Linear Programming

Linear Program (LP)

Input: $A \in \mathbb{R}^{m \times n}$, $b \in \mathbb{R}^m$, $c \in \mathbb{R}^n$.

A: m×n matrix; b, c: column vectors

Output: $x \in \mathbb{R}^n$ s.t. (1) $Ax \le b$, (2) $x \ge 0$, (3) $\langle c, x \rangle$ is maximized.

x: decision varaiable(s)

(1), (2): (linear) constraints — $(Ax)_i \le b_i \quad \forall\, 1 \le i \le m$

(3): (linear) objective function. $\langle c, x \rangle = \sum_{i=1}^{n} c_i x_i = c^T x$.

* "continuous optimization" (output is a set of real values).

* Can model numerous scenarios in business, economics, and engineering. (1975 Nobel Prize in Economics, Simplex method: "top 10 algos" in 20^{th} century)

* Basic problem in mathematical optimization (convex programming)

* Useful for "discrete problems" too

Ex) Optimized diet

	meat	rice
protein	20	15
calorie	400	400
price	4	2

calorie limit: 2000

price : 12

maximize protein while satisfying calorie & price constraints?

Let x_1 = amount of meat
x_2 = " " rice.

Then, we want to maximize $20x_1 + 15x_2$

subject to $400x_1 + 400x_2 \le 2000$

$4x_1 + 2 \cdot x_2 \le 12$

$x_1, x_2 \ge 0$.

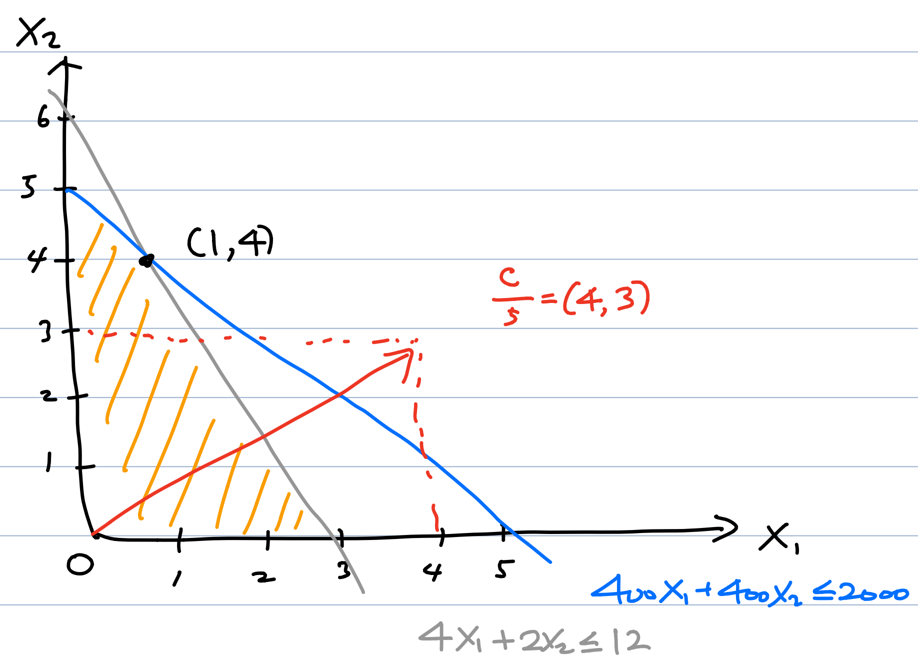
When $c = \begin{bmatrix} 20 \\ 15 \end{bmatrix}$, $b = \begin{bmatrix} 2000 \\ 12 \end{bmatrix}$, $A = \begin{bmatrix} 400 & 400 \\ 4 & 2 \end{bmatrix}$, $x = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$,

equivalent to maximize $\langle c, x \rangle$

s.t. $Ax \le b$

(subject to) $x \ge 0$

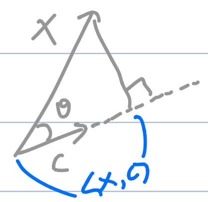
Geometrically,



Find the point in feasible region $\{x \in \mathbb{R}^n : Ax \leq b, x \geq 0\}$ that is

longest in c's direction (largest $\langle x, c \rangle$)

(if $\|c\|_2 = 1$ WLOG, $\langle x, c \rangle = \|x\| \cdot \|c\| \cdot \cos\theta$ is the length of the projection of x to c's direction.)



Algebraic Aspects of LP

Linearity,

When $x \in \mathbb{R}^n$, $f(x): \mathbb{R}^n \to \mathbb{R}$ is "linear in x" if $f(x) = \sum_{i=1}^{n} a_i x_i$. Generally $f(x): \mathbb{R}^n \to \mathbb{R}^m$ is linear if $f(x) = Ax$ for some matrix $A \in \mathbb{R}^{m \times n}$.

If f is linear, $f(ax+by) = f(ax) + f(by) = af(x) + bf(y)$
$\forall a, b \in \mathbb{R}$, $\forall x, y \in \mathbb{R}^n$

Constraint $f(x) \le b$ is linear if f is linear.

Generality,

Is $\max \langle c, x \rangle$ s.t. $Ax \le b$, $x \ge 0$ general enough? YES.

Suppose we want to solve

$\min \langle c, x \rangle$ s.t. $Ax \le b$, $Dx = e$, $Fx \ge g$

⓪ Replace $Fx \ge g$ by $-Fx \le -g$.

① Replace $Dx = e$ by $Dx \le e$ and $-Dx \le -e$. (all constraints inequality)

② Replace each original variable x_i by $x_i^+ - x_i^-$ where $x_i^+, x_i^- \ge 0$.

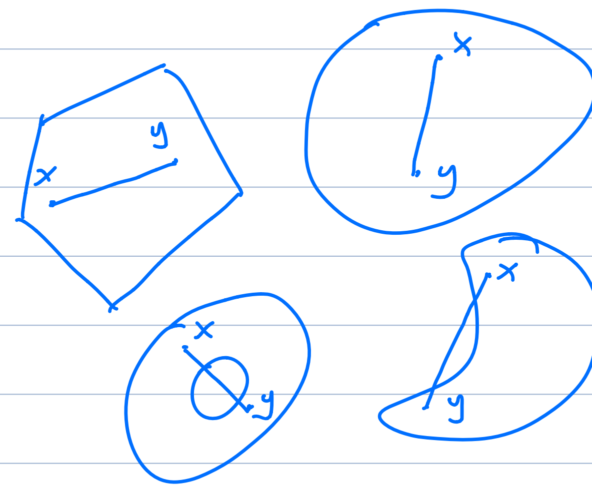
③ $\min \langle c, x \rangle$ is same as $\max \langle -c, x \rangle$

Ex,

$$\begin{aligned} \min\ & x_1 + x_2 + x_3 \\ \text{s.t. } & x_1 + 4x_2 \ge 4 \\ & x_1 - 2x_3 = 1 \end{aligned} \iff \begin{aligned} \max\ & -(x_1^+ - x_1^-) - (x_2^+ - x_2^-) - (x_3^+ - x_3^-) \\ \text{s.t. } & -(x_1^+ - x_1^-) - 4(x_2^+ - x_2^-) \le -4 \\ & (x_1^+ - x_1^-) - 2(x_3^+ - x_3^-) \le 1 \\ & -(x_1^+ - x_1^-) + 2(x_3^+ - x_3^-) \le -1 \\ & x_1^+, x_1^-, x_2^+, x_2^-, x_3^+, x_3^- \ge 0 \end{aligned}$$

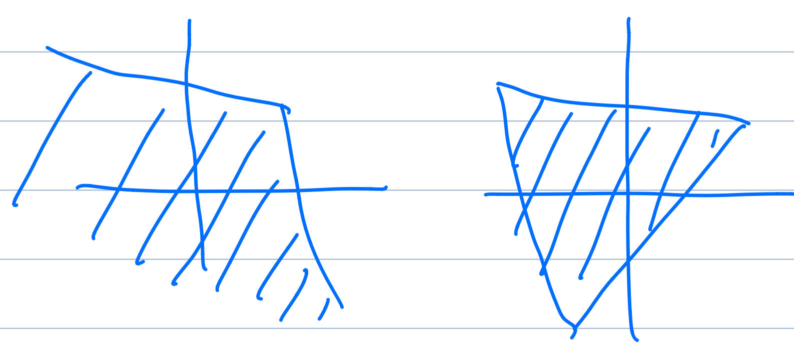
Basic Convex Geometry

Def. A set $S \subseteq \mathbb{R}^n$ is "convex" if $\forall x, y \in S$ and $\lambda \in [0,1]$,
$\lambda \cdot x + (1-\lambda) y \in S$.



Def. $S \subseteq \mathbb{R}^n$ is a "polyhedron" if $S = \{x : Px \le q\}$ for some $P \in \mathbb{R}^{m \times n}$, $q = \mathbb{R}^m$

and "polytope" if $\exists R \in \mathbb{R}$ s.t. $\forall x \in S$, $\|x\|_2 \le R$



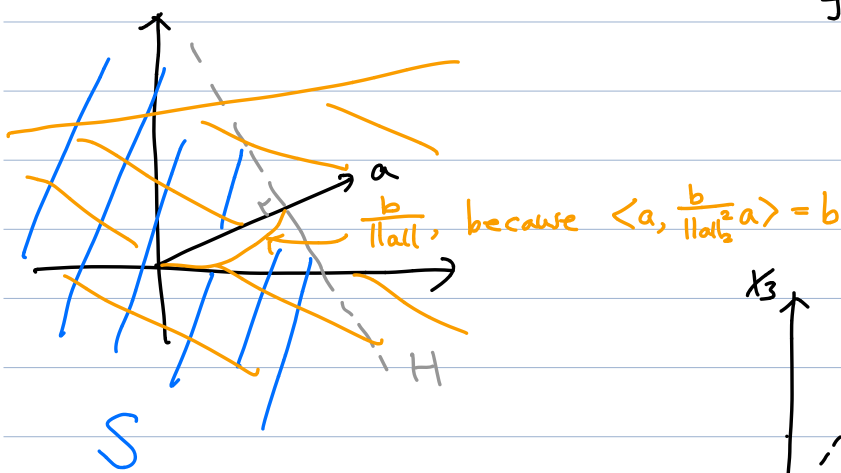
Of course, if $S = \{x : Ax \le b, x \ge 0\}$, then

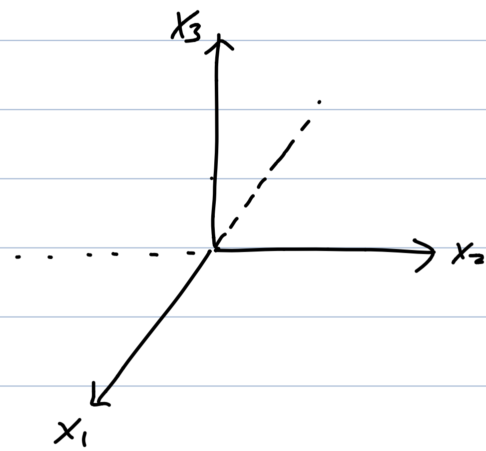
$$P = \underbrace{\begin{bmatrix} A \\ -I \end{bmatrix}}_{n} \begin{matrix} \}m \\ \}n \end{matrix} \quad q = \begin{bmatrix} b \\ 0 \end{bmatrix} \begin{matrix} \}m \\ \}n \end{matrix}$$

ensures $S = \{x : Px \le q\}$ is a polyhedron.

Def. $H \subseteq \mathbb{R}^n$ is called "hyperplane" if $H = \{x : \langle a, x\rangle = b\}$ for some $a \in \mathbb{R}^n$, $b \in \mathbb{R}$.

$S \subseteq \mathbb{R}^n$ is called "halfspace" if $S = \{x : \langle a, x\rangle \le b\}$ for some $a \in \mathbb{R}^n$, $b \in \mathbb{R}$.





hyperplane (i.e., $\{x : x_3 = 0\}$) is "of dimension $n-1$"

halfspace (i.e., $\{x : x_3 \ge 0\}$) is "of dimension n"

Note that given polyhedron $S = \{Px \le q\}$, $P = \begin{bmatrix} - p_1 - \\ \vdots \\ - p_m - \end{bmatrix}$, $q = \begin{bmatrix} q_1 \\ \vdots \\ q_m \end{bmatrix}$,
$S_i = \{x : \langle p_i, x\rangle \le q_i\}$ is a halfspace and
$S = \bigcap_{i=1}^{m} S_i$.

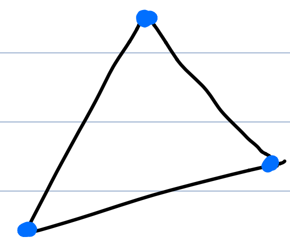
Claim. Every hyperplane/halfspace is convex.

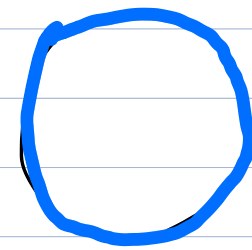
Claim. If $S, T \subseteq \mathbb{R}^n$ are convex, so is $S \cap T$.

LP Algorithms

Def: Given a convex set $S \subseteq \mathbb{R}^n$, $x \in S$ is an "extreme point" if $\nexists\, y, z \in S$, $\lambda \in [0,1]$ s.t.

$$x \neq y, z \quad \text{and} \quad x = \lambda y + (1-\lambda) z.$$

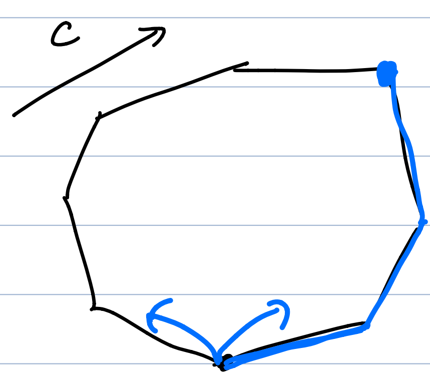




For polyhedra, "vertices"

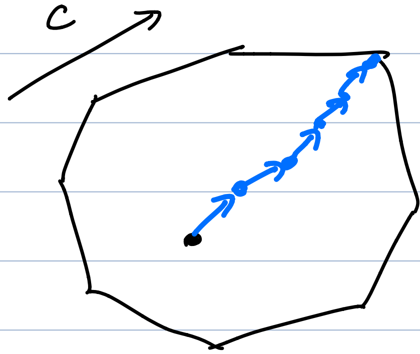
Lemma: For any LP, there is an optimal $x \in \mathbb{R}^n$, which is an extreme point of the feasible set.

Simplex: Move to a better extreme point.



- works great in practice
- can take exponential # steps.
- but bad instances are gone after small perturbations (smoothed analysis)

Interior Point: In each iteration, convert it to "unconstrained" problem and use iterative method for that.

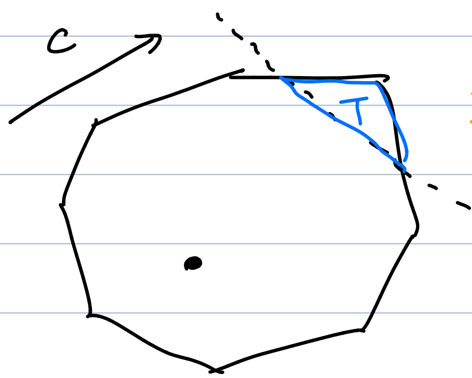


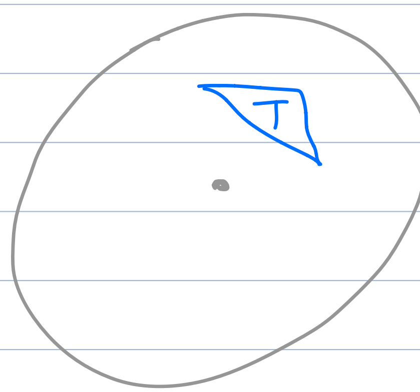
- works well in practice
- polynomial time.

Ellipsoid: Check $\overbrace{S \cap \{x : \langle c, x\rangle \ge d\}}^{=T}$ is nonempty or not by maintaining "ellipsoid" $E \supseteq T$,

$\{x \in \mathbb{R}^n : \|B(x-c)\|_2 \le \delta\}$ for some $B \in \mathbb{R}^{m\times n}$, $c \in \mathbb{R}^n$, $\delta \in \mathbb{R}$

c: center.

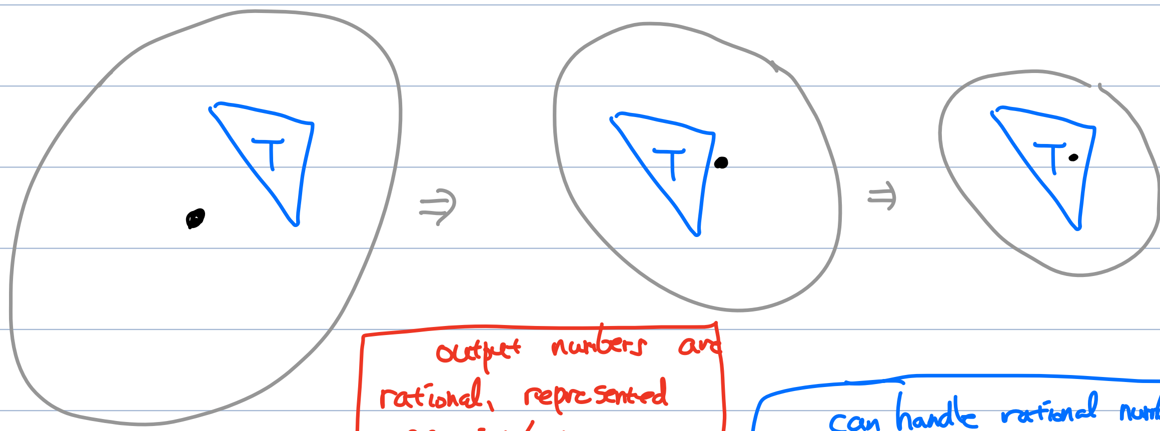




Check center $c \in T$.

YES: done,

NO: "Shrink" E to a smaller ellipsoid.



output numbers are rational, represented as int/int

can handle rational numbers by multiplying least common mult. to denominators

Running time: if all numbers of A, b, c are integers whose absolute value is $\leq M$, can be solved in time $\text{poly}(n, m, \log M)$.

Fastest runtime: $\max(n,m)^{\omega} \cdot \log M$ — ω: matrix mult. constant $\approx 2.37\ldots$

technically, polynomial time since input size = (# bits) $= \Theta(n \cdot m \cdot \log M)$.

Generally, consider input with n integers with magnitude $\leq M$.

(# bits in input $= n \cdot \log M$).

Pseudopolynomial time : runs in $\text{poly}(n, M)$.

Polynomial time : runs in $\text{poly}(n, \log M)$.

Strongly poly time : runs in $\text{poly}(n)$ time

when we assume basic operations of large numbers takes $O(1)$ time

Open Question, strongly poly-time algo for LP?

(YES for Max-Flow via Edmonds-Karp!)

Max Flow

s-t Max Flow

Input: Directed graph $G=(V,E)$ with capacity $c: E \to \mathbb{R}^{\geq 0}$, $s, t \in V$.

Output: Find a s-t flow f feasible for G s.t. $\|f\|$ is maximized.

A **flow** is a function $f: E \to \mathbb{R}^{\geq 0} \leftarrow \{x \in \mathbb{R} : x \geq 0\}$

f is **feasible** (for G) if $\forall e \in E$, $0 \leq f(e) \leq c(e)$

Given a flow $f: E \to \mathbb{R}^{\geq 0}$ and $v \in V$, define "net flow at v" to be

$$\Delta f(v) := \underbrace{\sum_{(v,w) \in E} f(v,w)}_{\text{outgoing flow}} - \underbrace{\sum_{(u,v) \in E} f(u,v)}_{\text{incoming flow}}$$

f is **s-t flow** if $\Delta f(v) = 0 \quad \forall v \neq s, t$.

$$\|f\| = \Delta f(s).$$

Then, it can be represented as the following LP. Variables: $f \in \mathbb{R}^E$

$$\text{maximize} \sum_{w:(s,w)\in E} f_{(s,w)} - \sum_{u:(u,s)\in E} f_{(u,s)}$$

$$\text{subject to} \quad f_e \le c_e \qquad \forall e \in E.$$

$$\sum_{w:(s,w)\in E} f_{(s,w)} - \sum_{u:(u,s)\in E} f_{(u,s)} = 0 \qquad \forall v \in V \setminus \{s,t\}$$

$$f_e \ge 0 \qquad \forall e \in E.$$