satisfying $\bar{d} \neq 0, A\bar{d} \leq 0$.

Note: *When $X = \{x \in \mathbb{R}^n : Ax \leq b\}$ is known to be nonempty, its boundedness/unboundedness is independent of the particular value of b!*

Examples:

- To certify that the set

$$X = \{x \in \mathbb{R}^3 : -2 \le x_1 + x_2 \le 2, -2 \le x_2 + x_3 \le 2, \\ -2 \le x_3 + x_1 \le 2\}$$

is bounded, it suffices to certify that it belongs to the box $\{x \in \mathbb{R}^3 : |x_i| \le 3, 1 \le i \le 3\}$, which was already done.

- To certify that the set

$$X = \{x \in \mathbb{R}^4 : -2 \le x_1 + x_2 \le 2, -2 \le x_2 + x_3 \le 2, \\ -2 \le x_3 + x_4 \le 2, -2 \le x_4 + x_1 \le 2\}$$

is *un*bounded, it suffices to note that the vector $\bar{x} = [0; 0; 0; 0]$ belongs to X, and the vector $\bar{d} = [1; -1; 1; -1]$ when plugged into the inequalities defining X makes the bodies of the inequalities zero and thus is a recessive direction of X.

Certificates in LO

♣ Consider LO program in the form

$$\mathrm{Opt} = \max_x \left\{ c^T x : \left\{ \begin{array}{rcll} Px & \leq & p & (\ell) \\ Qx & \geq & q & (g) \\ Rx & = & r & (e) \end{array} \right\} \right. \qquad (P)$$

♠ *How to certify that (P) is feasible/infeasible?*

• To certify that (P) is feasible, it suffices to point out a feasible solution $\bar{x}$ to the program.

• To certify that (P) is *in*feasible, it suffices to point out aggregation weights $\lambda_\ell, \lambda_g, \lambda_e$ such that

$$\begin{array}{c} \lambda_\ell \geq 0, \lambda_g \leq 0 \\ P^T\lambda_\ell + Q^T\lambda_g + R^T\lambda_e = 0 \\ p^T\lambda_\ell + q^T\lambda_g + r^T\lambda_e < 0 \end{array}$$

$$\mathrm{Opt} = \max_x \left\{ c^T x : \left\{ \begin{array}{rcll} Px & \leq & p & (\ell) \\ Qx & \geq & q & (g) \\ Rx & = & r & (e) \end{array} \right\} \right. \qquad (P)$$

♠ *How to certify that (P) is bounded/unbounded?*

• (P) is bounded if either (P) is infeasible, or (P) is feasible and there exists a such that the inequality $c^T x \leq a$ is consequence of the system of constraints. Consequently, to certify that (P) is bounded, we should

— either point out an infeasibility certificate $\lambda_\ell \geq 0, \lambda_g \leq 0, \lambda_e : P^T\lambda_\ell + Q^T\lambda_g + R^T\lambda_e = 0, p^T\lambda_\ell + q^T\lambda_g + r^T\lambda_e < 0$ for (P),

— or point out a feasible solution $\bar{x}$ and $a, \lambda_\ell, \lambda_g, \lambda_e$ such that

$$\lambda_\ell \geq 0, \lambda_g \leq 0 \ \& \ P^T\lambda_\ell + Q^T\lambda_g + R^T\lambda_e = c \\ \& \ p^T\lambda_\ell + q^T\lambda_g + r^T\lambda_e \leq a$$

which, since a can be arbitrary, amounts to

$$\lambda_\ell \geq 0, \lambda_g \leq 0, P^T\lambda_\ell + Q^T\lambda_g + R^T\lambda_e = c$$

Note: We can skip the necessity to certify that (P) is feasible.

$$\mathrm{Opt} = \max_x \left\{ c^T x : \left\{ \begin{array}{rcll} Px & \leq & p & (\ell) \\ Qx & \geq & q & (g) \\ Rx & = & r & (e) \end{array} \right\} \right. \qquad (P)$$

- (P) is *un*bounded iff (P) is feasible and there is a recessive direction d such that $c^T d > 0$

$\Rightarrow$ to certify that (P) is unbounded, we should point out a feasible solution $\bar{x}$ to (P) *and* a vector d such that

$$Pd \leq 0, Qd \geq 0, Rd = 0, c^T d > 0.$$

Note: *If (P) is known to be feasible, its boundedness/unboundedness is independent of a particular value of $[p; q; r]$.*

$$\mathrm{Opt} = \max_x \left\{ c^T x : \left\{ \begin{array}{rcll} Px & \leq & p & (\ell) \\ Qx & \geq & q & (g) \\ Rx & = & r & (e) \end{array} \right\} \right. \qquad (P)$$

♠ *How to certify that* $\mathrm{Opt} \geq a$ *for a given* $a \in \mathbb{R}$*?*

A certificate is a feasible solution $\bar{x}$ with $c^T x \geq a$.

♠ *How to certify that* $\mathrm{Opt} \leq a$ *for a given* $a \in \mathbb{R}$*?*

$\mathrm{Opt} \leq a$ iff the linear inequality $c^T x \leq a$ is a consequence of the system of constraints. To certify this, we should

— either point out an infeasibility certificate $\lambda_\ell, \lambda_g, \lambda_e$:

$$\begin{gathered} \lambda_\ell \geq 0, \lambda_g \leq 0, \\ P^T \lambda_\ell + Q^T \lambda_g + R^T \lambda_e = 0, \\ p^T \lambda_e + q^T \lambda_g + r^T \lambda_e < 0 \end{gathered}$$

for (P),

— or point out $\lambda_\ell, \lambda_g, \lambda_e$ such that

$$\begin{gathered} \lambda_\ell \geq 0, \lambda_g \leq 0, \\ P^T \lambda_\ell + Q^T \lambda_g + R^T \lambda_e = c, \\ p^T \lambda_\ell + q^T \lambda_g + r^T \lambda_e \leq a \end{gathered}$$

$$\mathrm{Opt} = \max_x \left\{ c^T x : \left\{ \begin{array}{rcll} Px & \leq & p & (\ell) \\ Qx & \geq & q & (g) \\ Rx & = & r & (e) \end{array} \right\} \right. \qquad (P)$$

♣ *How to certify that $\bar{x}$ is an optimal solution to (P)?*

• $\bar{x}$ is optimal solution iff it is feasible and $\mathrm{Opt} \leq c^T \bar{x}$. The latter amounts to existence of $\lambda_\ell, \lambda_g, \lambda_e$ such that

$$\overbrace{\lambda_\ell \geq 0, \lambda_g \leq 0}^{(a)} \ \& \ \overbrace{P^T\lambda_\ell + Q^T\lambda_g + R^T\lambda_e = c}^{(b)} \ \& \ \underbrace{p^T\lambda_\ell + q^T\lambda_g + r^T\lambda_e = c^T\bar{x}}_{(c)}$$

• Multiplying both sides in (b) by $\bar{x}^T$ and subtracting the result from (c), we get

$$\underbrace{\lambda_\ell^T}_{\geq 0} \overbrace{[p - P\bar{x}]}^{\geq 0} + \underbrace{\lambda_g^T}_{\leq 0} \overbrace{[q - Q\bar{x}]}^{\leq 0} + \lambda_e^T \overbrace{[r - R\bar{x}]}^{=0} = 0$$

which is possible iff $(\lambda_\ell)_i[p_i - (P\bar{x})_i] = 0$ for all i and $(\lambda_g)_j[q_j - (Q\bar{x})_j] = 0$ for all j.

$$\mathrm{Opt} = \max_x \left\{ c^T x : \left\{ \begin{array}{rcll} Px & \leq & p & (\ell) \\ Qx & \geq & q & (g) \\ Rx & = & r & (e) \end{array} \right\} \right. \qquad (P)$$

♠ We have arrived at the *Karush-Kuhn-Tucker Optimality Conditions in LO:*

A feasible solution $\bar{x}$ to (P) is optimal iff the constraints of (P) can be assigned with vectors of Lagrange multipliers $\lambda_\ell, \lambda_g, \lambda_e$ in such a way that

- [signs of multipliers] *Lagrange multipliers associated with $\leq$-constraints are nonnegative, and Lagrange multipliers associated with $\geq$-constraints are nonpositive,*
- [complementary slackness] *Lagrange multipliers associated with non-active at $\bar{x}$ constraints are zero, and*
- [KKT equation] *One has*

$$P^T\lambda_\ell + Q^T\lambda_g + R^T\lambda_e = c$$

Example. To certify that the feasible solution

$$\bar{x} = [1; ...; 1] \in \mathbb{R}^n$$

to the LO program

$$\max_x \left\{ x_1 + ... + x_n : \begin{array}{llll} x_1 + x_2 \le 2, & x_2 + x_3 \le 2 & , ..., & x_n + x_1 \le 2 \\ x_1 + x_2 \ge -2, & x_2 + x_3 \ge -2 & , ..., & x_n + x_1 \ge -2 \end{array} \right\}$$

is optimal, it suffices to assign the constraints with Lagrange multipliers $\lambda_\ell = [1/2; 1/2; ...; 1/2]$, $\lambda_g = [0; ...; 0]$ and to note that

$$\begin{array}{c} \\ P^T\lambda_\ell + Q^T\lambda_g = \end{array} \begin{array}{c} \lambda_\ell \ge 0, \lambda_g \le 0 \\ \left[\begin{array}{ccccccc} 1 & & & & & & 1 \\ 1 & 1 & & & & & \\ & 1 & 1 & & & & \\ & & 1 & \ddots & & & \\ & & & \ddots & 1 & & \\ & & & & 1 & 1 & \end{array} \right] \end{array} \lambda_\ell^T = c := [1; ...; 1]$$

and complementary slackness takes place.

♣ **Application:** Faces of polyhedral set revisited. Recall that a face of a nonempty polyhedral set

$$X = \{x \in \mathbb{R}^n : a_i^T x \leq b_i, 1 \leq i \leq m\}$$

is a nonempty set of the form

$$X_I = \{x \in \mathbb{R}^n : a_i^T x = b_i, i \in I, a_i^T x \leq b_i, i \notin I\}$$

This definition is *not* geometric.

Geometric characterization of faces:

(i) *Let $c^T x$ be a linear function bounded from above on X. Then the set*

$$\mathrm{Argmax}_X\, c^T x := \{x \in X : c^T x = \max_{x' \in X} c^T x'\}$$

is a face of X. In particular, if the maximizer of $c^T x$ over X exists and is unique, it is an extreme point of X.

(ii) *Vice versa, every face of X admits a representation as $\mathrm{Argmax}_{x \in X}\, c^T x$ for properly chosen c. In particular, every vertex of X is the unique maximizer, over X, of some linear function.*

Proof:

(i): Let $c^T x$ be bounded from above on X. Then the set $X_* = \mathrm{Argmax}_{x \in X} c^T x$ is nonempty. Let $x_* \in X_*$. By KKT Optimality conditions, there exist $\lambda \geq 0$ such that

$$\sum_i \lambda_i a_i = c, \; a_i^T x < b_i \Rightarrow \lambda_i = 0.$$

Let $I_* = \{i : \lambda_i > 0\}$. We claim that $X_* = X_{I_*}$. Indeed,

— $x \in X_{I_*} \Rightarrow c^T x = [\sum_{i \in I_*} \lambda_i a_i]^T x$

$= \sum_{i \in I_*} \lambda_i a_i^T x = \sum_{i \in I_*} b_i$

$= \sum_{i \in I_*} \lambda_i a_i^T x_* = c^T x_*,$

$\Rightarrow x \in X_* := \mathrm{Argmax}_{y \in X} c^T y$, and

— $x \in X_* \Rightarrow c^T (x_* - x) = 0$

$\Rightarrow \sum_{i \in I_*} \lambda_i (a_i^T x_* - a_i^T x) = 0$

$\Rightarrow \sum_{i \in I_*} \underbrace{\lambda_i}_{>0} (b_i - \underbrace{a_i^T x}_{\leq b_i}) = 0$

$\Rightarrow a_i^T x = b_i \, \forall i \in I_* \Rightarrow x \in X_{I_*}$.

(ii): Let X_I be a face of X, and let us set $c = \sum_{i \in I} a_i$. Same as above, it is immediately seen that $X_I = \mathrm{Argmax}_{x \in X} c^T x$.

LO Duality

♣ Consider an LO program

$$\mathrm{Opt}(P) = \max_x \left\{ c^T x : Ax \leq b \right\} \qquad (P)$$

The dual problem stems from the desire to bound from above the optimal value of the *primal* problem (P), To this end, we use our aggregation technique, specifically,

- *assign the constraints $a_i^T x \leq b_i$ with nonnegative aggregation weights λ_i* ("Lagrange multipliers") *and sum them up with these weights, thus getting the inequality*

$$[A^T\lambda]^T x \leq b^T\lambda \qquad (!)$$

Note: by construction, this inequality is a consequence of the system of constraints in (P) and thus is satisfied at every feasible solution to (P).

- *We may be lucky to get in the left hand side of $(!)$ exactly the objective $c^T x$:*

$$A^T\lambda = c.$$

In this case, $(!)$ says that $b^T\lambda$ is an upper bound on $c^T x$ everywhere in the feasible domain of (P), and thus $b^T\lambda \geq \mathrm{Opt}(P)$.

$$\mathrm{Opt}(P) = \max_x \left\{ c^T x : Ax \le b \right\} \qquad (P)$$

♠ *We arrive at the problem of finding the best – the smallest – upper bound on* $\mathrm{Opt}(P)$ *achievable with our bounding scheme. This new problem is*

$$\mathrm{Opt}(D) = \min_\lambda \left\{ b^T\lambda : A^T\lambda = c, \lambda \ge 0 \right\}. \quad (D)$$

It is called the problem dual to (P)*.*

♣ **Note:** Our "bounding principle" can be applied to *every* LO program, independently of its format. For example, as applied to the primal LO program

$$\mathrm{Opt}(P) = \max_x \left\{ c^T x : \left\{ \begin{array}{lcll} Px & \underbrace{\le}_{\lambda_\ell} & p & (\ell) \\ Qx & \underbrace{\ge}_{\lambda_g} & q & (g) \\ Rx & \underbrace{=}_{\lambda_e} & r & (e) \end{array} \right\} \right. \qquad (P)$$

it leads to the dual problem in the form of

$$\mathrm{Opt}(D) = \min_{[\lambda_\ell;\lambda_g,\lambda_e]} \left\{ p^T\lambda_\ell + q^T\lambda_g + r^T\lambda_e : \left\{ \begin{array}{l} \lambda_\ell \ge 0, \lambda_g \le 0 \\ P^T\lambda_\ell + Q^T\lambda_g + R^T\lambda_e = c \end{array} \right\} \right. \qquad (D)$$

• *Pay attention to the specific notation: the signs* $\le$, $\ge$, $=$ *of constraints in* (P) *are marked by the associated vectors of Lagrange multipliers* λ_ℓ, λ_g, λ_e*.*

$$\mathrm{Opt}(P) = \max_x \left\{ c^T x : \left\{ \begin{array}{rcll} Px & \leq & p & (\ell) \\ Qx & \geq & q & (g) \\ Rx & = & r & (e) \end{array} \right\} \right. \qquad (P)$$

$$\mathrm{Opt}(D) = \min_{[\lambda_\ell;\lambda_g,\lambda_e]} \left\{ p^T\lambda_\ell + q^T\lambda_g + r^T\lambda_e : \left\{ \begin{array}{l} \lambda_\ell \geq 0, \lambda_g \leq 0 \\ P^T\lambda_\ell + Q^T\lambda_g + R^T\lambda_e = c \end{array} \right\} \right. \qquad (D)$$

LO Duality Theorem: *Consider a primal LO program (P) along with its dual program (D). Then*

(i) [Primal-dual symmetry] *The duality is symmetric: (D) is an LO program, and the program dual to (D) is (equivalent to) the primal problem (P).*

(ii) [Weak duality] *We always have $\mathrm{Opt}(D) \geq \mathrm{Opt}(P)$.*

Attention!*: the latter inequality holds true when (P) is a maximization problem. In general, Weak Duality says that the optimal value in the minimization problem of a primal-dual pair is $\geq$ the optimal value of the maximization problem of the pair.*

(iii) [Strong duality] *The following 3 properties are equivalent to each other:*

- *one of the problems is feasible and bounded*
- *both problems are solvable*
- *both problems are feasible*

and whenever these equivalent to each other properties take place, we have

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

$$\mathrm{Opt}(P) = \max_x \left\{ c^T x : \left\{ \begin{array}{lcll} Px & \leq & p & (\ell) \\ Qx & \geq & q & (g) \\ Rx & = & r & (e) \end{array} \right\} \right. \qquad (P)$$

$$\mathrm{Opt}(D) = \min_{[\lambda_\ell;\lambda_g,\lambda_e]} \left\{ p^T\lambda_\ell + q^T\lambda_g + r^T\lambda_e : \left\{ \begin{array}{l} \lambda_\ell \geq 0, \lambda_g \leq 0 \\ P^T\lambda_\ell + Q^T\lambda_g + R^T\lambda_e = c \end{array} \right\} \right. \qquad (D)$$

Proof of Primal-Dual Symmetry: We rewrite (D) is *exactly* the same form as (P), that is, as

$$-\mathrm{Opt}(D) = \max_{[\lambda_\ell;\lambda_g,\lambda_e]} \left\{ -p^T\lambda_\ell - q^T\lambda_g - r^T\lambda_e : \left\{ \begin{array}{l} \lambda_g \leq 0, \lambda_\ell \geq 0 \\ P^T\lambda_\ell + Q^T\lambda_g + R^T\lambda_e = c \end{array} \right\} \right.$$

and apply the recipe for building the dual, resulting in

$$\min_{[x_\ell;x_g;x_e]} \left\{ c^T x_e : \left\{ \begin{array}{l} x_\ell \geq 0, x_g \leq 0 \\ Px_e + x_g = -p \\ Qx_e + x_\ell = -q \\ Rx_e = -r \end{array} \right\} \right.$$

whence, setting $x_e = -x$ and eliminating x_g and x_e, the problem dual to dual becomes

$$\min_x \left\{ -c^T x : Px \leq p, Qx \leq q, Rx = r \right\}$$

which is equivalent to (P). $\square$

$$\mathrm{Opt}(P) = \max_x \left\{ c^T x : \left\{ \begin{array}{lcll} Px & \leq & p & (\ell) \\ Qx & \geq & q & (g) \\ Rx & = & r & (e) \end{array} \right\} \right. \qquad (P)$$

$$\mathrm{Opt}(D) = \min_{[\lambda_\ell;\lambda_g,\lambda_e]} \left\{ p^T\lambda_\ell + q^T\lambda_g + r^T\lambda_e : \left\{ \begin{array}{l} \lambda_\ell \geq 0, \lambda_g \leq 0 \\ P^T\lambda_\ell + Q^T\lambda_g + R^T\lambda_e = c \end{array} \right\} \right. \qquad (D)$$

Proof of Weak Duality $\mathrm{Opt}(D) \geq \mathrm{Opt}(P)$**:** by construction of the dual.

Proof of Strong Duality

$$\mathrm{Opt}(P) = \max_x \left\{ c^T x : \left\{ \begin{array}{lcll} Px & \leq & p & (\ell) \\ Qx & \geq & q & (g) \\ Rx & = & r & (e) \end{array} \right\} \right. \qquad (P)$$

$$\mathrm{Opt}(D) = \min_{[\lambda_\ell;\lambda_g,\lambda_e]} \left\{ p^T\lambda_\ell + q^T\lambda_g + r^T\lambda_e : \left\{ \begin{array}{l} \lambda_\ell \geq 0, \lambda_g \leq 0 \\ P^T\lambda_\ell + Q^T\lambda_g + R^T\lambda_e = c \end{array} \right\} \right. \qquad (D)$$

Main Lemma: *Let one of the problems (P), (D) be feasible and bounded. Then both problems are solvable with equal optimal values.*

Proof of Main Lemma: By Primal-Dual Symmetry, we can assume w.l.o.g. that the feasible and bounded problem is (P). By what we already know, (P) is solvable. Let us prove that (D) is solvable, and the optimal values are equal to each other.

• Observe that the linear inequality $c^T x \leq \mathrm{Opt}(P)$ is a consequence of the (solvable!) system of constraints of (P). By Inhomogeneous Farkas Lemma

$$\begin{array}{l} \exists \lambda_\ell \geq 0, \lambda_g \leq 0, \lambda_e : \\ P^T\lambda_\ell + Q^T\lambda_g + R^T\lambda_e = c \ \& \ p^T\lambda_\ell + q^T\lambda_g + r^T\lambda_e \leq \mathrm{Opt}(P). \end{array}$$

$\Rightarrow$ *λ is feasible for (D) with the value of dual objective $\leq \mathrm{Opt}(P)$. By Weak Duality, this value should be $\geq \mathrm{Opt}(P)$*

$\Rightarrow$ *the dual objective at λ equals to $\mathrm{Opt}(P)$*

$\Rightarrow$ *λ is dual optimal and $\mathrm{Opt}(D) = \mathrm{Opt}(P)$.* $\square$

Proof of Strong Duality (continued)

Main Lemma $\Rightarrow$ Strong Duality:

- By Main Lemma, if one of the problems (P), (D) is feasible and bounded, then both problems are solvable with equal optimal values
- If both problems are solvable, then both are feasible
- If both problems are feasible, then both are bounded by Weak Duality, and thus one of them (in fact, both of them) is feasible and bounded.

Immediate Consequences

$$\mathrm{Opt}(P) = \max_x \left\{ c^T x : \left\{ \begin{array}{lcll} Px & \leq & p & (\ell) \\ Qx & \geq & q & (g) \\ Rx & = & r & (e) \end{array} \right\} \right. \qquad (P)$$

$$\mathrm{Opt}(D) = \min_{[\lambda_\ell;\lambda_g,\lambda_e]} \left\{ p^T\lambda_\ell + q^T\lambda_g + r^T\lambda_e : \left\{ \begin{array}{l} \lambda_\ell \geq 0, \lambda_g \leq 0 \\ P^T\lambda_\ell + Q^T\lambda_g + R^T\lambda_e = c \end{array} \right\} \right. \qquad (D)$$

Theorem *Whenever at least one of the problems (P), (D) is feasible, we have*

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

Indeed, assuming w.l.o.g. that the feasible problem is (P), observe that

— if (P) is bounded, $\mathrm{Opt}(P) = \mathrm{Opt}(D)$ by Duality Theorem.

— if (P) is unbounded, $\mathrm{Opt}(P) = +\mathrm{inf}$ and (D) is infeasible by Weak Duality, meaning that $\mathrm{Opt}(D) = +\infty$ as well.

Immediate Consequences (continued)

$$\mathrm{Opt}(P) = \max_x \left\{ c^T x : \left\{ \begin{array}{lcll} Px & \leq & p & (\ell) \\ Qx & \geq & q & (g) \\ Rx & = & r & (e) \end{array} \right\} \right. \qquad (P)$$

$$\mathrm{Opt}(D) = \min_{[\lambda_\ell;\lambda_g,\lambda_e]} \left\{ p^T\lambda_\ell + q^T\lambda_g + r^T\lambda_e : \left\{ \begin{array}{l} \lambda_\ell \geq 0, \lambda_g \leq 0 \\ P^T\lambda_\ell + Q^T\lambda_g + R^T\lambda_e = c \end{array} \right\} \right. \qquad (D)$$

♣ **Optimality Conditions in LO:** *Let x and $\lambda = [\lambda_\ell; \lambda_g; \lambda_e]$ be a pair of feasible solutions to (P) and (D). This pair is comprised of optimal solutions to the respective problems*

- [zero duality gap] *if and only if the duality gap, as evaluated at this pair, vanishes:*

$$\begin{array}{rcl} \mathrm{DualityGap}(x,\lambda) & := & [p^T\lambda_\ell + q^T\lambda_g + r^T\lambda_e] - c^T x \\ & = & 0 \end{array}$$

- [complementary slackness] *if and only if the products of all Lagrange multipliers λ_i and the residuals in the corresponding primal constrains are zero:*

$$\forall i : [\lambda_\ell]_i [p - Px]_i = 0 \ \& \ \forall j : [\lambda_g]_j [q - Qx]_j = 0.$$

Proof

We are in the situation when both problems are feasible and thus both are solvable with equal optimal values. Therefore

$$\begin{aligned}\text{DualityGap}(x,\lambda) \;:=\; &\big[[p^T\lambda_\ell + q^T\lambda_g + r^T\lambda_e] - \text{Opt}(D)\big]\\ &+\big[\text{Opt}(P) - c^Tx\big]\end{aligned}$$

For a primal-dual pair of feasible solutions the expressions in the magenta and the red brackets are nonnegative

$\Rightarrow$ *Duality Gap, as evaluated at a primal-dual feasible pair, is nonnegative and can vanish iff both the expressions in the magenta and the red brackets vanish, that is, iff x is primal optimal and λ is dual optimal.*

- Observe that

$$\begin{aligned}&\text{DualityGap}(x,\lambda) = [p^T\lambda_\ell + q^T\lambda_g + r^T\lambda_e] - c^Tx\\ &= [p^T\lambda_\ell + q^T\lambda_g + r^T\lambda_e] - [P^T\lambda_\ell + Q^T\lambda_g + R^T\lambda_2]^Tx\\ &= \lambda_\ell^T[p - Px] + \lambda_g^T[q - Qx] + \lambda_e^T[r - Rx]\\ &= \textstyle\sum_i[\lambda_\ell]_i[p - Px]_i + \sum_j[\lambda_g]_j[q - Qx]_j\end{aligned}$$

Al terms in the resulting sums are nonnegative

$\Rightarrow$ *Duality Gap vanishes iff the complementary slackness holds true.*

Geometry of a Primal-Dual Pair of LO Programs

♣ Consider a primal-dual pair of LO programs

$$\mathrm{Opt}(P) = \max_x \left\{ c^T x : \left\{ \begin{array}{lcll} Px & \underbrace{\leq}_{\lambda_\ell} & p & (\ell) \\ Rx & \underbrace{=}_{\lambda_e} & r & (e) \end{array} \right\} \right. \qquad (P)$$

$$\mathrm{Opt}(D) = \min_{[\lambda_\ell;\lambda_e]} \left\{ p^T\lambda_\ell + r^T\lambda_e : \left\{ \begin{array}{l} \lambda_\ell \geq 0 \\ P^T\lambda_\ell + R^T\lambda_e = c \end{array} \right\} \right. \qquad (D)$$

Standing Assumption: *The systems of linear equations in (P), (D) are solvable:*

$$\exists \bar{x}, \bar{\lambda} = [\bar{\lambda}_\ell; \bar{\lambda}_e] : R\bar{x} = r,\ P^T\bar{\lambda}_\ell + R^T\bar{\lambda}_e = -c$$

♣ **Observation:** Whenever $Rx = r$, we have

$$\begin{array}{rcl} c^T x & = & -[P^T\bar{\lambda}_\ell + R^T\bar{\lambda}_e]^T x = -\bar{\lambda}_\ell^T[Px] - \bar{\lambda}_e^T[Rx] \\ & = & \bar{\lambda}_\ell^T[p - Px] + \left[-\bar{\lambda}_\ell^T p - \bar{\lambda}_e^T r\right] \end{array}$$

$\Rightarrow$ (P) is equivalent to the problem

$$\max_x \left\{ \bar{\lambda}_\ell^T[p - Px] : p - Px \geq 0, Rx = r \right\}.$$

$$\mathrm{Opt}(P) = \max_x \left\{ c^T x : \left\{ \begin{array}{lcll} Px & \underbrace{\leq}_{\lambda_\ell} & p & (\ell) \\ Rx & \underbrace{=}_{\lambda_e} & r & (e) \end{array} \right\} \right. \qquad (P)$$

$$\mathrm{Opt}(D) = \min_{[\lambda_\ell;\lambda_e]} \left\{ p^T\lambda_\ell + r^T\lambda_e : \left\{ \begin{array}{l} \lambda_\ell \geq 0 \\ P^T\lambda_\ell + R^T\lambda_e = c \end{array} \right\} \right. \qquad (D)$$

$[\bar{\lambda}_\ell; \bar{\lambda}_e]$ satisfies equations of (D)

$\Rightarrow$ (P) is equivalent to the problem

$$\max_x \left\{ \bar{\lambda}_\ell^T [p - Px] : p - Px \geq 0, Rx = r \right\}.$$

♠ Passing to the new variable ("primal slack")

$$\xi = p - Px,$$

the primal problem becomes

$$\max_\xi \left\{ \bar{\lambda}_\ell^T \xi : \xi \geq 0, \xi \in \mathcal{M}_P := \bar{\xi} + \mathcal{L}_P \right\}$$

$$\left[\begin{array}{rcl} \mathcal{L}_P & = & \{\xi = Px : Rx = 0\} \\ \bar{\xi} & = & p - P\bar{x} \quad [\bar{x} \text{ solves the equations of } (P)] \end{array} \right]$$

$\mathcal{M}_P$: primal feasible affine plane

$$\mathrm{Opt}(P) = \max_x \left\{ c^T x : \left\{ \begin{array}{rcll} Px & \leq & p & (\ell) \\ Rx & = & r & (e) \end{array} \right\} \right. \qquad (P)$$

$$\mathrm{Opt}(D) = \min_{[\lambda_\ell;\lambda_e]} \left\{ p^T\lambda_\ell + r^T\lambda_e : \left\{ \begin{array}{l} \lambda_\ell \geq 0 \\ P^T\lambda_\ell + R^T\lambda_e = c \end{array} \right\} \right. \qquad (D)$$

♣ Let us express (D) in terms of the *dual slack* λ_ℓ. If $[\lambda_\ell; \lambda_e]$ satisfies the equality constraints in (D), then

$$\begin{array}{l} p^T\lambda_\ell + r^T\lambda_e = p^T\lambda_\ell + [R\bar{x}]^T\lambda_e = p^T\lambda_\ell + \bar{x}^T[R^T\lambda_e] \\ = p^T\lambda_\ell + \bar{x}^T[c - P^T\lambda_\ell] = [p - P\bar{x}]^T\lambda_\ell + \bar{x}^T c \\ = \bar{\xi}^T\lambda_\ell + \bar{x}^T c \end{array}$$

$\Rightarrow$ (D) is equivalent to the problem

$$\min_{\lambda_\ell} \left\{ \bar{\xi}^T\lambda_\ell : \lambda_\ell \geq 0, \lambda_\ell \in \mathcal{M}_D := \mathcal{L}_D - \bar{\lambda}_\ell \right\}$$

$$\mathcal{L}_D = \{\lambda_\ell : \exists \lambda_e : P^T\lambda_\ell + R^T\lambda_e = 0\}$$

$\mathcal{M}_d$: dual feasible affine plane

$$\mathrm{Opt}(P) = \max_x \left\{ c^T x : \left\{ \begin{array}{rcll} Px & \leq & p & (\ell) \\ Rx & = & r & (e) \end{array} \right\} \right. \qquad (P)$$

$$\mathrm{Opt}(D) = \min_{[\lambda_\ell;\lambda_e]} \left\{ p^T\lambda_\ell + r^T\lambda_e : \left\{ \begin{array}{l} \lambda_\ell \geq 0 \\ P^T\lambda_\ell + R^T\lambda_e = c \end{array} \right\} \right. \qquad (D)$$

Bottom line: Problems (P), (D) are equivalent to problems

$$\max_\xi \left\{ \bar{\lambda}_\ell^T \xi : \xi \geq 0,\, \xi \in \mathcal{M}_P := \mathcal{L}_P + \bar{\xi} \right\} \qquad (\mathcal{P})$$

$$\min_{\lambda_\ell} \left\{ \bar{\xi}^T \lambda_\ell : \lambda_\ell \geq 0, \lambda_\ell \in \mathcal{M}_D := \mathcal{L}_D - \bar{\lambda}_\ell \right\} \qquad (\mathcal{D})$$

where

$$\mathcal{L}_P = \{\xi : \exists x : \xi = Px, Rx = 0\},$$
$$\mathcal{L}_D = \{\lambda_\ell : \exists \lambda_e : P^T\lambda_\ell + R^T\lambda_e = 0\}$$

Note:

- Linear subspaces $\mathcal{L}_P$ and $\mathcal{L}_D$ are orthogonal complements of each other
- The minus primal objective $-\bar{\lambda}_\ell$ belongs to the dual feasible plane $\mathcal{M}_D$, and the dual objective $\bar{\xi}$ belongs to the primal feasible plane $\mathcal{M}_P$. Moreover, replacing $\bar{\lambda}_\ell$, $\bar{\xi}$ with any other pair of points from $-\mathcal{M}_D$ and $\mathcal{M}_P$, problems remain essentially intact – on the respective feasible sets, the objectives get constant shifts

Problems (P), (D) are equivalent to problems

$$\max_{\xi}\left\{\bar{\lambda}_\ell^T\xi : \xi \geq 0,\, \xi \in \mathcal{M}_P := \mathcal{L}_P + \bar{\xi}\right\} \qquad (\mathcal{P})$$

$$\min_{\lambda_\ell}\left\{\bar{\xi}^T\lambda_\ell : \lambda_\ell \geq 0, \lambda_\ell \in \mathcal{M}_D := \mathcal{L}_D - \bar{\lambda}_\ell\right\} \qquad (\mathcal{D})$$

where

$$\mathcal{L}_P = \{\xi : \exists x : \xi = Px, Rx = 0\},$$
$$\mathcal{L}_D = \{\lambda_\ell : \exists \lambda_e : P^T\lambda_\ell + R^T\lambda_e = 0\}$$

• A primal-dual feasible pair $(x, [\lambda_\ell; \lambda_e])$ of solutions to (P), (D) induces a pair of feasible solutions $(\xi = p - Px, \lambda_\ell)$ to $(\mathcal{P}, \mathcal{D})$, and

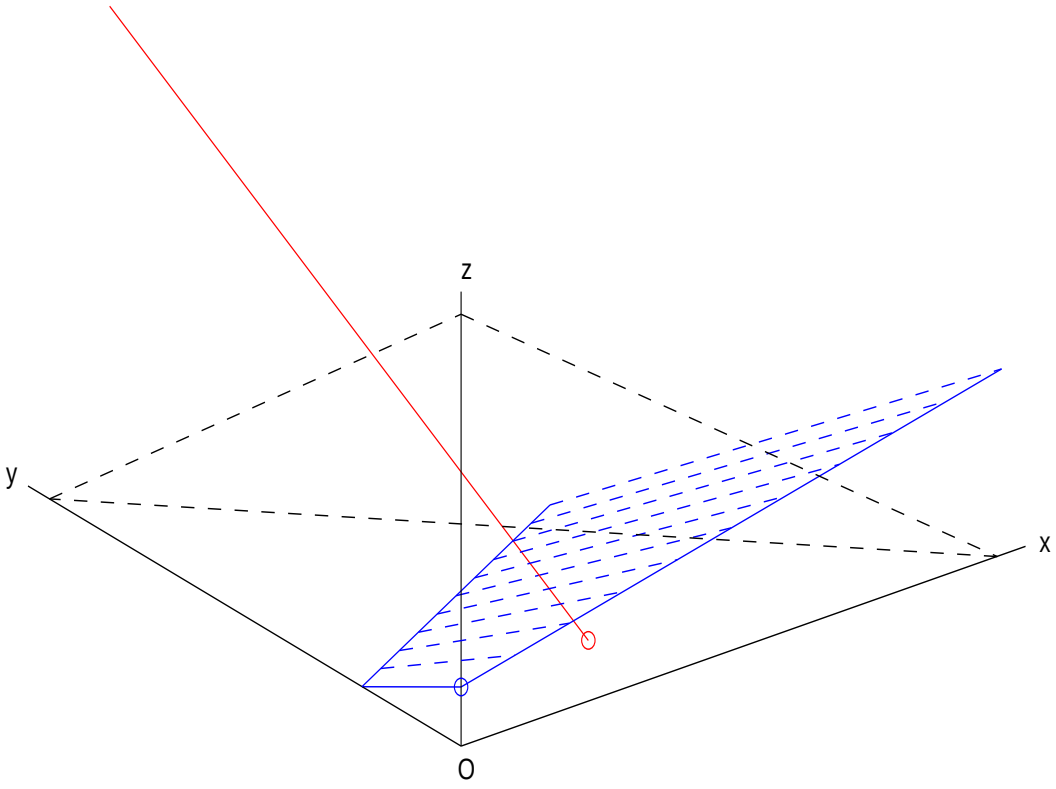
$$\mathrm{DualityGap}(x, [\lambda_\ell, \lambda_e]) = \lambda_\ell^T\xi.$$

Thus, *to solve $(P), (D)$ to optimality is the same as to pick a pair of orthogonal to each other feasible solutions to $(\mathcal{P})$, $(\mathcal{D})$.*

♣ We arrive at a wonderful perfectly symmetric and transparent geometric picture:

Geometrically, a primal-dual pair of LO programs is given by a pair of affine planes $\mathcal{M}_P$ and $\mathcal{M}_D$ in certain $\mathbb{R}^N$; these planes are shifts of linear subspaces $\mathcal{L}_P$ and $\mathcal{L}_D$ which are orthogonal complements of each other.
We intersect $\mathcal{M}_P$ and $\mathcal{M}_D$ with the nonnegative orthant $\mathbb{R}^N_+$, and our goal is to find in these intersections two orthogonal to each other vectors.

♠ Duality Theorem says that this task is feasible if and only if both $\mathcal{M}_P$ and $\mathcal{M}_D$ intersect the nonnegative orthant.



Geometry of primal-dual pair of LO programs:

Blue area: feasible set of $(\mathcal{P})$ — intersection of the 2D primal feasible plane $\mathcal{M}_P$ with the nonnegative orthant $\mathbb{R}^3_+$.

Red segment: feasible set of $(\mathcal{D})$ — intersection of the 1D dual feasible plane $\mathcal{M}_D$ with the nonnegative orthant $\mathbb{R}^3_+$.

Blue dot: primal optimal solution ξ^*.

Red dot: dual optimal solution λ^*_ℓ.

Pay attention to orthogonality of the primal solution (which is on the z-axis) and the dual solution (which is in the xy-plane).

The Cost Function of an LO program, I

♣ Consider an LO program

$$\mathrm{Opt}(b) = \max_x \left\{ c^T x : Ax \le b \right\}. \qquad (P[b])$$

Note: we treat the data A, c as fixed, and b as varying, and are interested in the properties of the optimal value $\mathrm{Opt}(b)$ as a function of b.

♠ **Fact:** *When b is such that $(P[b])$ is feasible, the property of problem to be/not to be bounded is independent of the value of b.*

Indeed, a feasible problem $(P[b])$ is unbounded iff there exists d: $Ad \le 0, c^T d > 0$, and this fact is independent of the particular value of b.

Standing Assumption: There exists b such that $P([b])$ is feasible and bounded

$\Rightarrow$ $P([b])$ is bounded whenever it is feasible.

Theorem *Under Assumption, $-\mathrm{Opt}(b)$ is a polyhedrally representable function with the polyhedral representation*

$$\begin{aligned} &\{[b;\tau] : -\mathrm{Opt}(b) \le \tau\} \\ &= \{[b;\tau] : \exists x : Ax \le b, -c^T x \le \tau\}. \end{aligned}$$

The function $\mathrm{Opt}(b)$ is monotone in b:

$$b' \le b'' \Rightarrow \mathrm{Opt}(b') \le \mathrm{Opt}(b'').$$

$$\mathrm{Opt}(b) = \max_x \left\{ c^T x : Ax \le b \right\}. \qquad (P[b])$$

♠ Additional information can be obtained from Duality. The problem dual to $(P[b])$ is

$$\min_\lambda \left\{ b^T \lambda : \lambda \ge 0, A^T \lambda = c \right\}. \qquad (D[b])$$

By LO Duality Theorem, under our Standing Assumption $(D[b])$ *is feasible for every* b*, and*

$$\mathrm{Opt}(b) = \min_\lambda \left\{ b^T \lambda : \lambda \ge 0, A^T \lambda = c \right\}. \qquad (*)$$

Observation: *Let* $\bar{b}$ *be such that* $\mathrm{Opt}(\bar{b}) > -\infty$*, so that* $(D[\bar{b}])$ *is solvable, and let* $\bar{\lambda}$ *be an optimal solution to* $(D[\bar{b}])$*. Then* $\bar{\lambda}$ *is a supergradient of* $\mathrm{Opt}(b)$ *at* $b = \bar{b}$*, meaning that*

$$\forall b : \mathrm{Opt}(b) \le \mathrm{Opt}(\bar{b}) + \bar{\lambda}^T [b - \bar{b}]. \qquad (!)$$

Indeed, by $(*)$ we have $\mathrm{Opt}(\bar{b}) = \bar{\lambda}^T \bar{b}$ and $\mathrm{Opt}(b) \le \bar{\lambda}^T b$, that is,

$\mathrm{Opt}(b) \le \bar{\lambda}^T \bar{b} + \bar{\lambda}^T [b - \bar{b}] = \mathrm{Opt}(\bar{b}) + \bar{\lambda}^T [b - \bar{b}]$. □

$$\begin{array}{rcll}\mathrm{Opt}(b) & = & \max\limits_x\left\{c^Tx : Ax\le b\right\} & (P[b])\\ & = & \min\limits_\lambda\left\{b^T\lambda : \lambda\ge 0, A^T\lambda = c\right\} & (D[b])\end{array}$$

$$\mathrm{Opt}(\bar b) > -\infty,\ \bar\lambda\in \mathrm{Argmin}_\lambda\{\bar b^T\lambda : \lambda\ge 0, A^T\lambda = c\}$$

$$\Rightarrow \mathrm{Opt}(b)\le \mathrm{Opt}(\bar b) + \bar\lambda[b-\bar b] \quad (!)$$

$\mathrm{Opt}(b)$ is a polyhedrally representable and thus piecewise linear function with the *full-dimensional* domain:

$$\mathrm{Dom}\,\mathrm{Opt}(\cdot) = \{b : \exists x : Ax\le b\}.$$

Representing the feasible set $\Lambda = \{\lambda : \lambda\ge 0, A^T\lambda = c\}$ of $(D[b])$ as

$$\Lambda = \mathrm{Conv}(\{\lambda_1,...,\lambda_N\}) + \mathrm{Cone}\,(\{r_1,...,r_M\})$$

we get

$$\begin{array}{c}\mathrm{Dom}\,\mathrm{Opt}(b) = \{b : b^Tr_j\ge 0,\ 1\le j\le M\},\\ b\in\mathrm{Dom}\,\mathrm{Opt}(b)\Rightarrow \mathrm{Opt}(b) = \min\limits_{1\le i\le m}\lambda_i^Tb\end{array}$$

$$\begin{aligned}\mathrm{Opt}(b) &= \max_x \left\{c^T x : Ax \le b\right\} && (P[b])\\ &= \min_\lambda \left\{b^T\lambda : \lambda \ge 0, A^T\lambda = c\right\} && (D[b])\end{aligned}$$

$$\mathrm{Opt}(\bar{b}) > -\infty,\ \bar{\lambda} \in \mathrm{Argmin}_\lambda\{\bar{b}^T\lambda : \lambda \ge 0, A^T\lambda = c\}$$

$$\Rightarrow \mathrm{Opt}(b) \le \mathrm{Opt}(\bar{b}) + \bar{\lambda}[b - \bar{b}] \quad (!)$$

$$\mathrm{Dom}\,\mathrm{Opt}(b) = \{b : b^T r_j \ge 0,\ 1 \le j \le M\},$$
$$b \in \mathrm{Dom}\,\mathrm{Opt}(b) \Rightarrow \mathrm{Opt}(b) = \min_{1\le i\le m} \lambda_i^T b$$

$\Rightarrow$ *Under our Standing Assumption,*

- $\mathrm{Dom}\,\mathrm{Opt}(\cdot)$ *is a full-dimensional polyhedral cone,*
- *Assuming w.l.o.g. that* $\lambda_i \ne \lambda_j$ *when* $i \ne j$*, the finitely many hyperplanes* $\{b : \lambda_i^T b = \lambda_j^T b\}$, $1 \le i < j \le N$, *split this cone into finitely many cells, and in the interior of every cell* $\mathrm{Opt}(b)$ *is a linear function of* b*.*
- *By* $(!)$*, when* b *is in the interior of a cell, the optimal solution* $\lambda(b)$ *to* $(D[b])$ *is unique, and* $\lambda(b) = \nabla\mathrm{Opt}(b)$*.*

Law of Diminishing Marginal Returns

♠ Consider a function of the form

$$\mathrm{Opt}(\beta) = \max_x \left\{ c^T x : Px \le p, q^T x \le \beta \right\} \qquad (P_\beta)$$

Interpretation: x is a production plan, $q^T x$ is the price of resources required by x, β is our investment in the resources, $\mathrm{Opt}(\beta)$ is the maximal return for an investment β.

♠ As above, for β such that (P_β) is feasible, the problem is either always bounded, or is always unbounded. Assume that the first is the case. Then

- The domain $\mathrm{Dom}\,\mathrm{Opt}(\cdot)$ of $\mathrm{Opt}(\cdot)$ is a nonempty ray $\underline{\beta} \le \beta < \infty$ with $\underline{\beta} \ge -\infty$, and
- $\mathrm{Opt}(\beta)$ is nondecreasing and *concave*.

Monotonicity and concavity imply that if

$$\underline{\beta} \le \beta_1 < \beta_2 < \beta_3,$$

then

$$\frac{\mathrm{Opt}(\beta_2) - \mathrm{Opt}(\beta_1)}{\beta_2 - \beta_1} \ge \frac{\mathrm{Opt}(\beta_3) - \mathrm{Opt}(\beta_2)}{\beta_3 - \beta_2},$$

that is, *the reward for an extra \$1 in the investment can only decrease* (or remain the same) *as the investment grows.* In Economics, this is called *the law of diminishing marginal returns.*

The Cost Function of an LO program, II

♣ Consider an LO program

$$\mathrm{Opt}(c) = \max_x \left\{ c^T x : Ax \le b \right\}. \qquad (P[c])$$

Note: we treat the data A, b as fixed, and c as varying, and are interested in the properties of $\mathrm{Opt}(c)$ as a function of c.

Standing Assumption: $(P[\cdot])$ is feasible (this fact is independent of the value of c).

Theorem *Under Assumption, $\mathrm{Opt}(c)$ is a polyhedrally representable function with the polyhedral representation*

$$\begin{aligned} &\{[c;\tau] : \mathrm{Opt}(c) \le \tau\} \\ &= \{[c;\tau] : \exists \lambda : \lambda \ge 0, A^T\lambda = c, b^T\lambda \le \tau\}. \end{aligned}$$

Proof. Since $(P[c])$ is feasible, by LO Duality Theorem the program is solvable if and only if the dual program

$$\min_\lambda \{b^T\lambda : \lambda \ge 0, A^T\lambda = c\} \qquad (D[c])$$

is feasible, and in this case the optimal values of the problems are equal

$\Rightarrow$ $\tau \ge \mathrm{Opt}(c)$ iff $(D[c])$ has a feasible solution with the value of the objective $\le \tau$. $\square$

$$\mathrm{Opt}(c) = \max_x \left\{ c^T x : Ax \le b \right\}. \qquad (P[c])$$

Theorem *Let $\bar{c}$ be such that $\mathrm{Opt}(\bar{c}) < \infty$, and $\bar{x}$ be an optimal solution to $(P[\bar{c}])$. Then $\bar{x}$ is a subgradient of $\mathrm{Opt}(\cdot)$ at the point $\bar{c}$:*

$$\forall c : \mathrm{Opt}(c) \ge \mathrm{Opt}(\bar{c}) + \bar{x}^T [c - \bar{c}]. \qquad (!)$$

Proof: We have $\mathrm{Opt}(c) \ge c^T \bar{x} = \bar{c}^T \bar{x} + [c - \bar{c}]^T \bar{x} = \mathrm{Opt}(\bar{c}) + \bar{x}^T [c - \bar{c}]$. □

♠ Representing

$$\{x : Ax \le b\} = \mathrm{Conv}(\{v_1, ..., v_N\}) + \mathrm{Cone}\,(\{r_1, ..., r_M\}),$$

we see that

- $\mathrm{Dom}\,\mathrm{Opt}(\cdot) = \{c : r_j^T c \le 0,\ 1 \le j \le M\}$ is a polyhedral cone, and
- $c \in \mathrm{Dom}\,\mathrm{Opt}(\cdot) \Rightarrow \mathrm{Opt}(c) = \max\limits_{1 \le i \le N} v_i^T c$.

In particular, *if $\mathrm{Dom}\,\mathrm{Opt}(\cdot)$ is full-dimensional and v_i are distinct from each other, everywhere in $\mathrm{Dom}\,\mathrm{Opt}(\cdot)$ outside finitely many hyperplanes $\{c : v_i^T c = v_j^T c\}$, $1 \le i < j \le N$, the optimal solution $x = x(c)$ to $(P[c])$ is unique and $x(c) = \nabla \mathrm{Opt}(c)$.*

♣ Let $X = \{x \in \mathbb{R}^n : Ax \leq b\}$ be a *nonempty* polyhedral set. The function

$$\mathrm{Opt}(c) = \max_{x \in X} c^T x : \mathbb{R}^n \to \mathbb{R} \cup \{+\infty\}$$

has a name - it is called the *support function* of X. Along with already investigated properties of the support function, an important one is as follows:

♠ *The support function of a nonempty polyhedral set X "remembers" X: if*

$$\mathrm{Opt}(c) = \max_{x \in X} c^T x,$$

then

$$X = \{x \in \mathbb{R}^n : c^T x \leq \mathrm{Opt}(c) \ \forall c\}.$$

Proof

Let $X^+ = \{x \in \mathbb{R}^n : c^T x \leq \mathsf{Opt}(c)\, \forall c\}$. We clearly have $X \subset X^+$. To prove the inverse inclusion, let $\bar{x} \in X^+$; we want to prove that $x \in X$. To this end let us represent $X = \{x \in \mathbb{R}^n : a_i^T x \leq b_i, 1 \leq i \leq m\}$. For every i, we have

$$a_i^T \bar{x} \leq \mathsf{Opt}(a_i) \leq b_i,$$

and thus $\bar{x} \in X$. □

Quiz: What are the support functions of

- Unit box $\{x \in \mathbb{R}^n : \|x\|_\infty := \max_i |x_i| \leq 1\}$?
- Unit $\|\cdot\|_1$ ball $\{x \in \mathbb{R}^n : \|x\|_1 := \sum_i |x_i| \leq 1\}$?

Antagonistic Games

♣ Consider the situation as follows. Given are

- two nonempty sets $X \subset \mathbb{R}^n$, $\Lambda \subset \mathbb{R}^m$
- real-valued *cost function* $\phi(x, \lambda) : X \times \Lambda \to \mathbb{R}$

These data define a *game* of two players, A (you) and B (me). A selects a point $x \in X$, and B selects a point $x \in X$. As a result of these choices I (B) pay to you (A) the sum $\phi(x, \lambda)$.

Naturally, I am interested to minimize my payment, and you are interested to maximize it.

How should we act ?

♠ **I.** Assume that you make your selection first, and I know your choice when making my selection. When you select $x \in X$, you should be ready to get as low as

$$\underline{\phi}(x) = \inf_{\lambda \in \Lambda} \phi(x, \lambda),$$

and your natural policy is *to maximize your worst-case profit by selecting as x an optimal solution to the primal problem*

$$\mathrm{Opt}(\mathcal{P}) = \sup_{x \in X} \left[\underline{\phi}(x) := \inf_{\lambda \in \Lambda} \phi(x, \lambda) \right] \qquad (\mathcal{P})$$

♠ **II.** Now assume that I make my selection first, and you know it when making your selection. When selecting λ, I should be ready to pay as much as

$$\overline{\phi}(\lambda) = \sup_{x \in X} \phi(x, \lambda),$$

and my natural policy is *to minimize my worst-case loss by selecting as λ an optimal solution to the dual problem*

$$\mathrm{Opt}(\mathcal{D}) = \inf_{\lambda \in \Lambda} \left[\overline{\phi}(\lambda) := \sup_{x \in X} \phi(x, \lambda) \right] \qquad (\mathcal{D})$$

$$\begin{array}{rcll}\mathrm{Opt}(\mathcal{P}) & = & \sup_{x\in X}\left[\underline{\phi}(x) := \inf_{\lambda\in\Lambda}\phi(x,\lambda)\right] & (\mathcal{P})\\ \mathrm{Opt}(\mathcal{D}) & = & \inf_{\lambda\in\Lambda}\left[\overline{\phi}(\lambda) := \sup_{x\in X}\phi(x,\lambda)\right] & (\mathcal{D})\end{array}$$

• Intuitively, the situation when I make my selection first and you know it when making your selection is *worse* for me than the one when you select your choice first and I know it when making my selection

$\Rightarrow$ *We can guess that*

$$\mathrm{Opt}(\mathcal{P}) := \sup_{x\in X}\inf_{\lambda\in\Lambda\in X}\phi(x,\lambda) \le \inf_{\lambda\in\Lambda}\sup_{x\in X}\phi(x,\lambda) =: \mathrm{Opt}(\mathcal{D})$$

This guess is indeed true, and is called *Weak Duality*.

(?) *What is natural behavior of the players when they are making their selections simultaneously, knowing only "game's data" $X, \Lambda, \phi(\cdot,\cdot)$, but not knowing the selection of the adversary?*

♠ **A natural answer** is offered by the notion of Nash Equilibrium: A pair of choices $x_* \in X$, $\lambda_* \in \Lambda$ such that *whenever one of the players sticks to x_* (or to λ_*), the other cannot gain when deviating from λ_* (respectively, x_*):*

$$\forall(x \in X, \lambda \in \Lambda) : \phi(x_*,\lambda) \ge \phi(x_*,\lambda_*) \ge \phi(x,\lambda_*)$$

*Points $(x_*,\lambda_*) \in X \times \Lambda$ with this property are called saddle points (*max *in $x \in X$,* min *in $\lambda \in \Lambda$) of $\phi(x,\lambda)$.*

♠ **Theorem.** *Saddle points exist iff both $(\mathcal{P})$ and $(\mathcal{D})$ are solvable with equal optimal values: $\mathrm{Opt}(\mathcal{P}) = \mathrm{Opt}(\mathcal{D})$. In this case, saddle points are exactly points (x_*,λ_*) comprised of optimal solutions to $(\mathcal{P})$ and $(\mathcal{D})$, and for every such point it holds*

$$\phi(x_*,\lambda_*) = \mathrm{Opt}(\mathcal{P}) = \mathrm{Opt}(\mathcal{D})$$

Example: Lagrange Duality in LO

♠ *Consider a primal-dual pair LO programs*

$$\mathrm{Opt}(P) = \max_{x\in\mathbb{R}^n}\left\{c^Tx : \left\{\begin{array}{rcll} Px & \le & p & (\ell)\\ Rx & = & r & (e)\end{array}\right\}\right. \qquad (P)$$

$$\mathrm{Opt}(D) = \min_{[\lambda_\ell;\lambda_e]}\left\{p^T\lambda_\ell + r^T\lambda_e : \left\{\begin{array}{l}\lambda_\ell \ge 0\\ P^T\lambda_\ell + R^T\lambda_e = c\end{array}\right\}\right. \qquad (D)$$

and let us associate with the primal problem (P) *its Lagrange function*

$$L(x,\lambda) = c^Tx - \lambda_\ell^T[Px-p] - \lambda_e[Rx-r]$$

with x *(primal variable) varying in* $X = \mathbb{R}^n$ *and the Lagrange multipliers* $\lambda = [\lambda_\ell;\lambda_e]$ *varying in*

$$\Lambda = \{[\lambda_\ell;\lambda_e] : \lambda_\ell \ge 0\}$$

(?) *What are the primal and the dual problems associated with* X, Λ *and the cost function* L *?*

What are the saddle points?

♠ *We have*

$$\underline{L}(x) := \inf_{\lambda_\ell\ge 0,\lambda_e}\left[c^Tx + \lambda_\ell{}^T[p-Px] + \lambda_e{}^T[r-Rx]\right]$$
$$= c^Tx + \inf_{\lambda_\ell\ge 0}\left[\lambda_\ell^T[p-Px]\right] + \inf_{\lambda_e}^T\left[\lambda_e^T[r-Rx]\right]$$
$$= \left\{\begin{array}{ll} c^Tx, & Px\le p \text{ and } Rx = r\\ -\infty, & \text{otherwise}\end{array}\right.$$

$$\overline{L}(\lambda_\ell,\lambda_e) := \sup_x\left[c^Tx + \lambda_\ell{}^T[p-Px] + \lambda_e{}^T[r-Rx]\right]$$
$$= \sup_x\left[[c-P^T\lambda_\ell - R^T\lambda_e]^Tx + p^T\lambda_\ell + r^T\lambda_e{}^T\right]$$
$$= \left\{\begin{array}{ll} p^T\lambda_\ell + r^T\lambda_e, & P^T\lambda_\ell + R^T\lambda_e = c\\ +\infty, & \text{otherwise}\end{array}\right.$$

$$\mathrm{Opt}(P) = \max_{x\in\mathbb{R}^n}\left\{c^Tx : \left\{\begin{array}{rcll} Px & \le & p & (\ell)\\ Rx & = & r & (e)\end{array}\right\}\right. \qquad (P)$$

$$\mathrm{Opt}(D) = \min_{[\lambda_\ell;\lambda_e]}\left\{p^T\lambda_\ell + r^T\lambda_e : \left\{\begin{array}{l}\lambda_\ell \ge 0\\ P^T\lambda_\ell + R^T\lambda_e = c\end{array}\right\}\right. \qquad (D)$$

$\Rightarrow$ *"Largange game" on the domains*

$$X = \mathbb{R}^n,\ \Lambda = \{[\lambda_\ell;\lambda_e] : \lambda_\ell \ge 0\}$$

with cost function

$$L(x,\lambda) = c^Tx - \lambda_\ell^T[Px-p] - \lambda_e[Rx-r]$$

$$\Rightarrow \left\{\begin{array}{l} \underline{L}(x) = \left\{\begin{array}{ll} c^Tx, & Px \le p \text{ and } Rx = r\\ -\infty, & \text{otherwise}\end{array}\right.\\ \overline{L}(\lambda_\ell,\lambda_e) = \left\{\begin{array}{ll} p^T\lambda_\ell + r^T\lambda_e, & P^T\lambda_\ell + R^T\lambda_e = c\\ +\infty, & \text{otherwise}\end{array}\right.\end{array}\right.$$

Conclusion: *The primal and the dual problems associated with our "Lagrange game"*

$$\begin{array}{rcll}\mathrm{Opt}(\mathcal{P}) & = & \max_x \underline{L}(x) & (\mathcal{P})\\ \mathrm{Opt}(\mathcal{D}) & = & \min_{\lambda=[\lambda_\ell;\lambda_e]\in\Lambda} \overline{L}(\lambda) & (\mathcal{D})\end{array}$$

are nothing but (P) and (D) (in slight disguise). $\Rightarrow$ Lagrange game has saddle points if and only if (P) and (D) are solvable with equal optimal values which, by Duality Theorem, takes place iff both (P) and (D) are solvable, same as iff one of the problems (P), (D) is solvable. Whenever this is the case, saddle points of the Lagrange game are exactly the primal-dual optimal pairs of (P), (D).

$$\mathrm{Opt}(P) = \max_{x\in\mathbb{R}^n}\left\{c^Tx : \left\{\begin{array}{rcll} Px & \le & p & (\ell)\\ Rx & = & r & (e)\end{array}\right\}\right. \qquad (P)$$

$$\mathrm{Opt}(D) = \min_{[\lambda_\ell;\lambda_e]}\left\{p^T\lambda_\ell + r^T\lambda_e : \left\{\begin{array}{l} \lambda_\ell \ge 0\\ P^T\lambda_\ell + R^T\lambda_e = c\end{array}\right\}\right. \qquad (D)$$

♠ *Let (P), (D) be solvable, so that the primal-dual optimal solutions $(x^*, \lambda^* = [\lambda^*_\ell; \lambda^*_e])$ are exactly the saddle points of the Lagrange function*

$$\begin{array}{rcl} L(x,\lambda) & = & c^Tx - \lambda_\ell^T[Px-p] - \lambda_e^T[Rx-r]\\ & = & x^T[c - P^T\lambda_\ell - R^T\lambda_e] + p^T\lambda_\ell + r^T\lambda_e\end{array}$$

*(*min *in $\lambda = [\lambda_\ell; \lambda_e]$ with $\lambda_\ell \ge 0$,* max *in $x \in \mathbb{R}^n$).*

Question: *When $(x^*, \lambda^* = [\lambda^*_\ell; \lambda^*_e])$ is a saddle point of the Lagrange function?*

Answer: *We should have*

- $\lambda^*_\ell \ge 0$
- *$L(x, \lambda^*)$ as a function of $x \in \mathbb{R}^n$ should attain its maximum in x at x^*, which is the case iff the KKT equation*

$$P^T\lambda^*_\ell + R^T\lambda^*_\ell = c$$

takes place

- *$L(x^*, \lambda)$ as a function of $\lambda = [\lambda_\ell; \lambda_e]$ with $\lambda_\ell \ge 0$ should attain its minimum at λ^*, which is the case iff*

$$Rx^* = r,\ Px^* \le p \ \text{ and } \ [\lambda^*_\ell]^T[Px^* - p] = 0.$$

♡ *We have recovered KKT optimality conditions in LO!*

Polyhedral Games

♣ "Lagrange game" is a very special case of *polyhedral game*, where both X and Λ are polyhedral sets and the cost function is *bilinear*, that is, of the form

$$\phi(x, \lambda) = p^T x + q^T \lambda + \lambda^T R x$$

(?) *What can be said on saddle points of a general polyhedral game*

♠ Let $X = \{x \in \mathbb{R}^n : Ax \leq b\}$ and $\Lambda = \{\lambda \in \mathbb{R}^m : C\lambda \leq d\}$.

Standing assumption: *Both X and Λ are nonempty and bounded.*

♠ **Fact:** *Under Standing assumption, a polyhedral game reduces to LO and has saddle points.*

$$X = \{x : Ax \leq b\}, \Lambda = \{\lambda : C\lambda \leq d\}, \phi(x, \lambda) = p^T x + q^T \lambda + \lambda^T R x$$

Explanation: We have

$$\begin{aligned}
&\underline{\phi}(x) = \inf_{\lambda: C\lambda \leq d} \left[p^T x + q^T \lambda + \lambda^T R x\right] \\
&= p^T x + \min_\lambda \left\{[q + Rx]^T \lambda : C\lambda \leq d\right\} \\
&= p^T x - \max_\lambda \left\{[-q - Rx]^T \lambda : C\lambda \leq d\right\} \\
&= p^T x - \min_w \left\{d^T w : w \geq 0, C^T w + q + Rx = 0\right\} \\
&\qquad\qquad \text{[Duality; note that } \{\lambda : C^T \lambda \leq d\} \neq \emptyset\text{]} \\
&= p^T x + \max_w \left\{-d^T w : w \geq 0, C^T w + q + Rx = 0\right\}
\end{aligned}$$

$\Rightarrow$ Problem $(\mathcal{P})$ of maximizing $\underline{\phi}(x)$ over $x \in X$ is nothing but the LO program

$$\max_{x,w} \left\{p^T x - d^T w : Ax \leq b, w \geq 0, C^T w + Rx = -q\right\} \qquad (!)$$

both $(\mathcal{P})$ and (!) are solvable/unsolvable simultaneously, and optimal solutions to $(\mathcal{P})$ are exactly the x-components of optimal solutions to (!).

Note: $\Lambda = \{\lambda : C\lambda \leq d\}$ is nonempty and bounded, whence $\underline{\phi}$ is a finite everywhere polyhedral function. Taking into account that X is nonempty and bounded, the problem $(\mathcal{P})$ of maximizing $\underline{\phi}$ over X is solvable, whence (!) is solvable as well.

- By completely similar reasoning, *problem $(\mathcal{D})$ of minimizing over $\lambda \in \Lambda$ the function $\overline{\phi}(\lambda) = \sup_{x \in X} \phi(x, \lambda)$ is nothing but the LO program*

$$\min_{\lambda,z} \left\{b^T z + q^T \lambda : C\lambda \underbrace{\leq}_{w} d, z \geq 0, -A^T z + R^T \lambda \underbrace{=}_{x} -p\right\} \qquad (!!)$$

It is immediately seen that (!!) is the dual of the *solvable* problem (!)

$\Rightarrow$ (!), (!!) are solvable with equal optimal values

$\Rightarrow$ *$(\mathcal{P})$ and $(\mathcal{D})$ are solvable with equal optimal values*

$\Rightarrow$ *Saddle points exist and are of the form (x^*, λ^*), where x^* is a component of an optimal solution to (!), and λ^* is a component of an optimal solution to (!!).*

Application: von Neumann Lemma

♣ Let $X = \{x \in \mathbb{R}^n : Ax \leq b\}$ be a *nonempty and bounded* polyhedral set, let

$$f_i(x) = b_i^T x + c_i,\ i = 1, ..., m$$

be a finite collection of *affine* functions on X.

Consider the *maximin* problem

$$\mathrm{Opt} = \max_{x \in X} \left[f(x) := \min_{i=1,...,m} f_i(x) \right]$$

Observation: *The problem is nothing but the primal problem* $(\mathcal{P})$ *associated with the polyhedral game where*

$$X = \{x \in \mathbb{R}^n : Ax \leq b\}, \Lambda = \{\lambda \in \mathbb{R}^m : \lambda \geq 0, \textstyle\sum_i \lambda_i = 1\},$$
$$\phi(x, \lambda) = \textstyle\sum_i \lambda_i f_i(x) = \sum_i \lambda_i [b_i^T x + c_i]$$

Indeed,

$$\underline{\phi}(x) = \min_{\lambda} \left\{ \sum_i \lambda_i f_i(x) : \lambda \geq 0, \sum_i \lambda_i = 1 \right\} = \min_{1 \leq i \leq m} f_i(x) = f(x).$$

• By the above, the resulting game has a saddle point x^*, λ^*, implying that *for some* $\lambda^* \in \Lambda$,

$$\max_{x \in X} \textstyle\sum_i \lambda_i^* f_i(x) = \overline{\phi}(\lambda^*) = \mathrm{Opt}(\mathcal{D})$$
$$= \mathrm{Opt}(\mathcal{P}) = \max_{x \in X} f(x) = \max_{x \in X} \left[\min_{1 \leq i \leq m} f_i(x) \right]$$

In words: *Maximum of the minimum of several affine functions taken over a bounded and nonempty polyhedral domain in the space of arguments, is the same as the maximum over the same domain of a properly selected convex combination of these functions.*

Really surprising – *any* convex combination of a finite collection of functions is, at every point, $\geq$ the smallest of these functions!

Matrix Games and Mixed Strategies

♣ Consider a game with *finite set* $\{1, ..., m\}$ of your choices and *finite set* $\{1, ..., n\}$ of my choices. In this case the cost function ϕ can be identified with $m \times n$ matrix $M = [M_{ij} = \phi(i, j)]$. The resulting *matrix game* is as follows:

Two players – you and me – are given an $m \times n$ matrix M. You select a row, I select a column; when you select row i, and I - column j, your win (and my loss) is M_{ij}.

♠ In a matrix game, saddle points are pairs $(\overline{i}, \overline{j})$ such that the entry $M_{\overline{i},\overline{j}}$ of the game matrix M is the largest in its column and the smallest in its row, like element $M_{2,3}$ in the matrix

$$M = \begin{bmatrix} 1 & 2 & 3 & 4 \\ 9 & 7 & 7 & 8 \\ 9 & 10 & 4 & 12 \end{bmatrix}$$

However: Existence of a saddle point in a matrix is a "rare commodity." For example, the matrix

$$M = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

has no saddle point.

Quiz: *Assume an $n \times n$ game matrix is selected at random, with entries assigned the values 0 and 1 with probability 0.5 independently across the entries. What is the probability $p(n)$ for the game to have a saddle point?*

n	2	4	8	16	32
$p(n)$	0.875	≈0.4547	≈0.060		

Quiz: *Assume an $n \times n$ game matrix is selected at random, with entries assigned the values 0 and 1 with probability 0.5 independently across the entries. What is the probability $p(n)$ for the game to have a saddle point?*

A simple *upper bound* on $p(n)$ is $2n^2/2^n$ (why?) resulting in

n	2	4	8	16	32
$p(n)$	0.875	≈0.4547	≈0.060	<0.008	<4.8e-7

(?) *What to do if a matrix game does not have a saddle point?*

Partial answer [von Neumann and Morgenstern, late 1940's]: *pass to mixed strategies.*

♣ Imagine players are playing the game round by round and are interested in their average outcomes over large time horizon. In this case, they could use *randomized strategies* as follows

• you select a probability distribution x on the set $\{1,...,m\}$ of your choices; x is just a nonnegative m-dimensional vector with unit sum of entries:

$$\sum_{i=1}^{m} x_i = 1 \quad .$$

In every round, you draw your choice $\imath \in \{1,...,m\}$ from this distribution, so that the probability to select 1 is x_1, the probability to select 2 is x_2, etc.

• likewise, I select a probability distribution on the set $\{1,...,n\}$ of my choices – a nonnegative n-dimensional vector λ with entries summing up to 1, and in every round draw my choice from $\{1,...,n\}$ at random according to this distribution.

♠ With the outlined *mixed strategies*, the probability in a particular round the choices to be (i,j) is $x_i\lambda_j$, and your *expected* win (my expected loss) will be

$$\sum_{i=1}^{m}\sum_{j=1}^{n} M_{ij}x_i\lambda_j = x^T M\lambda$$

You are interested to maximize your expected win over x, and I am interested to minimize it over λ.

♠ We arrive at *matrix game in mixed strategies* given by

$$X = \{x \in \mathbb{R}^m : x \geq 0, \sum_i x_i = 1\},$$
$$\Lambda = \{\lambda \in \mathbb{R}^n : \lambda \geq 0, \sum_j \lambda_j = 1\},$$
$$\phi(x, \lambda) = x^T M \lambda = \sum_{i,j} M_{ij} x_i \lambda_j$$

Note: Matrix game in fixed strategies is a game with polyhedral, nonempty and bounded X, Λ and with a bilinear cost function

$\Rightarrow$ *In mixed strategies, Nash equilibrium (a.k.a. saddle point) always exists!*

Informal Comment: In a "you or him" situation, it is crucial to keep your intended actions secret from your adversary. Mixed strategy is the ultimate implementation of this principle: *you yourself do not know what you will do tomorrow!*

Quiz: Bankrupt the Banker!

• Alicubi is a small African country. The currency there is Alicubi dollars (AD's).

♠ **Advertizement** (*Alicubi Evening News,* January 7, 2014)

If you are smart, you definitely will earn at least AD 99 by playing 12 rounds of the game Bankrupt the Banker!

Terms and conditions:

• At the beginning, 4 playing cards of 4 different suits are shuffled and placed in line in front of you, backs up. Their order is never changed later.

• In every round, you point at a card in the row. The banker

--- takes the pointed card and looks at its suit

--- tells you your win in the round,

--- returns the card to its place, still back up.

• Your win in the round is determined by the suit of the card you have selected and Banker's decision in the round. The list of legitimate Banker's decisions and the dependence of your win on card's suit and Banker's decision remain the same in all rounds.

• Rules of the game guarantee that if you are smart enough, your total win in 12 rounds will be at least AD 99.

♠ Assuming the advertizement truthful, are you ready to play? How would you play?

♠ **Explanation:** When translating the advertizement from natural language to Math, it reads as follows:

• In the nature there exists a matrix $M = [M_{ij}]_{i,j}$ with 4 rows indexed by the 4 suits, and N columns indexed by Banker's decisions.

In a round where you select card with suit i and Banker selects decision j, you win is M_{ij}.

• All you know about M is that M has 4 rows and that there exists a policy specifying your selections in such a way that *independently of the results of initial shuffling and of Banker's behavior, your total win in 12 rounds will be at least 99.*

The problem is to find this policy.

The solution. Let $a_i = \min\limits_{1 \le j \le N} M_{ij}$ be your *guaranteed* win when selecting card of suit i, and let $a^* = \max\limits_{1 \le i \le 4} a_i$ be your maximin win.

Proposition. **(i)** *No policy can guarantee you total win in 12 rounds larger than*

$$S = [a_1 + a_2 + a_3 + a_4] + 8a^*$$

(ii) *Total win at least S is guaranteed by the policy as follows:*

— initially, assign every card with the estimate $+\infty$;

— in course of the game, update estimates of the card as follow: *the current estimate of a card is the minimum of the wins you got when selecting this card in the past; if this card was never used before, keep its estimate at its initial value* $+\infty$;

— in every round, select the card with the largest current estimate.

Conclusion: Since the advertizement is truthful, you are in the case of $S \ge 99$

$\Rightarrow$ *Apply policy from* (ii) *and enjoy your AD 99 !*

Proof:

(i): Let the Banker be greedy, so that when you select card with suit i, your win is exactly a_i. Assume w.l.o.g. that $a_1 \leq a_2 \leq a_3 \leq a_4$, so that $a^* = a_4$.

• When selecting the card in the first round, you have no specific information. Whatever card you decide to select, the initial shuffling can be such that the suit of this card will be 1 (the worst, as far as guaranteed win is concerned)

$\Rightarrow$ *Your guaranteed win in the first round cannot be larger than* a_1.

• Assume the shuffling indeed made the suit of your first selection equal to 1, so that you won a_1 in the first round. In the second round, you could select the same card again, or select another card. In the second case, the initial shuffling can be such that your second selection is the second worst card (with suit $i = 2$).

$\Rightarrow$ *Your guaranteed total win in the first two rounds is at most* $\max[2a_1, a_1 + a_2] = a_1 + a_2$.

• Iterating this reasoning, we conclude that *your guaranteed total win in the first 4 rounds is at most* $a_1 + a_2 + a_3 + a_4$. *Since greedy Banker never pays you more that* $a^* = a_4 = \max[a_1, a_2, a_2, a_4]$, *you guaranteed total win is at most*

$$S = a_1 + a_2 + a_3 + a_4 + (12 - 4)a^*$$

(ii): With the policy described in Proposition,

- *A card which once yielded win $< a^*$ will never be used in the future*

Indeed, since the win on card $\#s$ happened to be $< a^*$, the estimate of this card is $< a^*$, and you always have a card with estimate $\geq a^*$ (namely, card with suit 4)

- *If card $\#s$ once brought you win $< a_*$, this win is at least a_{i_s}, where i_s is the suit of card $\#s$.*

$\Rightarrow$ *Your win will be $< a^*$ in at most three rounds, and the total win over these rounds is at least $a_1 + a_2 + a_3$ $\Rightarrow$ Your total win is at least*

$$a_1 + a_2 + a_3 + (12 - 3)a^* = a_1 + a_2 + a_3 + a_4 + (12 - 4)a^*$$

Note: In the game we played,

$$a_1 = a_2 = a_3 = -267, a_4 = 100$$

$\Rightarrow S = 3 \cdot (-267) + 9 \cdot 100 = 99$

However: with Greedy Banker and just 4 (instead of 3) "bad choices," your total win will be

$$4 \cdot (-267) + 8 \cdot 100 = -268$$

Conclusions:

- *To earn and to learn are, in general, two different goals!*

If your goal is to guarantee a positive win in round # 12 (or # 2013), this goal is inachievable: Banker could cheat you by paying all the time, say, AD 100 whatever be your choice, and become greedy only in the "critical" round.

Whether you learn something or nothing, it depends on Banker; but you earn *as if* you were learning something!

- *Cheating is expensive: to keep you ignorant, Banker should pay you extras!*

Applications of Duality in Robust LO

♣ Data uncertainty: Sources

Typically, the data of real world LOs

$$\max_x \left\{c^T x : Ax \le b\right\} \quad [A = [a_{ij}] : m \times n] \qquad \text{(LO)}$$

is not known exactly when the problem is being solved. The most common reasons for data uncertainty are:

• Some of data entries (future demands, returns, etc.) do not exist when the problem is solved and hence are replaced with their forecasts. These data entries are subject to *prediction errors*

• Some of the data (parameters of technological devices/processes, contents associated with raw materials, etc.) cannot be measured exactly, and their true values drift around the measured "nominal" values. These data are subject to *measurement errors*

• Some of the decision variables (intensities with which we intend to use various technological processes, parameters of physical devices we are designing, etc.) cannot be implemented exactly as computed. The resulting *implementation errors* are equivalent to appropriate artificial data uncertainties.

A typical implementation error can be modeled as $x_j \mapsto (1+\xi_j)x_j + \eta_j$, and effect of these errors on a linear constraint

$$\sum_{j=1}^{n} a_{ij}x_j \le b_j$$

is *as if* there were no implementation errors, but the data a_{ij} got the multiplicative perturbations:

$$a_{ij} \mapsto a_{ij}(1+\xi_j)$$

,

and the data b_i got the perturbation

$$b_i \mapsto b_i - \textstyle\sum_j \eta_j a_{ij}.$$

Data uncertainty: Dangers.

In the traditional LO methodology, a small data uncertainty (say, 0.1% or less) is just ignored: the problem is solved *as if* the given ("nominal") data were exact, and the resulting *nominal* optimal solution is what is recommended for use.

Rationale: we hope that small data uncertainties will not affect too badly the feasibility/optimality properties of the nominal solution when plugged into the "true" problem.

Fact: *The above hope can be by far too optimistic, and the nominal solution can be practically meaningless.*

♣ **Example: Antenna Design**

♠ [Physics:] *Directional density of energy transmitted by an monochromatic antenna placed at the origin is proportional to $|D(\delta)|^2$, where the antenna's diagram $D(\delta)$ is a complex-valued function of 3-D direction (unit 3-D vector) δ.*

♠ [Physics:] For an *antenna array* — a complex antenna comprised of a number of antenna elements, the diagram is

$$D(\delta) = \sum_j x_j D_j(\delta) \qquad (*)$$

- $D_j(\cdot)$: diagrams of elements
- x_j: complex *weights* – design parameters responsible for how the elements in the array are invoked.

♠ **Antenna Design problem:** *Given diagrams*

$$D_1(\cdot), ..., D_n(\cdot)$$

and a target diagram $D_*(\cdot)$, find complex weights x_i which make the synthesized diagram $(*)$ as close as possible to the target diagram $D_*(\cdot)$.

♡ When $D_j(\cdot)$, $D_*(\cdot)$ and the weights are real and the "closeness' is quantified by the maximal deviation along a finite grid Γ of directions, Antenna Design becomes the LO problem

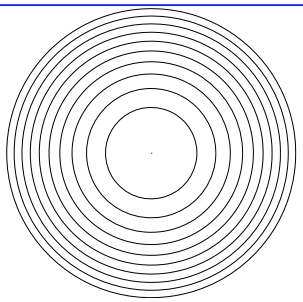
$$\min_{x\in\mathbb{R}^n,\tau} \left\{\tau : -\tau \le D_*(\delta) - \sum_j x_j D_j(\delta) \le \tau \ \forall \delta \in \Gamma\right\}.$$

♠ **Example:** Consider planar antenna array comprised of 10 elements (circle surrounded by 9 rings of equal areas) in the plane XY (Earth's surface"), and our goal is to send most of the energy "up," along the 12^o cone around the Z-axis:

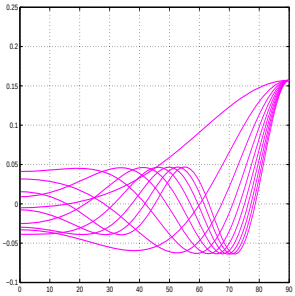
- Diagram of a ring $\{z = 0, a \le \sqrt{x^2 + y^2} \le b\}$:

$$D_{a,b}(\theta) = \frac{1}{2} \int_a^b \left[\int_0^{2\pi} r \cos\left(2\pi r \lambda^{-1} \cos(\theta) \cos(\phi)\right) d\phi \right] dr,$$

- θ: altitude angle
- λ: wavelength



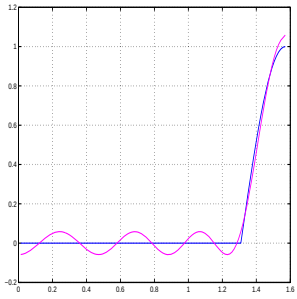
10 elements, equal areas, outer radius 1 m



Diagrams of the elements vs the altitude angle θ, λ =50 cm

- Nominal design problem:

$$\tau_* = \min_{x \in \mathbb{R}^{10}, \tau} \left\{ \tau : -\tau \le D_*(\theta_i) - \sum_{j=1}^{10} x_j D_j(\theta_i) \le \tau, \right.$$
$$\left. 1 \le i \le 240 \right\}, \quad \theta_i = \frac{i\pi}{480}$$

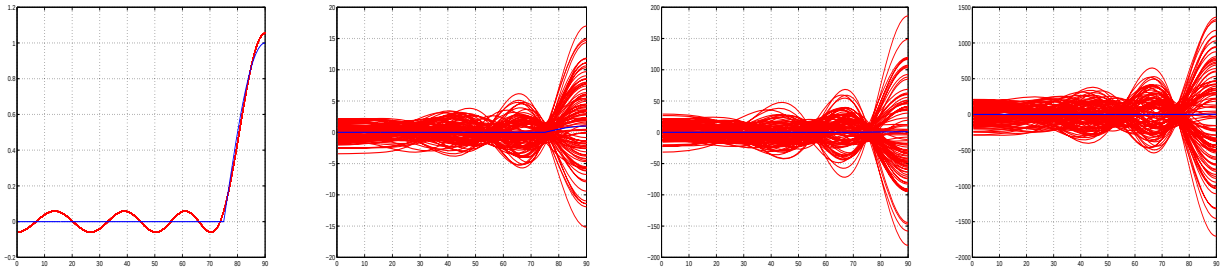


Target (blue) and nominal optimal (magenta) diagrams, $\tau_* = 0.0589$

But: The design variables are characteristics of physical devices and as such they cannot be implemented exactly as computed. *What happens when there are implementation errors:*

$$x_j^{\text{fact}} = (1+\epsilon_j)x_j^{\text{comp}},\ \epsilon_j \sim \text{Uniform}[-\rho,\rho]$$

with small ρ?



$\rho = 0$ $\rho = 0.0001$ $\rho = 0.001$ $\rho = 0.01$

"Dream and reality," nominal optimal design: samples of 100 actual diagrams (red) for different uncertainty levels. Blue: the target diagram

	Dream	Reality		
	$\rho = 0$	$\rho = 0.0001$	$\rho = 0.001$	$\rho = 0.01$
	value	mean	mean	mean
$\|\cdot\|_\infty$-distance to target	0.059	5.671	56.84	506.5
energy concentration	85.1%	16.4%	16.5%	14.9%

Quality of nominal antenna design: dream and reality. Data over 100 samples of actuation errors per each uncertainty level.

♠ Conclusion: *Nominal optimal design is completely meaningless...*

NETLIB Case Study: Diagnosis

♣ NETLIB is a collection of about 100 not very large LPs, mostly of real-world origin. To motivate the methodology of our "case study", here is constraint # 372 of the NETLIB problem PILOT4:

$$\begin{array}{rcl} a^T x & \equiv & -15.79081x_{826} - 8.598819x_{827} - 1.88789x_{828} - 1.362417x_{829} \\ & & -1.526049x_{830} - 0.031883x_{849} - 28.725555x_{850} - 10.792065x_{851} \\ & & -0.19004x_{852} - 2.757176x_{853} - 12.290832x_{854} + 717.562256x_{855} \\ & & -0.057865x_{856} - 3.785417x_{857} - 78.30661x_{858} - 122.163055x_{859} \\ & & -6.46609x_{860} - 0.48371x_{861} - 0.615264x_{862} - 1.353783x_{863} \\ & & -84.644257x_{864} - 122.459045x_{865} - 43.15593x_{866} - 1.712592x_{870} \\ & & -0.401597x_{871} + x_{880} - 0.946049x_{898} - 0.946049x_{916} \\ & \geq & b \equiv 23.387405 \end{array}$$

The related *nonzero* coordinates in the optimal solution x^* of the problem, as reported by CPLEX, are:

$$\begin{array}{ll} x^*_{826} = 255.6112787181108 & x^*_{827} = 6240.488912232100 \\ x^*_{828} = 3624.613324098961 & x^*_{829} = 18.20205065283259 \\ x^*_{849} = 174397.0389573037 & x^*_{870} = 14250.00176680900 \\ x^*_{871} = 25910.00731692178 & x^*_{880} = 104958.3199274139 \end{array}$$

This solution makes the constraint an equality within machine precision.

♣ Most of the coefficients in the constraint are "ugly reals" like -15.79081 or -84.644257. We can be sure that these coefficients characterize technological devices/processes, and as such *hardly are known to high accuracy*.

⇒ "ugly coefficients" can be assumed uncertain and coinciding with the "true" data within accuracy of 3-4 digits.

The only exception is the coefficient 1 of x_{880}, which perhaps reflects the structure of the problem and is exact.

$$
\begin{array}{rcl}
a^T x & \equiv & -15.79081x_{826} - 8.598819x_{827} - 1.88789x_{828} - 1.362417x_{829} \\
& & -1.526049x_{830} - 0.031883x_{849} - 28.725555x_{850} - 10.792065x_{851} \\
& & -0.19004x_{852} - 2.757176x_{853} - 12.290832x_{854} + 717.562256x_{855} \\
& & -0.057865x_{856} - 3.785417x_{857} - 78.30661x_{858} - 122.163055x_{859} \\
& & -6.46609x_{860} - 0.48371x_{861} - 0.615264x_{862} - 1.353783x_{863} \\
& & -84.644257x_{864} - 122.459045x_{865} - 43.15593x_{866} - 1.712592x_{870} \\
& & -0.401597x_{871} + x_{880} - 0.946049x_{898} - 0.946049x_{916} \\
& \geq & b \equiv 23.387405
\end{array}
$$

♣ Assume that the uncertain entries of a are 0.1%-accurate approximations of unknown entries in the "true" data $\tilde{a}$. How does data uncertainty affect the validity of the constraint *as evaluated at the nominal solution x^*?*

• *The worst case*, over all 0.1%-perturbations of uncertain data, violation of the constraint is as large as 450% of the right hand side!

• With *random* and *independent* of each other 0.1% perturbations of the uncertain coefficients, the statistics of the "relative constraint violation"

$$V = \frac{\max[b - \tilde{a}^T x^*, 0]}{b} \times 100\%$$

also is disastrous:

Prob$\{V > 0\}$	Prob$\{V > 150\%\}$	Mean(V)
0.50	0.18	125%

Relative violation of constraint # 372 in PILOT4

(1,000-element sample of 0.1% perturbations)

♣ We see that *quite small (just 0.1%) perturbations of "obviously uncertain" data coefficients can make the "nominal" optimal solution x^* heavily infeasible and thus – practically meaningless.*

♣ In Case Study, we choose a "perturbation level" $\rho \in \{1\%, 0.1\%, 0.01\%\}$, and, for every one of the NETLIB problems, measure the "reliability index" of the nominal solution at this perturbation level:

• We compute the optimal solution x^* of the program

• For every one of the *in*equality constraints

$$a^T x \leq b$$

— we split the left hand side coefficients a_j into "certain" (rational fractions p/q with $|q| \leq 100$) and "uncertain" (all the rest). Let J be the set of all uncertain coefficients of the constraint.

— we compute the *reliability index* of the constraint

$$\frac{\max[a^T x^* + \rho\sqrt{\sum_{j\in J} a_j^2 (x_j^*)^2} - b, 0]}{\max[1, |b|]} \times 100\%$$

Note: *the reliability index is of order of typical violation* (measured in percents of the right hand side) *of the constraint, as evaluated at x^*, under independent random perturbations, of relative magnitude ρ, of the uncertain coefficients*.

• We treat the nominal solution as *unreliable*, and the problem - as *bad*, the level of perturbations being ρ, if the worst, over the inequality constraints, reliability index is worse than 5%.

♣ The results of the Diagnosis phase of Case Study are as follows.

• From the total of 90 NETLIB problems processed,

— in 27 problems the nominal solution turned out to be unreliable at the largest ($\rho = 1\%$) level of uncertainty;

— 19 of these 27 problems were already bad at the 0.01%-level of uncertainty

— in 13 problems, 0.01% perturbations of the uncertain data can make the nominal solution more than 50%-infeasible for some of the constraints.

Problem	Size[a]	$\rho = 0.01\%$		$\rho = 0.1\%$	
		#bad[b]	Index[c]	#bad	Index
80BAU3B	2263×9799	37	84	177	842
25FV47	822×1571	14	16	28	162
ADLITTLE	57×97			2	6
AFIRO	28×32			1	5
CAPRI	272×353			10	39
CYCLE	1904×2857	2	110	5	1,100
D2Q06C	2172×5167	107	1,150	134	11,500
FINNIS	498×614	12	10	63	104
GREENBEA	2393×5405	13	116	30	1,160
KB2	44×41	5	27	6	268
MAROS	847×1443	3	6	38	57
PEROLD	626×1376	6	34	26	339
PILOT	1442×3652	16	50	185	498
PILOT4	411×1000	42	210,000	63	2,100,000
PILOT87	2031×4883	86	130	433	1,300
PILOTJA	941×1988	4	46	20	463
PILOTNOV	976×2172	4	69	13	694
PILOTWE	723×2789	61	12,200	69	122,000
SCFXM1	331×457	1	95	3	946
SCFXM2	661×914	2	95	6	946
SCFXM3	991×1371	3	95	9	946
SHARE1B	118×225	1	257	1	2,570

[a] # of linear constraints (excluding the box ones) plus 1 and # of variables

[b] # of constraints with index $> 5\%$

[c] The worst, over the constraints, reliability index, in %

♣ **Conclusions:**

♢ *In real-world applications of Linear Programming one cannot ignore the possibility that a small uncertainty in the data* (intrinsic for the majority of real-world LP programs) *can make the usual optimal solution of the problem completely meaningless from practical viewpoint.*

Consequently,

♢ *In applications of LP, there exists a real need of a technique capable of detecting cases when data uncertainty can heavily affect the quality of the nominal solution, and in these cases to generate a "reliable" solution, one which is immune against uncertainty.*

Robust LO is aimed at meeting this need.

Robust LO: Paradigm

♣ In Robust LO, one considers an *uncertain LO problem*

$$\mathcal{P} = \left\{\max_x \left\{c^T x : Ax \le b\right\} : (c, A, b) \in \mathcal{U}\right\},$$

— a *family* of *all* usual LO instances of common sizes m (number of constraints) and n (number of variables) with the *data* (c, A, b) running through a given *uncertainty set* $\mathcal{U} \subset \mathbb{R}^n_c \times \mathbb{R}^{m\times n}_A \times \mathbb{R}^m_b$.

♠ We consider the situation where

• *The solution should be built before the "true" data reveals itself and thus cannot depend on the true data.* All we know when building the solution is the uncertainty set $\mathcal{U}$ to which the true data belongs.

• *The constraints are hard: we cannot tolerate their violation.*

♠ In the outlined "decision environment," the only meaningful candidate solutions x are the *robust feasible ones* – those *which remain feasible whatever be a realization of the data from the uncertainty set:*

$$x \in \mathbb{R}^n \text{ is robust feasible for } \mathcal{P} \Leftrightarrow Ax \le b \; \forall (c, A, b) \in \mathcal{U}$$

♡ We characterize the objective at a candidate solution x by the *guaranteed value*

$$t(x) = \min\{c^T x : (c, A, b) \in \mathcal{U}\}$$

of the objective.

$$\mathcal{P} = \left\{\max_x \left\{c^T x : Ax \le b\right\} : (c, A, b) \in \mathcal{U}\right\},$$

♡ Finally, we associate with the uncertain problem $\mathcal{P}$ its *Robust Counterpart*

$$\begin{array}{l}\mathrm{ROpt}(\mathcal{P}) \\ = \max\limits_{t,x} \left\{t : t \le c^T x,\ Ax \le b\, \forall (c, A, b) \in \mathcal{U}\right\}\end{array} \qquad \text{(RC)}$$

where one seeks for the best (with the largest *guaranteed* value of the objective) *robust feasible* solution to $\mathcal{P}$.

The optimal solution to the RC is treated as the best among "immunized against uncertainty" solutions and is recommended for actual use.

Basic question: *Unless the uncertainty set $\mathcal{U}$ is finite, the RC is not an LO program, since it has infinitely many linear constraints. Can we convert* (RC) *into an explicit LO program?*

$$\mathcal{P} = \left\{\max_x \left\{c^T x : Ax \le b\right\} : (c, A, b) \in \mathcal{U}\right\}$$
$$\Rightarrow \max_{t,x} \left\{t : t \le c^T x,\ Ax \le b \,\forall (c, A, b) \in \mathcal{U}\right\} \quad \text{(RC)}$$

Observation: *The RC remains intact when the uncertainty set $\mathcal{U}$ is replaced with its convex hull.*

Theorem: *The RC of an uncertain LO program with nonempty polyhedrally representable uncertainty set is equivalent to an LO program. Given a polyhedral representation of $\mathcal{U}$, the LO reformulation of the RC is easy to get.*

$$\mathcal{P} = \left\{\max_x \left\{c^T x : Ax \le b\right\} : (c, A, b) \in \mathcal{U}\right\}$$
$$\Rightarrow \max_{t,x} \left\{t : t \le c^T x,\ Ax \le b \,\forall (c, A, b) \in \mathcal{U}\right\} \quad \text{(RC)}$$

Proof of Theorem. Let

$$\mathcal{U} = \{\zeta = (c, A, b) \in \mathbb{R}^N : \exists w : P\zeta + Qw \le r\}$$

be a polyhedral representation of the uncertainty set. Setting $y = [x; t]$, the constraints of (RC) become

$$q_i(\zeta) - p_i^T(\zeta) y \le 0 \,\forall \zeta \in \mathcal{U},\ 0 \le i \le m \quad (C_i)$$

with $p_i(\cdot)$, $q_i(\cdot)$ affine in ζ. We have

$$q_i(\zeta) - p_i^T(\zeta) y \equiv \pi_i^T(y)\zeta - \theta_i(y),$$

with $\theta_i(y)$, $\pi_i(y)$ affine in y. Thus, i-th constraint in (RC) reads

$$\max_{\zeta, w_i} \left\{\pi_i^T(y)\zeta : P\zeta + Qw_i \le r\right\} = \max_{\zeta \in \mathcal{U}} \pi_i^T(y)\zeta \le \theta_i(y).$$

Since $\mathcal{U} \neq \emptyset$, by the LO Duality we have

$$\max_{\zeta, w_i} \left\{\pi_i^T(y)\zeta : P\zeta + Qw_i \le r\right\}$$
$$= \min_{\eta_i} \left\{r^T \eta_i : \eta_i \ge 0, P^T \eta_i = \pi_i(y), Q^T \eta_i = 0\right\}$$

$\Rightarrow$ *y satisfies (C_i) if and only if there exists η_i such that*

$$\eta_i \ge 0, P^T \eta_i = \pi_i(y), Q^T \eta_i = 0, r^T \eta_i \le \theta_i(y) \qquad (R_i)$$

$\Rightarrow$ *(RC) is equivalent to the LO program of maximizing $e^T y \equiv t$ in variables $y, \eta_0, \eta_1, ..., \eta_m$ under the linear constraints (R_i), $0 \le i \le m$.*

♠ **Example:** The Robust Counterpart of uncertain LO with *interval uncertainty*:

$$\begin{array}{rcl} \mathcal{U}_{\mathsf{obj}} & = & \{c : |c_j - c_j^0| \le \delta c_j,\ j = 1, ..., n\} \\ \mathcal{U}_i & = & \{(a_{i1}, , , .a_{im}, b_i) : |a_{ij} - a_{ij}^0| \le \delta a_{ij}, |b_i - b_i^0| \le \delta b_i\} \end{array}$$

is the LO program

$$\max_{x,y,t} \left\{ t : \begin{array}{l} \sum_j c_j^0 x_j - \sum_j \delta c_j y_j \ge t \\ \sum_j a_{ij}^0 x_j + \sum_j \delta a_{ij} y_j \le b_i - \delta b_i^0 \\ -y_j \le x_j \le y_j \end{array} \right\}$$

How it works? – Antenna Example

$$\min_{x,\tau}\left\{\tau : -\tau \le D_*(\theta_\ell) - \textstyle\sum_{j=1}^{10} x_j D_j(\theta_\ell) \le \tau,\ \ell = 1,...,L\right\}$$

$$\Updownarrow$$

$$\boxed{\min_{x,\tau}\{\tau : Ax + \tau a + b \ge 0\} \qquad \text{(LO)}}$$

- The influence of "implementation errors"

$$x_j \mapsto (1+\epsilon_j)x_j$$

with $|\epsilon_j| \le \rho \in [0,1]$ is *as if* there were no implementation errors, but the part A of the constraint matrix was uncertain and known "up to multiplication by a diagonal matrix with diagonal entries from $[1-\rho, 1+\rho]$":

$$\mathcal{U} = \left\{A = A^{\text{nom}}\text{Diag}\{1+\epsilon_1,...,1+\epsilon_{10}\} : |\epsilon_j| \le \rho\right\} \qquad \text{(U)}$$

Note that *as far as a particular constraint is concerned, the uncertainty is an interval one with $\delta A_{ij} = \rho|A_{ij}|$. The remaining coefficients (and the objective) are certain.*

♣ To improve reliability of our design, we replace the uncertain LO program (LO), (U) with its robust counterpart, which is nothing but an explicit LO program.

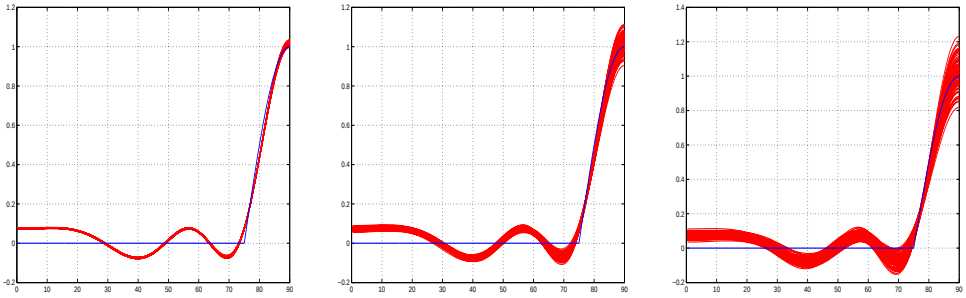
How it Works: Antenna Design (continued)

$$\min_{\tau,x}\left\{\tau : -\tau \le D_*(\theta_i) - \sum_{j=1}^{10} x_j D_j(\theta_i) \le \tau,\ 1 \le i \le I\right\}$$
$$x_j \mapsto (1+\epsilon_j)x_j,\ -\rho \le \epsilon_j \le \rho$$

$$\Downarrow$$

$$\min_{\tau,x}\left\{\tau : \begin{array}{c} D_*(\theta_i) - \sum_j x_j D_j(\theta_i) - \rho \sum_j |x_j||D_j(\theta_i)| \ge -\tau \\ D_*(\theta_i) - \sum_j x_j D_j(\theta_i) + \rho \sum_j |x_j||D_j(\theta_i)| \le \tau \end{array},\ 1 \le i \le I\right\}$$

♠ Solving the Robust Counterpart at uncertainty level $\rho = 0.01$, we arrive at *robust design*. The robust optimal value is 0.0815 (39% more than the nominal optimal value 0.0589).



$\rho = 0.01$ $\rho = 0.05$ $\rho = 0.1$

Robust optimal design: samples of 100 actual diagrams (red).

	Reality	
	$\rho = 0.01$	$\rho = 0.1$
$\|\cdot\|_\infty$ distance to target	max = 0.081 mean = 0.077	max = 0.216 mean = 0.113
energy concentration	min = 70.3% mean = 72.3%	min = 52.2% mean = 70.8%

Robust optimal design, data over 100 samples of actuation errors.

- For *nominal* design with $\rho = 0.001$, the average $\|\cdot\|_\infty$-distance to target is 56.8, and average energy concentration is 16.5%.

♣ *Why the "nominal design" is that unreliable?*

• The basic diagrams $D_j(\cdot)$ are "nearly linearly dependent". As a result, the nominal problem is "ill-posed" – it possesses a huge domain comprised of "nearly optimal" solutions. Indeed, look what are the optimal values in the nominal Antenna Design LO with added box constraints $|x_j| \leq L$ on the variables:

L	1	10	10^2	10^3	10^4	10^5	10^6
Opt_Val	0.0945	0.0800	0.0736	0.0696	0.0659	0.0627	0.0622

The "exactly optimal" solution to the nominal problem is very large, and therefore even small *relative* implementation errors may completely destroy the design.

• In the robust counterpart, magnitudes of candidate solutions are penalized, and RC implements a smart trade-off between the optimality and the magnitude (i.e., the stability) of the solution.

j	1	2	3	4	5	6	7	8	9	10
x_j^{nom}	2e3	-1e4	6e4	-1e5	1e5	2e4	-1e5	1e6	-7e4	1e4
x_j^{rob}	-0.3	5.0	-3.4	-5.1	6.9	5.5	5.3	-7.5	-8.9	13

How it works? NETLIB Case Study

♣ When applying the RO methodology to the bad NETLIB problems, assuming interval uncertainty of (relative) magnitude $\rho \in \{1\%, 0.1\%, 0.01\%\}$ in "ugly coefficients" of *in*equality constraints (*no uncertainty in equations*!), it turns out that

- Reliable solutions do exist, except for 4 cases corresponding to the highest ($\rho = 1\%$) perturbation level.
- The "price of immunization" in terms of the objective value is surprisingly low: when $\rho \leq 0.1\%$, it never exceeds 1% and it is less than 0.1% in 13 of 23 cases. Thus, *passing to the robust solutions, we gain a lot in the ability of the solution to withstand data uncertainty, while losing nearly nothing in optimality.*

Problem	Nominal optimal value	Objective at robust solution	
		$\rho = 0.1\%$	$\rho = 1\%$
80BAU3B	987224.2		1009229 (2.2%)
25FV47	5501.846	5502.191 (0.0%)	5505.653 (0.1%)
ADLITTLE	225495.0		228061.3 (1.1%)
AFIRO	-464.7531	-464.7500 (0.0%)	-464.2613 (0.1%)
BNL2	1811.237	1811.237 (0.0%)	1811.338 (0.0%)
BRANDY	1518.511		1518.581 (0.0%)
CAPRI	1912.621	1912.738 (0.0%)	1913.958 (0.1%)
CYCLE	1913.958	1913.958 (0.0%)	1913.958 (0.0%)
D2Q06C	122784.2	122893.8 (0.1%)	Infeasible
E226	-18.75193		-18.75173 (0.0%)
FFFFF800	555679.6		555715.2 (0.0%)
FINNIS	172791.1	173269.4 (0.3%)	178448.7 (3.3%)
GREENBEA	-72555250	-72192920 (0.5%)	-68869430 (5.1%)
KB2	-1749.900	-1749.638 (0.0%)	-1746.613 (0.2%)
MAROS	-58063.74	-58011.14 (0.1%)	-57312.23 (1.3%)
NESM	14076040		14172030 (0.7%)
PEROLD	-9380.755	-9362.653 (0.2%)	Infeasible
PILOT	-557.4875	-555.3021 (0.4%)	Infeasible
PILOT4	-64195.51	-63584.16 (1.0%)	-58113.67 (9.5%)
PILOT87	301.7109	302.2191 (0.2%)	Infeasible
PILOTJA	-6113.136	-6104.153 (0.2%)	-5943.937 (2.8%)
PILOTNOV	-4497.276	-4488.072 (0.2%)	-4405.665 (2.0%)
PILOTWE	-2720108	-2713356 (0.3%)	-2651786 (2.5%)
SCFXM1	18416.76	18420.66 (0.0%)	18470.51 (0.3%)
SCFXM2	36660.26	36666.86 (0.0%)	36764.43 (0.3%)
SCFXM3	54901.25	54910.49 (0.0%)	55055.51 (0.3%)
SHARE1B	-76589.32	-76589.32 (0.0%)	-76589.29 (0.0%)

Objective values at nominal and robust solutions to bad NETLIB problems.
Percent in $(\cdot)$: Excess of robust optimal value over the nominal optimal value

Quiz: Alicubi is a small country in Africa. All $n = 128$ types of diary products consumed by the population are supplied by a single company Diary Co which frequently and abruptly changes the prices of its products. There is, however, Consumer Protection Law stating that whenever buying $m = 5$ distinct from each other diary products, in unit amount each, the total cost cannot exceed 5 Alicubi dollars (AD).
Mr. Nemo wants to order diaries. His utility function to be maximized is

$$u = \sum_{i=1}^{n} p_i x_i$$

• $p_i \geq 0$: per unit "utility" of diary # i • x_i: order for diary # i
and he wants to maximize the utility by selecting order $x = [x_1; ...; x_{128}]$ under the constraints that

• $x \geq 0$

• Whatever be the diary prices (obeying the Consumer Protection Law) at the time of order's delivery, Mr. Nemo's AD 1 will be enough to pay the bill.

(?) How Mr. Nemo should act?

♠ Mr. Nemo should solve the Robust Optimization problem

$$\max_{x\in\mathbb{R}^n}\left\{p^Tx : x\geq 0, c^Tx\leq 1\ \forall c\in\mathcal{U}_m\right\} \qquad [n=128]$$

where $\mathcal{U}_m$ is the set of all price vectors obeying the Consumer Protection Law:

$$\mathcal{U}_m=\{c\in\mathbb{R}^n : c\geq 0, c_{j_1}+c_{j_2}+...+c_{j_m}\leq m \text{ for all } 1\leq j_1<j_2<...<j_m\leq n\} \qquad [m=5]$$

• The uncertainty set is polyhedral $\Rightarrow$ *Mr. Nemo's problem can be converted to LO*

But: $\mathcal{U}_m$ is given by a huge number of linear inequalities. *Can we find a "compact" polyhedral representation of* $\mathcal{U}_m$ *?*

$$\mathcal{U}_m = \{c \in \mathbb{R}^n : c \geq 0, c_{j_1} + c_{j_2} + ... + c_{j_m} \leq m \text{ for all } 1 \leq j_1 < j_2 < ... < j_m \leq n\}$$

$[m = 5]$

(?) *Can we find a "compact" polyhedral representation of* $\mathcal{U}_m$ *?*

Observation: $\mathcal{U}_m = \{c \geq 0 : e^T c \leq m \text{ for all } e \in \mathcal{E}\}$

• $\mathcal{E}$: the set of all Boolean vectors from $\mathbb{R}^n$ with m nonzero entries

♠ As we know, $\mathcal{E}$ is exactly the set of all extreme points of the polytope

$$Y = \{y \in \mathbb{R}^n : 0 \leq y_j \leq 1 \, \forall j, \textstyle\sum_{j=1}^n y_j = m\}$$

$\Rightarrow \mathcal{U}_m$ is the set

$$\{c \in \mathbb{R}^n_+ : \overbrace{\max_y \{c^T y : -y \underbrace{\leq}_{u} 0, y \underbrace{\leq}_{v} [1; ...; 1], \textstyle\sum_j y_j \underbrace{=}_{w} m\}}^{(*)} \leq m\}$$

$$= \left\{c \in \mathbb{R}^n_+ : \min_{u,v,w} \{mw + \textstyle\sum_j v_j : w[1; ...; 1] + v - u = c, \, u \geq 0, v \geq 0\} \leq m\right\}$$

[we have passed to the dual of $(*)$]

$$= \left\{c \in \mathbb{R}^n_+ : \min_{u,v,w} \{mw + \textstyle\sum_j v_j : w[1; ...; 1] + v \geq c, \, v \geq 0\} \leq m\right\}$$

$$= \left\{c \in \mathbb{R}^n_+ : \exists v, w : c \leq w[1; ...; 1] + v, v \geq 0, \textstyle\sum_j v_j + mw \leq m\right\}$$

- Mr. Nemo's problem becomes

$\max_x p^T x$

s.t. $x \geq 0$

$$\max_{c,v,w} \left\{ x^T c : \left\{ \begin{array}{l} c - v - w[1;...1] \underbrace{\leq}_{q} 0 \\ -c \underbrace{\leq}_{r} 0,\ -v \underbrace{\leq}_{s} 0 \\ [1;...;1]^T v + mw \underbrace{=}_{t} m \end{array} \right\} \leq 1 \right.$$

$$\Leftrightarrow \quad \min_{q,r,s,t} \left\{ mt : \left\{ \begin{array}{l} q - r = x,\ q + s = t[1;...;1] \\ [1;...;1]^T q = mt \\ q \geq 0, r \geq 0, s \geq 0 \end{array} \right\} \leq 1 \right.$$

$$\Leftrightarrow \quad \min_q \left\{ \sum_{j=1}^n q_j : q \geq x, q \geq 0, q_j \leq \frac{\sum_k q_k}{m},\ 1 \leq j \leq n \right\} \leq 1$$

$$\Leftrightarrow \quad \exists q : q \geq x, q \geq 0, \sum_j q_j \leq 1, q_j \leq \frac{\sum_k q_k}{m}, 1 \leq j \leq n$$

$$\Leftrightarrow \quad \exists q : q \geq x, q \geq 0, \sum_{j=1}^n q_j = 1,\ q_j \leq \frac{1}{m},\ 1 \leq j \leq n$$

or, which is the same,

$$\boxed{\max_x \{ \sum_{j=1}^n p_j x_j : 0 \leq x_j \leq \frac{1}{m}\ \forall j, \sum_{j=1}^n x_j \leq 1 \}}$$

- Since $p_j \geq 0$ for all j, *a robustly optimal order is to request* $\frac{1}{m} = \frac{1}{5}$ *units of every one of the* $m = 5$ *most useful for Mr. Nemo (with the largest* p_j*) diaries and zero amounts of all other diaries. The robust optimal value in Mr. Nemo's problem is the average of the 5 largest* p_j*'s.*

Affinely Adjustable Robust Counterpart

♣ The rationale behind the Robust Optimization paradigm as applied to LO is based on two assumptions:

A. *Constraints of an uncertain LO program is a "must": a meaningful solution should satisfy all realizations of the constraints allowed by the uncertainty set.*

B. *All decision variables should be defined before the true data become known and thus should be independent of the true data.*

♣ In many cases, Assumption **B** is too conservative:

• In dynamical decision-making, only part of decision variables correspond to "here and now" decisions, while the remaining variables represent "wait and see" decisions to be made when part of the true data will be already revealed.

(!) *"Wait and see" decision variables may – and should! – depend on the corresponding part of the true data.*

• Some of decision variables do not represent actual decisions at all; they are artificial "analysis variables" introduced to convert the problem into the LO form.

(!) *Analysis variables may – and should! – depend on the entire true data.*

Example: Consider the problem of the best $\|\cdot\|_1$-approximation

$$\min_{x,t}\left\{t : \sum_i |b_i - \sum_j a_{ij}x_j| \le t\right\}. \qquad \text{(P)}$$

When the data are certain, this problem is equivalent to the LP program

$$\min_{x,y,t}\left\{t : \sum_i y_i \le t,\ -y_i \le b_i - \sum_j a_{ij}x_j \le y_i\ \forall i\right\}. \quad \text{(LP)}$$

With uncertain data, the Robust Counterpart of (P) becomes the semi-infinite problem

$$\min_{x,t}\left\{t : \sum_i |b_i - \sum_j a_{ij}x_j| \le t\ \forall (b_i, a_{ij}) \in \mathcal{U}\right\},$$

or, which is the same, the problem

$$\min_{x,t}\left\{t : \forall (b_i, a_{ij}) \in \mathcal{U} :\ \exists y : \sum_i y_i \le t,\ -y_i \le b_i - \sum_j a_{ij}x_j \le y_i\right\},$$

while the RC of (LP) is the much more conservative problem

$$\min_{x,t}\left\{t : \exists y :\ \forall (b_i, a_{ij}) \in \mathcal{U} : \sum_i y_i \le t,\ -y_i \le b_i - \sum_j a_{ij}x_j \le y_i\right\}.$$

Adjustable Robust Counterpart of an Uncertain LO

♣ Consider an uncertain LO. Assume w.l.o.g. that the data of LO are affinely parameterized by a "perturbation vector" ζ running through a given *perturbation set* $\mathcal{Z}$:

$$\mathcal{LP} = \left\{\max_x \left\{c^T[\zeta]x : A[\zeta]x - b[\zeta] \leq 0\right\} : \zeta \in \mathcal{Z}\right\}$$
$$\left[c_j[\zeta], A_{ij}[\zeta], b_i[\zeta] : \text{ affine in } \zeta\right]$$

♠ Assume that every decision variable may depend on a given "portion" of the true data. Since the latter is affine in ζ, this assumption says that x_j *may depend on* $P_j\zeta$, where P_j are given matrices.

- $P_j = 0 \Rightarrow x_j$ *is non-adjustable*: x_j represents an independent of the true data "here and now" decision;
- $P_j \neq 0 \Rightarrow x_j$ *is adjustable*: x_j represents a "wait and see' decision or an analysis variable which may adjust itself – fully or partially, depending on P_j – to the true data.

$$\mathcal{LP} = \left\{\max_x \left\{c^T[\zeta]x : A[\zeta]x - b[\zeta] \le 0\right\} : \zeta \in \mathcal{Z}\right\}$$
$$\left[c_j[\zeta], A_{ij}[\zeta], b_i[\zeta] : \text{ affine in } \zeta\right]$$

♣ Under circumstances, a natural Robust Counterpart of $\mathcal{LP}$ is the problem

Find t and functions $\phi_j(\cdot)$ such that the decision rules $x_j = \phi_j(P_j\zeta)$ make all the constraints feasible for all perturbations $\zeta \in \mathcal{Z}$, while minimizing the guaranteed value t of the objective:

$$\max_{t,\{\phi_j(\cdot)\}} \left\{ t : \begin{array}{l} \sum_j c_j[\zeta]\phi_j(P_j\zeta) \ge t \;\forall \zeta \in \mathcal{Z} \\ \sum_j \phi_j(P_j\zeta)A_j[\zeta] - b[\zeta] \le 0 \;\forall \zeta \in \mathcal{Z} \end{array} \right\} \qquad \text{(ARC)}$$

♣ **Bad news:** The *Adjustable Robust Counterpart*

$$\max_{t,\{\phi_j(\cdot)\}} \left\{ t : \begin{array}{l} \sum_j c_j[\zeta]\phi_j(P_j\zeta) \geq t \, \forall \zeta \in \mathcal{Z} \\ \sum_j \phi_j(P_j\zeta)A_j[\zeta] - b[\zeta] \leq 0 \, \forall \zeta \in \mathcal{Z} \end{array} \right\} \quad \text{(ARC)}$$

of uncertain LP is an *infinite-dimensional* optimization program and as such typically is absolutely intractable: How could we represent efficiently general-type functions of many variables, not speaking about how to optimize with respect to these functions?

♠ **Partial Remedy (???):** Let us restrict the decision rules $x_j = \phi_j(P_j\zeta)$ to be easily representable – specifically, *affine* – functions:

$$\phi_j(P_j\zeta) \equiv \mu_j + \nu_j^T P_j \zeta.$$

With this dramatic simplification, (ARC) becomes a *finite-dimensional* (still semi-infinite) *optimization problem in new non-adjustable variables* μ_j, ν_j

$$\max_{t,\{\mu_j,\nu_j\}} \left\{ t : \begin{array}{l} \sum_j c_j[\zeta](\mu_j + \nu_j^T P_j\zeta) \geq t \, \forall \zeta \in \mathcal{Z} \\ \sum_j (\mu_j + \nu_j^T P_j\zeta)A_j[\zeta] - b[\zeta] \leq 0 \, \forall \zeta \in \mathcal{Z} \end{array} \right\} \quad \text{(AARC)}$$

♣ We have associated with uncertain LO

$$\mathcal{LP} = \left\{\max_x \left\{c^T[\zeta]x : A[\zeta]x - b[\zeta] \le 0\right\} : \zeta \in \mathcal{Z}\right\}$$
$$\left[c_j[\zeta], A_{ij}[\zeta], b_i[\zeta] : \text{ affine in } \zeta\right]$$

and the "information matrices" $P_1, ..., P_n$ the *Affinely Adjustable Robust Counterpart*

$$\max_{t,\{\mu_j,\nu_j\}} \left\{ t : \begin{array}{l} \sum_j c_j[\zeta](\mu_j + \nu_j^T P_j \zeta) \ge t \, \forall \zeta \in \mathcal{Z} \\ \sum_j (\mu_j + \nu_j^T P_j \zeta) A_j[\zeta] - b[\zeta] \le 0 \, \forall \zeta \in \mathcal{Z} \end{array} \right\} \quad \text{(AARC)}$$

♠ **Relatively good news:**

• AARC is by far more flexible than the usual (non-adjustable) RC of $\mathcal{LP}$.

• As compared to ARC, AARC has much more chances to be computationally tractable:

— *In the case of simple recourse*, where the coefficients of adjustable variables are certain, AARC has the same tractability properties as RC:

If the perturbation set $\mathcal{Z}$ is given by polyhedral representation, (AARC) *can be straightforwardly converted into an explicit LO program.*

— *In the general case*, (AARC) may be computationally intractable; however, under mild assumptions on the perturbation set, (AARC) admits "tight" computationally tractable approximation.

♣ **Example: simple Inventory model.** There is a single-product inventory system with

• a single warehouse which should at any time store at least $V_{\min}$ and at most $V_{\max}$ units of the product;

• *uncertain* demands d_t of periods $t = 1, ..., T$ known to vary within given bounds:

$$d_t \in [d_t^*(1-\theta), d_t^*(1+\theta)], \; t = 1, ..., T$$

• $\theta \in [0, 1]$: uncertainty level

No backlogged demand is allowed!

• I factories from which the warehouse can be replenished:

— at the beginning of period t, you may order $p_{t,i}$ units of product from factory i. Your orders should satisfy the constraints

$$\begin{array}{ll} 0 \le p_{t,i} \le P_i(t) & \text{[bounds on capacities per period]} \\ \sum_t p_{t,i} \le Q_i & \text{[bounds on cumulative capacities]} \end{array}$$

— an order is executed with no delay

— order $p_{t,i}$ costs you $c_i(t)p_{t,i}$.

• The goal: *to minimize the total cost of the orders.*

♠ *With certain demand*, the problem can be modeled as the LO program

$$\min_{\substack{p_{t,i},:i\le I,t\le T,\\ v_t,2\le t\le T+1}} \sum_{t,i} c_i(t)p_{t,i} \qquad \text{[total cost]}$$

s.t.

$$\begin{array}{ll}
v_{t+1} - v_t - \sum_i p_{t,i} = d_t,\ t = 1,...,T & \left[\begin{array}{c}\text{state equations}\\ (v_1 \text{ is given})\end{array}\right] \\
V_{\min} \le v_t \le V_{\max}, 2 \le t \le T+1 & \text{[bounds on states]} \\
0 \le p_{t,i} \le P_i(t),\ i \le I, t \le T & \text{[bounds on orders]} \\
\sum_t p_{t,i} \le Q_i,\ i \le I & \left[\begin{array}{c}\text{cumulative bounds}\\ \text{on orders}\end{array}\right]
\end{array}$$

♠ *With uncertain demand*, it is natural to assume that the orders $p_{t,i}$ may depend on the demands of the preceding periods $1,...,t-1$. The *analysis variables* v_t are allowed to depend on the entire actual data. In fact, it suffices to allow for v_t to depend on $d_1,...,d_{t-1}$.

♠ Applying the AARC methodology, we make $p_{t,i}$ and v_t affine functions of past demands:

$$\begin{array}{rcl}
p_{t,i} & = & \phi^0_{t,i} + \sum\limits_{1\le\tau<t} \phi^\tau_{t,i} d_\tau \\
v_t & = & \psi^0_t + \sum\limits_{1\le\tau<t} \psi^\tau_t d_\tau
\end{array}$$

- *ϕ's and ψ's are our new decision variables...*

$$\begin{array}{|c|}\hline \min_{\{p_{t,i},v_t\}} \sum_{t,i} c_i(t)p_{t,i} \quad \text{s.t.} \\ v_{t+1} - v_t - \sum_i p_{t,i} = d_t,\ t = 1,...,T \\ V_{\min} \le v_t \le V_{\max}, 2 \le t \le T+1 \\ 0 \le p_{t,i} \le P_i(t),\ i \le I, t \le T \\ \sum_t p_{t,i} \le Q_i,\ i \le I \\ \hline \begin{array}{rcl} p_{t,i} & = & \phi^0_{t,i} + \sum\limits_{1\le\tau<t} \phi^\tau_{t,i} d_\tau \\ v_t & = & \psi^0_t + \sum\limits_{1\le\tau<t} \psi^\tau_t d_\tau \end{array} \\ \hline \end{array}$$

♠ The AARC is the following *semi-infinite* LO in non-adjustable decision variables ϕ's and ψ's:

$$\min_{C,\{\phi^\tau_{t,i},\psi^\tau_t\}} C \quad \text{s.t.}$$

$$\sum_{t,i} c_i(t) \left[\phi^0_{t,i} + \sum_{1\le\tau<t} \phi^\tau_{t,i} d_\tau\right] \le C$$

$$\left[\psi^0_{t+1} + \sum_{\tau=1}^{t} \psi^\tau_{t+1} d_\tau\right] - \left[\psi^0_t + \sum_{\tau=1}^{t-1} \psi^\tau_t d_\tau\right] - \sum_i \left[\phi^0_{t,i} + \sum_{\tau=1}^{t-1} \phi^\tau_{t,i} d_\tau\right] = d_t$$

$$V_{\min} \le \left[\psi^0_t + \sum_{\tau=1}^{t-1} \psi^\tau_t d_\tau\right] \le V_{\max}$$

$$0 \le \left[\phi^0_{t,i} + \sum_{\tau=1}^{t-1} \phi^\tau_{t,i} d_\tau\right] \le P_i(t)$$

$$\sum_t \left[\phi^0_{t,i} + \sum_{\tau=1}^{t-1} \phi^\tau_{t,i} d_\tau\right] \le Q_i$$

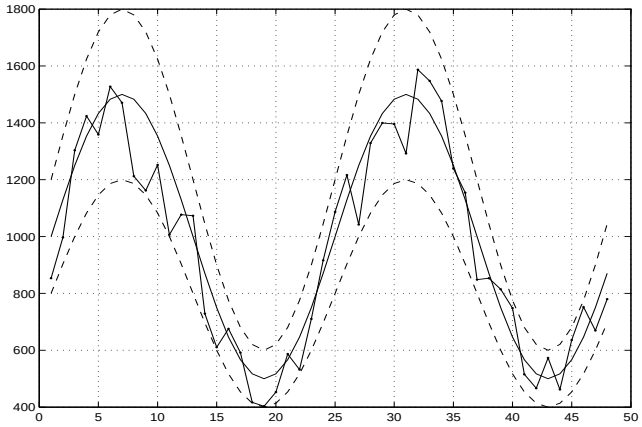
• The constraints should be valid for all values of "free" indexes and *all demand trajectories* $d = \{d_t\}_{t=1}^T$ *from the "demand uncertainty box"*

$$\mathcal{D} = \{d : d^*_t(1-\theta) \le d_t \le d^*_t(1+\theta), 1 \le t \le T\}.$$

♠ *The AARC can be straightforwardly converted to a usual LP and easily solved.*

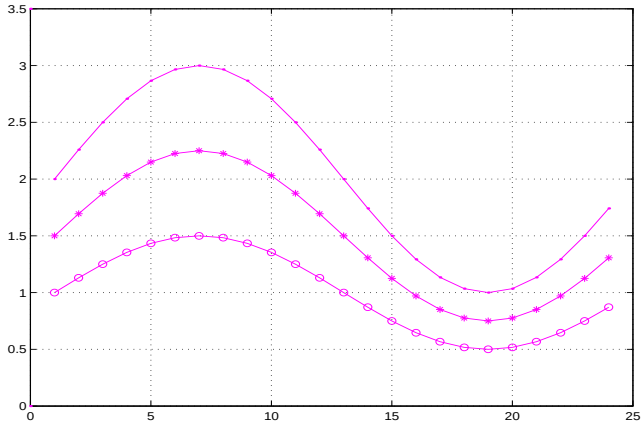
♣ In the numerical illustration to follow:

- the planning horizon is $T = 24$
- there are $I = 3$ factories with per period capacities $P_i(t) = 567$ and cumulative capacities $Q_i = 13600$
- the nominal demand d_t^* is seasonal:



$$d_t^* = 1000\left(1 + 0.5\sin\left(\frac{\pi(t-1)}{12}\right)\right)$$

- the ordering costs also are seasonal:



$$c_i(t) = c_i\left(1 + 0.5\sin\left(\frac{\pi(t-1)}{12}\right)\right),\ c_1 = 1, c_2 = 1.5, c_3 = 2$$

- $v_1 = V_{\text{min}} = 500,\ V_{\text{max}} = 2000$
- demand uncertainty $\theta = 20\%$

♣ **Results:**

• Opt(AARC) = 35542.

Note: The *non-adjustable* RC is infeasible already at 5% uncertainty level!

• With uniformly distributed in the range ±20% demand perturbations, the average, over 100 simulations, AARC management cost is 35121.

Note: Over the same 100 simulations, the average *"utopian"* management cost (optimal for *a priori known* demand trajectories) is 33958, i.e., is by just 3.5% (!) less than the average AARC management cost.

♣ **Comparison with Dynamic Programming.** *When applicable*, DP is the technique for dynamical decision-making under uncertainty – in (worst-case-oriented) DP, one solves the Adjustable Robust Counterpart of uncertain LO, *with no ad hoc simplifications like "let us restrict ourselves with affine decision rules."*

♠ Unfortunately, DP suffers from *"curse of dimensionality"* – with DP, the computational effort blows up rapidly as the state dimension of the dynamical process grows. Usually state dimension 4 is already "too big".

Note: There is no "curse of dimensionality" in AARC!

Quiz: *What is state dimension in our toy inventory model?*

However: Reducing the number of factories to 1, increasing the per period capacity of the remaining factory to 1800 and making its cumulative capacity $+\infty$, we reduce the state dimension to 1 and make DP easily implementable. With this setup,

- the DP (that is, the "absolutely best") optimal value is 31270
- the AARC optimal value is 31514 – just by 0.8% worse!

ALGORITHMS OF LINEAR OPTIMIZATION

♣ The existing algorithmic "working horses" of LO fall into two major categories:

♠ Pivoting methods, primarily the *Simplex-type algorithms* which heavily exploit the polyhedral structure of LO programs, in particular, move along the vertices of the feasible set.

♠ Interior Point algorithms, primarily the *Primal-Dual Path-Following Methods*, much less "polyhedrally oriented" than the pivoting algorithms and, in particular, traveling along interior points of the feasible set of LO rather than along its vertices. In fact, IPM's have a much wider scope of applications than LO.

♠ Theoretically speaking (and modulo rounding errors), pivoting algorithms solve LO programs *exactly* in *finitely many* arithmetic operations. The operation count, however, can be astronomically large already for small LO's.

In contrast to the disastrously bad theoretical worst-case-oriented performance estimates, *Simplex-type algorithms seem to be extremely efficient in practice*. In 1940's — early 1990's these algorithms were, essentially, *the only* LO solution techniques.

♠ Interior Point algorithms, discovered in 1980's, entered LO practice in 1990's. These methods combine high practical performance (quite competitive with the one of pivoting algorithms) with nice theoretical worst-case-oriented efficiency guarantees.

Simplex Method – Executive Summary

S.I. Simplex method works with an LO problem in the *standard form*

$$\mathrm{Opt}(P) = \max_x \left\{ c^T x : Ax = b,\ x \geq 0 \right\} \qquad (P)$$

Standing Assumption:

- *A is an $m \times n$ matrix with linearly independent rows.*

$\Rightarrow$ *The system of linear equations $Ax = b$ has a solution* (not necessary nonnegative).

Note: This assumption is not restrictive: checking whether a system of linear *equations* is solvable is an easy task of Linear Algebra. When this is the case, it is equally easy to eliminate from A, one by one, rows which are linear combinations of the remaining rows, and this does not affect the solution set of the system.

Terminology: The set $\{x : Ax = b\}$ of solutions to the system of primal equations will be called the *primal feasible plane*.

$$\mathrm{Opt}(P) = \max_x \left\{ c^T x : Ax \underbrace{=}_{\lambda_e} b,\ x \underbrace{\geq}_{\lambda_g} 0 \right\} \qquad (P)$$

S.2. The problem dual to (P) reads

$$\mathrm{Opt}(D) = \min_{\lambda=[\lambda_g;\lambda_e]} \left\{ b^T\lambda_e : \lambda_g + A^T\lambda_e = c, \lambda_g \leq 0 \right\} \qquad (D)$$

Terminology: The set $\{\lambda = [\lambda_g; \lambda_e] : \lambda_g = c - A^T\lambda_\ell\}$ of solutions to the system of dual equations will be called the *dual feasible plane*. It always is nonempty.

Fact: *Under Standing Assumption, primal and dual feasible sets do not contain lines.* (why?) $\Rightarrow$ If (P), (D) are solvable, among the optimal solutions x^*, λ^* there are those which are extreme points of the respective feasible sets.

Fact: By Optimality Conditions, a pair

$$(x, \lambda = [\lambda_g; \lambda_e])$$

of *feasible* solutions to (P), (D) is comprised of optimal solutions to the respective problems iff the solutions are *complementary*:

$$(\lambda_g)_j x_j = 0,\ j = 1, ..., n.$$

Intermediate Summary:

- In order to find optimal solutions to (P), (D), we need to ensure *primal-dual feasibility* and *complementarity*.
- When achieving this goal, we can work with candidates to the role of extreme point solutions. The key role in the description of these candidates is played by the notion of a *basis* of A.

$$\begin{array}{rcll} \mathrm{Opt}(P) & = & \max_x \left\{c^T x : Ax = b,\, x \geq 0\right\} & (P) \\ \mathrm{Opt}(D) & = & \min_{\lambda=[\lambda_g;\lambda_e]} \left\{b^T \lambda_e : \lambda_g + A^T \lambda_e = c, \lambda_g \leq 0\right\} & (D) \end{array}$$

Definition: *A subset J of m distinct from each other indexes of columns in the $m \times n$ matrix A is called basis, if these m columns A_j, $j \in J$, are linearly independent or, which is the same, the $m \times m$ submatrix*

$$A_J = [A_{j_1}, A_{j_2}, ..., A_{j_m}]$$
$$[J = \{j_1, ..., j_m\},\ A_j \text{ is } j\text{-th column of } A]$$

is invertible.

Simple facts: For every basis J of A, there exists

— *exactly one solution x^J to the system $Ax = b$ of primal equations for which all nonbasic – with indexes not from J – entries are zeros.* This solution is called *basic primal solution* associated with basis J.

The basic part of x^J is $[A_J]^{-1}b$.

— *exactly one solution $\lambda^J = [\lambda_g^J; \lambda_e^J]$ to the system $\lambda_g = c - A^T\lambda_e$ of dual equations for which all basic entries in λ_g are zero.* This solution is called *basic dual solution* associated with basis J, and its λ_g-component is called *vector of reduced costs* associated with J.

λ^J is given by $\lambda_e^J = [A_J^T]^{-1}c_J$, $\lambda_g^J = c - A^T\lambda_e^J$, where c_J is comprised of basic entries of c (those with indexes from J).

♠ **Important observation:** Basic primal and dual solutions associated with the same basis always are complementary.

$$\begin{array}{rcll} \mathrm{Opt}(P) & = & \max_x \left\{c^T x : Ax = b,\, x \geq 0\right\} & (P) \\ \mathrm{Opt}(D) & = & \min_{\lambda=[\lambda_g;\lambda_e]} \left\{b^T \lambda_e : \lambda_g + A^T \lambda_e = c, \lambda_g \leq 0\right\} & (D) \end{array}$$

S.III. Crucial fact: *Under Standing Assumptions, the extreme points of the primal and the dual feasible sets are "parameterized" by bases of A:*

- *extreme points of the primal feasible set are exactly basic primal solutions which happen to be primal feasible (i.e., non-negative);*
- *extreme points of the dual feasible set are exactly basic dual solutions which happen to be dual feasible (i.e., to have non-positive reduced costs).*

$$\begin{array}{rcll} \mathrm{Opt}(P) & = & \max_x \left\{c^T x : Ax = b,\, x \geq 0\right\} & (P) \\ \mathrm{Opt}(D) & = & \min_{\lambda=[\lambda_g;\lambda_e]} \left\{b^T\lambda_e : \lambda_g + A^T\lambda_e = c, \lambda_g \leq 0\right\} & (D) \end{array}$$

S.IV. Simplex strategy:

♠ In the **Primal Simplex method**, we build a sequence of *feasible basic primal solutions* along with sequence of (perhaps infeasible) basic dual solutions associated with the same bases (and thus complementary to our primal solutions).

• The consecutive bases we build are *neighbouring*: each time we drop out one "old" basic index and make basic one "old" *non*basic index.

• The process terminates when

— either the current basic dual solution becomes feasible $\Rightarrow$ we get a pair of complementary primal-dual feasible (and thus optimal) solutions

— or unboundedness of (P) (i.e., infeasibility of (D)) is discovered.

$$\begin{array}{rcll} \mathrm{Opt}(P) & = & \max_x \left\{c^T x : Ax = b,\, x \geq 0\right\} & (P) \\ \mathrm{Opt}(D) & = & \min_{\lambda=[\lambda_g;\lambda_e]} \left\{b^T \lambda_e : \lambda_g + A^T \lambda_e = c, \lambda_g \leq 0\right\} & (D) \end{array}$$

Simplex strategy (continued):

♠ In the **Dual Simplex method**, we build a sequence of *feasible basic dual solutions* along with sequence of (perhaps infeasible) basic primal solutions associated with the same bases (and thus complementary to our dual solutions).

- The consecutive bases we build are *neighbouring*.
- The process terminates when

— either the current basic primal solution becomes feasible $\Rightarrow$ we end up with a pair of primal-dual feasible complementary (and thus optimal) solutions

— or unboundedness of (D) (and thus infeasibility of (P)) is discovered.

Implementing the Strategy: Primal Simplex

♠ At the beginning of a step, we have at our disposal

- current basis J
- associated with J *feasible* basic primal solution $\bar{x}$

♠ We start the step with computing the associated with J basic dual solution $\bar{\lambda}$, in particular, the vector of reduced costs $\bar{\lambda}_g$.

♡ It may happen that $\bar{\lambda}$ is dual feasible: $\bar{\lambda}_g \leq 0 \Rightarrow$ *we have a complementary pair of primal-dual feasible (and thus – primal-dual optimal) solutions $\bar{x}$, $\bar{\lambda}$ and terminate.*

Otherwise we proceed with the step.

$$\begin{array}{rcll} \mathrm{Opt}(P) & = & \max_x \left\{c^T x : Ax = b,\, x \geq 0\right\} & (P) \\ \mathrm{Opt}(D) & = & \min_{\lambda=[\lambda_g;\lambda_e]} \left\{b^T\lambda_e : \lambda_g + A^T\lambda_e = c, \lambda_g \leq 0\right\} & (D) \end{array}$$

Situation: We have at our disposal basis J, associated feasible basic primal solution $\bar{x}$, and associated vector $\bar{\lambda}_g$ of reduced costs *with some of the reduced costs positive.*

♠ We select index j_* of a positive reduced cost, and try to pass to a better feasible primal solution.

Note: j_* is nonbasic (basic reduced costs are zeros!)

Note: When replacing the original costs c with the vector of reduced costs $\bar{\lambda}_g$, we get an equivalent problem: on the entire primal feasible plane, the objective is just shifted by a constant.

Indeed, $\bar{\lambda}_g$ differs from c by a linear combination of rows a_i of A, and the linear forms $a_i^T x$ are constant on the primal feasible plane.

• Let us try to replace the j_*-th entry in $\bar{x}$ (which is zero) with some $t \geq 0$, *compensating this change by updating basic entries in $\bar{x}$ in order to satisfy the primal equations.* We get a ray $\{x(t) : t \geq 0\}$ in the primal feasible plane such that

$$\begin{cases} \text{basic part } x^J(t) \text{ of } x(t) \text{ is affine in } t \\ x_{j_*}(t) \equiv t \\ x_j(t) \equiv 0 \text{ for all nonbasic } j \text{ different from } j_* \end{cases}$$

Note: *When moving along the ray $\{x(t) : t \geq 0\}$ and increasing t, the primal objective strictly grows:*

$$c^T(x(t) - x(0)) = \bar{\lambda}_g^T(x(t) - x(0)) = \underbrace{(\bar{\lambda}_g)_{j_*}}_{>0} t$$

$$\begin{array}{rcll} \mathrm{Opt}(P) & = & \max_x \left\{c^T x : Ax = b,\, x \geq 0\right\} & (P) \\ \mathrm{Opt}(D) & = & \min_{\lambda=[\lambda_g;\lambda_e]} \left\{b^T\lambda_e : \lambda_g + A^T\lambda_e = c, \lambda_g \leq 0\right\} & (D) \end{array}$$

Situation: We have at our disposal basis J, associated feasible basic primal solution $\bar{x}$, associated reduced costs $\bar{\lambda}_g$, index j_* of a positive reduced cost, and an "improving ray" $\{x(t) : t \geq 0\}$ with the following properties:

- the ray lies in the primal feasible plane and emanates from $\bar{x}$;
- when moving along the ray, the primal objective grows;
- along the ray:

— j_*-th coordinate of $x(t)$ is t,

— basic coordinates in $x(t)$ affinely depend on t,

— all other coordinates in $x(t)$ stay zeros.

A. It may happen that as t grows, all basic coordinates in $x(t)$ stay nonnegative $\Rightarrow$ *we have discovered a primal feasible ray along which the primal objective goes to* $+\infty \Rightarrow (P)$ *is unbounded,* (D) *is infeasible, we terminate.*

$$\begin{array}{rcll} \mathrm{Opt}(P) & = & \max_x \left\{c^T x : Ax = b,\, x \geq 0\right\} & (P) \\ \mathrm{Opt}(D) & = & \min_{\lambda=[\lambda_g;\lambda_e]} \left\{b^T\lambda_e : \lambda_g + A^T\lambda_e = c, \lambda_g \leq 0\right\} & (D) \end{array}$$

Situation: We have at our disposal basis J, associated feasible basic primal solution $\bar{x}$, associated reduced costs $\bar{\lambda}_g$, index j_* of positive reduced cost, and an "improving ray" $x(t)$ with the following properties:

- the ray lies in the dual feasible plane and emanates from $\bar{\lambda}$;
- when moving along the ray, the dual objective strictly decreases;
- along the ray:

— j_*-th coordinate of x_t is t,

— basic coordinates in $x(t)$ affinely depend on t,

— all other coordinates in $x(t)$ stay zeros.

B. It may happen that as t grows, some basic coordinates in $x(t)$ (all nonnegative at $t = 0$!) eventually become negative.

- We identify the largest $t = \bar{t}$ for which all basic coordinates in $x(t)$ still are nonnegative. When $t = \bar{t}$, one of the basic coordinates of $x(t)$, let its index be i_*, "is about to become negative" – $x_{i_*}(\bar{t}) = 0$ and $x_{i_*}(t) < 0$ when $t > \bar{t}$.
- We take
  - $J^+ = [J - \{i_*\}] \cup \{j_*\}$ as our new basis – "i_* leaves the basis, j_* enters the basis" (it can be shown that J^+ indeed is a basis),
  - $x(\bar{t})$ as the basic solution associated with J^+ (it indeed is so!),

compute the basic dual solution associated with J^+ and pass to the next step.

Note: When passing to the next step, we

— *strictly improve the primal objective, if* $\bar{t} > 0$ (which definitely is the case when all basic entries in $\bar{x}$ are positive; such a basic solution $\bar{x}$ is called *nondegenerate*)

— *keep the basic solution and the value of the objective intact, if* $\bar{t} = 0$. This may happen only when $\bar{x}$ is degenerate.

In all cases, the basis does change.

Conclusion: *If all feasible basic primal solutions are nondegenerate, no one of them can be visited twice* (since the primal objective strictly grows at every step)

$\Rightarrow$ *We terminate with primal and dual optimal solutions (or with certificate of primal unboundedness) after finitely many steps*

Indeed, there are finitely many feasible basic primal solutions, and no one of them can be visited twice.

However: When the problem admits degenerate feasible basic primal solutions, the method can "loop for ever" – after several steps in which the primal feasible basic solution remains intact, and only basis changes, we can come back to the basis we started with and then "loop forever."

Remedies:

- In problems with "real data," chances to meet degeneracy are nonexistent.
- In problems of combinatorial origin, where the entries in A are moderate integers, the chances to meet degeneracy could be high, but actual cycling is a very rare phenomenon. Thus, cycling is *not* a practical issue.
- It often happens that there are several candidates to be entered to/discarded from the basis. It turns out that properly selected rules for "resolving ties" *provably eliminate* cycling.

How to Find Initial Feasible Primal Basic Solution?

$$\mathrm{Opt}(P) \quad = \quad \max_x \left\{c^T x : Ax = b,\, x \geq 0\right\} \quad (P)$$

♠ In order to start Primal Simplex, we need an initial basis associated with a *feasible* basic solution to (P). The standard way to achieve this goal is to run *Phase 0* as follows.

• Multiplying, if necessary, the equations of (P) by -1, we can ensure that $b \geq 0$.

• Consider auxiliary LO program in variables x, s:

$$\min_{x,s} \left\{\sum_{i=1}^m s_i : Ax + s = b,\, x \geq 0, s \geq 0\right\}$$

Note:

• Problem is in the standard form, and a feasible basic solution to it is readily available: the basis is comprised of the indexes of s-variables, and the basic part of the basic solution is just b.

• The optimal value in the problem is either 0 (meaning that (P) is feasible), or is strictly positive (meaning that (P) is infeasible).

$\Rightarrow$ *Solving the auxiliary problem by Primal Simplex, we find out whether (P) is feasible, and if it is the case, at the optimal solution x^*, s^* we have $s^* = 0$, meaning that x^* is a basic feasible solution to (P). We can now solve (P) starting with this feasible basic solution and corresponding basis.*

"Column Generation"

$$\mathrm{Opt}(P) \quad = \quad \max_x \left\{c^T x : Ax = b,\, x \geq 0\right\} \quad (P)$$

♣ When solving an $m \times n$ standard form maximization LO by Primal Simplex, computational effort per step reduces to

A. Identifying, given current basis J, the index j_* of "bad" – positive – reduced cost, if any exists;

B. Updating the basis J and the associated basic solution, provided positive reduced cost was found.

Note: *The only part of A which participates in a step is comprised of the "old" basic columns A_j, $j \in J$, and the column A_{j_*}.*

This is in sharp contrast with computing the vector of reduced costs – this computation requires to process *every one* of the columns of A.

$$\mathrm{Opt}(P) \quad = \quad \max_x \left\{ c^T x : Ax = b,\, x \geq 0 \right\} \quad (P)$$

♠ Given basis J and feasible basic solution, we want

A. To identify the index j_* of a positive reduced cost, or to conclude that no such cost exists;

B. To update the basis and the basic feasible solution, provided a positive reduced cost was found.

♠ With good implementation, task **B** requires at most $O(m^2)$ arithmetic operations; this cost is not affected by the magnitude of n. **B** is "doable" when m is "moderate" (with modern hardware, tens of thousands and perhaps even millions, but not billions or billions of billions)

♡ In contrast to this, *computing the entire vector of reduced costs takes something like $O(m^2 n)$ arithmetic operations*. For problem in standard form, $n \geq m$ (why?), and in typical applications $n \gg m$. *What to do when n is huge, so that computing the entire vector of reduced costs becomes prohibitively time consuming?*

Note: In some applications, n is that large, that we just cannot store the matrix A, or even a single n-dimensional vector.

$$\text{Opt}(P) \;=\; \max_x \left\{ c^T x : Ax = b,\, x \geq 0 \right\} \quad (P)$$

♠ **Observation.** In applications, huge n usually means that the constraint matrix A is given by a specific "short description" rather than by the standard listing of nonzero entries and their indexes (for a huge matrix, where can we take data to fill such a list, unless the matrix is "well organized" ?)

♠ **The idea** of column generation is to use "short description" of columns of a "well-organized" A in order to identify the column with positive reduced cost (or to conclude that no such column exists), thus avoiding computing the entire vector of reduced costs.

Implementation of this idea is "problem specific" – it depends on what is the "short description" of A.

$$\mathrm{Opt}(P) \quad = \quad \max_x \left\{c^T x : Ax = b,\, x \geq 0\right\} \quad (P)$$

♠ Given basis J, the vector of reduced costs is $\lambda_g = c - A^T[A_J^T]^{-1}c_J$, where c_J is the basic part of c.

$\Rightarrow (\lambda_g)_j = c_j - e^T A_j$, where $e = [A_J^T]^{-1}c_J$ and A_j is j-th column of A.

Note: *Computing e requires to operate with basic columns of A and basic entries of c only!*

♠ After e is computed, identifying the index of a positive reduced cost, if any, reduces to solving the discrete optimization problem

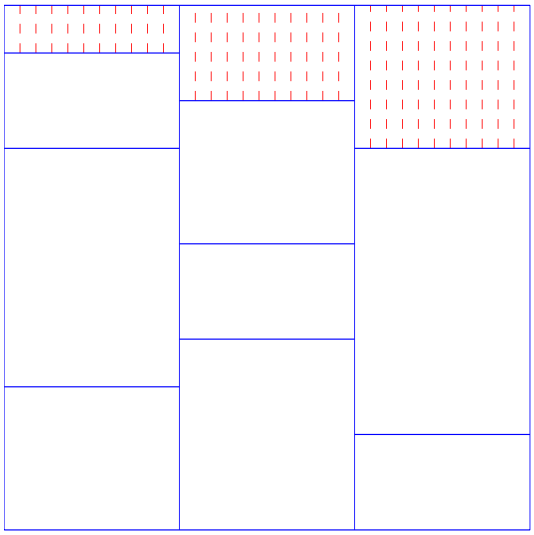
$$\max_j \left[c_j - e^T A_j\right] \qquad (!)$$

When A admits a good description, utilizing this description could result in a much more efficient algorithm for solving (!) than the exhaustive search through all the columns A_j!

Example: Cutting Stock problem. We should produce metal rectangle plates of m types. A plate of type j must have width w_j and height h_j, and we need b_j of these plates, $1 \leq j \leq m$.

• Plates are cut off a band with height H and infinite width. *How to arrange the plates on the band in order to minimize the waste?*

♠ We can group the plates according to their width and solve the problem for every one of the groups, thus reducing it to the case when all the plates have common width w and distinct from each other heights; this is called the *Cutting Stock* problem. In this case, we can split the band into vertical rectangles of width w and decide what should be "plate patterns" in every rectangle – how many plates of type $i = 1, 2, ...$ we place on the rectangle.



3 plate patterns in 3 vertical rectangles, Red: waste

♠ Let us arrange heights h_i of plates of types $i = 1, 2, ..., m$ into m-dimensional vector h, and identify a plate pattern with m-dimensional vector $p = [p_1; ...; p_m]$, where p_i is the number of plates of type i in the pattern.

Example: Assume we have $m = 3$ types of plates with heights $h_1 = 10$, $h_2 = 20$, $h_3 = 30$. In this case, $h = [10; 20; 30]$. Pattern $p = [1; 2; 1]$ describes vertical rectangle from which we cut

— 1 plate of height $h_1 = 10$

— 2 plates of height $h_2 = 20$

— 1 plate of height $h_3 = 30$.

Note: The total height of plates in pattern p is

$$p_1 h_1 + p_2 h_2 + ... + p_m h_m = h^T p$$

$\Rightarrow$ *Feasible patterns are nonnegative integer vectors p satisfying $h^T p \leq H$*

Example (continued): In our Example, pattern $p = [1; 2; 1]$ to be feasible requires the height of the band to be at least

$$1 \cdot 10 + 2 \cdot 20 + 1 \cdot 30 = h^T p = 80$$

♠ Let n be the number of all feasible plate patterns, let A_j, $j = 1, ..., n$, be the list of all these patterns; we think of $A_1, ..., A_n$ as of the columns of $m \times n$ matrix

$$A = [A_1, ..., A_n]$$

Quiz: Let there be $m = 3$ types of plates with heights

$$h_1 = 10, h_2 = 20, h_3 = 30,$$

and let $H = 80$. Which of the following vectors are columns of A ?

- $\begin{bmatrix} 3 \\ 1 \\ 1 \end{bmatrix}$
- $\begin{bmatrix} 0 \\ 0 \\ 1 \\ 1 \end{bmatrix}$
- $\begin{bmatrix} 4 \\ 1 \\ 1 \end{bmatrix}$
- $\begin{bmatrix} 1 \\ 1 \\ 2 \end{bmatrix}$
- $\begin{bmatrix} 0 \\ 1 \\ 2 \end{bmatrix}$

• Let our decision variables x_j be the numbers of rectangles where patterns A_j will be used.

• The total yield of plates of type i is $(Ax)_i$, and we should have $Ax = b$

• The waste in a rectangle where we use plate pattern A_j is $H - h^T A_j$

$\Rightarrow$ The total waste is

$$\sum_j [H - h^T A_j] x_j$$

$\Rightarrow$ Our problem (in maximization form: maximize minus waste) becomes

$$\max_x \left\{ \sum_j \underbrace{[h^T A_j - H]}_{c_j} x_j : Ax = b,\ x \geq 0 \right\} \qquad (*)$$

♠ Our problem becomes

$$\max_x \left\{ \sum_j \underbrace{[h^T A_j - H]}_{c_j} x_j : Ax = b,\ x \geq 0 \right\} \qquad (*)$$

Note: We skip the natural requirement that x should be integer. This can be justified when

— we are speaking about mass production and can expect that in the optimal solution, nonzero x_j's will be large, and their rounding will not make much harm, or

– we are solving the problem with integrality constraints on x_j, and (!) is the relaxation of the "true" problem generated by the master branch and bound algorithm.

$$\max_x \left\{ \sum_j \underbrace{[h^T A_j - H]}_{c_j} x_j : Ax = b,\, x \geq 0 \right\} \qquad (*)$$

♠ When h_i are small as compared to H, n could be astronomically large. However, the problem of identifying the largest reduced cost:

$$\begin{aligned}
&\max_j [c_j - e^T A_j] \\
&= \max_j \left[\underbrace{h^T A_j - H}_{c_j} - e^T A_j \right] \\
&= \max_p \left\{ [h - e]^T p - H : p \geq 0,\, h^T p \leq H, p \text{ is integer} \right\}
\end{aligned}$$

is a knapsack type problem which can be solved efficiently by Dynamic Programming.

Quiz: Let $m = 20$, $h = [1; 2; ...; 20]$, $H = 100$. In your opinion, how large is the number n of all feasible plate patterns?

- $\leq 100,000$
- $\leq 1,000,000$
- $\leq 10,000,000$
- $\leq 100,000,000$
- $\leq 1,000,000,000$

Answer: $n = 928,321,174$

Dual Simplex Method

$$\begin{array}{rcll} \mathrm{Opt}(P) & = & \max_x \left\{c^T x : Ax = b,\, x \geq 0\right\} & (P) \\ \mathrm{Opt}(D) & = & \min_{\lambda=[\lambda_g;\lambda_e]} \left\{b^T \lambda_e : \lambda_g + A^T \lambda_e = c, \lambda_g \leq 0\right\} & (D) \end{array}$$

♠ In the **Dual Simplex method**, we build a sequence of *feasible basic dual solutions* along with sequence of (perhaps infeasible) basic primal solutions associated with the same bases (and thus complementary to our dual solutions).

- The consecutive bases we build are *neighbouring*.
- The process terminates when

— either the current basic primal solution becomes feasible $\Rightarrow$ we end up with a pair of primal-dual feasible complementary (and thus optimal) solutions

— or unboundedness of (D) (and thus infeasibility of (P)) is discovered.

Question: *Due to primal-dual symmetry, Dual Simplex looks exactly the same as Primal Simplex, with swapped primal and dual problems. Why a separate algorithm is necessary?*

Answer: *Geometrically*, primal-dual symmetry indeed is perfect, but *algorithmically* it is not. Algorithm works with *an analytical description* of a problem, and not with the problem as a geometrical entity! And analytically, (P) and (D) are in different formats...

$\Rightarrow$ *Algorithmic description of Primal Simplex cannot be "literally translated" into the description of Dual Simplex...*

$$\begin{array}{rcll} \mathrm{Opt}(P) & = & \max_x \left\{c^T x : Ax = b,\, x \geq 0\right\} & (P) \\ \mathrm{Opt}(D) & = & \min_{\lambda=[\lambda_g;\lambda_e]} \left\{b^T\lambda_e : \lambda_g + A^T\lambda_e = c, \lambda_g \leq 0\right\} & (D) \end{array}$$

♠ At the beginning of a step of Dual Simplex, we have at our disposal

- basis J
- *feasible* basic dual solution $\bar{\lambda} = [\bar{\lambda}_g; \bar{\lambda}_e]$ associated with the basis.

♠ We start the step with computing the basic primal solution $\bar{x}$ associated with J.

♡ It may happen that $\bar{x}$ is primal feasible: $\bar{x} \geq 0 \Rightarrow$ *we have a complementary pair of primal-dual feasible (and thus – primal-dual optimal) solutions $\bar{x}$, $\bar{\lambda}$ and terminate.*

Otherwise we proceed with the step.

$$\begin{array}{rcll} \mathrm{Opt}(P) & = & \max_x \left\{c^T x : Ax = b,\, x \geq 0\right\} & (P) \\ \mathrm{Opt}(D) & = & \min_{\lambda=[\lambda_g;\lambda_e]} \left\{b^T\lambda_e : \lambda_g + A^T\lambda_e = c, \lambda_g \leq 0\right\} & (D) \end{array}$$

Situation: We have at our disposal basis J, associated dual *feasible* basic solution $\bar{\lambda} = [\bar{\lambda}_g; \bar{\lambda}_e]$, and associated with J basic primal solution $\bar{x}$ with *not all* entries in x positive.

♠ We select index j_* of a negative entry in $\bar{x}$, and try to pass to a better feasible dual solution.

Note: j_* is basic (nonbasic entries in $\bar{x}$ are zeros!)

Note: When replacing the dual objective $b^T\lambda_e$ with the objective $-\bar{x}^T\lambda_g$, we get an equivalent problem: on the entire dual feasible plane, the dual objective is just shifted by a constant.

Indeed, on the dual feasible plane we have

$$\bar{x}^T[A^T\lambda_e] = \bar{x}^T c - \bar{x}^T\lambda_g$$

which, due to $A\bar{x} = b$, reads $-\bar{x}^T\lambda_g = b^T\lambda_e + \mathrm{const}$.

• Let us try to replace j_*-th entry in $\bar{\lambda}_g$ (which is zero) with some $t \leq 0$, *compensating this change by updating nonbasic entries in $\bar{\lambda}_g$ and updating λ_e in order to satisfy the dual equations.* We get a *ray* $\{\lambda(t) = [\lambda_g(t); \lambda_e(t)] : t \leq 0\}$ *in the dual feasible plane* such that

$$\begin{cases} \text{nonbasic part } \lambda_g^J(t) \text{ of } \lambda_g(t) \text{ is affine in } t \\ (\lambda_g(t))_{j_*} \equiv t \\ (\lambda_g(t))_j \equiv 0 \quad \text{for all basic } j \text{ different from } j_* \end{cases}$$

Note: *When moving along the ray $\{\lambda(t) : t \leq 0\}$ and decreasing t, the dual objective strictly decreases:*

$$b^T(\lambda_e(t) - \lambda_e(0)) = [-\bar{x}]^T(\lambda_g(t) - \lambda_g(0)) = \underbrace{-\bar{x}_{j_*}}_{>0} t$$

$$\begin{array}{rcll} \mathrm{Opt}(P) & = & \max_x \left\{c^T x : Ax = b,\, x \geq 0\right\} & (P) \\ \mathrm{Opt}(D) & = & \min_{\lambda=[\lambda_g;\lambda_e]} \left\{b^T\lambda_e : \lambda_g + A^T\lambda_e = c, \lambda_g \leq 0\right\} & (D) \end{array}$$

Situation: We have at our disposal basis J, associated feasible basic dual solution $\bar{\lambda} = [\bar{\lambda}_g; \bar{\lambda}_e]$, associated basic primal solution $\bar{x}$, index j_* of negative entry in $\bar{x}$ and an "improving ray" $\{\lambda(t) : t \leq 0\}$ with the following properties:

- the ray lies in the dual feasible plane and emanates from $\bar{\lambda}$;
- when moving along the ray, the dual objective strictly decreases;
- along the ray:

— j_*-th coordinate of $\lambda_g(t)$ is t,

— nonbasic coordinates of $\lambda_g(t)$ affinely depend on t,

— all other coordinates in $\lambda_g(t)$ stay zeros.

A. It may happen that as $t \leq 0$ decreases, all nonbasic coordinates in $\lambda_g(t)$ stay nonpositive $\Rightarrow$ *we have discovered a dual feasible ray along which the dual objective goes to* $-\infty$ $\Rightarrow$ (D) *is unbounded,* (P) *is infeasible, we terminate.*

$$\begin{array}{rcll} \mathrm{Opt}(P) & = & \max_x \left\{c^T x : Ax = b,\, x \geq 0\right\} & (P) \\ \mathrm{Opt}(D) & = & \min_{\lambda=[\lambda_g;\lambda_e]} \left\{b^T\lambda_e : \lambda_g + A^T\lambda_e = c, \lambda_g \leq 0\right\} & (D) \end{array}$$

Situation: We have at our disposal basis J, associated feasible basic dual solution $\bar{\lambda} = [\bar{\lambda}_g; \bar{\lambda}_e]$, associated basic primal solution $\bar{x}$, index j_* of negative entry in $\bar{x}$ and an "improving ray" $\{\lambda(t) : t \leq 0\}$ with the following properties:

- the ray lies in the dual feasible plane and emanates from $\bar{\lambda}$;
- when moving along the ray, the dual objective strictly decreases;
- along the ray:

— j_*-th coordinate of $\lambda_g(t)$ is t,

— nonbasic coordinates of $\lambda_g(t)$ affinely depend on t,

— all other coordinates in $\lambda_g(t)$ stay zeros.

B. It may happen that as $t \leq 0$ decreases, some nonbasic coordinates $\lambda_g(t)$ (all nonpositive at $t = 0$!) eventually become positive.

- We identify the smallest $t = \bar{t} \leq 0$ for which all nonbasic coordinates in $\lambda_g(t)$ still are nonpositive. When $t = \bar{t}$, one of the nonbasic coordinates in $\lambda_g(t)$, let its index be i_*, "is about to become positive" – $(\lambda_g(\bar{t}))_{i_*} = 0$ and $(\lambda_g(t))_{i_*} > 0$ when $t < \bar{t}$.
- We take
  - $J^+ = [J - \{j_*\}] \cup \{i_*\}$ as our new basis – "j_* leaves the basis, i_* enters the basis" (it can be shown that J^+ indeed is a basis),
  - $\lambda(\bar{t})$ as the feasible basic dual solution associated with J^+ (it indeed is so!),

compute the basic primal solution associated with J^+ and pass to the next step.

Note: When passing to the next step, we

— *strictly improve the dual objective, if* $\bar{t} < 0$ (which definitely is the case when all nonbasic entries in $\bar{\lambda}_g$ are negative; such a basic dual solution $\bar{\lambda}$ is called *nondegenerate*)

— *keep the basic dual solution and the value of the dual objective intact, if* $\bar{t} = 0$. This may happen only when $\bar{\lambda}$ is degenerate.

In all cases, the basis does change.

Conclusion: *If all basic feasible dual solutions are nondegenerate, no one of them can be visited twice* (since the dual objective strictly decreases at every step)

$\Rightarrow$ *We terminate with primal and dual optimal solutions (or with certificate of dual unboundedness) after finitely many steps*

Indeed, there are finitely many basic feasible dual solutions, and no one of them can be visited twice.

However: When the problem admits degenerate dual solutions, the method can "loop for ever" – after several steps in which the dual feasible basic solution remains intact, and only basis changes, we can come back to the basis we started with and then "loop forever."

Part II: General Convex Optimization

- Convex sets, functions and problems
- Lagrange Duality and Optimality Conditions
- Ellipsoid Method

Convex Optimization Program

♣ A Convex Optimization problem is an extension of a LO problem

$$\text{Opt} = \min_{x \in \mathbb{R}^n} \left\{ c^T x : a_i^T x - b_i \leq 0,\ 1 \leq i \leq m \right\} \qquad (LO)$$

obtained by replacing the linear objective $c^T x$ by a *convex* objective $f(x)$, and the affine constraints $a_i^T x - b_i \leq 0$ – with convex constraints $g_i(x) \leq 0$, $i = 1, ..., m$, "convexity" of constraint $g_i(x) \leq 0$ being a shortcut for "*the left hand side $g_i(x)$ of the constraint is a convex function*." It makes sense also to add to the formulation of the problem the *domain constraint* $x \in X$, where X is a given convex set.

$\Rightarrow$ *Convex Optimization problem is a Mathematical Programming problem*

$$\text{Opt} = \min_{x \in X \subset \mathbb{R}^n} \{ f(x); g_i(x) \leq 0, i = 1, ..., m \} \qquad (P)$$

where the objective f, and the left hand sides $g_i(x)$ of the constraints $g_i(x) \leq 0$ are convex functions, and the domain $X \subset \mathbb{R}^n$ is convex.

$$\mathrm{Opt} = \min_{x \in X \subset \mathbb{R}^n} \{f(x); g_i(x) \leq 0, i = 1, ..., m\} \qquad (P)$$

Note: Usually, the domain X of (P) itself is given by a bunch of convex constraints

$\Rightarrow$ Adding the constraints describing X to the list of "functional constraints" $g_i(x) \leq 0$, we can "get rid" of the domain – to make $X = \mathbb{R}^n$.

However: "can" is not the same as "should." In some cases, presence of the domain is convenient, and we keep it in the problem.

Review of Elementary Calculus

♣ In the LO part of our course, our "working horse" was elementary Linear Algebra.

Just formulating facts about Convex Programming requires (minimal!) portion of Analysis. Review of this portion is as follows:

♣ *A sequence of vectors $x_t \in \mathbb{R}^n$, $t = 1, 2, ...$, is called converging to a vector $\bar{x}$, if for every positive r, all x_t, for all large enough values of t are at the distance at most r of $\bar{x}$:*

$$\forall r > 0 \exists t = t(r) : \|x_t - \bar{x}\|_2 \leq r \text{ whenever } t \geq t(r)$$

In this situation, $\bar{x}$ is called the limit of the system, denoted by $\bar{x} = \lim_{t\to\infty} x_t$, or by "$x_t \to \bar{x}$ as $t \to \infty$"..

Note:

• A sequence $x_1, x_2, ...$ of vectors converges to $\bar{x}$ iff the sequence of reals $\|x_t - \bar{x}\|_2$ converges to 0:

$$\bar{x} = \lim_{t\to\infty} x_t \quad \Leftrightarrow \quad \|x_t - \bar{x}\|_2 \to 0 \text{as } t \to \infty$$

• A sequence which has a limit is called *converging*. A converging sequence has *exactly one* limit.

♣ **Let X be a set in $\mathbb{R}^n$**

♠ *X is called closed, if it contains the limits of all converging sequences of its elements:*

$$\text{if } x_t \in X, t = 1, 2, ... \text{ and } x_t \to \bar{x} \text{ as } t \to \infty, \text{ then } \bar{x} \in X$$

Facts: When adding to X the limits of all converging sequences of the members of X, we get a *closed* set called the *closure* of X and denoted $\mathrm{cl}X$. A set is closed iff it coincides with its closure.

Informally: $\mathrm{cl}X$ *is the set of all points x which can be approximated to whatever high accuracy by points from X.*

♠ *A point x is called an interior point of X (notation: $x \in \mathrm{int}\, X$), if a ball of some positive radius centered at x is contained to X:*

$$\{x \in \mathrm{int}\, X\} \Leftrightarrow \{\exists r > 0 : B_r(x) := \{y : \|x - y\|_2 \le r\} \subset X\}$$

Informally: *The fact that $x \in \mathrm{int}\, X$ means that $x \in X$ and this inclusion is robust: all points close enough to x belong to X as well. Equivalently: $x \in \mathrm{int}\, X$ iff x cannot be approximated to high enough accuracy by points outside of X.*

- *The set of all interior points of X is called the interior of X, denoted $\mathrm{int}\, X$*
- *X is called open if every point of X is its interior point: $X = \mathrm{int}\, X$*

♠ *We always have* $\mathrm{int}\, X \subset X \subset \mathrm{cl} X$. *For open sets, the left of these inclusions is equality. For closed sets, the right of these inclusions is equality.*

• *The complement of the interior in the closure is called the boundary of* X, *denoted* ∂X:

$$\partial X = (\mathrm{cl} X) \backslash (\mathrm{int}\, X) = \{x : x \in \mathrm{cl} X \ \& \ x \notin \mathrm{int}\, X\}$$

Informally: *The boundary of* X *is the set of all points* x *which can be approximated to whatever high accuracy both by points from* X *and points from outside of* X.

Quiz: *What can be said about closedness, openess, closure, interior, and boundary of the following sets in* $\mathbb{R}^n$*:*

- $X = \emptyset$
- $X = \mathbb{R}^n$
- $X = [0, 1] := \{x : 0 \leq x \leq 1\} \subset \mathbb{R}$
- $X = (0, 1] := \{x : 0 < x \leq 1\} \subset \mathbb{R}$
- $X = (0, 1) := \{x : 0 < x < 1\} \subset \mathbb{R}$

Quiz: *What can be said about closedness, openness, closure, interior, and boundary of the following sets in* $\mathbb{R}^n$*:*

- $X = \emptyset$

$\emptyset$ is both closed and open and thus coincides with its interior, its closure and its boundary: all these sets are empty!

- $X = \mathbb{R}^n$

$\mathbb{R}^n$ is both closed and open (and thus coincides with its interior and its closure); the boundary of $\mathbb{R}^n$ is empty.

Note: *$\emptyset$ and $\mathbb{R}^n$ are the only subsets of $\mathbb{R}^n$ which are both open and closed!*

- $X = [0,1] := \{x : 0 \leq x \leq 1\} \subset \mathbb{R}$

X is closed and is not open; $\mathrm{cl}X = X = [0,1]$, $\mathrm{int}\, X = (0,1) := \{x : 0 < x < 1\}$, $\partial X = \{0,1\}$.

- $X = (0,1] := \{x : 0 < x \leq 1\} \subset \mathbb{R}$

X is neither closed nor open, $\mathrm{cl}X = [0,1]$, $\mathrm{int}\, X = (0,1)$, $\partial X = \{0,1\}$

- $X = \{x : 0 < x < 1\} \subset \mathbb{R}$

X is open and not closed, $\mathrm{cl}X = [0,1]$, $\mathrm{int}\, X = X = (0,1)$, $\partial X = \{0,1\}$

Elementary Calculus of Closedness and Openess

♣ **Facts:** *When speaking about subsets of common "universe" $\mathbb{R}^n$,*

- *closed sets are exactly the complements of open sets:*

$$X \text{ is closed iff } (\mathbb{R}^n \backslash X) \text{ is open}$$

- *intersection of whatever family of closed sets is closed*
- *union of whatever family of open sets is open*
- *finite unions of closed sets are closed*
- *finite intersections of open sets are open.*
- *the closure of a set X is the intersection of all closed sets containing X, and thus is the smallest closed set containing X – whenever a closed set Y contains X, Y contains $\mathrm{cl}X$ as well.*

Continuity

♣ **Let X be a subset of $\mathbb{R}^n$ and $f(x)$ be a real-valued function defined on X.**

♠ *f is called continuous on X at a point $\bar{x} \in X$, if for every sequence x_t of points from X converging to $\bar{x}$, $f(x_t)$ converges to $f(\bar{x})$ as $t \to \infty$*

Informally: *f is continuous on X at a point $\bar{x} \in X$, if $f(x)$ approximates $f(\bar{x})$ to a whatever high desired accuracy, provided that $x \in X$ is close enough to $\bar{x}$.*

Formally: *f is continuous on x at $\bar{x} \in X$, iff for every $\epsilon > 0$ there exists $\delta = \delta(\epsilon) > 0$ such that*

$$\text{if } x \in X \text{ satisfies } \|x - \bar{x}\|_2 \leq \delta, \text{ then } |f(x) - f(\bar{x})| < \epsilon.$$

♠ *f is called continuous on X, if it is continuous on x at every point of X:*

$$f \text{ is continuous on } X$$
$$\Updownarrow$$
$$\text{whenever } \bar{x} = \lim_{t\to\infty} x_t \text{ with } \bar{x},\ x_t \text{ from } X,$$
$$\text{it holds } f(\bar{x}) = \lim_{t\to\infty} f(x_t)$$

In words: f is continuous on X iff along every sequence x_i of points of X converging, as $i \to \infty$, to a point $\bar{x} \in X$, the values $f(x_i)$ of f converge to $f(\bar{x})$.

How to Recognize Continuity

Fact: *Elementary functions of one variable*, like constants, $\exp\{x\}$, x^p, $\sin(x)$, $\ln(x)$, *are continuous on their natural domains.*

This is established by "bare hands" – by case-by-case verifying the definition of continuity.

Fact: *Continuity of more complicated functions is established via "calculus of continuity" – general statements which state that such and such operations with functions preserve continuity.*

♣ Basic rules of "calculus of continuity" are as follows:

♠ Stability of continuity w.r.t. arithmetic operations: if f_1, f_2 are real-valued functions on $X \subset \mathbb{R}^n$, $\bar{x} \in X$, and all f_i are continuous on X at $\bar{x}$, then

- linear combinations $af_1(x) + bf_2(x)$ of functions f_i with constant coefficients a, b are continuous on X at $\bar{x}$
- the product $f_1(x)f_2(x)$ is continuous on X at $\bar{x}$
- if $f_2(\bar{x}) \neq 0$, then the ratio $f_1(x)/f_2(x)$ is continuous on X at $\bar{x}$

♠ **Theorem on superposition:** *Let*

— $f_1, ..., f_m$ be real-valued functions on $X \subset \mathbb{R}^n$, $\bar{x} \in X$, and let all f_i be continuous on X at $\bar{x}$;

— F be a real-valued function on $Y \subset \mathbb{R}^m$.

Assume that

(a) $f(x) := [f_1(x); ...; f_m(x)] \in Y$ whenever $x \in X$

(b) F is continuous on Y at the point $\bar{y} = f(\bar{x})$

Then the function $g(x) = F(f(x))$ is continuous on X at the point $\bar{x}$.

Closedness/Openess and Continuity

Fact: *Let $X \subset \mathbb{R}^n$ be* **nonempty and closed**, *and f be a real-valued continuous function on X. Then*

- *The subsets of X given by nonstrict (in)equalities involving f:*

$$\{x \in X : f(x) \leq a\} \quad \textit{[a Lebesque set of } f\textit{]}$$
$$\{x \in X : f(x) = a\} \quad \textit{[a level set of } f\textit{]}$$
$$\{x \in X : f(x) \geq a\}$$

where a is a real, are closed.

In particular: *All polyhedral sets are closed*

Note: Replacing nonstrict inequalities with strict ones, we may get non-closed sets (which could be non-open).

- *X, in addition to closedness and nonemptiness, is bounded, then f is bounded on X and attains its maximum and its minimum on X.*
- *If f, in addition to continuity on X, is coercive, meaning that every Lebesque set of f is bounded, or, equivalently, that*

$$f(x_t) \to \infty \text{ as } t \to \infty \text{ whenever } x_t \in X$$
$$\text{are such that } \|x_t\|_2 \to \infty \text{ as } t \to \infty,$$

then f attains its minimum on X.

Fact: *Let $X \subset \mathbb{R}^n$ be* **nonempty and open**, *and f be a real-valued continuous function on X. Then the subsets of X given by strict inequalities involving f:*

$$\{x \in X : f(x) < a\}$$
$$\{x \in X : f(x) > a\}$$

where a is a real, are open.

Convex Sets: Second Acquaintance
Calculus of Convex Sets

♣ Calculus of convex sets is very similar to calculus of polyhedrally representable sets. Specifically:

S.1. Taking intersections: *If the sets $X_\alpha \subset \mathbb{R}^n$, $\alpha \in A$* (where A can be infinite), *are convex sets, so is their intersection* $\bigcap\limits_{\alpha\in A} X_\alpha$

Note: In "calculus of polyhedral representations," similar rule was restricted to *finite* index sets A.

S.2. Taking direct products. *The direct product of K convex sets $X_k \subset \mathbb{R}^{n_k}$ – the set*

$$X = \{[x^1; ...; x^K] : x^k \in X_k, 1 \le k \le K\} \subset \mathbb{R}^{n_1+...+n_K}$$

is convex.

S.3. Taking affine image. *If $X \subset \mathbb{R}^n$ is a convex set and $y = Ax + b : \mathbb{R}^n \to \mathbb{R}^m$ is an affine mapping, then the set $Y = AX + b := \{y = Ax + b : x \in X\} \subset \mathbb{R}^m$ is convex.*

S.4. Taking inverse affine image. *If $X \subset \mathbb{R}^n$ is a convex set and $x = Ay + b : \mathbb{R}^m \to \mathbb{R}^n$ is an affine mapping, then the set $Y = \{y \in \mathbb{R}^m : Ay + b \in X\} \subset \mathbb{R}^m$ is convex.*

S.5. Taking arithmetic sum: *If the sets $X_i \subset \mathbb{R}^n$, $1 \le i \le k$, are convex, so is their arithmetic sum*

$$X_1 + ... + X_k := \{x = x_1 + ... + x_k : x_i \in X_i, 1 \le i \le k\}$$

Topological Properties of Convex Sets

Facts: *Let X be a convex set in $\mathbb{R}^n$.* Then

- *the interior and the closure of X are convex*
- *If nonempty, the interior "well approximates" the closure: whenever $\bar{x} \in \mathrm{int}\, X$ and $x \in \mathrm{cl} X$, the vectors*

$$(1-\lambda)\bar{x} + \lambda x,\ 0 \leq \lambda < 1$$

belong to $\mathrm{int}\, X$. In particular, every point from $\mathrm{cl} X$ can be approximated within whatever accuracy by a point from $\mathrm{int}\, X$.

♣ The second of the above statements is void when $\mathrm{int}\, X = \emptyset$ (which well may happen for a nonempty convex X). There are many other situations when the presence of interior (or the fact that a point in consideration belongs to the interior of the domain in question) is important.

How to "compensate" for potential emptiness of the interior?

Remedy: *relative interior* – interior taken w.r.t. the affine hull of the set.

♠ **Definition:** *Let X be a nonempty subset in $\mathbb{R}^n$ and $\mathrm{Aff}(X)$ be the affine hull of X. We say that a point $x \in X$ is relatively interior point of X, if all close enough to x points from $\mathrm{Aff}(X)$ belong to X, that is, if*

$$\exists r > 0\colon\ \|y - x\|_2 \le r \text{ and } y \in \mathrm{Aff}(X) \text{ imply } y \in X.$$

• The set of all relatively interior points of X is called the *relative interior* $\mathrm{ri}\, X$ *of* X.

Facts: *Let X be a nonempty convex set in $\mathbb{R}^n$. Then*

• *the relative interior of X is nonempty and convex, and*

• *the relative interior "well approximates" the closure: whenever $\bar{x} \in \mathrm{ri}\, X$ and $x \in \mathrm{cl} X$, the vectors*

$$(1 - \lambda)\bar{x} + \lambda x,\ 0 \le \lambda < 1$$

belong to $\mathrm{ri}\, X$. In particular, every point from $\mathrm{cl} X$ can be approximated within whatever accuracy by a point from $\mathrm{ri}\, X$.

Main Facts about Convex Sets

♣ **Basic facts** about polyhedral sets (which are convex and closed) extend, with some losses, to general *closed* convex sets.

Facts: ♡ Every polyhedral set X is intersection of finitely many sets of the form $\{x : a_i^T x \leq b_i\}$, $1 \leq i \leq M$.

♡ *Every closed convex set X is intersection of a sequence of sets of the form $\{x : a_i^T x \leq b_i\}$, $i = 1, 2, ...$*

Facts: ♡ If r is a recessive direction of polyhedral set X, meaning that for some $\bar{x} \in X$, the ray $\{\bar{x} + tr : t \geq 0\}$ is contained in X, then for every $x \in X$, the ray $\{x + tr : t \geq 0\}$ is contained in X.

All recessive directions of a polyhedral set X form a polyhedral cone $\mathrm{Rec}(X)$, and $X + \mathrm{Rec}(X) = X$.

♡ *If r is a recessive direction of closed convex set X, meaning that for some $\bar{x} \in X$, the ray $\{\bar{x} + tr : t \geq 0\}$ is contained in X, then for every $x \in X$, the ray $\{x + tr : t \geq 0\}$ is contained in X.*

All recessive directions of a closed convex set X form a closed convex cone $\mathrm{Rec}(X)$, and $X + \mathrm{Rec}(X) = X$.

Facts: ♡ A nonempty polyhedral set is bounded iff its recessive cone is trivial: $\mathrm{Rec}(X) = \{0\}$.

♡ A closed convex set is bounded iff its recessive cone is trivial: $\mathrm{Rec}(X) = \{0\}$.

Facts: ♡ A nonempty polyhedral set X is the sum of a polyhedral set $\widehat{X}$ not containing lines and the linear subspace $\mathrm{Rec}(X) \cap [-\mathrm{Rec}(X)]$

♡ *A nonempty closed convex set X is the sum of a closed convex set $\widehat{X}$ not containing lines and the linear subspace* $\mathrm{Rec}(X) \cap [-\mathrm{Rec}(X)]$

Facts: ♡ Let X be a nonempty polyhedral set which does not contain lines. Then

• X has extreme points, and the set $\mathrm{Ext}(X)$ of these points is finite: $\mathrm{Ext}(x) = \{v_1, ..., v_N\}$.

• The recessive cone of X is the conic hull of finitely many vectors $r_1, ..., r_N$: $\mathrm{Rec}(X) = \mathrm{Cone}\,\{r_1, ..., r_M\}$.

• We have

$$\begin{aligned} X &= \mathrm{Conv}(\mathrm{Ext}(X)) + \mathrm{Rec}(X) \\ &= \mathrm{Conv}\{v_1, ..., v_N\} + \mathrm{Cone}\,\{r_1, ..., r_M\} \end{aligned}$$

• In particular, a nonempty bounded polyhedral set is the convex hull of the finite set $\mathrm{Ext}(X)$.

♡ *Let X be a nonempty closed convex set which does not contain lines. Then*

• *X has extreme points (perhaps, infinitely many)*

• *We have $X = \mathrm{Conv}(\mathrm{Ext}(X)) + \mathrm{Rec}(X)$*

• *In particular, a nonempty closed convex and bounded set X is the convex hull of the set $\mathrm{Ext}(X)$.*

Facts: ♡ Let K be a polyhedral cone.

• K admits a base B – a set of the form $\{x \in K : f^T x = 1\}$ which intersects all nontrivial rays from K – iff K is nontrivial and pointed. This is exactly the case when K has extreme rays.

• The recessive cone of a base B of K is trivial.

• The number of extreme rays, if any, of K is finite, and these rays are exactly the rays spanned by the extreme points of a base B of K.

• If K is pointed, the number of extreme rays of K is finite, and K is the conic hull of the generators of all extreme rays of K.

♡ *Let K be a closed convex cone.*

• *K admits a base B – a set of the form $\{x \in K : f^T x = 1\}$ which intersects all nontrivial rays from K – iff K is nontrivial and pointed.*

This is exactly the case when K has extreme rays.

• *The recessive cone of a base B of K is trivial.*

• *The extreme rays of K are exactly the rays spanned by the extreme points of a base B of K.*

• *If K is pointed, then K is the conic hull of the generators of all extreme rays of K.*

Facts: ♡ Let K be a polyhedral cone. Then the cone K_* dual to K also is polyhedral, and the cone dual to the dual is K itself.

♡ *Let K be a closed convex cone. Then the cone dual to K also is a closed convex cone, and the cone dual to the dual is K itself.*

♣ While the above results on closed convex sets resemble their polyhedral prototypes, the proofs are more technical.

♠ In retrospect, all polyhedral results stem from Fourier-Motzkin elimination, which is a purely polyhedral fact: the projection of a closed convex set, while being convex, *not necessarily* is closed!

♠ *The* key tool in extending the above results from polyhedral to "closed convex" case is the notion of *separation by linear form* and associated *Separation Theorem*.

Definition. *Let S, T be nonempty sets in $\mathbb{R}^n$. we say that a linear form $a^T x$ separates S and T, if, for some real a, the sets are on the different sides of the hyperplane $\{x : f^T x = a\}$ and not both of them belong to this hyperplane, or, which is the same, if*

$$\begin{array}{rcl} \sup_{x\in S} f^T x & \leq & \inf_{x\in T} f^T x \\ \inf_{x\in S} f^T x & < & \sup_{x\in T} f^T x \end{array}$$

Separation Theorem: *Two nonempty convex sets S, T in $\mathbb{R}^n$ can be separated by a linear form iff their relative interiors do not intersect.*

• As many most useful theorems in Math, this statement at the first glance seems completely esoteric.

Convex Functions: Second Acquaintance

♠ Recall that a convex function $f(x)$ on $\mathbb{R}^n$ is, in general, only partially defined; this is a function on $\mathbb{R}^n$ which takes real values and value $+\infty$ and possesses the following *equivalent to each other* properties:

(a) The epigraph

$$\mathrm{Epi}\{f\} = \{[x;t] \in \mathbb{R}^n \times x : f(x) \le t\}$$

is a convex set

(b) Convexity inequality

$$\forall (x, y \in \mathbb{R}^n, \lambda \in [0,1]) : f((1-\lambda)x + \lambda y) \le (1-\lambda) f(x) + \lambda f(y)$$

♠ *The set* $\mathrm{Dom} f = \{x : f(x) < \infty\}$ *where* f *takes real values is called the domain of* f.

- In (b), we use our standard conventions:
  - $a + (+\infty) = +\infty$ for all $a \in \mathbb{R} \cup \{+\infty\}$
  - $0 \times (+\infty) = 0$ and $a \times (+\infty) = +\infty$ whenever a is a positive real

Operations like $(-\infty) + (+\infty)$ or $a \times (+\infty)$ with negative reals a, which in fact will never arise in our context, are undefined.

Calculus of Convex Functions

♣ Calculus of convex functions is very similar to calculus of polyhedrally representable functions. Specifically:

F.1. Taking linear combinations with positive coefficients. *If $f_i : \mathbb{R}^n \to \mathbb{R} \cup \{+\infty\}$ are convex functions and $\lambda_i \geq 0$, $1 \leq i \leq k$, then $f(x) = \sum_{i=1}^k \lambda_i f_i(x)$ is convex.*

F.2. Direct summation. *If $f_i : \mathbb{R}^{n_i} \to \mathbb{R} \cup \{+\infty\}$, $1 \leq i \leq k$, are convex functions, then so is their direct sum*

$$f([x^1; ...; x^k]) = \sum_{i=1}^k f_i(x^i) : \mathbb{R}^{n_1+...+n_k} \to \mathbb{R} \cup \{+\infty\}$$

F.3. Taking supremum. *If $f_\alpha : \mathbb{R}^n \to \mathbb{R} \cup \{+\infty\}$, $\alpha \in A$ (A can be infinite!) are convex functions, so is their supremum*

$$f(x) = \sup_{\alpha \in A} f_\alpha(x).$$

F.4. Affine substitution of argument. *If function*

$$f(x) : \mathbb{R}^n \to \mathbb{R} \cup \{+\infty\}$$

is convex and $x = Ay + b : \mathbb{R}^m \to \mathbb{R}^n$ is an affine mapping, then the function

$$g(y) = f(Ay + b) : \mathbb{R}^m \to \mathbb{R} \cup \{+\infty\}$$

is convex.

F.5. Partial minimization. *Let function*

$$f(x,y) : \mathbb{R}^n_x \times \mathbb{R}^m_y \to \mathbb{R} \cup \{+\infty\}$$

be convex, and let

$$g(x) = \inf_y f(x,y) : \mathbb{R}^n_x \to \{-\infty\} \cup \mathbb{R} \cup \{+\infty\}$$

For every convex set Q such that $g > -\infty$ at every point of Q, the restriction of g on Q – the function

$$g_Q(x) = \begin{cases} g(x), & x \in Q \\ +\infty, & x \notin Q \end{cases}$$

is convex.

F.6. Projective transformation. *Let a function $f(x) : \mathbb{R}^n \to \mathbb{R} \cup \{+\infty\}$ be convex. Then the function*

$$g(x,\alpha) = \begin{cases} \alpha f(x/\alpha), & \alpha > 0 \\ +\infty, & \alpha \le 0 \end{cases}$$

is convex.

Example: $f(x) = x^2$ is convex

$\Rightarrow$ $g(x,\alpha) = \alpha f(x/\alpha) = \frac{x^2}{\alpha}$ is convex in the domain $\alpha > 0$.

F.7. Theorem on Superposition. *Let*

- $f_i(x) : \mathbb{R}^n \to \mathbb{R} \cup \{+\infty\}$ *be convex functions,*
- $F(y) : \mathbb{R}^m \to \mathbb{R} \cup \{+\infty\}$ *be a convex function, such that* $F(y_1, ..., y_m)$ *is monotonically nondecreasing in every one of* $y_1, ..., y_m$. *Then the superposition*

$$g(x) = \begin{cases} F(f_1(x), ..., f_m(x)), & f_i(x) < +\infty \, \forall i \\ +\infty, & \text{otherwise} \end{cases}$$

of F *and* $f_1, ..., f_m$ *is a convex function.*

Note: *if some of* f_i*, say,* $f_1, ..., f_k$*, are affine, then the Superposition Theorem remains valid when we require the monotonicity of* F *w.r.t. the variables* $y_{k+1}, ..., y_m$ *only.*

Note: One can slightly relax the monotonicity requirement. Specifically, *assume that for some convex set* $Q \subset \mathbb{R}^m$,

— $f(x)$*, when finite,* (i.e., all $f_i(x)$ are reals) *belongs to* Q*, and*

— F *is monotone on* Q *only: whenever* $y \geq y'$ *and* $y, y' \in Q$*, we have* $F(y) \geq F(y')$.

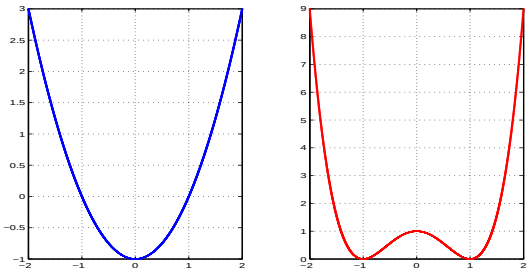
Then the superposition $F(f_1(x), ..., f_k(x))$ *is convex.*

Here again, if $f_1, ..., f_k$, are affine, the monotonicity on Q can be further relaxed to the following property: *whenever* $y \geq y'$ *are such that* $y, y' \in Q$ *and* $y_j = y'_j$ *for* $j \leq k$*, one has* $F(y) \geq F(y')$.

Illustration: Let $f_1, ..., f_m$ be *nonnegative* convex functions, and let $F(y_1, ..., y_m) = \sum_{i=1}^m y_i^2$. *Is the function* $g(x) = F(f_1(x), ..., f_m(x)) = \sum_{i=1}^m f_i^2(x)$ *convex?*

- "Plain" Theorem on superposition is not applicable, since F, while convex, is not monotone.
- However, *on the nonnegative orthant* $Q = \{y \geq 0\}$, F *is* monotone, and *since all* f_i *are nonnegative,* the conditions of the "relaxed" Theorem on Superposition are met, proving that g *is convex*.

Note: nonnegativity of f_i in this illustration is important. The square of an arbitrary convex function can be nonconvex. For example, $f(x) = x^2 - 1$ is convex, and its square is nonconvex!



Left: $x^2 - 1$ Right: $(x^2 - 1)^2$

What does the definition of convexity actually mean?

♣ The Convexity inequality

$f((1-\lambda)x+\lambda y) \le (1-\lambda)f(x)+\lambda f(y),\ 0 \le \lambda \le 1$

is *automatically* satisfied when $\lambda = 0$ and $\lambda = 1$, same as is *automatically* satisfied in x and/or y is not in the domain of f, same as it is *automatically* satisfied when $x = y$.

$\Rightarrow$ Thus, the Convexity inequality says something only when $x, y \in \mathrm{Dom}(f)$, $x \neq y$ and $0 < \lambda < 1$.

What does it say in this case?

• *First,* it says that whenever $0 < \lambda < 1$, the point $z = (1-\lambda)x + \lambda y$ is in the domain of $f \Rightarrow \mathrm{Dom} f$ *is convex*.

• *Second*, when $0 < \lambda < 1$, $z = (1-\lambda)x + \lambda y$ is a (relative) interior point of the segment $[x, y]$, and

$$\|y-x\|_2 : \|y-z\|_2 : \|z-x\|_2 = 1 : (1-\lambda) : \lambda$$

whence

$$f(z) \le (1-\lambda)f(x)+\lambda f(y) \qquad (*)$$
$$\Leftrightarrow f(z)-f(x) \le \underbrace{\lambda}_{\frac{\|z-x\|_2}{\|y-x\|_2}} (f(y)-f(x))$$
$$\Leftrightarrow \frac{f(z)-f(x)}{\|z-x\|} \le \frac{f(y)-f(x)}{\|y-x\|}$$

Similarly,

$$f(z) \le (1-\lambda)f(x)+\lambda f(y) \qquad (*)$$
$$\Leftrightarrow \underbrace{(1-\lambda)}_{\frac{\|y-z\|_2}{\|y-x\|_2}}(f(y)-f(x)) \le f(y)-f(z)$$
$$\Leftrightarrow \frac{f(y)-f(x)}{\|y-x\|_2} \le \frac{f(y)-f(z)}{\|y-z\|_2}$$

Conclusion: *f is convex iff for every three distinct points x, y, z such that $x, y \in \mathrm{Dom} f$ and $z \in [x, y]$, we have $z \in \mathrm{Dom} f$ and*

$$\frac{f(z)-f(x)}{\|z-x\|_2} \leq \frac{f(y)-f(x)}{\|y-x\|_2} \leq \frac{f(y)-f(z)}{\|y-z\|_2} \qquad (!)$$

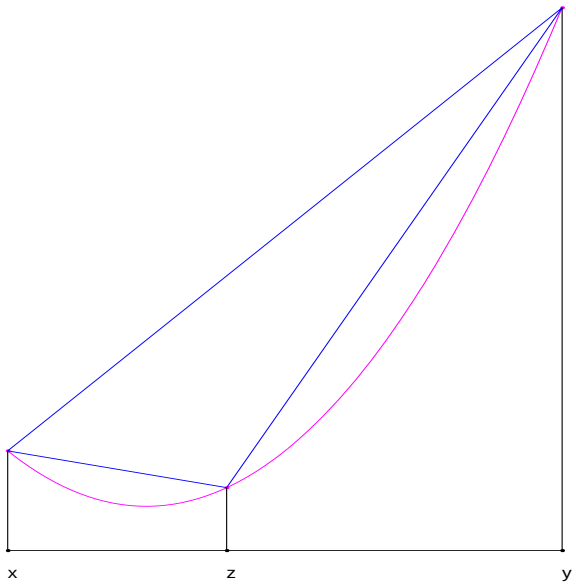
In words: *When moving along $[x; y]$, among the three average rates at which f changes*

(a) "at the beginning" ($x \to z$),

(b) "at average" ($x \to y$), and

(c) "at the end" ($z \to y$),

the first is the smallest, and the third is the largest.



Note: *From 3 inequalities in (!):*

$$\begin{array}{c} \frac{f(z)-f(x)}{\|z-x\|_2} \leq \frac{f(y)-f(x)}{\|y-x\|_2} \\ \frac{f(y)-f(x)}{\|y-x\|_2} \leq \frac{f(y)-f(z)}{\|y-z\|_2} \\ \frac{f(z)-f(x)}{\|z-x\|_2} \leq \frac{f(y)-f(z)}{\|y-z\|_2} \end{array}$$

every single one implies the other two.

Conclusions:

A. *Convexity of a function $f : \mathbb{R}^n \to \mathbb{R} \cup \{+\infty\}$ is one-dimensional property: f is convex iff $\mathrm{Dom} f$ is a convex set, and for every line segment $[x, y] \in \mathrm{Dom} f$ the restriction of f on the segment $[x, y]$ – the univariate function*

$$g(\lambda) = f((1-\lambda)x + \lambda y) = f(x + \lambda[y - x])$$

is convex on $[0, 1]$.

B. For a univariate real-valued *differentiable* function g on $[0, 1]$, average rate of change when moving from a to b, $0 \le a < b \le 1$, is the derivative of g at certain point of (a, b) (Lagrange's Theorem). With minimal effort, from our story about average rates it follows that *a differentiable on $[0, 1]$ univariate function g is convex iff its derivative is monotonically nondecreasing on $[0, 1]$.*

This can be easily extended to the claim that *A continuous function on a one-dimensional convex set and differentiable on the interior of this set, is convex on the set iff its derivative is nondecreasing on the interior of the set.*

C. Recalling what is a *necessary and sufficient* condition for monotonicity of a smooth univariate function, we arrive at the following claim: *A univariate function g which is continuous on a convex one-dimensional set Δ and twice differentiable on $\mathrm{int}\,\Delta$ is convex on Δ iff $g''(\lambda) \ge 0$ for all $\lambda \in \mathrm{int}\,\Delta$.*

D. Combining **C** and **A**, we arrive at the following simple and extremely useful result:

Let $f : \mathbb{R}^n \to \mathbb{R} \cup \{+\infty\}$ be a function such that

- $\mathrm{Dom} f$ *is convex,*
- f *is continuous on* $\mathrm{Dom} f$*, and*
- *for every $x \in \mathrm{ri}\,\mathrm{Dom} f$, and every direction $h \in \mathrm{Dom} f - x$, the second order directional derivative taken at x along the direction h – the quantity $\frac{d^2}{d\lambda^2}\Big|_{\lambda=0} f(x+\lambda h)$ – exists.*

f is convex iff all these second order directional derivatives are nonnegative.

♠ Recall that at the beginning of the course we have developed "calculus" of convex functions and convex sets. We know, e.g., that the operations

- taking linear combinations with nonnegative constant coefficients,
- taking pointwise supremum of a *whatever* family of functions,
- *affine* substitution of argument in a function,

as applied to *convex* functions, produce *convex* results.

$\Rightarrow$ *to justify convexity of a function, it suffices to show that it is obtained from known to be convex "raw materials" by convexity-preserving operations.*

♠ The question is, where to take "raw materials." In the LO part of the course, we need only one kind of "raw material" – affine functions.
In the general case, most of "raw materials" are yielded by **D.**
♠ "Modulo calculus of convex functions" most of the materials we need are just univariate functions. Applying elementary calculus and **C**, we conclude, e.g., that the following univariate functions are convex on the indicated domains:
• for every nonnegative integer p, x^{2p} is convex on the entire $\mathbb{R}$
• for every real $p \geq 1$, the function $|x|^p$ is convex on $\mathbb{R}$, and x^p – on $\mathbb{R}_+ = [0, \infty)$
• when $0 < p < 1$, the function $-x^p$ is convex on $\mathbb{R}^+$ (i.e., the function x^p is *concave* on $\mathbb{R}^+$)
• when $p < 0$, the function x^p is convex on $(0, \infty)$
• the exponent $\exp\{x\}$ is convex on the entire $\mathbb{R}$, and the function $\ln(1/x)$ is convex on $(0, \infty)$
• the *entropy* $f(x) = \begin{cases} x \ln x, & x > 0 \\ 0, & x = 0 \end{cases}$ is convex on $[0, \infty)$
• .....

♣ Usually, "calculus of convexity" along with "univariate convex raw materials" is enough to establish convexity of actually convex multivariate functions.

♠ In relatively rare cases, convexity of useful multivariate functions should be established by "bare hands" — directly via **D.**

Example: *The function* $f(x) = \ln(e^{x_1} + ... + e^{x_n})$ *is convex.*

Indeed,

$$\begin{aligned}
&\frac{d}{d\lambda} f(x+\lambda h) = \frac{\sum_i e^{x_i+\lambda h_i} h_i}{\sum_i e^{x_i+\lambda h_i}} \\
\Rightarrow\ &\frac{d^2}{d\lambda^2} f(x+\lambda h) = \frac{\sum_i e^{x_i+\lambda h_i} h_i^2}{\sum_i e^{x_i+\lambda h_i}} - \left[\frac{\sum_i e^{x_i+\lambda h_i} h_i}{\sum_i e^{x_i+\lambda h_i}}\right]^2 \\
\Rightarrow\ &\frac{d^2}{d\lambda^2}\Big|_{\lambda=0} f(x+\lambda h) = \sum_i p_i h_i^2 - [\sum_i p_i h_i]^2,\ p_i = \frac{e^{x_i}}{\sum_j e^{x_j}}.
\end{aligned}$$

Note: p_i are nonnegative and sum up to 1

$\Rightarrow \frac{d^2}{d\lambda^2}\Big|_{\lambda=0} f(x+\lambda h)$ is the variance of random variable taking values $h_1, ..., h_n$ with probabilities $p_1, ..., p_n$, *and the variance always is nonnegative*.

Corollary: *When* $c_i > 0$, *the function*

$$g(y) = \ln\left(\sum_i c_i \exp\{a_i^T y\}\right)$$

is convex.

Indeed, $g(y) = \ln\left(\sum_i \exp\{\ln c_i + a_i^T y\}\right)$ is obtained from the convex function $\ln\left(\sum_i e^{x_i}\right)$ by affine substitution of argument, and this operation preserves convexity.

Quiz: *Which of the following functions are convex?*

- $\ln(e^{2x+3y} + 2e^{y-x})$
- $\ln(e^{x^2} + e^{y^2})$
- $\ln(e^{-x^2} + e^{y^2})$
- $\ln(e^{x^2} + 2e^{-3x^2})$
- $\ln(e^{x^2} + e^{-x^2})$

Quiz: *Which of the following functions are convex?*

- $\ln(e^{2x+3y}+2e^{y-x})$ – Convex along with $\ln(e^{x_1}+e^{x_2})$ (affine substitution of argument)
- $\ln(e^{x^2}+e^{y^2})$ – Convex along with $\ln(e^{x_1}+e^{x_2})$ and x^2, y^2 (Theorem on Superposition; note that $\ln(e^{x_1}+e^{x_2})$ is nondecreasing in x_1 and x_2)
- $\ln(e^{-x^2}+e^{y^2})$ – Non-convex: look what happens when $y=0$: $\frac{d}{dx}f(x,0)=\frac{-2xe^{-x^2}}{e^{-x^2}+1}$, and the derivative is *not* nondecreasing in x!
- $\ln(e^{x^2}+2e^{-3x^2})$ – Non-convex: $\frac{d}{dx}f(x)=\frac{-x(6e^{-3x^2}-2e^{x^2})}{e^{x^2}+e^{-3x^2}}$, and the derivative is *not* nondecreasing around $x=0$
- $\ln(e^{x^2}+e^{-x^2})$ — Convex: The function $\ln(e^s+e^{-s})$ is convex *and nondecreasing* in the domain $s\geq 0$, and x^2 is convex *and nonnegative*

Interesting convex functions: norms

♣ A real-valued function $\|x\| : \mathbb{R}^n \to \mathbb{R}$ is called *norm*, if

- [homogeneity] $\|\lambda x\| = \lambda \|x\|$ for all $x \in \mathbb{R}^n$ and all nonnegative real λ
- [symmetry] $\|x\| = \|-x\|$ for all x
- [positivity] $\|x\|$ is positive unless $x = 0$ ($\|0\| = 0$ by homogeneity)
- [triangle inequality] $\|x + y\| \leq \|x\| + \|y\|$ for all x, y

Fact: *A homogeneous real-valued function $p(\cdot)$ is convex iff it satisfies the triangle inequality.*

Indeed, when p is homogeneous, Convexity Inequality reads

$$\begin{array}{rcl} p((1-\lambda)u + \lambda v) & \leq & (1-\lambda)p(u) + \lambda p(v) \\ & = & p((1-\lambda)u) + p(\lambda v) \end{array}$$

and thus is nothing but triangle inequality.

Fact: *Every two norms $\|\cdot\|$, $\|\cdot\|'$ on $\mathbb{R}^n$ are within constant factor of each other: for some $\theta > 0$, it holds*

$$\theta \|x\| \leq \|x\|' \leq \theta^{-1} \|x\| \text{ for all } x$$

In particular, all norms define the same notion of convergence: $\|x_t - x\| \to 0$, as $t \to \infty$, is exactly the same as $\|x_t - x\|_2 \to 0$ as $t \to \infty$.

This property *characterises* finite-dimensional linear spaces $\mathbb{R}^n$.

♠ The standard norms on $\mathbb{R}^n$ are the ℓ_p *norms*

$$\|x\|_p = (\sum_i |x_i|^p)^{1/p}$$

Here $1 \leq p \leq \infty$. When $p = \infty$, the right hand side, by definition, is $\max_i |x_i|$ (which is also $\lim_{p\to\infty} \|x\|_p$).

Note: $\|x\|_1 = \sum_i |x_i|$, $\|x\|_2 = \sqrt{\sum_i x_i^2}$ is the standard Euclidean norm, $\|x\|_\infty = \max_i |x_i|$.

♠ $\|\cdot\|_p$ clearly is homogeneous, symmetric, and positive outside the origin. Triangle inequality (equivalent to convexity) stems from the

Hölder Inequality: *For $p \in [1, \infty]$, let $p_* \in [1, \infty]$ be given by $\frac{1}{p} + \frac{1}{p_*} = 1$* (e.g., $1_* = \infty, 2_* = 2, \infty_* = 1$). *Then*

$$\textit{for all } x, y \in \mathbb{R}^n\textit{: } x^T y \leq \|x\|_p \|y\|_{p_*}$$

and the inequality is tight: for every x,

$$\|x\|_p = \max_y \{x^T y : \|y\|_{p_*} \leq 1\}$$

• Tightness says that $\|x\|_p$ is the supremum of a family of linear functions of x and thus is convex by calculus of convexity.

Note: When $p = 2$, $p_* = 2$ as well, and Hölder Inequality becomes the Cauchy Inequality:

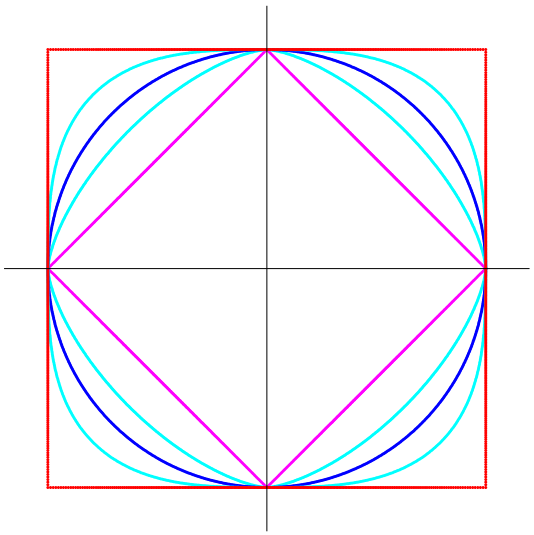
$$x^T y \leq \|x\|_2 \|y\|_2.$$

♠ A function $f : \mathbb{R}^n \to \mathbb{R} \cup \{+\infty\}$ produces *Lebesque sets* $\{x : f(x) \le a\}$ where a is a real. *The Lebesque sets of a convex function f always are convex:*

$$f(x) \le a, f(y) \le a, \lambda \in [0,1], \ f \text{ is convex}$$
$$\Downarrow$$
$$f((1-\lambda)x + \lambda y) \le (1-\lambda)f(x) + \lambda f(y) \le (1-\lambda)a + \lambda a = a$$

• The Lebesque set $\{x : \|x\| \le 1\}$ of a norm is called the *unit ball* of the norm.

Here are the unit balls of several ℓ_p-norms on $\mathbb{R}^2$:



From inside outside: $p = 1, 3/2, 2, 3, \infty$

• In every dimension, $\|x\|_p$ *decreases as p grows*

$\Rightarrow$ unit ball of $\|\cdot\|_p$ *extends as p grows*.

Jensen's Inequality

• Convexity implies the *Jensen's Inequality: The value of a convex function* $h : \mathbb{R}^n \to \mathbb{R} \cup \{+\infty\}$ *at a convex combination of points is* $\leq$ *the convex combination, with the same weights, of the values of* h *at these points:*

$$\forall (m, x_i \in \mathbb{R}^n, \lambda_i \geq 0,\ 1 \leq i \leq m, \text{ with } \textstyle\sum_i \lambda_i = 1) : \\ h(\textstyle\sum_i \lambda_i x_i) \leq \textstyle\sum_i \lambda_i h(x_i)$$

For example, *for a random variable* ξ *taking real values* $a_1, ..., a_N$ *with probabilities* $p_1, ..., p_N$*, its variance*

$$\mathit{Var}\{\xi\} := \mathbf{E}\{\xi^2\} - [\mathbf{E}\{\xi\}]^2 = \sum_i p_i a_i^2 - \left[\sum_i p_i a_i\right]^2$$

is always nonnegative. This is just Jensen's inequality with x^2 in the role of $f(x)$, a_i in the role of x_i and p_i in the role of λ_i.

Continuity of a Convex Function

♣ **Fact:** *Convex function is continuous on its domain at every relative interior point of the domain. In particular, a real-valued convex function on $\mathbb{R}^n$ always is continuous on the entire $\mathbb{R}^n$.*

In fact, we can say much more:

♠ *Let f be convex, and Y be a closed and bounded subset of the relative interior of $\mathrm{Dom} f$. Within Y, change in f is at most proportional to the change in the argument: for some $L = L(Y) < \infty$, we have*

$$|f(x) - f(y)| \leq L\|x - y\|_2 \text{ for all } x, y \in Y$$

Scientifically: With f and Y as above, f is *Lipschitz continuous* on Y.

• At boundary points of its domain, a convex function can be discontinuous: it can "jump up," as is the case with the function

$$f(x) = \begin{cases} 0, & x < 0 \\ 1, & x = 0 \\ +\infty, & x > 0 \end{cases}$$

Gradient Inequality

♣ **Fact:** *Let f be a convex function on $\mathbb{R}^n$, and $\bar{x} \in \mathrm{Dom} f$ be a point where f is differentiable. Then the linearization of f, taken at $\bar{x}$, underestimates f:*

$$f(x) \geq f(\bar{x}) + \langle \nabla f(\bar{x}), x - \bar{x} \rangle \text{ for all } x$$

• The inequality clearly holds true when $f(x) = +\infty$, as well as when $x = \bar{x}$. When $f(x) < \infty$ and $x \neq \bar{x}$, for all $\lambda \in (0, 1)$ it holds

$$\frac{f(\bar{x}+\lambda[x-\bar{x}])-f(\bar{x})}{\lambda\|x-\bar{x}\|_2} \leq \frac{f(x)-f(\bar{x})}{\|x-\bar{x}\|_2}$$

as $\lambda \to +0$, the left hand side in this inequality tends to $\frac{\langle \nabla f(\bar{x}), x-\bar{x} \rangle}{\|x-\bar{x}\|_2}$, and the right hand side remains intact. Passing to limit as $\lambda \to +0$, we get

$$\frac{\langle \nabla f(\bar{x}), x-\bar{x} \rangle}{\|x-\bar{x}\|_2} \leq \frac{f(x)-f(\bar{x})}{\|x-\bar{x}\|_2}.$$

as claimed.

Note: Gradient Inequality is the source of many useful inequalities, like

• $p \geq 1, x \geq -1 \Rightarrow (1+x)^p \geq 1+px$ $(f = (1+x)^p, \bar{x} = 0)$

• $p > 0, x > -1 \Rightarrow \frac{1}{(1+x)^p} \geq 1 - px$ $(f(x) = \frac{1}{(1+x)^p}, \bar{x} = 0)$

• $\mathrm{e}^x \geq 1 + x$ $(f(x) = \mathrm{e}^x, \bar{x} = 0)$

Subgradients of Convex Functions

♠ Let f be a convex function. If $\bar{x} \in \mathrm{Dom} f$ and f is differentiable at x, then $\nabla f(\bar{x})$ is the "slope" (the vector of coefficients) of an *affine* function

$$h(x) = f(\bar{x}) + \langle \nabla f(\bar{x}), x - \bar{x} \rangle$$

which *everywhere underestimates* f and *coincides with* f *at the point* $\bar{x}$.

However: It may happen that such an "exact at $\bar{x}$ affine lower bound on f" exists at a point $\bar{x}$ where f is *not* differentiable.

Example: $f(x) = |x| : \mathbb{R} \to \mathbb{R}$ is convex and differentiable outside of $x = 0$, and is *not* differentiable at $\bar{x} = 0$. However, there are many affine lower bounds on f which are exact at $\bar{x} = 0$: whenever $|g| \leq 1$, we have

$$\forall x : f(x) = |x| \geq gx = f(0) + g(x - 0)$$

and slopes g of these lower bounds fill the entire segment $[-1, 1]$.

♠ As far as a convex function f is concerned, the slopes of exact at $\bar{x}$ affine lower bounds on $f(\cdot)$ are, in many respects, quite satisfactory *surrogates* of $\nabla(\bar{x})$. These slopes have name: they are called *subgradients* of f at $\bar{x}$.

Definition: *let f be a convex function on $\mathbb{R}^n$, and $\bar{x} \in \mathrm{Dom} f$. A vector $g \in \mathbb{R}^n$ is called subgradient of f at $\bar{x}$, if*

$$\forall x : f(x) \geq f(\bar{x}) + \langle g, x - \bar{x} \rangle.$$

The set of all subgradients of f at $\bar{x}$ is called the *sub-differential* of f at $\bar{x}$, denoted $\partial f(\bar{x})$.

Note: By Gradient inequality, *if $\nabla f(\bar{x})$ exists, then $\nabla f(\bar{x})$ is a subgradient of f at $\bar{x}$.*

♠ The main property of subgradients is that they "nearly always" exist:

Fact: *Let f be a convex function on $\mathbb{R}^n$ and $\bar{x}$ belong to relative interior of $\mathrm{Dom} f$. Then f admits a subgrdient at $\bar{x}$: $\partial f(\bar{x}) \neq \emptyset$*

• At a point $\bar{x}$ from the relative boundary of $\mathrm{Dom} f$, $\partial f(\bar{x})$ can be empty even when $\mathrm{Dom} f$ is closed and f is continuous on $\mathrm{Dom} f$. For example, the convex univariate function $f(x) = -\sqrt{x}$ with the domain $[0, \infty)$ admits *no* subgradients at $\bar{x} = 0$.

♠ **Elementary calculus** of subgradients is as follows.

A. *Let a convex function f be differentiable at $\bar{x} \in \mathrm{Dom} f$. Then $\nabla f(\bar{x}) \in \partial f(\bar{x})$, and if $\bar{x} \in \mathrm{int}\,\mathrm{Dom} f$, then $\nabla f(\bar{x})$ is the only subgradient of f at $\bar{x}$.*

B. Behavior of subgradients with respect to linear operations and change of variables is very similar to the one of gradients:

- *Let $f(x) = \sum_{i=1}^m \lambda_i f_i(x)$ with nonnegative λ_i and convex f_i, and let $f_i, ..., f_m$ admit subgradients g_i at a point $\bar{x}$. Then $\sum_i \lambda_i g_i \in \partial f(\bar{x})$.*
- *Let f be a convex function on $\mathbb{R}^n$, and let*

$$x = Ay + b : \mathbb{R}^m \to \mathbb{R}^n,\ h(y) = f(Ay + b).$$

Given $\bar{y} \in \mathbb{R}^m$, let $\bar{x} = A\bar{y} + b$ and $g \in \partial f(\bar{x})$. Then $A^T g \in \partial h(\bar{y})$.

C. Smoothness not always "survives" passing from several functions to their maximum, while subgradients are well suited for this operation:

Let $\{f_\alpha(\cdot)\}_{\alpha \in A}$ be a family of convex functions on $\mathbb{R}^n$, and $f(x) = \sup_{\alpha \in A} f_\alpha(x)$. Given $\bar{x} \in \mathrm{dom} f$, assume that there exists $\bar{\alpha} \in A$ such that $f(\bar{x}) = f_{\bar{\alpha}}(\bar{x})$. Then any subgradient g of $f_{\bar{\alpha}}(\cdot)$ at $\bar{x}$ is a subgradient of f at $\bar{x}$.

Indeed, $f(x) \geq f_{\bar{\alpha}}(x) \geq f_{\bar{\alpha}}(\bar{x}) + \langle g, x - \bar{x} \rangle = f(\bar{x}) + \langle g, x - \bar{x} \rangle$

Minima of Convex Functions

♣ The following simple facts explain why minimizing *convex* functions over *convex* sets is much easier than nonconvex minimization.

♠ **Facts:** *Let X be a convex set in $\mathbb{R}^n$, and f be a convex function on $\mathbb{R}^n$. consider optimization problem*

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

• *Every local minimizer x_* of f on X is a global minimizer of f on X: if $x_* \in [\mathrm{Dom} f] \cap X$ is a local minimizer, meaning that for some positive r, $f(x) \geq f(x_*)$ whenever $x \in X$ and $\|x - x_*\|_2 \leq r$, then x_* is a global minimizer of f on X: $f(x) \geq f(x_*)$ for all $x \in X$.*

Indeed, let x_* be a local minimizer of f on X; to see that $f(x) \geq f(x_*)$ for every $x \in X$, it suffices to verify this inequality when $x \in [\mathrm{Dom} f] \cap X$ and $x \neq x_*$. In this case, by convexity

$$\frac{f(x_* + \lambda[x - x_*]) - f(x_*)}{\lambda \|x - x_*\|_2} \leq \frac{f(x) - f(x_*)}{\|x - x_*\|_2}$$

for all $\lambda \in (0, 1)$. Since x_* is a local minimizer of f on X, the left hand side ratio is nonnegative when λ is positive and small $\Rightarrow$ the right hand side ratio is nonnegative $\Rightarrow$ $f(x) \geq f(x_*)$ □

• *The set of minimizers of f on X is convex.*

Indeed, when nonempty, this set is the Lebesque set $\{x : f(x) \leq \mathrm{Opt}\}$ (which is convex) intersected with convex set X.

Question: *Let X be a convex set in $\mathbb{R}^n$, f be a convex function, and let $x_* \in X \cap \mathrm{Dom} f$ be a point such that f is differentiable at x_*. When x_* is a global minimizer of f on X?*

Answer: *This is the case iff the directional derivative of f, taken at x_* along any direction leading from x_* into X, is nonnegative:*

$$\forall (x \in X) : \langle \nabla f(x_*), x - x_* \rangle \geq 0$$

• In one direction: for every $x \in X$, we should have $g(\lambda) := f(x_* + \lambda(x - x_*)) \geq f(x_*) = g(0)$ when $0 \leq \lambda \leq 1$, whence $0 \leq g'(0) = \langle \nabla f(x_*), x - x_* \rangle$. This should be so for all $x \in X$, implying that $\langle \nabla f(x_*), x - x_* \rangle \geq 0$ for all $x \in X$.

• In the opposite direction: By Gradient Inequality, $f(x) \geq f(x_*) + \langle \nabla f(x_*), x - x_* \rangle$ for all x, implying that $f(x) \geq f(x_*)$ when $x \in X$ and $\langle \nabla f(x_*), x - x_* \rangle \geq 0$ for all $x \in X$.

Note: Given $x \in X$, the set of all vectors h such that $\langle h, y - x \rangle \geq 0$ whenever $y \in X$, is called the *normal cone $N_X(x)$ of X taken at x*; this set indeed is a closed convex cone. This cone is dual to the *radial cone*

$$T_X(x) = \mathrm{Cone}\,\{X - x\}$$

spanned by the directions leading from x into X.

The above necessary and sufficient optimality condition reads: *In the situation in question, x_* is a global minimizer of f on X iff $\nabla f(x) \in N_X(x_*)$.*

What this condition actually means, it depends on what is the normal cone $N_X(x_*)$.

$$x_* \;??\in??\; \mathrm{Argmin}_X f$$

Examples:

• $x_* \in \mathrm{int}\, X$. In this case, $T_X(x_*) = \mathbb{R}^n$, whence $N_X(x_*) = \{0\}$

$\Rightarrow$ *When X is a convex set, f is convex and is differentiable at a point $x_* \in [\mathrm{int}\, X] \cap [\mathrm{int}\, \mathrm{Dom} f]$, the point x_* is a global minimizer of f on X iff*

$$\nabla f(x_*) = 0 \quad \text{[Fermat equation]}$$

• $x_* \in \mathrm{ri}\, X$. In this case, $T_X(x_*)$ is the linear subspace L parallel to the affine hull $\mathrm{Aff}(X)$ of X, whence $N_X(x_*) = L^\perp$.

$\Rightarrow$ *When X is a convex set, f is convex and is differentiable at a point $x_* \in [\mathrm{ri}\, X] \cap [\mathrm{int}\, \mathrm{Dom} f]$, the point x_* is a global minimizer of f on X iff $\nabla f(x_*)$ is orthogonal to $L = \mathrm{Lin}(X - x_*)$.*

$$x_* \text{ ?? } \in \text{?? } \mathrm{Argmin}_X f$$

Examples (continued):

• $X = \{x : a_i^T x \le b_i,\ 1 \le i \le m\}$ is a polyhedral set. In this case, the radial cone is $\{h : a_i^T h \le 0\ \forall i \in I(x_*)\}$, where $I(x_*)$ is the set of indexes of all constraints $a_i^T x \le b_i$ which are *active at* x_* – are satisfied at x_*as equalities. By Homogeneous Farkas Lemma, the normal cone (the dual to the radial one) is

$$N_X(x_*) = \mathrm{Cone}\,\{-a_i : i \in I(x_*)\}$$

⇒ *When* $X = \{x : a_i^T x \le b_i,\ 1 \le i \le m\}$, f *is convex and is differentiable at a point* $x_* \in X \cap [\mathrm{Dom} f]$, *the point* x_* *is a global minimizer of* f *on* X *iff there are nonnegative Lagrange multipliers* λ_i^* *associated with active at* x_* *constraints* $a_i^T x \le b_i$, $i \in I(x_*)$ *such that*

$$\nabla f(x_*) + \textstyle\sum_{i \in I(x_*)} \lambda_i^* a_i = 0.$$

Setting $\lambda_i^* = 0$ for $i \notin I(x_*)$, our optimality condition reads:

When $X = \{x : a_i^T x \le b_i,\ 1 \le i \le m\}$, f *is convex and is differentiable at a point* $x_* \in X \cap [\mathrm{Dom} f]$, *the point* x_* *is a global minimizer of* f *on* X *iff there are nonnegative Lagrange multipliers* λ_i^* *such that*

$$\lambda_i^*[b_i - a_i^T x_*] = 0\ \forall i \qquad \text{[complementary slackness]}$$
$$\nabla f(x_*) + \textstyle\sum_i \lambda_i^* a_i = 0 \qquad \text{[KKT equation]}$$

This is exact analogy of the KKT optimality condition in LO, with $\nabla f(x_*)$ in the role of the vector of coefficients of the LO objective.

♠ One of the beauties of Convex Optimization is the presence of *local* conditions for *global* optimality. In simple cases, these conditions allow for explicit solving of convex problems "on paper."

Examples:

A. *Let $a_1, ..., a_n$ be given positive reals. What is*

$$\mathrm{Opt} = \min_x \left\{ f(x) = \sum_{j=1}^n \frac{a_j}{x_j} : x > 0, \sum_{j=1}^n x_j \le 1 \right\} ?$$

To represent the problem as a convex program $\min_X f$, we set

$$X = \{x \in \mathbb{R}^n : x > 0, \sum_j x_j \le 1\},\ \mathrm{Dom} f = \{x \in \mathbb{R}^n : x > 0\}$$

Let us make an *educated guess* that there exist an optimal solution where $x_j > 0$ for all j and $\sum_j x_j = 1$. The optimality condition reads:

For some $\lambda \ge 0$, it holds $\nabla f(x) + \lambda[1; ...; 1] = 0$ and $\sum_j x_j = 1$, which amounts to

$-\frac{a_j}{x_j^2} + \lambda = 0 \, \forall j$ and $\sum_j x_j = 1$. in other words,

$x_j = \sqrt{a_j/\lambda}$ and $\sum_j \sqrt{a_j/\lambda} = 1$, whence

$$\lambda = \sqrt{a_1} + ... + \sqrt{a_n}, x_j = \frac{\sqrt{a_j}}{\sqrt{a_1} + ... + \sqrt{a_n}}, f(x) = [\sqrt{a_1} + ... + \sqrt{a_n}]^2$$

We have satisfied the optimality condition *which in the convex case is sufficient for global optimality* and thus have found a global optimal solution. *There is no need to justify our educated guess – we are in the situation when eating indeed is a proof of the pudding!*

B. *Given n-dimensional vector a, we want to solve the problem*

$$\mathrm{Opt} = \min_x \left\{ f(x) = \ln\left(\sum_{i=1}^n \mathrm{e}^{x_i}\right) - a^T x \right\}$$

We are minimizing smooth convex function over the entire $\mathbb{R}^n$ $\Rightarrow$ minimizers are exactly the solutions to the Fermat equation $\nabla f(x) = 0$, that is,

$$\frac{\mathrm{e}^{x_i}}{\sum_{j=1}^n \mathrm{e}^{x_j}} = y_i,\ 1 \le i \le n$$

• It is seen by naked eye that the necessary condition for the system to have a solution is $y_i > 0$ for all i and $\sum_i y_i = 1$. This condition is also *sufficient:* when it is satisfied, the Fermat equation is satisfied when setting

$$x_j = \ln(y_j),\ 1 \le j \le n$$

resulting in $\mathrm{Opt} = -\sum_j y_j \ln y_j$.

• All we know so far when our condition

$$y > 0, \sum_j y_j = 1$$

is *not* satisfied, is that *in this case, the problem has no optimal solutions*. What about the optimal value and near-optimal solutions?

With some dedicated effort, it could be seen that

— if y is not nonnegative and/or $\sum_j y_j \neq 1$, the problem is unbounded: $\mathrm{Opt} = -\infty$

— if $\sum_j y_j = 1$, $y \ge 0$, but *not* $y > 0$, we still have $\mathrm{Opt} = -\sum_j y_j \ln y_j$, but there is no optimal solution. To get a near-optimal solution, it suffices to make the entries in x corresponding to $y_j > 0$ equal to $\ln y_j$, and make the entries corresponding to $y_j = 0$ large in magnitude negative reals.

Here eating gives only partial proof of the pudding, but already this partial proof allows to guess what is the correct answer and how to justify it.

♢When a Minimizer of a Convex Function is Unique?

♣ Sometimes it makes sense to know when the optimal solution to a convex program

$$\mathrm{Opt} = \min_{x \in X \subset \mathbb{R}^n} f(x) \qquad (P)$$

is unique.

The standard *sufficient* condition for uniqueness is *strict convexity* of f, defined as follows:

♠ *A convex function $f : \mathbb{R}^n \to \mathbb{R} \cup \{+\infty\}$ is called strictly convex, when f strictly satisfies the Convexity Inequality "in all nontrivial cases:"*

$$x, y \in \mathrm{Dom} f, x \neq y, 0 < \lambda < 1 \Rightarrow f((1-\lambda)x + \lambda y) < (1-\lambda) f(x) + \lambda f(y).$$

Fact: *If X is convex subset of $\mathbb{R}^n$, f is strictly convex and (P) is solvable, the optimal solution to (P) is unique.*

Indeed, if x', x'' were two distinct optimal solutions, the point $\bar{x} = \frac{1}{2}(x' + x'')$ would be feasible and, by strict convexity, would satisfy $f(\bar{x}) < \frac{1}{2}(f(x') + f(x'') = \mathrm{Opt}$, which is impossible.

♠ Assuming a convex function f to possess second order directional derivatives, taken at every point $x \in \mathrm{ri}\,\mathrm{Dom} f$ along every direction $h \in [\mathrm{Dom} f] - x$, the standard *sufficient* condition for f to be strictly convex is that when $h \neq 0$, these derivatives are strictly positive: for all $(x \in \mathrm{ri}\,\mathrm{Dom} f, h \in [\mathrm{Dom} f] - x, h \neq 0)$:

$$\frac{d^2}{d\lambda^2}\Big|_{\lambda=0} f(x + \lambda h) > 0.$$

Maxima of Convex Functions

♣ **Fact:** *Let f be a convex function. Then*

- *If f attains its maximum over $\mathrm{Dom} f$ at a point $x_* \in \mathrm{ri}\,\mathrm{Dom} f$, then f is constant on $\mathrm{Dom} f$*
- *If $\mathrm{Dom} f$ is closed and does not contain lines and f attains its maximum on $\mathrm{Dom} f$, then among the maximizers there is an extreme point of $\mathrm{Dom} f$*
- *If $\mathrm{Dom} f$ is polyhedral and f is bounded from above on $\mathrm{Dom} f$, then f attains its maximum on $\mathrm{Dom} f$.*

♠ Assume we want to maximize a convex function f which is real valued on a nonempty, bounded and polyhedral set X, over this set.

Good news: *The optimal solution does exist and can be found among the extreme points of X (i.e., in "finite time").*

For example: it is easy to maximize a convex function over the standard simplex $\{x \in \mathbb{R}^n : x \geq 0, \sum_i x_1 = 1\}$ of reasonable dimension.

Very bad news: *X may have astronomically many extreme points, and huge number of them could be local, but not global, maximizers of f. In general, local information on f does not allow to understand whether a local maximizer is global.*

While it is relatively easy, starting with a point $x \in X$, to find an extreme points v of X with $f(v) \geq f(x)$, it, in general, is impossible to avoid exhaustive search (usually completely unrealistic) through all, or a "significant part" of, extreme points of X.

Conclusion: Convex functions are badly suited for maximization.

We shall see, however, that convex functions *are well suited for minimization*.

Optimality Conditions in Convex Programming

♣ Consider an optimization problem

$$\mathrm{Opt} = \min_{x \in X \subset \mathbb{R}^n} \{f(x); g_i(x) \leq 0, i = 1, ..., m\} \qquad (P)$$

Standing Assumption: *The problem is convex, meaning that X is a convex set, and $f, g_1, ..., g_m$ are convex functions. We always assume X to be nonempty, and $f, g_1, ..., g_m$ to be real-valued on X.*

♠ The main theoretical questions related to (P) are

A. *Is the problem solvable?*

B. *Is the optimal solution unique?*

C. *How to characterize an optimal solution – what are optimality conditions?*

♠ The main practical question is how to find an optimal solution, or, more realistically, *how to find, in reasonable time, a near-optimal and near-feasible solution?*

♠ We already know some answers to **A** and to **B**:

• **The standard answer to A** is:

If the feasible set $X_* = \{x \in X : g_i(x) \leq 0, 1 \leq i \leq m\}$ is nonempty and closed (closedness definitely takes place when X is closed, and $g_1, ..., g_m$ are continuous on X) *and f is continuous and coercive on X_*, an optimal solution does exist.*

Note: convexity here is irrelevant.

• **The standard answer to B** is:

If f is strictly convex on its domain, the optimal solution, if it exists, is unique.

♠ What is ahead of us now, is **C.**

Lagrange Function and Lagrange Duality

$$\mathrm{Opt}(P) = \min_{x \in X \subset \mathbb{R}^n} \{f(x); g_i(x) \le 0, i = 1, ..., m\} \quad (P)$$

♠ *The Lagrange function of problem (P) is the function*

$$L(x, \lambda) := f(x) + \sum_{i=1}^m \lambda_i g_i(x) : X \to \mathbb{R}^m_+ \to \mathbb{R}$$

Note: *When speaking about the Lagrange function,*

- *x-argument is restricted to vary in X*
- *λ-argument is restricted to vary in $\mathbb{R}^m_+$ – we want the Lagrange multipliers $\lambda_1, ..., \lambda_m$ to be nonnegative.*

Note: Essentially, we have already met Lagrange function in the LO case, where $X = \mathbb{R}^n$, f is linear, and $g_1, ..., g_m$ are affine. "Essentially" reflects the current swap of min and max as compared to the LO case: in LO, our problem of interest was to maximize, and now it is to minimize.

Observation: *When $\lambda \ge 0$, the Lagrange function underestimates $f(\cdot)$ on the feasible set of (P)*

$\Rightarrow$ *In the domain $\lambda \ge 0$, the function*

$$\underline{L}(\lambda) = \inf_{x \in X} L(x, \lambda) : \mathbb{R}^m_+ \to \mathbb{R} \cup \{-\infty\}$$

is $\le \mathrm{Opt}(P)$.

The problem

$$\begin{array}{rcll} \mathrm{Opt}(D) & = & \max_{\lambda \ge 0} \underline{L}(\lambda) & (D) \\ & = & \max_{\lambda \ge 0} \inf_{x \in X} L(x, \lambda) & \end{array}$$

$$
\begin{array}{rcll}
\mathrm{Opt}(P) & = & \min\limits_{x\in X\subset\mathbb{R}^n}\{f(x);\, g_i(x)\le 0, i=1,...,m\} & (P)\\
\Rightarrow L(x,\lambda) & := & f(x)+\sum\limits_{i=1}^m \lambda_i g_i(x) : X\to\mathbb{R}^m_+\to\mathbb{R} & \\
\Rightarrow \underline{L}(\lambda) & = & \inf\limits_{x\in X} L(x,\lambda) : \mathbb{R}^m_+\to\mathbb{R}\cup\{-\infty\} & \\
\Rightarrow \mathrm{Opt}(D) & = & \max\limits_{\lambda\ge 0}\underline{L}(\lambda), & (D)\\
 & = & \max\limits_{\lambda\ge 0}\inf\limits_{x\in X} L(x,\lambda) &
\end{array}
$$

Note: We have seen that in the LO case, (D) is the LO dual of (P) (in a slight disguise).

Fact [Weak Duality]: *By construction,*

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

Note: Convexity is irrelevant here.

Course of actions: We will show that *in the convex case*, under mild assumptions $\mathrm{Opt}(D)=\mathrm{Opt}(P)$, and will extract from this fact optimality conditions for (P).

♠ The "mild assumption," in its simplest form, is

Slater condition: *(P) admits a strictly feasible solution $\bar{x}$,* meaning that $\bar{x}\in X$ and $g_i(\bar{x})<0$ *for all* $i=1,...,m$.

• A more advanced version of "mild assumption" is

Relaxed Slater condition: *(P) admits a feasible solution $\bar{x}\in \mathrm{ri}\, X$ where all non-affine constraints $g_i(x)\le 0$ are satisfied as strict inequalities.*

Note: For convex (P), the Relaxed Slater condition is weaker than the plain Slater condition.

Lagrange Duality Theorem

$$\begin{array}{rcll} \mathrm{Opt}(P) & = & \min\limits_{x\in X\subset\mathbb{R}^n}\{f(x); g_i(x)\le 0, i=1,...,m\} & (P)\\ \Rightarrow L(x,\lambda) & := & f(x)+\sum\limits_{i=1}^m \lambda_i g_i(x) : X\to\mathbb{R}^m_+\to\mathbb{R} & \\ \Rightarrow \underline{L}(\lambda) & = & \inf\limits_{x\in X} L(x,\lambda) : \mathbb{R}^m_+\to\mathbb{R}\cup\{-\infty\} & \\ \Rightarrow \mathrm{Opt}(D) & = & \max\limits_{\lambda\ge 0}\underline{L}(\lambda), & (D)\\ & = & \max\limits_{\lambda\ge 0}\inf\limits_{x\in X} L(x,\lambda) & \end{array}$$

♠ **Lagrange Duality Theorem:** *Under our Standing Assumptions* (which include convexity of (P)) *and Relaxed Slater condition,* (D) *is solvable, and*

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

Illustration:

- Let (P) be the problem

$$\mathrm{Opt}(P)=\min_{x\in X=[0,\infty)}\left\{f(x)=\frac{1}{1+x}:, g_1(x):=20-x\le 0\right\}. \qquad (P)$$

Here $\mathrm{Opt}(P)=\inf_x\{\frac{1}{1+x} : x\ge 20\}=0$, but (P) is *unsolvable*.

However, problem is convex and satisfies Slater condition. We have

$$\underline{L}(\lambda)=\inf_{x\ge 0}\left\{\frac{1}{1+x}+\lambda(20-x)\right\}=\begin{cases} 0, & \lambda=0\\ -\infty, & \lambda>0\end{cases}$$

and (D) is solvable with the optimal solution $\lambda=0$ and optimal value $\mathrm{Opt}(D)=0=\mathrm{Opt}(P)$.

• In LDT all assumptions are essential. For example, the problem

$$\mathrm{Opt}(P) = \min_{x \in X = \mathbb{R}} \left\{ x : g_1(x) := \frac{1}{2}x^2 \leq 0 \right\}, \qquad (P)$$

is convex and solvable with $\mathrm{Opt}(P) = 0$. It, however, does *not* satisfy Slater condition.

We have

$$\underline{L}(x) = \min_x \left\{ x + \frac{\lambda}{2}x^2 \right\} = \begin{cases} -\infty, & \lambda = 0 \\ -\frac{1}{2\lambda}, & \lambda > 0 \end{cases}$$

$\Rightarrow \mathrm{Opt}(D) = 0 = \mathrm{Opt}(P)$ ("by chance"), but the dual problem has no solutions!

Optimality Conditions in CO, Saddle Point From

$$\begin{array}{rcl} \mathrm{Opt}(P) & = & \min\limits_{x\in X\subset\mathbb{R}^n}\{f(x); g_i(x)\le 0, i=1,...,m\} \quad (P) \\ \Rightarrow L(x,\lambda) & := & f(x)+\sum\limits_{i=1}^m \lambda_i g_i(x) : X\to \mathbb{R}^m_+ \to \mathbb{R} \end{array}$$

Theorem [optimality conditions for (P), saddle point form] *Let $x_*\in X$. Then*

(i) *If x_* can be augmented by $\lambda^*\ge 0$ to yield a saddle point (x_*,λ^*) of $L(x,\lambda)$ (*min *in $x\in X$,* max *in $\lambda\ge 0$), that is,*

$$\forall(x\in X,\lambda\ge 0): L(x,\lambda^*)\underbrace{\ge}_{(1)} L(x_*,\lambda^*)\underbrace{\ge}_{(2)} L(x_*,\lambda)$$

then x_* is an optimal solution to (P), and x_*,λ_* satisfy the complementary slackness:

$$\lambda_i^* g_i(x_*)=0,\ 1\le i\le m$$

(ii) *Assume that (P) is convex and satisfies the Relaxed Slater condition. Then x_* is an optimal solution to (P) iff x_* can be augmented, by a properly selected $\lambda^*\ge 0$, to yield a saddle point (x_*,λ^*) of the Lagrange function, and x_*,λ^* satisfy the complementary slackness.*

$$\begin{array}{rcl}\mathrm{Opt}(P) & = & \min\limits_{x\in X\subset\mathbb{R}^n}\{f(x);\, g_i(x)\le 0, i=1,...,m\} \quad (P)\\ \Rightarrow L(x,\lambda) & := & f(x)+\sum\limits_{i=1}^m \lambda_i g_i(x) : X\to \mathbb{R}^m_+\to\mathbb{R}\end{array}$$

"**(i)** *If $x_*\in X$ can be augmented by $\lambda^*\ge 0$ to yield a saddle point (x_*,λ^*) of $L(x,\lambda)$ (*min *in $x\in X$,* max *in $\lambda\ge 0$), that is,*

$$\forall(x\in X,\lambda\ge 0): L(x,\lambda^*)\underbrace{\ge}_{(1)} L(x_*,\lambda^*)\underbrace{\ge}_{(2)} L(x_*,\lambda)$$

then x_* is an optimal solution to (P), and x_*, λ_* satisfy the complementary slackness:

$$\lambda_i^* g_i(x_*)=0,\ 1\le i\le m"$$

Explanation, (i): Let $x_*\in X$ and $\lambda^*\ge 0$ are such that (x_*,λ^*) form a saddle point of L on $X\times\{\lambda\ge 0\}$. Then

• by (2), λ^* is a maximizer of $L(x_*,\lambda)$ as a function of $\lambda\ge 0$. This function is linear in λ, and therefore its minimum in $\lambda\ge 0$ can be achieved at λ^* iff $g_i(x_*)\ge 0$ for all i and complementary slackness holds. By complementary slackness,

$$L(x_*,\lambda^*)\underbrace{=}_{(3)} f(x_*)$$

Note: as a byproduct of our reasoning, we get that *feasibility of x_* for (P) plus complementary slackness is a necessary and sufficient condition for (2) to hold for all $\lambda\ge 0$.*

Now, if x is a feasible solution to (P), then $f(x)\ge L(x,\lambda^*)$ (since $\lambda^*\ge 0$), which combines with (1), (3) to imply that $f(x)\ge f(x_*)$. The bottom line is that x_* is an optimal solution to (P). □

$$\begin{array}{rcll} \mathrm{Opt}(P) & = & \min\limits_{x\in X\subset\mathbb{R}^n}\{f(x); g_i(x)\le 0, i=1,...,m\} & (P)\\ \Rightarrow L(x,\lambda) & := & f(x)+\sum\limits_{i=1}^m \lambda_i g_i(x) : X\to \mathbb{R}^m_+\to\mathbb{R} & \\ \Rightarrow \mathrm{Opt}(D) & = & \max\limits_{\lambda\ge 0}\underline{L}(\lambda), & (D)\\ & = & \max\limits_{\lambda\ge 0}\inf\limits_{x\in X} L(x,\lambda) & (S) \end{array}$$

"**(ii)** *Assume that* (P) *is convex and satisfies the Relaxed Slater condition. Then* $x_*\in X$ *is an optimal solution to* (P) *iff* x_* *can be augmented, by a properly selected* $\lambda^*\ge 0$*, to yield a saddle point* (x_*,λ^*) *of the Lagrange function, and* x_*,λ^* *satisfy the complementary slackness."*

Explanation, (ii): "If x_* can be augmented ... then x_* is an optimal solution to (P) and ..." was already proved in (i) and when checking that the complementary slackness is implied by the fact that (x_*,λ^*) is a saddle point of L on $X\times\{\lambda\ge 0\}$.

Now assume that x_* is an optimal solution to (P), and let us check that then "x_* can be augmented...". Consider the saddle point problem (S). The associated dual problem is (D), and the associated primal problem (P') is

$$\min_{x\in X}\left[\overline{L}(x):=\sup_{\lambda\ge 0}L(x,\lambda)=\begin{cases} f(x), & g_i(x)\le 0\ \forall i\\ +\infty, & \text{otherwise}\end{cases}\right]$$

$\Rightarrow$ x_* *is an optimal solution to* (P')*, and* $\mathrm{Opt}(P')=\mathrm{Opt}(P)$.

By Lagrange Duality Theorem, (D) is solvable with an optimal solution λ^* and optimal value $\mathrm{Opt}(D)=\mathrm{Opt}(P)$.

$\Rightarrow$ (P') *is solvable with optimal solution* x_**,* (D) *is solvable with optimal solution* λ^**, and* $\mathrm{Opt}(P')=\mathrm{Opt}(D)$

$\Rightarrow$ By what we know about saddle points, (x_*,λ^*) is a saddle point of L on $X\times\{\lambda\ge 0\}$. This, as we have seen when proving (i), implies complementary slackness. $\square$

Optimality Conditions in CO, Karush-Kuhn-Tucker Form

$$\begin{aligned} \mathrm{Opt}(P) &= \min_{x\in X\subset\mathbb{R}^n}\{f(x); g_i(x)\le 0, i=1,...,m\} \quad (P)\\ \Rightarrow L(x,\lambda) &:= f(x)+\sum_{i=1}^m \lambda_i g_i(x) : X\to \mathbb{R}^m_+\to\mathbb{R}\end{aligned}$$

Theorem [optimality conditions for (P), KKT form] *Let x_* be a feasible solution to a convex problem (P), and let $f, g_1, ..., g_m$ be differentiable at x_*. Then*

(i) *If x_* is a KKT point of (P), meaning that x_* can be augmented by a properly selected $\lambda^*\ge 0$ to satisfy*

$\lambda^*_j g_j(x_*)=0\,\forall j$ [complementary slackness]

$\nabla f(x_*)+\sum_{i=1}^m \lambda^*_i\nabla g_i(x_*)\in N_X(x_*)$ [KKT equation]

$N_X(x_*)=\{h : \langle h, x'-x\rangle\ge 0\,\forall x'\in X\}$: *normal cone of X at x*

*then (x_*,λ^*) is a saddle point of $L(x,\lambda)$ (*min *in $x\in X$,* max *in $\lambda\ge 0$), whence x_* is an optimal solution to (P).*

(ii) *Assume that, in addition to what stated in the premise of Theorem, (P) satisfies the Relaxed Slater condition. Then x_* is an optimal solution to (P) iff x_* is a KKT point of (P).*

Explanation, (i): If x_* is a KKT point and $\lambda^*\ge 0$ is the associated vector of Lagrange multipliers, then

- x_* is feasible for (P) by assumption, and x_*, λ^* satisfy complementary slackness

  $\Rightarrow$ $L(x_*,\lambda)$ *as a function of $\lambda\ge 0$ attains it maximum at λ^*.*

- The function $h(x)=f(x)+\sum_i\lambda^*_i g_i(x)$ is convex and differentiable at $x_*\in X$ and satisfies $\nabla h(x_*)\in N_X(x_*)$

  $\Rightarrow$ $L(x,\lambda^*)$ *as a function of $x\in X$ attains its minimum at x_**

$\Rightarrow$ (x_*,λ^*) *is a saddle point of L on $X\times\{\lambda\ge 0\}$*

$\Rightarrow$ [previous theorem] x_* *is an optimal solution to (P).* $\square$

$$\begin{array}{rcl} \mathrm{Opt}(P) & = & \min\limits_{x\in X\subset\mathbb{R}^n}\{f(x); g_i(x)\le 0, i=1,...,m\} \quad (P) \\ \Rightarrow L(x,\lambda) & := & f(x)+\sum\limits_{i=1}^m \lambda_i g_i(x) : X\to\mathbb{R}^m_+\to\mathbb{R} \end{array}$$

"**(ii)** *Assume that, in addition to what stated in the premise of Theorem, (P) the Relaxed Slater condition. Then x_* is an optimal solution to (P) iff x_* is a KKT point of (P).*"

Explanation, (ii): "If x_* is a KKT point ... then x_* is an optimal solution to (P)" is stated by (i).

All we need to verify is the claim

"If (P) is convex and satisfies Relaxed Slater condition, f, g_i are differentiable at x_* and x_* is an optimal solution to (P), then x_* is a KKT point of (P)."

• By the Saddle Point form of Optimality Conditions, under the premise of our claim x_* can be augmented by λ^* to yield a saddle point (x_*,λ^*) of L on $X\times\{\lambda\ge 0\}$ and x^*,λ^* satisfy complementary slackness. All we need to verify is the validity of KKT equality.

• The function $h(x)=f(x)\sum_i\lambda_i^* g_i(x)$ is convex and differentiable at $x_*\in X$. Since (x_*,λ^*) is a saddle point of L on $X\times\{\lambda\ge 0\}$, $h(x)$ attains its minimum on X at x_*

$\Rightarrow$ [what we know about minimizing convex function over a convex set] $\nabla h(x_*)\in N_X(x_*)$, which is the KKT equality $\square$

Solving Convex Problems: Ellipsoid Algorithm

♣ There is a wide spectrum of algorithms capable to approximate *global* solutions of convex problems to *high accuracy* in *"reasonable" time.*
We will start with one of the "universal" algorithms of this type – the *Ellipsoid method* imposing only minimal additional to convexity requirements on the problem.
♣ The Ellipsoid method is aimed at solving convex problem in the form

$$\mathrm{Opt}(P) = \min_{x \in X \subset \mathbb{R}^n} f(x)$$

where

• f is a real-valued continuous convex function on X which admits subgradients at every point of X.
f is given by *First Order oracle* – a procedure ("black box") which, given on input a point $x \in X$, returns the value $f(x)$ and a subgradient $f'(x)$ of f at x.
For example, when f is differentiable, it is enough to be able to compute the value and the gradient of f at a point from X.

• X is a *closed and bounded* convex set in $\mathbb{R}^n$ with *nonempty interior*.
X is given by *Separation oracle* – a procedure Sep_X which, given on input a point $x \in \mathbb{R}^n$, reports whether $x \in X$, and if it is not the case, returns a *separator* – a *nonzero* vector $e \in \mathbb{R}^n$ such that

$$\max_{y \in X} e^T y \leq e^T x.$$

$$\mathrm{Opt}(P) = \min_{x \in X \subset \mathbb{R}^n} f(x)$$

♠ Usually, the original description of the feasible domain X of the problem is as follows:

$$X = \{x \in Y : g_i(x) \leq 0,\ 1 \leq i \leq m\}$$

where

- Y is a nonempty convex set admitting a simple Separation oracle Sep_Y.

Example: Let Y be nonempty and given by a list of linear inequalities $a_k^T x \leq b_k$, $1 \leq k \leq K$. Here Sep_Y is as follows:

Given a query point x, we check validity of the inequalities $a_k^T x \leq b_k$. If all of them are satisfied, we claim that $x \in Y$, otherwise claim that $x \notin Y$, take a violated inequality – one with $a_k^T x > b_k$ – and return a_k as the required separator e.

Note: We have $\max_{y \in Y} a_k^T y \leq b_k < a_k^T x$, implying that $e := a_k$ separates x and Y and is nonzero (since $Y \neq \emptyset$).

- $g_i : Y \to \mathbb{R}$ are convex functions on Y given by First Order oracles and such that *given $x \in Y$, we can check whether $g_i(x) \leq 0$ for all i, and if it is not the case, we can find $i_* = i_*(x)$ such that $g_{i_*}(x) > 0$.*

$$X = \{x \in Y : g_i(x) \leq 0,\ 1 \leq i \leq m\}$$

♠ In the outlined situation, assuming X nonempty, Separation oracle Sep_X for X can be built as follows: Given query point $x \in \mathbb{R}^n$, we

— call Sep_Y to check whether $x \in Y$. If it is not the case, $x \notin X$, and the separator of x and Y separates x and X as well. Thus, when Sep_Y reports that $x \notin Y$, we are done.

— when Sep_Y reports that $x \in Y$, we check whether $g_i(x) \leq 0$ for all i. If it is the case, $x \in X$, and we are done. Otherwise we claim that $x \notin X$, find a constraint $g_{i_*}(\cdot) \leq 0$ violated at x: $g_{i_*}(x) > 0$, call First Oracle to compute a subgradient e of $g_{i_*}(\cdot)$ at x and return this e as the separator of x and X.

Note: In the latter case, e is nonzero and separates x and X: since $g_{i_*}(y) \geq g_{i_*}(x) + e^T(y - x) > e^T(y - x)$ and $g_{i_*}(y) \leq 0$ when $y \in X$, we have

$$y \in X \Rightarrow e^T(y - x) < 0$$

It follows that $e \neq 0$ (X is nonempty!) and $\max_{y \in X} e^T y \leq e^T x$.

$$\mathrm{Opt}(P) = \min_{x \in X \subset \mathbb{R}^n} f(x) \qquad (P)$$

Assumptions:

- X is convex, closed and bounded set with $\mathrm{int}\, X \neq \emptyset$ given by Separation oracle Sep_X.
- f is convex and continuous function on X given by First Order oracle $\mathcal{O}_f$.
- [new] *We have an "upper bound" on X – we know $R < \infty$ such that the ball B of radius R centered at the origin contains X,*

(?) *How to solve (P) ?*

To get an idea, let us start with univariate case.

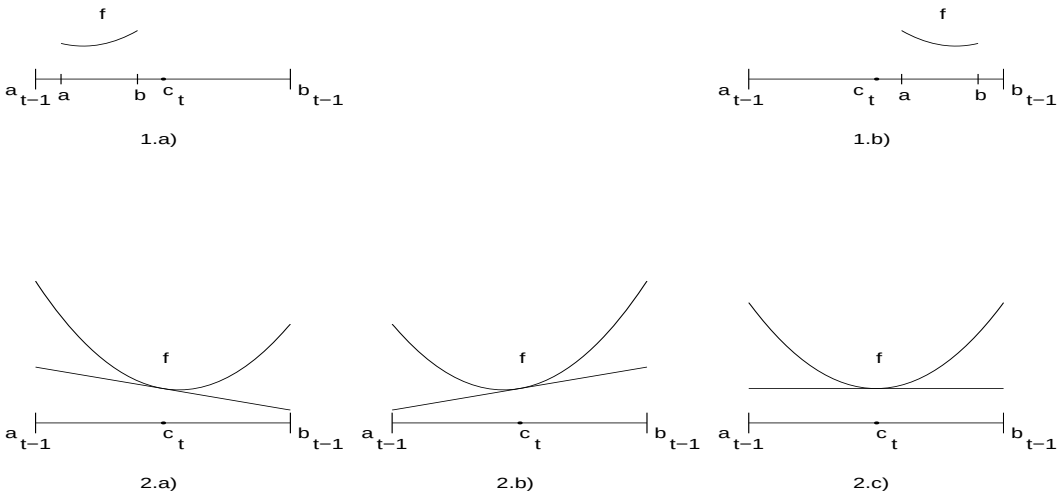
Univariate Case: Bisection

♣ When solving a problem

$$\min_x \{f(x) : x \in X = [a, b] \subset [-R, R]\},$$

by bisection, we recursively update *localizers* – segments $\Delta_t = [a_t, b_t]$ containing the optimal set $X_{\rm opt}$.

- **Initialization:** Set $\Delta_1 = [-R, R]$ $[\supset X_{\rm opt}]$
- **Step** t**:** Given $\Delta_t \supset X_{\rm opt}$ let c_t be the midpoint of Δ_t. Calling Separation and First Order oracles at e_t, we replace Δ_t by *twice smaller* localizer Δ_{t+1}.



1)	Sep_X says that $c_t \notin X$ and reports, via separator e, on which side of c_t X is. 1.a): $\Delta_{t+1} = [a_t, c_t]$; 1.b): $\Delta_{t+1} = [c_t, b_t]$
2)	Sep_X says that $c_t \in X$, and $\mathcal{O}_f$ reports, via $\mathrm{sign} f'(c_t)$, on which side of c_t $X_{\rm opt}$ is. 2.a): $\Delta_{t+1} = [a_t, c_t]$; 2.b): $\Delta_{t+1} = [c_t, b_t]$; 2.c): $c_t \in X_{\rm opt}$

♠ *Since the localizers rapidly shrink and X is of positive length, eventually some of search points will become feasible, and the nonoptimality of the best found so far feasible search point will rapidly converge to 0 as process goes on.*

♠ Bisection admits multidimensional extension, called *Generinc Cutting Plane Algorithm,* where one builds a sequence of "shrinking" *localizers* G_t – closed and bounded convex domains containing the optimal set X_{opt} of (P).

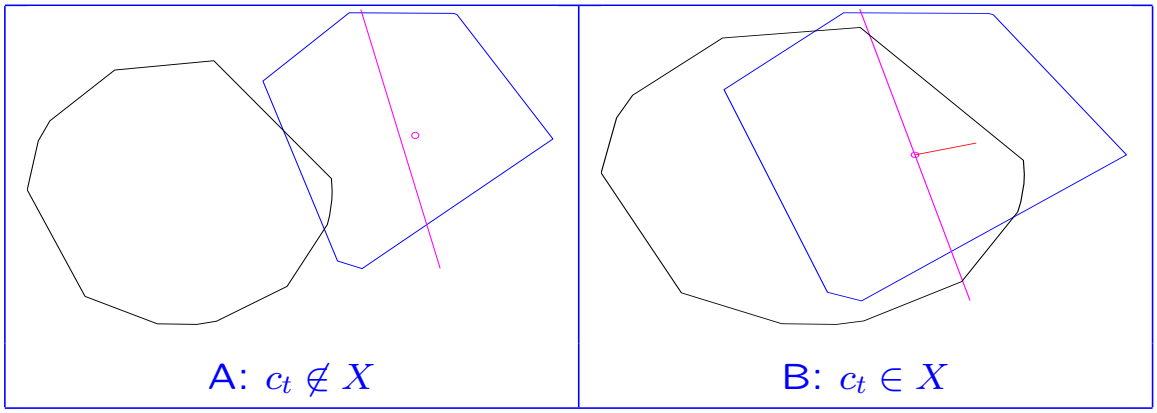
Generic Cutting Plane Algorithm is as follows:

♠ **Initialization** Select as G_1 a closed and bounded convex set containing X and thus being a localizer.

$$\mathrm{Opt}(P) = \min_{x \in X \subset \mathbb{R}^n} f(x) \qquad (P)$$

♠ **Step** $t = 1, 2, ...$**:** Given current localizer G_t,

• Select current *search point* $c_t \in G_t$ and call Separation and First Order oracles to form a *cut* – to find $e_t \neq 0$ such that $X_{\mathrm{opt}} \subset \widehat{G}_t := \{x \in G_t : e_t^T x \leq e_t^T c_t\}$



Black: X; Blue: G_t; Magenta: Cutting hyperplane

To this end

— call Sep_X, c_t being the input. If Sep_X says that $c_t \notin X$ and returns a separator, take it as e_t (case A on the picture).

Note: $c_t \notin X \Rightarrow$ *all points from* $G_t \backslash \widehat{G}_t$ *are infeasible*

— if $c_t \in X_t$, call $\mathcal{O}_f$ to compute $f(c_t)$, $f'(c_t)$. If $f'(c_t) = 0$, terminate, otherwise set $e_t = f'(c_t)$ (case B on the picture).

Note: *When* $f'(c_t) = 0$, c_t *is optimal for* (P), *otherwise* $f(x) > f(c_t)$ *at all feasible points from* $G_t \backslash \widehat{G}_t$

• By the two "Note" above, $\widehat{G}_t$ is a localizer along with G_t. Select a closed and bounded convex set $G_{t+1} \supset \widehat{G}_t$ (it also will be a localizer) and pass to step $t + 1$.

$$\mathrm{Opt}(P) = \min_{x \in X \subset \mathbb{R}^n} f(x) \qquad (P)$$

♠ *Approximate solution x^t built in course of $t = 1, 2, ...$ steps is the best – with the smallest value of f – of the feasible search points $c_1, ..., c_t$ built so far.*

If in course of the first t steps no feasible search points were built, x^t is undefined.

♣ **Analysing Cutting Plane algorithm**

• Let $\mathrm{Vol}(G)$ be the n-dimensional volume of a closed and bounded convex set $G \subset \mathbb{R}^n$.

Note: For convenience, we use, as the unit of volume, the volume of n-dimensional unit ball $\{x \in \mathbb{R}^n : \|x\|_2 \leq 1\}$, and not the volume of n-dimensional unit box.

• Let us call the quantity $\rho(G) = [\mathrm{Vol}(G)]^{1/n}$ the *radius* of G. $\rho(G)$ is the radius of n-dimensional ball with the same volume as G, and this quantity can be thought of as the average linear size of G.

Theorem. *Let convex problem (P) satisfying our standing assumptions be solved by Generic Cutting Plane Algorithm generating localizers G_1, G_2,... and ensuring that $\rho(G_t) \to 0$ as $t \to \infty$. Let $\bar{t}$ be the first step where $\rho(G_{t+1}) < \rho(X)$. Starting with this step, approximate solution x^t is well defined and obeys the "error bound"*

$$f(x^t) - \mathrm{Opt}(P) \leq \min_{\tau \leq t} \left[\frac{\rho(G_{\tau+1})}{\rho(X)}\right] \left[\max_X f - \min_X f\right]$$

$$\mathrm{Opt}(P) = \min_{x \in X \subset \mathbb{R}^n} f(x) \qquad (P)$$

Explanation: Since $\mathrm{int}\, X \neq \emptyset$, $\rho(X)$ is positive, and since X is closed and bounded, (P) is solvable. Let x_* be an optimal solution to (P).

• Let us fix $\epsilon \in (0,1)$ and set $X_\epsilon = x_* + \epsilon(X - x_*)$. X_ϵ is obtained X by similarity transformation which keeps x_* intact and "shrinks" X towards x_* by factor ϵ. This transformation multiplies volumes by ϵ^n $\Rightarrow$ $\rho(X_\epsilon) = \epsilon\rho(X)$.

• Let t be such that $\rho(G_{t+1}) < \epsilon\rho(X) = \rho(X_\epsilon)$. Then $\mathrm{Vol}(G_{t+1}) < \mathrm{Vol}(X_\epsilon)$ $\Rightarrow$ *the set $X_\epsilon \backslash G_t$ is nonempty* $\Rightarrow$ *for some $z \in X$, the point*

$$y = x_* + \epsilon(z - x_*) = (1-\epsilon)x_* + \epsilon z$$

does not belong to G_{t+1}.

• G_1 contains X and thus y, and G_{t+1} does not contain y, implying that *for some $\tau \leq t$, it holds*

$$e_\tau^T y > e_\tau^T c_\tau \qquad (!)$$

• We definitely have $c_\tau \in X$ – otherwise e_τ separates c_τ and $X \ni y$, and (!) witnesses otherwise.

• Thus, $c_\tau \in X$ and therefore $e_\tau = f'(x_\tau)$. By the definition of subgradient, we have $f(y) \geq f(c_\tau) + e_\tau^T(y - c_\tau)$ $\Rightarrow$ [by (!)] $f(c_\tau) \leq f(y) = f((1-\epsilon)x_* + \epsilon z) \leq (1-\epsilon)f(x_*) + \epsilon f(z)$

$$\Rightarrow f(c_\tau) - f(x_*) \leq \epsilon[f(z) - f(x_*)] \leq \epsilon\left[\max_X f - \min_X f\right].$$

Bottom line: *If $0 < \epsilon < 1$ and $\rho(G_{t+1}) < \epsilon\rho(X)$, then x^t is well defined* (since $\tau \leq t$ and c_τ is feasible) *and*

$$f(x^t) - \mathrm{Opt}(P) \leq \epsilon\left[\max_X f - \min_X f\right].$$

$$\mathrm{Opt}(P) = \min_{x \in X \subset \mathbb{R}^n} f(x) \qquad (P)$$

"Starting with the first step $\bar{t}$ where $\rho(G_{t+1}) < \rho(X)$, x^t is well defined, and

$$f(x^t) - \mathrm{Opt} \leq \underbrace{\min_{\tau \leq t} \left[\frac{\rho(G_{\tau+1})}{\rho(X)}\right]}_{\epsilon_t} \underbrace{\left[\max_X f - \min_X f\right]}_{V}"$$

♣ We are done. Let $t \geq \bar{t}$, so that $\epsilon_t < 1$, and let $\epsilon \in (\epsilon_t, 1)$. Then for some $t' \leq t$ we have

$$\rho(G_{t'+1}) < \epsilon \rho(X)$$

$\Rightarrow$ [by bottom line] $x^{t'}$ *is well defined and*

$$f(x^{t'}) - \mathrm{Opt}(P) \leq \epsilon V$$

$\Rightarrow$ [since $f(x^t) \leq f(x^{t'})$ due to $t \geq t'$] x^t *is well defined and* $f(x^t) - \mathrm{Opt}(P) \leq \epsilon V$

$\Rightarrow$ [passing to limit as $\epsilon \to \epsilon_t + 0$] x^t *is well defined and* $f(x^t) - \mathrm{Opt}(P) \leq \epsilon_t V$ □

$$\mathrm{Opt}(P) = \min_{x \in X \subset \mathbb{R}^n} f(x) \qquad (P)$$

♠ **Corollary:** *Let (P) be solved by cutting Plane Algorithm which ensures, for some $\vartheta \in (0,1)$, that*

$$\rho(G_{t+1}) \leq \vartheta \rho(G_t)$$

Then, for every desired accuracy $\epsilon > 0$, finding feasible ϵ-optimal solution x_ϵ to (P) (i.e., a feasible solution x_ϵ satisfying $f(x_\epsilon) - \mathrm{Opt} \leq \epsilon$) *takes at most*

$$N = \frac{1}{\ln(1/\vartheta)} \ln\left(\mathcal{R}\left[1 + \frac{V}{\epsilon}\right]\right) + 1$$

steps of the algorithm. Here

$$\mathcal{R} = \frac{\rho(G_1)}{\rho(X)}$$

says how well, in terms of volume, the initial localizer G_1 approximates X, and

$$V = \max_X f - \min_X f$$

is the variation of f on X.

Note: $\mathcal{R}$, and V/ϵ are under log, implying that high accuracy and poor approximation of X by G_1 cost "nearly nothing."

What matters, is the factor *at the log* which is the larger the closer $\vartheta < 1$ is to 1.

"Academic" Implementation: Centers of Gravity

♠ In high dimensions, to ensure progress in volumes of subsequent localizers in a Cutting Plane algorithm is not an easy task: we do *not* know how the cut through c_t will pass, and thus should select c_t in G_t in such a way that *whatever be the cut*, it cuts off the current localizer G_t a "meaningful" part of its volume.

♠ The most natural choice of c_t in G_t is the *center of gravity*:

$$c_t = \left[\int_{G_t} x dx\right] / \left[\int_{G_t} 1 dx\right],$$

the expectation of the random vector uniformly distributed on G_t.

Good news: The Center of Gravity policy with $G_{t+1} = \widehat{G}_t$ results in

$$\vartheta = \left(1 - \left[\tfrac{1}{n+1}\right]\right)^{1/n} \leq [0.632...]^{1/n}$$

This results in the complexity bound (# of steps needed to build ϵ-solution)

$$N = 2.2n \ln\left(\mathcal{R}\left[1 + \tfrac{V}{\epsilon}\right]\right) + 1$$

Note: It can be proved that *within absolute constant factor*, like 4, *this is the best complexity bound achievable by whatever algorithm for convex minimization which can "learn" the objective via First Order oracle only.*

Disastrously bad news: Centers of Gravity are *not* implementable, unless the dimension n of the problem is like 2 or 3.

Reason: In the method, we have no control on the shape of localizers. Perhaps the best we can say is that if we started with a polytope G_1 given by M linear inequalities, even as simple as a box, then G_t, for meaningful t's, is a more or less arbitrary polytope given by at most $M+t-1$ linear inequalities. And computing center of gravity of a general-type high-dimensional polytope is a computationally intractable task – it requires astronomically many computations already in the dimensions like $5-10$.

Remedy: *Maintain the shape of G_t simple and convenient for computing centers of gravity,* sacrificing, if necessary, the value of ϑ.

The most natural implementation of this remedy is enforcing G_t to be ***ellipsoids***. As a result,

- c_t becomes computable in $O(n^2)$ operations (nice!)
- $\vartheta = [0.632...]^{1/n} \approx \exp\{-0.367/n\}$ increases to $\vartheta \approx \exp\{-0.5/n^2\}$, spoiling the complexity bound

$$N = 2.2n \ln\left(\mathcal{R}\left[1+\tfrac{V}{\epsilon}\right]\right)+1$$

to

$$N = 4n^2 \ln\left(\mathcal{R}\left[1+\tfrac{V}{\epsilon}\right]\right)+1$$

(unpleasant, but survivable...)

Practical Implementation - Ellipsoid Method

♠ *Ellipsoid in* $\mathbb{R}^n$ is the image of the unit n-dimensional ball under one-to-one affine mapping:

$$E = E(B, c) = \{x = Bu + c : u^T u \leq 1\}$$

where B is $n \times n$ nonsingular matrix, and $c \in \mathbb{R}^n$.

• c is the center of ellipsoid $E = E(B, c)$: when $c + h \in E$, $c - h \in E$ as well

• When multiplying by $n \times n$ matrix B, n-dimensional volumes are multiplied by $|\mathrm{Det}(B)|$

$\Rightarrow \mathrm{Vol}(E(B, c)) = |\mathrm{Det}(B)|$, $\rho(E(B, c)) = |\mathrm{Det}(B)|^{1/n}$.

Simple fact: *Let $E(B, c)$ be ellipsoid in $\mathbb{R}^n$ and $e \in \mathbb{R}^n$ be a nonzero vector. The "half-ellipsoid"*

$$\widehat{E} = \{x \in E(B, c) : e^T x \leq e^T c\}$$

is covered by the ellipsoid $E^+ = E(B^+, c^+)$ given by

$$c^+ = c - \tfrac{1}{n+1} Bp, \; p = B^T e / \sqrt{e^T B B^T e}$$

$$B^+ = \tfrac{n}{\sqrt{n^2-1}} B + \left(\tfrac{n}{n+1} - \tfrac{n}{\sqrt{n^2-1}} \right) (Bp) p^T,$$

• *$E(B^+, c^+)$ is the ellipsoid of the smallest volume containing the half-ellipsoid $\widehat{E}$, and the volume of $E(B^+, c^+)$ is strictly smaller than the one of $E(B, c)$:*

$$\vartheta := \tfrac{\rho(E(B^+, c^+))}{\rho(E(B, c))} \leq \exp\{-\tfrac{1}{2n^2}\}.$$

• *Given B, c, e, computing B^+, c^+ costs $O(n^2)$ arithmetic operations.*

$$\mathrm{Opt}(P) = \min_{x \in X \subset \mathbb{R}^n} f(x) \qquad (P)$$

♣ **Ellipsoid method** is the Cutting Plane Algorithm where

- all localizers G_t are ellipsoids:
$$G_t = E(B_t, c_t),$$
- the search point at step t is c_t, and
- G_{t+1} is the smallest volume ellipsoid containing the half-ellipsoid
$$\widehat{G}_t = \{x \in G_t : e_t^T x \le e_t^T c_t\}$$

Computationally, at every step of the algorithm we once call the Separation oracle Sep_X, (at most) once call the First Order oracle $\mathcal{O}_f$ and spend $O(n^2)$ operations to update (B_t, c_t) into (B_{t+1}, c_{t+1}) by explicit formulas.

♠ **Complexity bound** of the Ellipsoid algorithm is
$$N = 4n^2 \ln\left(\mathcal{R}\left[1 + \frac{V}{\epsilon}\right]\right) + 1$$
$$\mathcal{R} = \frac{\rho(G_1)}{\rho(X)}, \; V = \max_{x \in X} f(x) - \min_{x \in X} f(x)$$

Pay attention:

- $\mathcal{R}, V, \epsilon$ are under log $\Rightarrow$ *large magnitudes in data entries and high accuracy are not issues*
- *the factor at the log depends only on the structural parameter of the problem* (its design dimension n) *and is independent of the remaining data.*

What is Inside Simple Fact

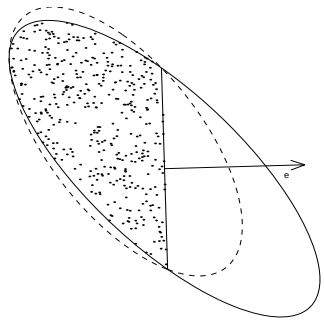
♠ Messy formulas describing the updating

$$(B_t, c_t) \to (B_{t+1}, c_{t+1})$$

in fact are easy to get.

• Ellipsoid E is the image of the unit ball B under affine transformation. *Affine transformation preserves ratio of volumes*

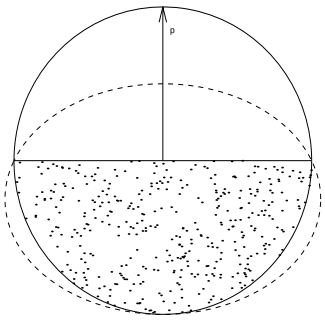
$\Rightarrow$ *Finding the smallest volume ellipsoid containing a given half-ellipsoid $\widehat{E}$ reduces to finding the smallest volume ellipsoid B^+ containing half-ball $\widehat{B}$:*



E, $\widehat{E}$ and E^+

$\Leftrightarrow$

$x = c + Bu$



B, $\widehat{B}$ and B^+

• The "ball" problem is highly symmetric, and solving it reduces to a simple exercise in elementary Calculus.

Why Ellipsoids?

(?) *When enforcing the localizers to be of "simple and stable" shape, why we make them ellipsoids (i.e., affine images of the unit Euclidean ball), and not something else, say parallelotopes (affine images of the unit box)?*

Answer: In a "simple stable shape" version of Cutting Plane Scheme all localizers are affine images of some fixed n-dimensional *solid* $\mathbf{C}$ (closed and bounded convex set in $\mathbb{R}^n$ with a nonempty interior). To allow for reducing step by step volumes of localizers, $\mathbf{C}$ cannot be arbitrary. What we need is the following property of $\mathbf{C}$:

One can fix a point $\mathbf{c}$ *in* $\mathbf{C}$ *in such a way that whatever be a cut*

$$\widehat{\mathbf{C}} = \{x \in \mathbf{C} : e^T x \leq e^T \mathbf{c}\} \quad [e \neq 0]$$

this cut can be covered by the affine image of $\mathbf{C}$ *with the volume less than the one of* $\mathbf{C}$*:*

$$\exists B, b : \widehat{\mathbf{C}} \subset B\mathbf{C} + b \;\&\; |\mathrm{Det}(B)| < 1 \qquad (!)$$

Note: The Ellipsoid method corresponds to unit Euclidean ball in the role of $\mathbf{C}$ and to $\mathbf{c} = 0$, which allows to satisfy (!) with $|\mathrm{Det}(B)| \leq \exp\{-\frac{1}{2n}\}$, finally yielding $\vartheta \leq \exp\{-\frac{1}{2n^2}\}$.

• Solids $\mathbf{C}$ with the above property are "rare commodity." For example, n-dimensional box does *not* possess it.

• Another "good" solid is n-dimensional simplex (this is not that easy to see!). Here (!) can be satisfied with $|\mathrm{Det}(B)| \leq \exp\{-O(1/n^2)\}$, finally yielding $\vartheta = (1 - O(1/n^3))$.

$\Rightarrow$ *From the complexity viewpoint, "simplex" Cutting Plane algorithm is worse than the Ellipsoid method.*

The same is true for handful of other known so far (and quite exotic) "good solids."

Ellipsoid Method: pro's & con's

♣ **Academically speaking,** *Ellipsoid method is an indispensable tool underlying basically all results on efficient solvability of generic convex problems*, most notably, the famous theorem of L. Khachiyan (1978) on *efficient* (scientifically: *polynomial time*, whatever it means) *solvability of Linear Programming with rational data.*

♠ *What matters from theoretical perspective,* is "universality" of the algorithm (nearly no assumptions on the problem except for convexity) and complexity bound of the form *"structural parameter outside of log, all else, including required accuracy, under the log."*

♠ Another theoretical (and to some extent, also practical) advantage of the Ellipsoid algorithm is that *as far as the representation of the feasible set X is concerned, all we need is a Separation oracle, and not the list of constraints describing X.* The number of these constraints can be astronomically large, making impossible to check feasibility by looking at the constraints one by one; however, in many important situations the constraints are "well organized," allowing to implement Separation oracle efficiently.

♠ Theoretically, the only (and minor!) drawbacks of the algorithm is the necessity for the feasible set X to be bounded, with known "upper bound," and to possess nonempty interior.

As of now, there is not way to cure the first drawback without sacrificing universality. The second "drawback" is artifact: given nonempty

$$X = \{x : g_i(x) \leq 0, 1 \leq i \leq m\},$$

we can extend it to

$$X^\epsilon = \{x : g_i(x) \leq \epsilon, 1 \leq i \leq m\},$$

thus making the interior nonempty, and minimize the objective within accuracy ϵ on this larger set, seeking for ϵ-optimal ϵ-feasible solution instead of ϵ-optimal and *exactly feasible* one.

This is quite natural: to find a feasible solution is, in general, not easier than to find an optimal one. Thus, *either ask for exactly feasible and exactly optimal solution* (which beyond LO is unrealistic), or allow for controlled violation in *both* feasibility and optimality!

♠ **From practical perspective**, theoretical drawbacks of the Ellipsoid method become irrelevant: for all practical purposes, bounds on the magnitude of variables like 10^{100} is the same as no bounds at all, and infeasibility like 10^{-10} is the same as feasibility. And since the bounds on the variables and the infeasibility are under log in the complexity estimate, 10^{100} and 10^{-10} are not a disaster.

♠ **Practical limitations** (rather severe!) of Ellipsoid algorithm stem from method's sensitivity to problem's design dimension n. Theoretically, with $\epsilon, V, \mathcal{R}$ fixed, the number of steps grows with n as n^2, and the effort per step is *at least* $O(n^2)$ a.o.

$\Rightarrow$ *Theoretically, computational effort grows with n at least as $O(n^4)$,*

$\Rightarrow$ *n like 1000 and more is beyond the "practical grasp" of the algorithm.*

Note: *Nearly all modern applications of Convex Optimization deal with n in the range of tens and hundreds of thousands!*

♠ By itself, growth of *theoretical* complexity with n as n^4 is not a big deal: for Simplex method, this growth is exponential rather than polynomial, and nobody dies – in reality, Simplex does *not* work according to its disastrous theoretical complexity bound.

Ellipsoid algorithm, unfortunately, works more or less according to its complexity bound.

⇒ *Practical scope of Ellipsoid algorithm is restricted to convex problems with few tens of variables.*

However: Low-dimensional convex problems from time to time do arise in applications. More importantly, these problems arise "on a permanent basis" as auxiliary problems within some modern algorithms aimed at solving *extremely large-scale* convex problems.

⇒ *The scope of practical applications of Ellipsoid algorithm is nonempty, and within this scope, the algorithm, due to its ability to produce high-accuracy solutions* (and surprising stability to rounding errors) *can be considered as the method of choice.*

How It Works

$$\text{Opt} = \min_x f(x),\ X = \{x \in \mathbb{R}^n : a_i^T x - b_i \le 0,\ 1 \le i \le m\}$$

♠ Real-life problem with $n = 10$ variables and $m = 81,963,927$ "well-organized" linear constraints:

CPU, sec	t	$f(x^t)$	$f(x^t) - \text{Opt} \le$	$\rho(G_t)/\rho(G_1)$
0.01	1	0.000000	6.7e4	1.0e0
0.53	63	0.000000	6.7e3	4.2e-1
0.60	176	0.000000	6.7e2	8.9e-2
0.61	280	0.000000	6.6e1	1.5e-2
0.63	436	0.000000	6.6e0	2.5e-3
1.17	895	-1.615642	6.3e-1	4.2e-5
1.45	1250	-1.983631	6.1e-2	4.7e-6
1.68	1628	-2.020759	5.9e-3	4.5e-7
1.88	1992	-2.024579	5.9e-4	4.5e-8
2.08	2364	-2.024957	5.9e-5	4.5e-9
2.42	2755	-2.024996	5.7e-6	4.1e-10
2.66	3033	-2.024999	9.4e-7	7.6e-11

Note: My implementation of Ellipsoid algorithm utilizes simple tricks described in the beginning of "Optional Project," including on-line upper bounding of "optimality gaps" $f(x^t) - \text{Opt}$.

♠ Similar problem with $n = 30$ variables and $m = 1,462,753,730$ "well-organized" linear constraints:

CPU, sec	t	$f(x^t)$	$f(x^t) - \text{Opt} \le$	$\rho(G_t)/\rho(G_1)$
0.02	1	0.000000	5.9e5	1.0e0
1.56	649	0.000000	5.9e4	5.0e-1
1.95	2258	0.000000	5.9e3	8.1e-2
2.23	4130	0.000000	5.9e2	8.5e-3
5.28	7080	-19.044887	5.9e1	8.6e-4
10.13	10100	-46.339639	5.7e0	1.1e-4
15.42	13308	-49.683777	5.6e-1	1.1e-5
19.65	16627	-50.034527	5.5e-2	1.0e-6
25.12	19817	-50.071008	5.4e-3	1.1e-7
31.03	23040	-50.074601	5.4e-4	1.1e-8
37.84	26434	-50.074959	5.4e-5	1.0e-9
45.61	29447	-50.074996	5.3e-6	1.2e-10
52.35	31983	-50.074999	1.0e-6	2.0e-11

Part III: Conic Optimization

- From Linear to Conic Optimization
- Conic Duality
- Interior Point Methods for Linear and Semidefinite Optimization

Conic Optimization: Why?

♠ "Universal" Convex Optimization algorithm, like Ellipsoids method, are *blind* (scientifically: "black box oriented") – they do *not* utilize problem's structure, aside of convexity, and "learn" the problem via local information (values and (sub)gradients of objective and constraints along search points).

At present level of our knowledge, this implies severe limitations on the sizes of convex problems amenable to "universal" algorithms.

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 LO problem. Although *by far* inferior to the Ellipsoid method *in the worst case*, Simplex Method in reality is capable to solve LO'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.

From Linear to Conic Optimization

♠ Before utilizing structure of a convex program, one should "reveal" it.

Revealing structure is a highly challenging task: *it is unclear what we are looking for until we find it!*

♠ The most useful, as of now, "structure revealing" form of convex program – *Conic Optimization* – was found in early 1990's. The idea behind looks really striking (if not crazy):

• Traditionally, when passing from a LO problem

$$\min\{c^T x : Ax - b \leq 0\} \qquad (P)$$

to a convex one,

• linear objective $c^T x$ is replaced with convex objective, and

• affine in x left hand side $Ax-b$ in the vector inequality constraint $Ax - b \leq 0$ is replaced with entrywise convex vector-valued function $A(x)$, yielding the vector inequality constraint $A(x) \leq 0$. *In Conic Optimization, we keep the objective and the left hand side in the vector inequality $Ax - b \leq 0$ linear/affine, and "introduce nonlinearity" in what "≤ 0" means!*

Note: This is not as crazy as it looks. *When comparing numbers*, there is only one meaningful notion of $\leq$. Inequality $\leq$ in (P) is something different: it is specific "entrywise" inequality between *vectors*, with "$a \leq 0$" meaning "all entries in vector a are nonpositive."

On a closed inspection, *the entrywise vector inequality "$\leq$" is neither the only possible, nor the only useful way to compare vectors,* so why to stick to the entrywise $\leq$?

♣ A *Conic Programming* optimization program is

$$\mathrm{Opt} = \min_x \left\{ c^T x : Ax - b \in \mathbf{K} \right\}, \qquad (C)$$

where $\mathbf{K} \subset \mathbb{R}^m$ is a *regular* cone.

♠ *Regularity* of $\mathbf{K}$ means that

- $\mathbf{K}$ *is convex cone:*
  $(x_i \in \mathbf{K}, \lambda_i \geq 0, 1 \leq i \leq p) \Rightarrow \sum_i \lambda_i x_i \in \mathbf{K}$
- $\mathbf{K}$ *is pointed:* $\pm a \in \mathbf{K} \Leftrightarrow a = 0$
- $\mathbf{K}$ *is closed:* $x_i \in \mathbf{K}, \lim_{i\to\infty} x_i = x \Rightarrow x \in \mathbf{K}$
- $\mathbf{K}$ *has a nonempty interior* int $\mathbf{K}$*:*
  $\exists(\bar{x} \in \mathbf{K}, r > 0) : \{x : \|x - \bar{x}\|_2 \leq r\} \subset \mathbf{K}$

Example: The nonnegative orthant

$$\mathbb{R}^m_+ = \{x \in \mathbb{R}^m : x_i \geq 0,\ 1 \leq i \leq m\}$$

is a regular cone, and the associated conic problem (C) is just the usual LO program.

Fact: *When passing from LO programs* (i.e., conic programs associated with nonnegative orthants) *to conic programs associated with properly chosen wider families of cones, we extend dramatically the scope of applications we can process, while preserving the major part of LO theory and preserving our abilities to solve problems efficiently.*

• Let $\mathbf{K} \subset \mathbb{R}^m$ be a regular cone. We can associate with $\mathbf{K}$ two relations between vectors of $\mathbb{R}^m$:

- *"nonstrict $\mathbf{K}$-inequality" $\geq_{\mathbf{K}}$:*

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

- *"strict $\mathbf{K}$-inequality" $>_{\mathbf{K}}$:*

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

Example: when $\mathbf{K} = \mathbb{R}^m_+$, $\geq_{\mathbf{K}}$ is the usual "coordinate-wise" nonstrict inequality "$\geq$" between vectors $a, b \in \mathbb{R}^m$:

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

while $>_{\mathbf{K}}$ is the usual "coordinate-wise" strict inequality "$>$" between vectors $a, b \in \mathbb{R}^m$:

$$a > b \Leftrightarrow a_i > b_i,\ 1 \leq i \leq m$$

♣ $\mathbf{K}$-inequalities share the basic algebraic and topological properties of the usual coordinate-wise $\geq$ and $>$, for example:

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

- $a \geq_{\mathbf{K}} a$ (reflexivity),
- $a \geq_{\mathbf{K}} b$ and $b \geq_{\mathbf{K}} a \Rightarrow a = b$ (anti-symmetry)
- $a \geq_{\mathbf{K}} b$ and $b \geq_{\mathbf{K}} c \Rightarrow a \geq_{\mathbf{K}} c$ (transitivity)

♠ $\geq_{\mathbf{K}}$ is compatible with linear operations:

- $a \geq_{\mathbf{K}} b$ and $c \geq_{\mathbf{K}} d \Rightarrow a + c \geq_{\mathbf{K}} b + d$,
- $a \geq_{\mathbf{K}} b$ and $\lambda \geq 0 \Rightarrow \lambda a \geq_{\mathbf{K}} \lambda b$

♠ $\geq_{\mathbf{K}}$ is stable w.r.t. passing to limits:

$$a_i \geq_{\mathbf{K}} b_i, a_i \to a, b_i \to b \text{ as } i \to \infty \Rightarrow a \geq_{\mathbf{K}} b$$

♠ $>_{\mathbf{K}}$ satisfies the usual arithmetic properties, like

- $a >_{\mathbf{K}} b$ and $c \geq_{\mathbf{K}} d \Rightarrow a + c >_{\mathbf{K}} b + d$
- $a >_{\mathbf{K}} b$ and $\lambda > 0 \Rightarrow \lambda a >_{\mathbf{K}} \lambda b$

and is stable w.r.t perturbations: if $a >_{\mathbf{K}} b$, then $a' >_{\mathbf{K}} b'$ whenever a' is close enough to a and b' is close enough to b.

♣ **Note:** *Conic program associated with a regular cone* $\mathbf{K}$ *can be written down as*

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

Note: Every convex program can be equivalently reformulated as a conic one.

Data and Structure of Conic Program

$$\min_{x\in\mathbb{R}^n}\left\{c^Tx : Ax-b\geq_{\mathbf{K}}0\right\} \qquad \text{(CP)}$$

♠ When asked "what is the data, and what is the structure in (CP)", everybody will give the same answer:

The structure "sits" in the cone K (and in n), *and the data are the entries in c, A, b.*

But: General type convex cone is as "unstructured" as a general type convex function. Why not to say that in a convex program of in the MP form

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

the structure "sits" in the convex functions $f, g_1, ..., g_m$ (and m, n) — definitely true and absolutely useless!

♠ **Fact:** *Conic problems associated with just three specific families of cones cover nearly all* (for all practical purposes – just all) *applications of Convex Optimization.*

Cones from the three "magic" families possess transparent structure fully utilized by theoretically (and practically!) efficient *Interior Point* methods "tailored" to these cones.

⇒ *Reformulating convex program as a conic program from a "magic family" allows to process the problem by highly efficient dedicated algorithms.*

Linear/Conic Quadratic/Semidefinite Optimization

♣ The three magic families of cones are

- Direct products of nonnegative rays – *nonnegative orthants* giving rise to *Linear Optimization*,
- Direct products of *Lorentz cones* giving rise to *Conic Quadratic Optimization*, a.k.a. *Second Order Cone Optimization*,
- Direct products of *Semidefinite cones* giving rise to *Semidefinite Optimization*.

♣ **Linear Optimization.** Let $\mathcal{K} = \mathcal{LO}$ be the family of all nonnegative orthants, i.e., all direct products of nonnegative rays. Conic programs associated with cones from $\mathcal{K}$ are exactly the LO programs

$$\min_x \left\{ c^T x : \underbrace{a_i^T x - b_i \geq 0, 1 \leq i \leq m}_{\Leftrightarrow Ax - b \geq_{\mathbb{R}^m_+} 0} \right\}$$

♣ **Conic Quadratic Optimization.** *Lorentz cone* $\mathbf{L}^m$ of dimension m is the regular cone in $\mathbb{R}^m$ given by

$$\mathbf{L}^m = \{x \in \mathbb{R}^m : x_m \geq \sqrt{x_1^2 + ... + x_{m-1}^2}\}$$

This cone is self-dual.

♠ Let $\mathcal{K} = \mathcal{CQP}$ be the family of all direct products of Lorentz cones. Conic programs associated with cones from $\mathcal{K}$ are called *conic quadratic* programs.

"Mathematical Programming" form of a conic quadratic program is

$$\min_x \left\{c^T x : \underbrace{\|P_i x - p_i\|_2 \leq q_i^T x + r_i}_{\Leftrightarrow [P_i x - p_i; q_i^T x - r_i] \in \mathbf{L}^{m_i}}, 1 \leq i \leq m\right\}$$

Note: According our convention "sum over empty set is 0", $\mathbf{L}^1 = \mathbb{R}_+$ is the nonnegative ray

$\Rightarrow$ *All LO programs are Conic Quadratic ones.*

♣ **Semidefinite Optimization.**

♠ *Semidefinite cone* $\mathbf{S}^m_+$ of order m "lives" in the space $\mathbf{S}^m$ of real symmetric $m\times m$ matrices and is comprised of *positive semidefinite* $m \times m$ *matrices*, i.e., symmetric $m \times m$ matrices A such that $d^T Ad \geq 0$ for all d.

♡ **Equivalent descriptions of positive semidefiniteness:** *A symmetric $m \times m$ matrix A is positive semidefinite (notation: $A \succeq 0$) if and only if it possesses any one of the following properties:*

- *All eigenvalues of A are nonnegative, that is,*
$$A = U\mathrm{Diag}\{\lambda\}U^T$$
with orthogonal U and nonnegative λ.

Note: *In the representation $A = U\mathrm{Diag}\{\lambda\}U^T$ with orthogonal U, $\lambda = \lambda(A)$ is the vector of eigenvalues of A taken with their multiplicities*

- *$A = D^T D$ for a rectangular matrix D, or, equivalently, A is the sum of dyadic matrices: $A = \sum_\ell d_\ell d_\ell^T$*
- *All principal minors of A are nonnegative.*

♡ *The semidefinite cone $\mathbf{S}^m_+$ is regular and self-dual*, provided that the inner product on the space $\mathbf{S}^m$ where the cone lives is inherited from the natural embedding $\mathbf{S}^m$ into $\mathbb{R}^{m\times m}$:

$$\forall A, B \in \mathbf{S}^m : \langle A, B\rangle = \sum_{i,j} A_{ij}B_{ij} = \mathrm{Tr}(AB)$$

♠ Let $\mathcal{K} = \mathcal{SDP}$ be the family of all direct products of Semidefinite cones. Conic programs associated with cones from $\mathcal{K}$ are called *semidefinite programs*. Thus, *a semidefinite program* is an optimization program of the form

$$\min_x \left\{ c^T x : \mathcal{A}_i x - B_i := \sum_{j=1}^n x_j A^{ij} - B_i \succeq 0,\ 1 \le i \le m \right\}$$
$$A^{ij}, B_i : \text{ symmetric } k_i \times k_i \text{ matrices}$$

Note: A collection of symmetric matrices $A_1, ..., A_m$ is comprised of positive semidefinite matrices iff the block-diagonal matrix $\mathrm{Diag}\{A_1, ..., A_m\}$ is $\succeq 0$

$\Rightarrow$ an SDO program can be written down as a problem with a *single* $\succeq$ constraint (called also a *Linear Matrix Inequality* (LMI)):

$$\min_x \left\{ c^T x : \mathcal{A}x - B := \mathrm{Diag}\{\mathcal{A}_i x - B_i, 1 \le i \le m\} \succeq 0 \right\}.$$

♣ Three generic conic problems – Linear, Conic Quadratic and Semidefinite Optimization — posses intrinsic mathematical similarity allowing for deep unified theoretical and algorithmic developments, including design of theoretically *and practically* efficient polynomial time solution algorithms — *Interior Point Methods*.

♠ At the same time, *"expressive abilities" of Conic Quadratic and especially Semidefinite Optimization are incomparably stronger than those of Linear Optimization.* For all practical purposes, *the entire Convex Programming is within the grasp of Semidefinite Optimization*.

LO/CQO/SDO Hierarchy

♠ $\mathbf{L}^1 = \mathbb{R}_+ \Rightarrow \mathcal{LO} \subset \mathcal{CQO} \Rightarrow$ *Linear Optimization is a particular case of Conic Quadratic Optimization.*

♠ **Fact:** *Conic Quadratic Optimization is a particular case of Semidefinite Optimization.*

♡ **Explanation:** The relation $x \geq_{\mathbf{L}^k} 0$ is equivalent to the relation

$$\mathrm{Arrow}(x) = \left[\begin{array}{c|ccccc} x_k & x_1 & x_2 & \cdots & x_{k-1} \\ \hline x_1 & x_k & & & \\ x_2 & & x_k & & \\ \vdots & & & \ddots & \\ x_{k-1} & & & & x_k \end{array}\right] \succeq 0.$$

As a result, a system of conic quadratic constraints

$$A_i x - b_i \geq_{\mathbf{L}^{k_i}} 0,\ 1 \leq i \leq m$$

is equivalent to the system of LMIs

$$\mathrm{Arrow}(A_i x - b_i) \succeq 0,\ 1 \leq i \leq m.$$

Why

$$x \succeq_{\mathbf{L}^k} 0 \Leftrightarrow \mathrm{Arrow}(x) \succeq 0 \qquad (!)$$

Schur Complement Lemma: *A symmetric block matrix* $\begin{bmatrix} P & Q \\ Q^T & R \end{bmatrix}$ *with positive definite* R *is* $\succeq 0$ *if and only if the matrix* $P - QR^{-1}Q^T$ *is* $\succeq 0$.

Proof. We have

$$\begin{bmatrix} P & Q \\ Q^T & R \end{bmatrix} \Leftrightarrow [u;v]^T \begin{bmatrix} P & Q \\ Q^T & R \end{bmatrix} [u;v] \geq 0 \, \forall [u;v]$$
$$\Leftrightarrow u^T P u + 2u^T Q v + v^T R v \geq \forall [u;v]$$
$$\Leftrightarrow \forall u : u^T P u + \min_v \left\{ 2u^T Q v + v^T R v \right\} \geq 0$$
$$\Leftrightarrow \forall u : u^T P u - u^T Q R^{-1} Q^T u \geq 0$$
$$\Leftrightarrow P - QR^{-1}Q^T \succeq 0 \qquad \square$$

♠ **Schur Complement Lemma $\Rightarrow$ (!):**

• In one direction: Let $x \in \mathbf{L}^k$. Then either $x_k = 0$, whence $x = 0$ and $\mathrm{Arrow}(x) \succeq 0$, or $x_k > 0$ and $\sum_{i=1}^{k-1} \frac{x_i^2}{x_k} \leq x_k$, meaning that the matrix

$$\left[\begin{array}{c|ccc} x_k & x_1 & \cdots & x_{k-1} \\ \hline x_1 & x_k & & \\ \vdots & & \ddots & \\ x_{k-1} & & & x_k \end{array}\right]$$

satisfies the premise of the SCL and thus is $\succeq 0$.

- In another direction: let $\left[\begin{array}{c|ccc} x_k & x_1 & \cdots & x_{k-1} \\ \hline x_1 & x_k & & \\ \vdots & & \ddots & \\ x_{k-1} & & & x_k \end{array}\right] \succeq 0.$

Then either $x_k = 0$, and then $x = 0 \in \mathbf{L}^k$, or $x_k > 0$ and $\sum_{i=1}^{k-1} \frac{x_i^2}{x_k} \leq x_k$ by the SCL, whence $x \in \mathbf{L}^k$. $\square$

♣ **Example of CQO program: Control of Linear Dynamical system.** Consider a discrete time linear dynamical system given by

$$\begin{aligned} x(0) &= 0; \\ x(t+1) &= Ax(t) + Bu(t) + f(t), 0 \le t \le T-1 \end{aligned}$$

- $x(t)$: state at time t
- $u(t)$: control at time t
- $f(t)$: given external input

Goal: Given time horizon T, bounds on control $\|u(t)\|_2 \le 1$ for all t and desired destination x_*, find a control which makes $x(T)$ as close as possible to x_*.

The model: From state equations,

$$x(T) = \sum_{t=0}^{T-1} A^{T-t-1}[Bu(t) + f(t)],$$

so that the problem in question is

$$\min_{\tau, u(0), \ldots, u(T-1)} \left\{ \tau : \begin{array}{l} \|x_* - \sum_{t=0}^{T-1} A^{T-t-1}[Bu(t) + f(t)]\|_2 \le \tau \\ \|u(t)\|_2 \le 1,\ 0 \le t \le T-1 \end{array} \right\}$$

♣ **Example of SDO program: Relaxation of a Combinatorial Problem.**

♠ Numerous NP-hard combinatorial problems can be posed as problems of quadratic minimization under quadratic constraints:

$$\begin{array}{c}\mathrm{Opt}(P) = \min\limits_{x}\{f_0(x) : f_i(x) \le 0,\ 1 \le i \le m\} \\ f_i(x) = x^T Q_i x + 2b_i^T x + c_i,\ 0 \le i \le m\end{array} \qquad (P)$$

Example: One can model Boolean constraints $x_i \in \{0;1\}$ as quadratic equality constraints $x_i^2 = x_i$ and then represent them by pairs of quadratic inequalities $x_i^2 - x_i \le 0$ and $-x_i^2 + x_i \le 0$

$\Rightarrow$ *Boolean Programming problems reduce to* (P)*.*

$$\begin{array}{c}\mathrm{Opt}(P) = \min\limits_x \{f_0(x) : f_i(x) \le 0,\ 1 \le i \le m\} \\ f_i(x) = x^T Q_i x + 2b_i^T x + c_i,\ 0 \le i \le m\end{array} \qquad (P)$$

♠ In branch-and-bound algorithms, an important role is played by efficient bounding of $\mathrm{Opt}(P)$ from below. To this end one can use *Semidefinite relaxation* as follows:

- We set $F_i = \left[\begin{array}{c|c} Q_i & b_i \\ \hline b_i^T & c_i \end{array}\right]$, $0 \le i \le m$, and $X[x] = \left[\begin{array}{c|c} xx^T * x & \\ \hline x^T & 1 \end{array}\right]$, so that

$$f_i(x) = \mathrm{Tr}(F_i X[x]).$$

$\Rightarrow$ (P) is equivalent to the problem

$$\min_x \{\mathrm{Tr}(F_0 X[x]) : \mathrm{Tr}(F_i X[x]) \le 0,\ 1 \le i \le m\} \qquad (P')$$

$$\begin{array}{l}\mathrm{Opt}(P) = \min\limits_x \{f_0(x) : f_i(x) \le 0,\ 1 \le i \le m\} \\ \quad \left[f_i(x) = x^T Q_i x + 2b_i^T x + c_i,\ 0 \le i \le m\right] \\ \Leftrightarrow \min\limits_x \{\mathrm{Tr}(F_0 X[x]) : \mathrm{Tr}(F_i X[x]) \le 0,\ 1 \le i \le m\} \\ \quad \left[F_i = \left[\begin{array}{c|c} Q_i & b_i \\ \hline b_i^T & c_i \end{array}\right],\ 0 \le i \le m\right]\end{array} \qquad \begin{array}{c}(P) \\ \\ \\ (P')\end{array}$$

• The objective and the constraints in (P') are linear in $X[x]$, and the only difficulty is that as x runs through $\mathbb{R}^n$, $X[x]$ runs through a difficult for minimization manifold $\mathcal{X} \subset \mathbf{S}^{n+1}$ given by the following restrictions:

A. $X \succeq 0$

B. $X_{n+1,n+1} = 1$

C. $\mathrm{Rank}\, X = 1$

• Restrictions **A**, **B** are simple constraints specifying a nice convex domain

• Restriction **C** is the "troublemaker" – it makes the feasible set of (P) difficult

♠ *In SDO relaxation, we just eliminate the rank constraint* **C**, *thus ending up with the SDO program*

$$\mathrm{Opt(SDO)} = \min_{X \in \mathbf{S}^{n+1}} \left\{ \mathrm{Tr}(F_0 X) : \begin{array}{l} \mathrm{Tr}(F_i X) \le 0,\ 1 \le i \le m, \\ X \succeq 0, X_{n+1,n+1} = 1 \end{array} \right\}.$$

♠ *When passing from $(P) \equiv (P')$ to the SDO relaxation, we extend the domain over which we minimize*

$\Rightarrow \mathrm{Opt(SDO)} \le \mathrm{Opt}(P)$.

What Can Be Expressed via $\mathcal{LO}/\mathcal{CQO}/\mathcal{SDO}$?

♣ Consider a family $\mathcal{K}$ of regular cones such that

• $\mathcal{K}$ is closed w.r.t. taking direct products of cones: $\mathbf{K}_1, ..., \mathbf{K}_m \in \mathcal{K} \Rightarrow \mathbf{K}_1 \times ... \times \mathbf{K}_m \in \mathcal{K}$

• $\mathcal{K}$ is closed w.r.t. passing from a cone to its dual: $\mathbf{K} \in \mathcal{K} \Rightarrow \mathbf{K}_* \in \mathcal{K}$

Examples: $\mathcal{LO}$, $\mathcal{CQO}$, $\mathcal{SDO}$.

Question: *When an optimization program*

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

can be posed as a conic problem associated with a cone from $\mathcal{K}$?

Answer: This is the case when the set X and the function f are *$\mathcal{K}$-representable*, i.e., admit representations of the form

$$\begin{array}{rcl} X & = & \{x : \exists u : Ax + Bu + c \in \mathbf{K}_X\} \\ \mathrm{Epi}\{f\} & := & \{[x;\tau] : \tau \geq f(x)\} \\ & = & \{[x;\tau] : \exists w : Px + \tau p + Qw + d \in \mathbf{K}_f\} \end{array}$$

where $\mathbf{K}_X \in \mathcal{K}$, $\mathbf{K}_f \in \mathcal{K}$.

Indeed, if

$$\begin{array}{rcl} X & = & \{x : \exists u : Ax + Bu + c \in \mathbf{K}_X\} \\ \mathrm{Epi}\{f\} & := & \{[x;\tau] : \tau \geq f(x)\} \\ & = & \{[x;\tau] : \exists w : Px + \tau p + Qw + d \in \mathbf{K}_f\} \end{array}$$

then problem

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

is equivalent to

$$\min_{x,\tau,u,w} \left\{ \tau : \begin{array}{c} \overbrace{Ax + Bu + c \in \mathbf{K}_X}^{\text{says that } x \in X} \\ \underbrace{Px + \tau p + Qw + d \in \mathbf{K}_F}_{\text{says that } \tau \geq f(x)} \end{array} \right\}$$

and the constraints read

$$[Ax + bu + c; Px + \tau p + Qw + d] \in \mathbf{K} := \mathbf{K}_X \times \mathbf{K}_f \in \mathcal{K} \qquad .$$

♣ *$\mathcal{K}$-representable sets/functions always are convex.*

♣ *$\mathcal{K}$-representable sets/functions admit fully algorithmic calculus* completely similar to the one we have developed in the particular case $\mathcal{K} = \mathcal{LO}$.

♠ **Example of $\mathcal{CQO}$-representable function:** convex quadratic form

$$f(x) = x^T A^T A x + 2b^T x + c$$

Indeed,

$$\begin{aligned}
&\tau \geq f(x) \Leftrightarrow [\tau - c - 2b^T x] \geq \|Ax\|_2^2 \\
&\Leftrightarrow \left[\frac{1+[\tau-c-2b^T x]}{2}\right]^2 - \left[\frac{1-[\tau-c-2b^T x]}{2}\right]^2 \geq \|Ax\|_2^2 \\
&\Leftrightarrow \left[Ax; \frac{1-[\tau-c-2b^T x]}{2}; \frac{1+[\tau-c-2b^T x]}{2}\right] \in \mathbf{L}^{\dim b+2}
\end{aligned}$$

♠ **Examples of $\mathcal{SDO}$-representable functions/sets:**

• the maximal eigenvalue $\lambda_{\max}(X)$ of a symmetric $m \times m$ matrix X:

$$\tau \geq \lambda_{\max}(X) \Leftrightarrow \underbrace{\tau I_m - X \succeq 0}_{\text{LMI}}$$

• the sum of k largest eigenvalues of a symmetric $m \times m$ matrix X

• $\mathrm{Det}^{1/m}(X)$, $X \in \mathbf{S}^m_+$

• the set P_d of (vectors of coefficients of) nonnegative on a given segment Δ algebraic polynomials $p(x) = p_d x^d + p_{d-1} x^{d-1} + ... + p_1 x + p_0$ of degree $\leq d$.

Conic Duality Theorem

♣ Consider a conic program

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

As in the LO case, the concept of the dual problem stems from the desire to find a systematic way to bound from below the optimal value $\mathrm{Opt}(P)$.

♠ In the LO case $\mathbf{K} = \mathbb{R}^m_+$ this mechanism was built as follows:

• We observe that for properly chosen vectors of "aggregation weights" λ (specifically, for $\lambda \in \mathbb{R}^m_+$) the aggregated constraint $\lambda^T Ax \geq \lambda^T b$ is the consequence of the vector inequality $Ax \geq_{\mathbf{K}} b$ and thus $\lambda^T Ax \geq \lambda^T b$ for all feasible solutions x to (P)

• In particular, when admissible vector of aggregation weights λ is such that $A^T\lambda = c$, then the aggregated constraint reads "$c^T x \geq b^T\lambda$ for all feasible x" and thus $b^T\lambda$ is a lower bound on $\mathrm{Opt}(P)$. The dual problem is the problem of maximizing this bound:

$$\mathrm{Opt}(D) = \max_\lambda \{ b^T\lambda : \begin{array}{l} A^T\lambda = c \\ \lambda \text{ is admissible for aggregation} \end{array} \}$$

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

♠ The same approach works in the case of a general cone $\mathbf{K}$. The only issue to be resolved is:

What are admissible weight vectors λ for (P)? When a valid vector inequality $a \geq_{\mathbf{K}} b$ always implies the inequality $\lambda^T a \geq \lambda^T b$?

Answer: is immediate: *the required λ's are exactly the vectors from the cone $\mathbf{K}_*$ dual to $\mathbf{K}$.*

Indeed,

- If $\lambda \in \mathbf{K}_*$, then

$$a \geq_{\mathbf{K}} b \Rightarrow a - b \in \mathbf{K} \Rightarrow \lambda^T(a-b) \geq 0 \Rightarrow \lambda^T a \geq \lambda^T b,$$

that is, λ is an admissible weight vector.

- If λ is admissible weight vector and $a \in \mathbf{K}$, that is, $a \geq_{\mathbf{K}} 0$, we should have $\lambda^T a \geq \lambda^T 0 = 0$, so that $\lambda^T a \geq 0$ for all $a \in \mathbf{K}$, i.e., $\lambda \in \mathbf{K}_*$.

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

♠ We arrive at the following construction:

• Whenever $\lambda \in \mathbf{K}_*$, the scalar inequality $\lambda^T Ax \geq \lambda^T b$ is a consequence of the constraint in (P) and thus is valid everywhere on the feasible set of (P).

• In particular, when $\lambda \in \mathbf{K}_*$ is such that $A^T\lambda = c$, the quantity $b^T\lambda$ is a lower bound on $\mathrm{Opt}(P)$, and the dual problem is to maximize this bound:

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

As it should be, in the LO case, where $\mathbf{K} = \mathbb{R}^m_+ = (\mathbb{R}^m_+)_* = \mathbf{K}_*$, (D) is nothing but the LP dual of (P).

♣ Our "aggregation mechanism" can be applied to conic problems in a slightly more general format:

$$\mathrm{Opt}(P) = \min_{x \in X} \left\{ c^T x : \begin{array}{l} A_\ell x \geq_{\mathbf{K}^\ell} b_\ell,\ 1 \leq \ell \leq L \\ Px = p \end{array} \right\} \qquad (P)$$

Here the dual problem is built as follows:

- We associate with every vector inequality constraint

$$A_\ell x \geq_{\mathbf{K}^\ell} b_\ell$$

dual variable ("Lagrange multiplier") $\lambda_\ell \geq_{\mathbf{K}_*^\ell} 0$, so that the scalar inequality constraint $\lambda_\ell^T A_\ell x \geq \lambda_\ell^T b_\ell$ is a consequence of $A_\ell x \geq_{\mathbf{K}^\ell} b_\ell$ and $\lambda_\ell \geq_{\mathbf{K}_\ell^*} 0$;

- We associate with the system $Px = p$ a "free" vector μ of Lagrange multipliers of the same dimension as p, so that the scalar inequality $\mu^T P x \geq \mu^T p$ is a consequence of the vector equation $Px = p$;

- We sum up all the scalar inequalities we got, thus arriving at the scalar inequality

$$\left[\sum_{\ell=1}^L A_\ell^T \lambda_\ell + P^T \mu\right]^T x \geq \sum_{\ell=1}^L b_\ell^T \lambda_\ell + p^T \mu \qquad (*)$$

$$\mathrm{Opt}(P) = \min_{x \in X} \left\{ c^T x : \begin{array}{l} A_\ell x \geq_{\mathbf{K}^\ell} b_\ell,\ 1 \leq \ell \leq L \\ Px = p \end{array} \right\} \qquad (P)$$

Whenever x is feasible for (P) and $\lambda_\ell \geq_{\mathbf{K}_*^\ell} 0$, $1 \leq \ell \leq L$, we have

$$\left[\sum_{\ell=1}^L A_\ell^T \lambda_\ell + P^T \mu\right]^T x \geq \sum_{\ell=1}^L b_\ell^T \lambda_\ell + p^T \mu \qquad (*)$$

- If we are lucky to get in the left hand side of $(*)$ the expression $c^T x$, that is, if $\sum_{\ell=1}^L A_\ell^T \lambda_\ell + P^T \mu = c$, then the right hand side of $(*)$ is a lower bound on the objective of (P) everywhere in the feasible domain of (P) and thus is a lower bound on $\mathrm{Opt}(P)$. The dual problem is to maximize this bound:

$$\begin{array}{l} \mathrm{Opt}(D) \\ = \max\limits_{\lambda,\mu} \left\{ \sum_{\ell=1}^L b_\ell^T \lambda_\ell + p^T \mu : \begin{array}{l} \lambda_\ell \geq_{\mathbf{K}_*^\ell} 0,\ 1 \leq \ell \leq L \\ \sum_{\ell=1}^L A_\ell^T \lambda_\ell + P^T \mu = c \end{array} \right\} \end{array} \qquad (D)$$

Note: When all cones $\mathbf{K}^\ell$ are self-dual (as it is the case in Linear/Conic Quadratic/Semidefinite Optimization), the dual problem (D) involves *exactly the same cones* $\mathbf{K}^\ell$ as the primal problem.

Example: Dual of a Semidefinite program.

Consider a Semidefinite program

$$\min_x \left\{ c^T x : \begin{array}{l} \sum_{i=1}^n A_\ell^j x_j \succeq B_\ell^j, 1 \le \ell \le L \\ Px = p \end{array} \right\}$$

The cones $\mathbf{S}_+^k$ are self-dual, so that the Lagrange multipliers for the $\succeq$-constraints are matrices $\Lambda_\ell \succeq 0$ of the same size as the symmetric data matrices A_ℓ^j, B_ℓ. Aggregating the constraints of our SDO program and recalling that the inner product $\langle A, B \rangle$ in $\mathbf{S}^k$ is $\mathrm{Tr}(AB)$, the aggregated linear inequality reads

$$\sum_{j=1}^n x_j \left[\sum_{\ell=1}^L \mathrm{Tr}(A_\ell^j \Lambda_\ell) + \sum_{j=1}^n (P^T \mu)_j \right] \ge \sum_{\ell=1}^L \mathrm{Tr}(B_\ell \Lambda_\ell) + p^T \mu$$

The equality constraints of the dual should say that the left hand side expression, identically in $x \in \mathbb{R}^n$, is $c^T x$, that is, the dual problem reads

$$\max_{\{\Lambda_\ell\}, \mu} \left\{ \sum_{\ell=1}^L \mathrm{Tr}(B_\ell \Lambda_\ell) + p^T \mu : \begin{array}{r} \mathrm{Tr}(A_\ell^j \Lambda_\ell) + (P^T \mu)_j = c_j, \\ 1 \le j \le n \\ \Lambda_\ell \succeq 0, 1 \le \ell \le L \end{array} \right\}$$

Symmetry of Conic Duality

$$\mathrm{Opt}(P) = \min_{x \in X} \left\{ c^T x : \begin{array}{l} A_\ell x \geq_{\mathbf{K}^\ell} b_\ell,\ 1 \leq \ell \leq L \\ Px = p \end{array} \right\} \qquad (P)$$

$$\begin{aligned} &\mathrm{Opt}(D) \\ &= \max_{\lambda, \mu} \left\{ \sum_{\ell=1}^{L} b_\ell^T \lambda_\ell + p^T \mu : \begin{array}{l} \lambda_\ell \geq_{\mathbf{K}_*^\ell} 0,\ 1 \leq \ell \leq L \\ \sum_{\ell=1}^{L} A_\ell^T \lambda_\ell + P^T \mu = c \end{array} \right\} \end{aligned} \qquad (D)$$

♠ Observe that (D) is, essentially, in the same form as (P), and thus we can build the dual of (D). To this end, we rewrite (D) as

$$\begin{aligned} &-\mathrm{Opt}(D) \\ &= \min_{\lambda_\ell, \mu} \left\{ -\sum_{\ell=1}^{L} b_\ell^T \lambda_\ell - p^T \mu : \begin{array}{l} \lambda_\ell \geq_{\mathbf{K}_*^\ell} 0,\ 1 \leq \ell \leq L \\ \sum_{\ell=1}^{L} A_\ell^T \lambda_\ell + P^T \mu = c \end{array} \right\} \end{aligned} \qquad (D')$$

$$-\mathrm{Opt}(D) = \min_{\lambda_\ell,\mu}\left\{-\sum_{\ell=1}^{L} b_\ell^T\lambda_\ell - p^T\mu : \begin{array}{l}\lambda_\ell \geq_{\mathbf{K}_*^\ell} 0,\ 1\leq\ell\leq L\\ \sum_{\ell=1}^{L} A_\ell^T\lambda_\ell + P^T\mu = c\end{array}\right\} \qquad (D')$$

Denoting by $-x$ the vector of Lagrange multipliers for the equality constraints in (D'), and by $\xi_\ell \geq_{[\mathbf{K}_*^\ell]_*} 0$ (i.e., $\xi_\ell \geq_{\mathbf{K}^\ell} 0$) the vectors of Lagrange multipliers for the $\geq_{\mathbf{K}_*^\ell}$-constraints in (D') and aggregating the constraints of (D') with these weights, we see that everywhere on the feasible domain of (D') it holds:

$$\textstyle\sum_\ell [\xi_\ell - A_\ell x]^T\lambda_\ell + [-Px]^T\mu \geq -c^T x$$

• When the left hand side in this inequality as a function of $\{\lambda_\ell\}, \mu$ is identically equal to the objective of (D'), i.e., when

$$\left\{\begin{array}{rl}\xi_\ell - A_\ell x = -b_\ell & 1\leq\ell\leq L,\\ -Px = -p & \end{array}\right.,$$

the quantity $-c^T x$ is a lower bound on $\mathrm{Opt}(D') = -\mathrm{Opt}(D)$, and the problem dual to (D) thus is

$$\max_{x,\xi_\ell}\left\{-c^T x : \begin{array}{l}A_\ell x = b_\ell + \xi_\ell, 1\leq\ell\leq L\\ Px = p\\ \xi_\ell \geq_{\mathbf{K}^\ell} 0,\ 1\leq\ell\leq L\end{array}\right\}$$

which is equivalent to (P).

$\Rightarrow$ *Conic duality is symmetric!*

Conic Duality Theorem

♠ A conic program in the form

$$\min_y \left\{ c^T y : Ry = r, \underbrace{\begin{array}{rcl} Py & \geq_{\mathbf{K}} & p \\ Sy & \geq & s \end{array}}_{Qy \geq_{\mathbf{M}} q} \right\}$$

is called *strictly feasible*, if there exists a *strictly feasible* solution $\bar{y}$ – a feasible solution where the vector inequality constraint is satisfied as strict: $Q\bar{y} >_{\mathbf{M}} q$. The program is called *essentially strictly feasible*, if there exists a feasible solution $\widehat{y}$ such that $P\widehat{y} >_{\mathbf{K}} p$.

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

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

Conic Duality Theorem

♠ [Weak Duality] *One has* $\mathrm{Opt}(D) \leq \mathrm{Opt}(P)$.

♠ [Symmetry] *duality is symmetric:* (D) *is a conic program, and the program dual to* (D) *is (equivalent to)* (P).

♠ [Strong Duality] *Let one of the problems* (P),(D) *be essentially strictly feasible and bounded. Then the other problem is solvable, and* $\mathrm{Opt}(D) = \mathrm{Opt}(P)$.

In particular, if both (P) *and* (D) *are strictly feasible, then both the problems are solvable with equal optimal values.*

Example: Dual of the SDO relaxation. Recall that given a (difficult to solve!) quadratic quadratically constrained problem

$$\mathrm{Opt}_* = \min_x \{f_0(x) : f_i(x) \geq 0,\ 1 \leq i \leq m\}$$
$$f_i(x) = x^T Q_i x + 2b_i^T x + c_i$$

we can bound its optimal value from below by passing to the *semidefinite relaxation* of the problem:

$$\mathrm{Opt}_* \geq \mathrm{Opt} := \min_X \left\{ \mathrm{Tr}(F_0 X) : \begin{array}{l} \mathrm{Tr}(F_i X) \geq 0, 1 \leq i \leq m \\ X \succeq 0,\ X_{n+1,n+1} \equiv \mathrm{Tr}(GX) = 1 \\ G = \left[\begin{array}{cc} & \\ & 1 \end{array}\right] \end{array} \right\} \qquad (P)$$

$$F_i = \left[\begin{array}{cc} Q_i & b_i \\ b_i^T & c_i \end{array}\right],\ 0 \leq i \leq m.$$

Let us build the dual to (P). Denoting by $\lambda_i \geq 0$ the Lagrange multipliers for the scalar inequality constraints, by $\Lambda \succeq 0$ the Lagrange multiplier for the LMI $X \succeq 0$, and by μ – the Lagrange multiplier for the equality constraint $X_{n+1,n+1} = 1$, and aggregating the constraints, we get the aggregated inequality

$$\mathrm{Tr}([\textstyle\sum_{i=1}^m \lambda_i F_i]X) + \mathrm{Tr}(\Lambda X) + \mu \mathrm{Tr}(GX) \geq \mu$$

Specializing the Lagrange multipliers to make the left hand side to be identically equal to $\mathrm{Tr}(F_0 X)$, the dual problem reads

$$\mathrm{Opt}(D) = \max_{\Lambda, \{\lambda_i\}, \mu} \left\{\mu : F_0 = \textstyle\sum_{i=1}^m \lambda_i F_i + \mu G + \Lambda, \lambda \geq 0, \Lambda \succeq 0\right\}$$

We can easily eliminate Λ, thus arriving at

$$\mathrm{Opt}(D) = \max_{\{\lambda_i\}, \mu} \left\{\mu : \textstyle\sum_{i=1}^m \lambda_i F_i + \mu G \preceq F_0, \lambda \geq 0\right\} \qquad (D)$$

Geometry of Primal-Dual Pair of Conic Problems

♣ Consider a primal-dual pair of conic problems in the form

$$\mathrm{Opt}(P) = \min_x \left\{c^T x : Ax \geq_{\mathbf{K}} b, Px = p\right\} \qquad (P)$$
$$\mathrm{Opt}(D) = \max_{\lambda,\mu} \left\{b^T\lambda + p^T\mu : \lambda \geq_{\mathbf{K}_*} 0, A^T\lambda + P^T\mu = c\right\} \qquad (D)$$

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

$$\exists \bar{x}, \bar{\lambda}, \bar{\mu} : P\bar{x} = p \ \& \ A^T\bar{\lambda} + P^T\bar{\mu} = c$$

♠ Let us pass in (P) from variable x to the slack variable $\xi = Ax - b$. For x satisfying the equality constraints $Px = p$ of (P) we have

$$c^T x = [A^T\bar{\lambda} + P^T\bar{\mu}]^T x = \bar{\lambda}^T Ax + \bar{\mu}^T Px = \bar{\lambda}^T\xi + \bar{\mu}^T p + \bar{\lambda}^T b$$

$\Rightarrow$ (P) is equivalent to

$$\begin{array}{rcll} \mathrm{Opt}(\mathcal{P}) & = & \min_\xi \left\{\bar{\lambda}^T\xi : \xi \in \mathcal{M}_P \cap \mathbf{K}\right\} & (\mathcal{P}) \\ & = & \mathrm{Opt}(P) - \left[b^T\bar{\lambda} + p^T\bar{\mu}\right] & \\ \mathcal{M}_P & = & \mathcal{L}_P - \underbrace{[b - A\bar{x}]}_{\bar{\xi}}, & \\ \mathcal{L}_P & = & \{\xi : \exists x : \xi = Ax, Px = 0\} & \end{array}$$

♠ Let us eliminate from (D) the variable μ. For $[\lambda; \mu]$ satisfying the equality constraint $A^T\lambda + P^T\mu = c$ of (D) we have

$$b^T\lambda + p^T\mu = b^T\lambda + \bar{x}^T P^T\mu = b^T\lambda + \bar{x}^T[c - A^T\lambda] = \underbrace{[b - A\bar{x}]}_{\bar{\xi}}{}^T\lambda + c^T\bar{\xi}$$

$\Rightarrow$ (D) is equivalent to

$$\mathrm{Opt}(\mathcal{D}) = \max_\lambda \left\{\bar{\xi}^T\lambda : \lambda \in \mathcal{M}_D \cap \mathbf{K}_*\right\} = \mathrm{Opt}(D) - c^T\bar{\xi} \quad (\mathcal{D})$$
$$\mathcal{M}_D = \mathcal{L}_D + \bar{\lambda}, \ \mathcal{L}_D = \{\lambda : \exists \mu : A^T\lambda + P^T\mu = 0\}$$

$$\mathrm{Opt}(P) = \min_x \left\{c^T x : Ax \geq_{\mathbf{K}} b, Px = p\right\} \qquad (P)$$
$$\mathrm{Opt}(D) = \max_{\lambda,\mu} \left\{b^T\lambda + p^T\mu : \lambda \geq_{\mathbf{K}_*} 0, A^T\lambda + P^T\mu = c\right\} \qquad (D)$$

♣ **Intermediate Conclusion:** The primal-dual pair (C), (D) of conic problems with feasible equality constraints is equivalent to the pair

$$\mathrm{Opt}(\mathcal{P}) = \min_\xi \left\{\bar{\lambda}^T\xi : \xi \in \mathcal{M}_P \cap \mathbf{K}\right\} = \mathrm{Opt}(P) - \left[b^T\bar{\lambda} + p^T\bar{\mu}\right] \qquad (\mathcal{P})$$
$$\mathcal{M}_P = \mathcal{L}_P - \bar{\xi},\ \mathcal{L}_P = \{\xi : \exists x : \xi = Ax, Px = 0\}$$

$$\mathrm{Opt}(\mathcal{D}) = \max_\lambda \left\{\bar{\xi}^T\lambda : \lambda \in \mathcal{M}_D \cap \mathbf{K}_*\right\} = \mathrm{Opt}(D) - c^T\bar{\xi} \qquad (\mathcal{D})$$
$$\mathcal{M}_D = \mathcal{L}_D + \bar{\lambda},\ \mathcal{L}_D = \{\lambda : \exists\mu : A^T\lambda + P^T\mu = 0\}$$

Observation: The linear subspaces $\mathcal{L}_P$ and $\mathcal{L}_D$ are orthogonal complements of each other.

Observation: Let x be feasible for (P) and $[\lambda, \mu]$ be feasible for (D), and let $\xi = Ax - b$ then the primal slack associated with x. Then

$$\begin{aligned}\mathrm{DualityGap}(x,\lambda,\mu) &= c^T x - [b^T\lambda + p^T\mu] \\ &= [A^T\lambda + P^T\mu]^T x - [b^T\lambda + p^T\mu] \\ &= \lambda^T[Ax - b] + \mu^T[Px - p] = \lambda^T[Ax - b] = \lambda^T\xi.\end{aligned}$$

Note: To solve $(P), (D)$ $\Leftrightarrow$ to minimize the duality gap over primal feasible x and dual feasible λ, μ

$\Leftrightarrow$ *to minimize the inner product of $\xi^T\lambda$ over ξ feasible for $(\mathcal{P})$ and λ feasible for $(\mathcal{D})$.*

♣ **Conclusion:** *A primal-dual pair of conic problems*

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

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

with feasible equality constraints is, geometrically, the problem as follows:

♠ **We are given**

- *a regular cone* $\mathbf{K}$ *in certain* $\mathbb{R}^N$ *along with its dual cone* $\mathbf{K}_*$
- *a linear subspace* $\mathcal{L}_P \subset \mathbb{R}^N$ *along with its orthogonal complement* $\mathcal{L}_D \subset \mathbb{R}^N$
- *a pair of vectors* $\bar{\xi}, \bar{\lambda} \in \mathbb{R}^N$.

These data define

- Primal feasible set $\Xi = [\mathcal{L}_P - \bar{\xi}] \cap \mathbf{K} \subset \mathbb{R}^N$
- Dual feasible set $\Lambda = [\mathcal{L}_D + \bar{\lambda}] \cap \mathbf{K}_* \subset \mathbb{R}^N$

♠ **We want** to find a pair $\xi \in \Xi$ and $\lambda \in \Lambda$ with as small as possible inner product. Whenever Ξ intersects $\mathrm{int}\,\mathbf{K}$ and Λ intersects $\mathrm{int}\,\mathbf{K}_*$, this geometric problem is solvable, and its optimal value is 0 (Conic Duality Theorem).

$$\mathrm{Opt}(P) = \min_x \left\{c^T x : Ax \geq_{\mathbf{K}} b, Px = p\right\} \qquad (P)$$
$$\mathrm{Opt}(D) = \max_{\lambda,\mu} \left\{b^T\lambda + p^T\mu : \lambda \geq_{\mathbf{K}_*} 0, A^T\lambda + P^T\mu = c\right\} \qquad (D)$$

♣ The data $\mathcal{L}_P$, $\bar{\xi}$, $\mathcal{L}_D$, $\bar{\lambda}$ of the geometric problem associated with $(P), (D)$ is as follows:

$\mathcal{L}_P = \{\xi = Ax : Px = 0\}$

$\bar{\xi}$: any vector of the form $Ax - b$ with $Px = p$

$\mathcal{L}_D = \mathcal{L}_P^\perp = \{\lambda : \exists\mu : A^T\lambda + P^T\mu = 0\}$

$\bar{\lambda}$: any vector λ such that $A^T\lambda + P^T\mu = c$ for some μ

• Vectors $\xi \in \Xi$ are exactly vectors of the form $Ax - b$ coming from feasible solutions x to (P), and vectors λ from Λ are exactly the λ-components of the feasible solutions $[\lambda; \mu]$ to (D).

• ξ_*, λ_* form an optimal solution to the geometric problem if and only if $\xi_* = Ax_* - b$ with $Px_* = p$, λ_* can be augmented by some μ_* to satisfy $A^T\lambda_* + P^T\mu_* = c$ and, in addition, x_* is optimal for (P), and $[\lambda_*; \mu_*]$ is optimal for (D).

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]

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

[Indeed,

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

]

and if and only if

- [Complementary Slackness]

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

[Indeed,

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

]

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

$$\Updownarrow$$

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

♣ 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

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

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*?*

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

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

$$\mathrm{Opt} = \min_X \left\{ \mathrm{Tr}(F_0 X) : \begin{array}{l} \mathrm{Tr}(F_i X) \le 0, 1 \le i \le m \\ X = X[x] \text{ for some } x \end{array} \right\} \quad \left[F_i = \begin{bmatrix} Q_i & b_i \\ b_i^T & c_i \end{bmatrix}\right] \qquad (QP')$$

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

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

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

$$\mathrm{Opt(SDO)} = \min_X \left\{ \mathrm{Tr}(F_0 X) : \begin{array}{c} \mathrm{Tr}(F_i X) \le 0, 1 \le i \le M \\ X \succeq 0, X_{n+1,n+1} = 1 \end{array} \right\}$$

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

$$\Updownarrow$$

$$\mathrm{Opt} = \min_X \left\{ \mathrm{Tr}(F_0 X) : \begin{array}{c} \mathrm{Tr}(F_i X) \le 0, 1 \le i \le m \\ X = \left[\begin{array}{cc} xx^T & x \\ x^T & 1 \end{array}\right] \text{ for some } x \end{array} \right\} \qquad (QP')$$
$$\left[F_i = \left[\begin{array}{cc} Q_i & b_i \\ b_i^T & c_i \end{array}\right]\right]$$

$$\Downarrow$$

$$\mathrm{Opt(SDO)} = \min_X \left\{ \mathrm{Tr}(F_0 X) : \begin{array}{c} \mathrm{Tr}(F_i X) \le 0, 1 \le i \le M \\ X \succeq 0, X_{n+1,n+1} = 1 \end{array} \right\} \mathrm{(SDO)}$$

♠ **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 $X_{n+1,n+1} = 1$. *The "randomized" version of* (QP) *is exactly* (SDO) (check it!)

♣ With outlined interpretation, *an optimal solution to* (SDO) *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**

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

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

Note: When $L \succeq 0$ or L has zero diagonal, Opt and Opt(SDO) 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:

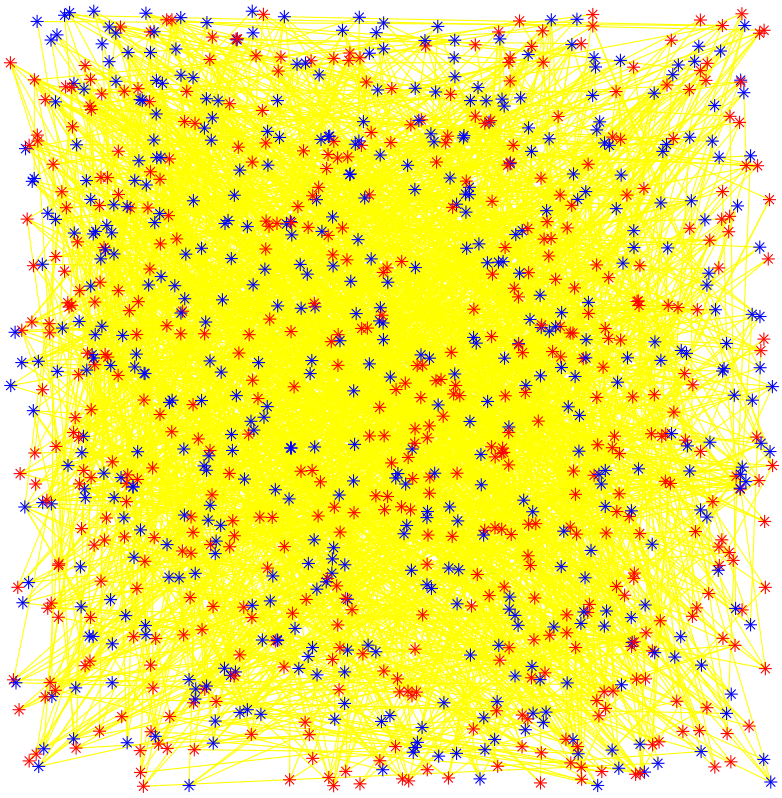
$$\text{Opt} \le \text{Opt(SDO)} \le 1.1383 \cdot \text{Opt} \qquad (!)$$

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

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

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

Illustration: MAXCUT, 1024 nodes, 2614 arcs.



Suboptimal cut, weight $\geq 0.9196 \cdot \mathrm{Opt(SDO)} \geq 0.9196 \cdot \mathrm{Opt}$

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

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

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

Answer [Nesterov'98]: $\mathrm{Opt} \le \mathrm{Opt(SDO)} \le \frac{\pi}{2} \cdot \mathrm{Opt}$.

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

$$\bar{x}^T L \bar{x} \ge 0.7867 \cdot \mathrm{Opt(SDO)} \ge 0.7867 \cdot \mathrm{Opt}$$

$$\begin{array}{rl} \mathrm{Opt} = \max_x\{x^TLx : x_i^2 \le 1,\, 1 \le i \le n\} & (QP) \\ \Rightarrow \mathrm{Opt(SDO)} = \max_X\{\mathrm{Tr}(XL) : X \succeq 0, X_{ii} \le 1\,\forall i\} & (\mathrm{SDO}) \end{array}$$

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

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

This is a particular case of the following result: *The SDP relaxation*

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

of the problem

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

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

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

$$\bar{x}^TL\bar{x} \ge 0.7649 \cdot \mathrm{Opt(SDO)} \ge 0.7649 \cdot \mathrm{Opt}$$

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

$$\bar{x}^TL\bar{x} \ge 0.9969 \cdot \mathrm{Opt(SDO)} \ge 0.9969 \cdot \mathrm{Opt}$$

Lagrangian Relaxation

♣ Recall that for every MP problem

$$\mathrm{Opt}(P) = \min_{x \in X} \{f(x) : g_i(x) \le 0,\ 1 \le i \le m\} \qquad (P)$$

its Lagrange function

$$L(x, \lambda) = f(x) + \sum_i \lambda_i g_i(x)$$

underestimates $f(x)$ on the feasible set of (P), provided
$\lambda \ge 0 \Rightarrow$

$$\mathrm{Opt}(D) = \max_{\lambda \ge 0} \left[\underline{L}(\lambda) := \inf_{x \in X} L(x, \lambda)\right] \le \mathrm{Opt}(P)$$

("Weak Lagrange duality").

♠ *Whenever $\underline{L}$ is efficiently computable, $\mathrm{Opt}(D)$ is an efficiently computable lower bound on $\mathrm{Opt}(P)$.*

♣ **Example:**

$$\text{Opt} = \min_{x \in \mathbb{R}^n} \{f_0(x) : f_i(x) \leq 0, 1 \leq i \leq m\} \quad (QP)$$

$$\left[f_i(x) = x^T Q_i x + 2b_i^T x + c_i,\ 0 \leq i \leq m\right]$$

- Applying Lagrange Relaxation Scheme, we get

$$\begin{aligned} \underline{L}(\lambda) &= \inf_x \left\{f_0(x) + \sum_{i=1}^m \lambda_i f_i(x)\right\} \\ &= \inf_x \left\{ x^T \left[Q_0 + \sum_{i=1}^m \lambda_i Q_i\right] x + 2\left[b_0 + \sum_{i=1}^m \lambda_i b_i\right]^T x \right. \\ &\qquad \left. + \left[c_0 + \sum_{i=1}^m \lambda_i c_i\right] \right\} \end{aligned}$$

Simple Fact: $x^T P x + 2q^T x + r \geq \tau$ *for all* $x \in \mathbb{R}^n$ *iff*

$$\begin{bmatrix} P & q \\ q^T & r - \tau \end{bmatrix} \succeq 0$$

- *Using Simple Fact, the Lagrange dual of* (QP) *becomes*

$$\text{Opt}(D) = \max_{\lambda, \tau} \left\{ \tau : \lambda \geq 0, \begin{bmatrix} Q_0 + \sum_{i=1}^m \lambda_i Q_i & b_0 + \sum_{i=1}^m \lambda_i b_i \\ b_0^T + \sum_{i=1}^m \lambda_i b_i^T & c_0 + \sum_{i=1}^m \lambda_i c_i - \tau \end{bmatrix} \succeq 0 \right\}$$

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

♠ **Note:** *The SDO relaxations of (QP) resulting from our two relaxation schemes read*

$$\mathrm{Opt(SDO)} = \min_X \left\{ \mathrm{Tr}(F_0 X) : \begin{array}{c} \mathrm{Tr}(Q_i X) \le 0, 1 \le i \le M \\ X \succeq 0, X_{n+1,n+1} = 1 \end{array} \right\} \quad (P)$$

$$\mathrm{SDP} = \max_{\lambda,\tau} \left\{ \tau : \left[\begin{array}{c|c} Q_0 + \sum_{i=1}^m \lambda_i Q_i & b_0 + \sum_{i=1}^m \lambda_i b_i \\ \hline b_0^T + \sum_{i=1}^m \lambda_i b_i^T & c_0 + \sum_{i=1}^m c_i\lambda_i - \tau \end{array}\right] \succeq 0, \ \lambda \ge 0 \right\} \quad (D)$$

On a closest inspection, they are just semidefinite duals of each other!

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:** *$\mathcal{SDP}$ 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

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

♠ **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*

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

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

$$x^T Ax + 2a^T x + \alpha \geq 0$$

iff

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

$\Rightarrow$ *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.

$$\exists \lambda_i \geq 0 : B \succeq \sum_{i=1}^m \lambda_i A_i \qquad (!)$$

$\Rightarrow x^T B x \geq 0$ 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

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

• 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.

Interior Point Methods for LO and SDO

Interior Point Methods for LO and SDO

♣ **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 LO, 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 new methods work in the interior of the feasible domain.

♠ Basic theory of IPMs remains the same when passing from LO to SDO

$\Rightarrow$ It makes sense to study this theory in the more general SDO case.

Primal-Dual Pair of SDO Programs

♣ Consider an SDO program in the form

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

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.

Note: In the diagonal case (with the block-diagonal structure in question, all diagonal blocks are of size 1), (P) becomes a LO program with m linear inequality constraints and n variables.

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

♠ **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

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

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

♠ Let $C \in \mathbf{S}^\nu$ satisfy the equality constraint in (D). Passing in (P) from x to the primal slack $X = \mathcal{A}x - b$, we can rewrite (P) equivalently as the problem

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

while (D) is the problem

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

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

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 $\mathrm{Tr}(XS) = 0$.

Fact: *For positive semidefinite 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 $\Rightarrow X^{1/2}SX^{1/2} = 0 \Rightarrow (S^{1/2}X^{1/2})^T(S^{1/2}X^{1/2}) = 0 \Rightarrow S^{1/2}X^{1/2} = 0 \Rightarrow$

$SX = S^{1/2}[S^{1/2}X^{1/2}]X^{1/2} = 0 \Rightarrow XS = (SX)^T = 0.$ $\square$

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

Theorem: *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*

$$XS = SX = 0$$

("SDO Complementary Slackness").

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

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

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

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

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

Back to Basic Analysis: Gradient and Hessian

♣ Consider a smooth (3 times continuously differentiable) function $f(x) : D \to \mathbb{R}$ defined on an open subset D of Euclidean space E.

♠ The *first order directional derivative of* f taken at a point $x \in D$ along a direction $h \in E$ is the quantity

$$Df(x)[h] := \frac{d}{dt}\Big|_{t=0} f(x+th)$$

Fact: *For a smooth* f, $Df(x)[h]$ *is linear in* h *and thus*

$$Df(x)[h] = \langle \nabla f(x), h \rangle \ \forall h$$

for a uniquely defined vector $\nabla f(x)$ *called the gradient of* f *at* x.

If E *is* $\mathbb{R}^n$ *with the standard Euclidean structure, then*

$$[\nabla f(x)]_i = \frac{\partial}{\partial x_i} f(x),\ 1 \le i \le n$$

♠ The *second order directional derivative* of f taken at a point $x \in D$ along a *pair* of directions g, h is defined as

$$D^2 f(x)[g,h] = \frac{d}{dt}\Big|_{t=0} [Df(x+tg)[h]]$$

Fact: *For a smooth f, $D^2 f(x)[g,h]$ is bilinear and symmetric in g, h, and therefore*

$$D^2 f(x)[g,h] = \langle g, \nabla^2 f(x) h\rangle = \langle \nabla^2 f(x) g, h\rangle \forall g, h \in E$$

for a uniquely defined linear mapping $h \mapsto \nabla^2 f(x) h : E \to E$, called the Hessian of f at x.

If E is $\mathbb{R}^n$ with the standard Euclidean structure, then

$$[\nabla^2 f(x)]_{ij} = \frac{\partial^2}{\partial x_i \partial x_j} f(x)$$

Fact: *Hessian is the derivative of the gradient:*

$$\nabla f(x+h) = \nabla f(x) + [\nabla^2 f(x)]h + R_x(h),$$
$$\|R_x(h)\| \le C_x \|h\|^2 \forall (h : \|h\| \le \rho_x), \rho_x > 0$$

Fact: *Gradient and Hessian define the second order Taylor expansion*

$$\widehat{f}(y) = f(x) + \langle y-x, \nabla f(x)\rangle + \tfrac{1}{2}\langle y-x, \nabla^2 f(x)[y-x]\rangle$$

of f at x which is a quadratic function of y with the same gradient and Hessian at x as those of f. This expansion approximates f around x, specifically,

$$|f(y) - \widehat{f}(y)| \le C_x \|y-x\|^3$$
$$\forall (y : \|y-x\| \le \rho_x), \rho_x > 0$$

Back to SDO

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

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

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

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

$$K(X) = -\ln \mathrm{Det} X : \mathbf{S}^\nu_{++} := \{X \in \mathbf{S}^\nu : X \succ 0\} \to \mathbb{R}$$

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*

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

In particular, K is strongly convex:

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

Proof:

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

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

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

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

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

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

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 to each other:*

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

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

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

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

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

Proof of the fact: A. Let us prove that the primal-dual central path is well defined. Let $\bar{S}$ be a strictly feasible solution to (D). For every pair of feasible solutions X, X' to $(\mathcal{P})$ we have

$$\langle X - X', \bar{S}\rangle = \langle X - X', C\rangle + \langle \underbrace{X - X'}_{\in\mathcal{L}_P}, \underbrace{S - C}_{\in\mathcal{L}_D = \mathcal{L}_P^\perp}\rangle = \langle X - X', C\rangle$$

$\Rightarrow$ On the feasible plane of $(\mathcal{P})$, the linear functions $\mathrm{Tr}(CX)$ and $\mathrm{Tr}(\bar{S}X)$ of X differ by a constant

$\Rightarrow$ To prove the existence of $X_*(\mu)$ is the same as to prove that the feasible problem

$$\mathrm{Opt} = \min_{X\in\mathcal{X}} \left[\mathrm{Tr}(\bar{S}X) + \mu K(X)\right] \qquad (R)$$

is solvable.

$$\text{Opt} = \min_{X \in \mathcal{X}} \left[\text{Tr}(\bar{S}X) + \mu K(X)\right] \qquad (R)$$

Let $X_i \in \mathcal{X}$ be such that

$$\left[\text{Tr}(\bar{S}X_i) + \mu K(X_i)\right] \to \text{Opt as } i \to \infty.$$

We claim that a properly selected subsequence $\{X_{i_j}\}_{j=1}^{\infty}$ of the sequence $\{X_i\}$ has a limit $\bar{X} \succ 0$.

Claim $\Rightarrow$ Solvability of (R): Since $X_{i_j} \to \bar{X} \succ 0$ as $j \to \infty$, we have $\bar{X} \in \mathcal{X}$ and

$$\text{Opt} = \lim_{j\to\infty} \left[\text{Tr}(\bar{S}X_{i_j}) + \mu K(X_{i_j})\right] = \left[\text{Tr}(\bar{S}\bar{X}) + \mu K(\bar{X})\right]$$

$\Rightarrow \bar{X}$ is an optimal solution to (R).

Proof of Claim *"Let $X_i \succ 0$ be such that*

$$\lim_{i\to\infty} [\mathrm{Tr}(\bar{S}X_i) + \mu K(X_i)] < +\infty.$$

Then a properly selected subsequence of $\{X_i\}_{i=1}^{\infty}$ has a limit which is $\succ 0$":

First step: Let us prove that X_i form a bounded sequence.

Lemma: *Let $\bar{S} \succ 0$. Then there exists $c = c(\bar{S}) > 0$ such that $\mathrm{Tr}(X\bar{S}) \geq c\|X\|$ for all $X \succeq 0$.*

Indeed, there exists $\rho > 0$ such that $\bar{S} - \rho U \succeq 0$ for all $U \in \mathbf{S}^{\nu}$, $\|U\| \leq 1$. Therefore for every $X \succeq 0$ we have

$$\begin{array}{c}\forall (U, \|U\| \leq 1) : \mathrm{Tr}([\bar{S} - \rho U]X) \geq 0 \\ \Rightarrow \mathrm{Tr}(\bar{S}X) \geq \rho \max\limits_{U:\|U\|\leq 1} \mathrm{Tr}(UX) = \rho\|X\|.\end{array}$$

Now let X_i satisfy the premise of our claim. Then

$$\mathrm{Tr}(\bar{S}X_i) + \mu K(X_i) \geq c(\bar{S})\|X_i\| - \mu \ln(\|X_i\|^m).$$

Since the left hand side sequence is above bounded and $cr - \mu \ln(r^m) \to \infty$ as $r \to +\infty$, the sequence $\|X_i\|$ indeed is bounded.

"Let $X_i \succ 0$ be such that $\lim_{i\to\infty}\left[\mathrm{Tr}(\bar{S}X_i)+\mu K(X_i)\right] < +\infty$. Then a properly selected subsequence of $\{X_i\}_{i=1}^{\infty}$ has a limit which is $\succ 0$"

Second step: Let us complete the proof of the claim. We have seen that the sequence $\{X_i\}_{i=1}^{\infty}$ is bounded, and thus we can select from it a converging subsequence X_{i_j}. Let $\bar{X} = \lim_{j\to\infty} X_{i_j}$. If $\bar{X}$ were a boundary point of $\mathbf{S}^{\nu}_{+}$, we would have

$$\mathrm{Tr}(\bar{S}X_{i_j}) + \mu K(X_{i_j}) \to +\infty, j \to \infty$$

which is not the case. Thus, $\bar{X}$ is an interior point of $\mathbf{S}^{\nu}_{+}$, that is, $\bar{X} \succ 0$.

The existence of $S_*(\mu)$ is proved similarly, with (D) in the role of $(\mathcal{P})$.

The uniqueness of $X_*(\mu)$ and $S_*(\mu)$ follows from the fact that these points are minimizers of strongly convex functions.

B. Let us prove that $S_*(\mu) = \mu X_*^{-1}(\mu)$. Indeed, since $X_*(\mu) \succ 0$ is the minimizer of $P_\mu(X) = \mathrm{Tr}(CX) + \mu K(X)$ on $\mathcal{X} = \{X \in [\mathcal{L}_P - B] \cap \mathbf{S}^\nu_{++}\}$, the first order directional derivatives of $P_\mu(X)$ taken at $X_*(\mu)$ along directions from $\mathcal{L}_P$ should be zero, that is, $\nabla P_\mu(X_*(\mu))$ should belong to $\mathcal{L}_D = \mathcal{L}_P^\perp$. Thus,

$$C - \mu X_*^{-1}(\mu) \in \mathcal{L}_D \Rightarrow S := \mu X_*^{-1}(\mu) \in C + \mathcal{L}_D \,\&\, S \succ 0$$

$\Rightarrow S \in \mathcal{S}$. Besides this,

$$\nabla K(S) = -S^{-1} = -\mu^{-1} X_*(\mu) \Rightarrow \mu \nabla K(S) = -X_*(\mu)$$
$$\Rightarrow \mu \nabla K(S) \in -[\mathcal{L}_P - B] \Rightarrow \mu \nabla K(S) - B \in \mathcal{L}_P = \mathcal{L}_D^\perp$$

$\Rightarrow \nabla D_\mu(S)$ is orthogonal to $\mathcal{L}_D$

$\Rightarrow S$ is the minimizer of $D_\mu(\cdot)$ on $\mathcal{S} = [\mathcal{L}_D + C] \cap \mathbf{S}^\nu_{++}$.

$\Rightarrow \mu X_*^{-1}(\mu) =: S = S_*(\mu)$. $\square$

Duality Gap on the Central Path

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

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

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

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

Therefore 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).

How to Trace 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

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

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

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

where $\mu_{i+1} \in (0, \mu_i)$ and $\Delta X_i, \Delta S_i$ are the solution to the Newton system

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

and $G^{(i)}_\mu(X, S) = 0$ 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

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

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

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

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

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

$$\Rightarrow \left\{\begin{array}{l}
\Delta x_i = -\mathcal{H}^{-1}\left[\mu_{i+1}^{-1} c + \mathcal{A}^*\nabla K(X_i)\right]\\
\Delta X_i = \mathcal{A}\Delta x_i\\
S_{i+1} = S_i + \Delta S_i = \mu_{i+1}\left[\nabla K(X_i) - \nabla^2 K(X_i)\mathcal{A}\Delta x_i\right]
\end{array}\right.$$

♠ $X_i = \mathcal{A}x_i - B$ 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.*

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

♠ 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*

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

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

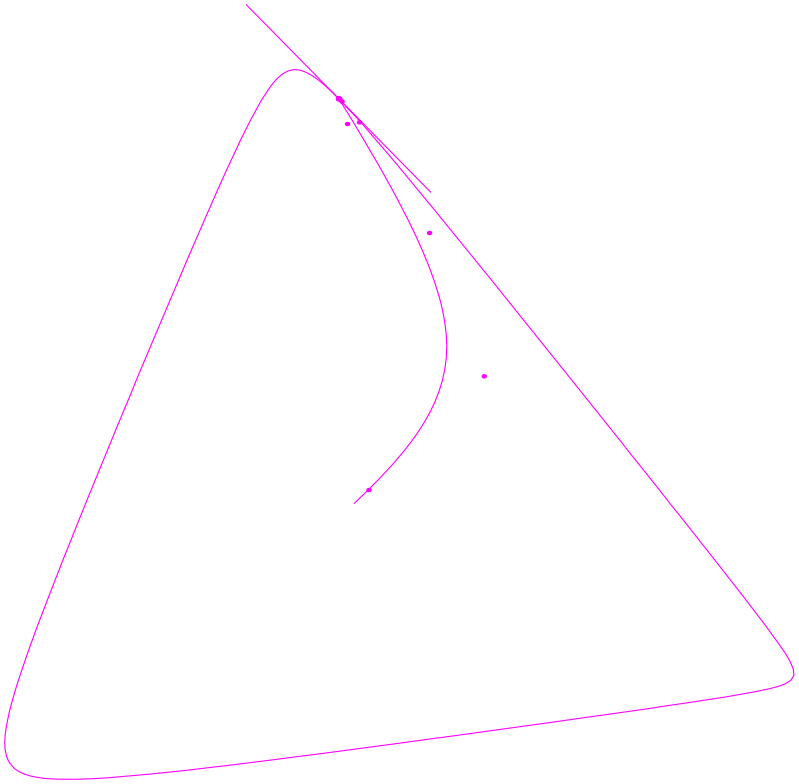
Theorem. *Let $(X_0 = \mathcal{A}x_0 - B, S_0, \mu_0)$ be close to the primal-dual central path, and let (P) be solved by the Primal path-following method where the path parameter μ is updated according to*

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

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 the 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 $\mathrm{Tr}(X_iS_i)$ of i-th iterate does not exceed $2m\mu_i$.

$\Rightarrow$ *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 SDO ($\mathbf{K} = \mathbf{S}^3_+$).
"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 LO and SDO.

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

♠ 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.*

⇒ 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

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

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

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

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

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

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

$\widehat{\mathcal{P}}$ 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 LO 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, $\widehat{X} = \widetilde{S}$).

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

Theorem: *Let 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. As a result, every $O(\sqrt{m})$ steps of the method reduce 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 a more aggressive fashion 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.