# Computational Geometry

## Polytopes, optimization and beyond

Vissarion Fisikopoulos

Department of Informatics & Telecommunications
National & Kapodistrian University of Athens

Spring 2025

# What is Linear Programming?

## Definition

A linear programming problem asks for a vector $x$ that maximizes or minimizes a given linear function, among all vectors $x$ that satisfy a given set of linear inequalities.

**Standard Form:**

$$\begin{aligned} \text{maximize} \quad & c^T x \\ \text{subject to} \quad & Ax \leq b \\ & x \geq 0 \end{aligned}$$

where $x \in \mathbb{R}^n$, $c \in \mathbb{R}^n$, $A \in \mathbb{R}^{m \times n}$, $b \in \mathbb{R}^m$

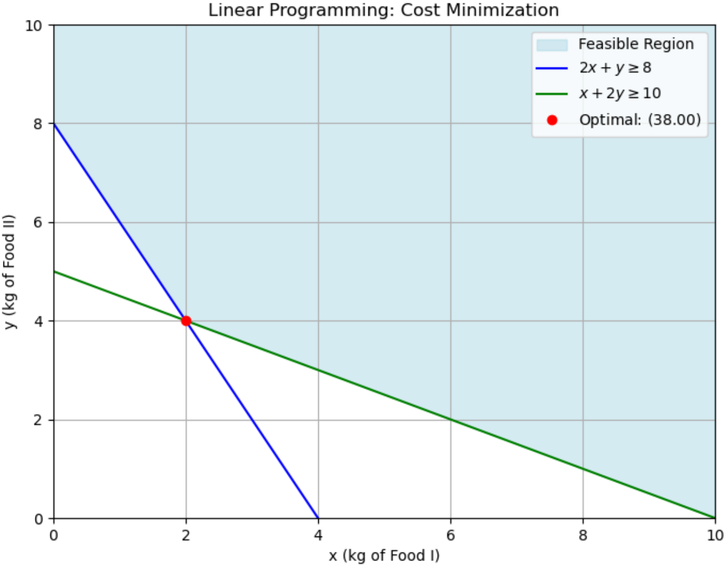
# Example: Minimizing Cost of a Nutritional Mix

- A doctor wants to mix two foods to meet vitamin requirements.
- At least 8 units of vitamin A and 10 units of vitamin C are needed.
- Food I: 2 units of A, 1 unit of C per kg, costs \$5/kg.
- Food II: 1 unit of A, 2 units of C per kg, costs \$7/kg.

**Let:** $x =$ kg of Food I, $y =$ kg of Food II

$$\begin{aligned} \text{Minimize} \quad & Z = 5x + 7y \\ \text{Subject to} \quad & 2x + y \geq 8 \quad \text{(Vitamin A)} \\ & x + 2y \geq 10 \quad \text{(Vitamin C)} \\ & x \geq 0,\ y \geq 0 \end{aligned}$$

# Example: Minimizing Cost of a Nutritional Mix



# Geometry of LP

The set of all feasible points of an LP is called the feasible region. The feasible region is a polytope.

- What is the dimension of this polytope?
- How many facets does this polytope have?
- Does an LP have always a (unique) solution? What are the polytopes for those edge cases?

# Finding Extreme Points in V-polytopes

- A finite set of points $\{x_1, x_2, \ldots, x_m\} \subset \mathbb{R}^n$
- Identify which points are *extreme points* of the convex hull:

$$P = \text{conv}(x_1, x_2, \ldots, x_m)$$

- For each point $x_i$, solve the LP:

$$\text{Find } \lambda_j \text{ for } j \neq i$$
$$\text{such that } x_i = \sum_{j \neq i} \lambda_j x_j$$
$$\sum_{j \neq i} \lambda_j = 1, \quad \lambda_j \geq 0$$

- Redundancy removal in H-polytopes via duality

# Maximum Inscribed Ball in an H-Polytope

- Find the largest Euclidean ball contained in a polytope $P$ defined by linear inequalities.
- $P = \{x \in \mathbb{R}^n \mid a_i^T x \leq b_i,\ i = 1, \ldots, m\}$
- Ball $B(x_0, r) = \{x_0 + u \mid \|u\| \leq r\}$
- LP formulation

$$\begin{aligned} \max \quad & r \\ \text{subject to} \quad & a_i^T x_0 + \|a_i\| r \leq b_i, \quad i = 1, \ldots, m \end{aligned}$$

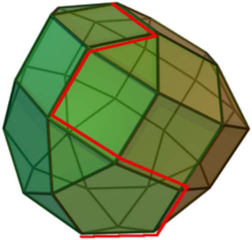
- Note: $x_0 + r\frac{a_i}{\|a_i\|}$ max point in $B$ w.r.t. $a_i$

# Simplex Method

- George Dantzig, 1947
- Moves along edges of the feasible polytope
- Exponential time worst-case, fast in practice

**Idea:**

1. Start at a basic feasible solution (vertex)
2. Move to adjacent vertex with better objective
3. Repeat until optimality



# Ellipsoid Method

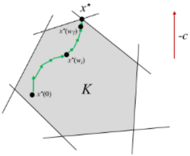
- First polynomial-time algorithm for LP (Khachiyan, 1979)
- Uses ellipsoids to enclose feasible region and iteratively shrink
- Theoretically important, but impractical

- Start: $E_0$ ellipsoid containing $P$
- While $x_i$ center of $E_i$ not in $P$ ($H_i$ separtes $x_i$ from $P$) do
- $\quad E_{i+1}$ ellipsoid contains $E_i \cap \{H_i\}$
- Property: the ellipsoids shrink in volume

# Interior Point Methods

- First efficient practical polynomial-time algorithm
- Moves through interior of feasible region, not on edges
- Popular in large-scale optimization

- First find an interior point (by solving a simpler LP with at trivial starting point)
- Defines a "central path" and computes points on it by solving "similar" optimization problems (typically by Newton's method)



# Further Reading

- Bertsimas - Introduction to Linear Optimization
- Boyd, Vandenberghe - Convex Optimization

# Outline

Linear Programming

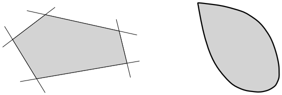
Randomized Algorithms for Convex Optimization

Polytopes and Applications

# Optimization

Given $P$ a convex body in $\mathbb{R}^n$:

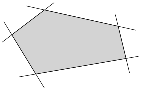
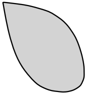
- minimize a convex function $f$ in $P$ (**convex optimization**).



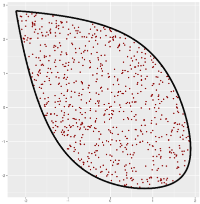
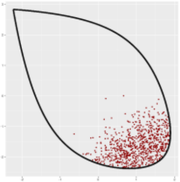
# Optimization

Given $P$ a convex body in $\mathbb{R}^n$:

- minimize a convex function $f$ in $P$ (**convex optimization**).

Goal: Randomized approximation algorithms based on sampling from $P$ with geometric random walks.

# Convex optimization - Special cases

## Linear program

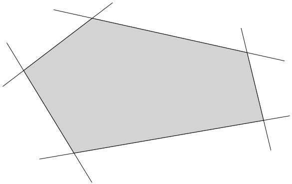
- The objective function is linear $f(\mathbf{x}) = \mathbf{c} \cdot \mathbf{x}$.
- The body is given as an intersection of $m$ half-spaces.

# Convex optimization - Special cases

## Linear program

- The objective function is linear $f(\mathbf{x}) = \mathbf{c} \cdot \mathbf{x}$.
- The body is given as an intersection of $m$ half-spaces.

H-polytope : $P = \{x \mid Ax \leq b,\ A \in \mathbb{R}^{m \times n},\ b \in \mathbb{R}^m\}$



# Convex optimization - Special cases

## Semidefinite program

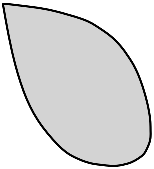
- The objective function is linear $f(\mathbf{x}) = \mathbf{c} \cdot \mathbf{x}$.
- The body is given as a Linear Matrix Inequality (LMI).

# Convex optimization - Special cases

## Semidefinite program

- The objective function is linear $f(\mathbf{x}) = \mathbf{c} \cdot \mathbf{x}$.
- The body is given as a Linear Matrix Inequality (LMI).

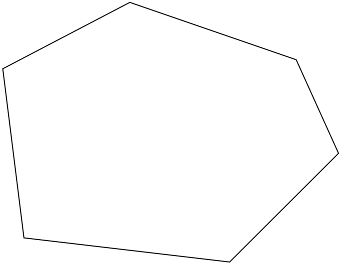
Spectrahedron : $K = \{x \mid A_0 + x_1 A_1 + \cdots + x_d A_d \succeq 0\}$,
where $A_i$: symmetric matrices, $B \succeq 0$: $B$ is positive semidefinite (symmetric with non-negative eigenvalues)



# Cutting planes

Dabbene, Shcherbakov, Polyak, 10'

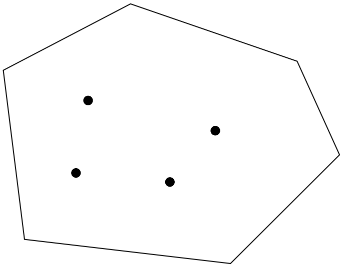
- Input: convex body $K$, objective function $c$.



# Cutting planes

Dabbene, Shcherbakov, Polyak, 10'

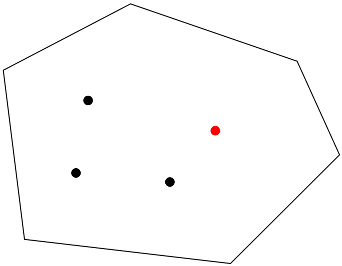
- Input: convex body $K$, objective function $c$.
- Sample $N$ points under the uniform distribution.



# Cutting planes

Dabbene, Shcherbakov, Polyak, 10'

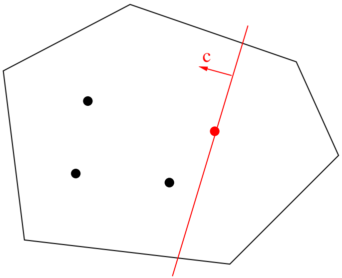
- Input: convex body $K$, objective function $c$.
- Sample $N$ points under the uniform distribution.
- Find the point $x$ minimizing the objective function.



# Cutting planes

Dabbene, Shcherbakov, Polyak, 10'

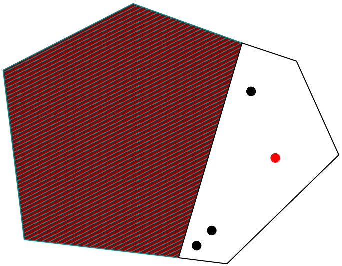
- Input: convex body $K$, objective function $c$.
- Sample $N$ points under the uniform distribution.
- Find the point $x$ minimizing the objective function.
- Cut the convex body at $x$.



# Cutting planes

Dabbene, Shcherbakov, Polyak, 10'

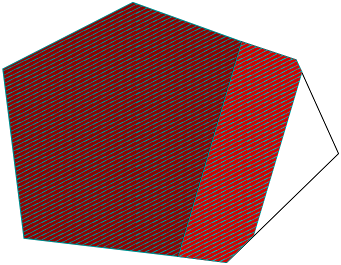
- Input: convex body $K$, objective function $c$.
- Sample $N$ points under the uniform distribution.
- Find the point $x$ minimizing the objective function.
- Cut the convex body at $x$.
- Repeat $I$ times.



# Cutting planes

Dabbene, Shcherbakov, Polyak, 10'

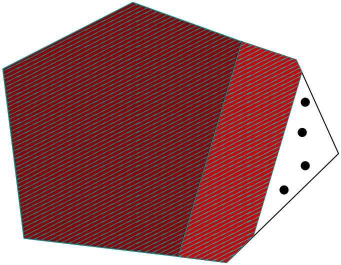
- Input: convex body $K$, objective function $c$.
- Sample $N$ points under the uniform distribution.
- Find the point $x$ minimizing the objective function.
- Cut the convex body at $x$.
- Repeat $I$ times.



# Cutting planes

Dabbene, Shcherbakov, Polyak, 10'

- Input: convex body $K$, objective function $c$.
- Sample $N$ points under the uniform distribution.
- Find the point $x$ minimizing the objective function.
- Cut the convex body at $x$.
- Repeat $I$ times.



# Cutting planes

- Let $rB_d \subseteq K \subseteq RB_d$.
- The expected number of phases s.t. $|f_I - f^*| < \epsilon$ is,

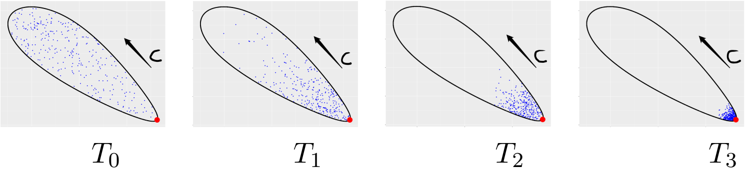
$$I = \left\lceil \frac{1}{\ln(N+1)} d \ln(R/\epsilon) \right\rceil = O^*(d)$$

# Exponential sampling and Simulated Annealing

Kalai, Vempala, 06’

Problem: Minimize a linear function $f(\mathbf{x}) = \mathbf{c} \cdot \mathbf{x}$ in body $K$.

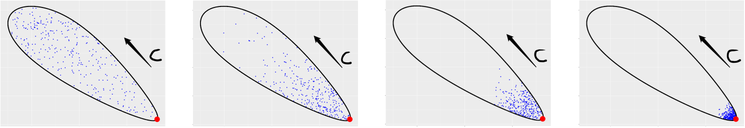
Answer: Sample from $\pi_T(\mathbf{x}) \propto e^{-\mathbf{c}\cdot\mathbf{x}/T}$, for $T = T_0 > \cdots > T_I$.




A sample from $\pi_{T_I}$ is $\epsilon$-close to the optimal solution with high probability.

# Simulated Annealing

## Fix the sequence of Temperatures




$T_0$ $T_1$ $T_2$ $T_3$

- The sequence $T_0 > \cdots > T_I$ is fixed s.t. the $L_2$ norm of $\pi_{T_i}$ w.r.t. $\pi_{T_{i+1}}$ is bounded by a constant,

$$||\pi_{T_i}/\pi_{T_{i+1}}|| = \mathbb{E}_{\pi_{T_i}}\left[\frac{d\pi_{T_i}}{d\pi_{T_{i+1}}}\right] = \int_K \frac{\pi_{T_i}(x)}{\pi_{T_i+1}(x)}\pi_{T_i}(x)dx = O(1)$$

- Then $\pi_{T_i}$ is a warm start for $\pi_{T_{i+1}}$ (Hit-and-Run).

# Simulated Annealing

## Convergence to the optimal solution

- Starting with $T_0 = R$ (uniform distribution is a warm start).

# Simulated Annealing

## Convergence to the optimal solution

- Starting with $T_0 = R$ (uniform distribution is a warm start).
- Set $T_i = T_0(1 - \frac{1}{\sqrt{d}})^i, \; i = 1, \ldots, I$ ($T_i$ is a warm start for $T_{i+1}$).

# Simulated Annealing

## Convergence to the optimal solution

- Starting with $T_0 = R$ (uniform distribution is a warm start).
- Set $T_i = T_0(1 - \frac{1}{\sqrt{d}})^i,\ i = 1, \ldots, I$ ($T_i$ is a warm start for $T_{i+1}$).
- Knowing that for a temp. $T$,

$$\mathbb{E}_{\pi_T}[\mathbf{c} \cdot \mathbf{x}] \leq dT + \min_{x \in K} \mathbf{c} \cdot \mathbf{x}$$

# Simulated Annealing

## Convergence to the optimal solution

- Starting with $T_0 = R$ (uniform distribution is a warm start).
- Set $T_i = T_0(1 - \frac{1}{\sqrt{d}})^i,\ i = 1, \ldots, I$ ($T_i$ is a warm start for $T_{i+1}$).
- Knowing that for a temp. $T$,

$$\mathbb{E}_{\pi_T}[\mathbf{c} \cdot \mathbf{x}] \leq dT + \min_{x \in K} \mathbf{c} \cdot \mathbf{x}$$

- $I = O^*(\sqrt{d})$ phases suffices to obtain a solution $|f_I - f^*| \leq \epsilon$.

# Simulated Annealing

## Convergence to the optimal solution

- Starting with $T_0 = R$ (uniform distribution is a warm start).
- Set $T_i = T_0(1 - \frac{1}{\sqrt{d}})^i, \ i = 1, \ldots, I$ ($T_i$ is a warm start for $T_{i+1}$).
- Knowing that for a temp. $T$,

$$\mathbb{E}_{\pi_T}[\mathbf{c} \cdot \mathbf{x}] \leq dT + \min_{x \in K} \mathbf{c} \cdot \mathbf{x}$$

- $I = O^*(\sqrt{d})$ phases suffices to obtain a solution $|f_I - f^*| \leq \epsilon$.
- No sequence of distributions $\propto f_i(\mathbf{c} \cdot \mathbf{x})$ can, in general, solve the problem in less than $\Omega(\sqrt{d})$ phases.

# Outline

Linear Programming

Randomized Algorithms for Convex Optimization

Polytopes and Applications

# Birkhoff polytopes

- Given the complete bipartite graph $K_{n,n} = (V, E)$ a perfect matching is $M \subseteq E$ s.t. every vertex meets exactly one member of $M$

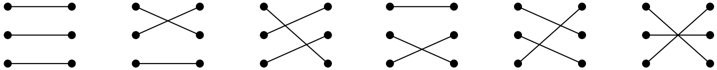
# Birkhoff polytopes

- Given the complete bipartite graph $K_{n,n} = (V, E)$ a perfect matching is $M \subseteq E$ s.t. every vertex meets exactly one member of $M$
- $S \subseteq E,\ \chi_e^S = \{1 \text{ if } e \in S, 0 \text{ otherwise}\}$

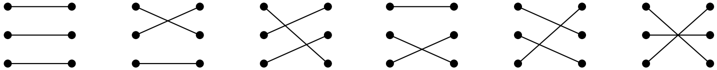
# Birkhoff polytopes

- Given the complete bipartite graph $K_{n,n} = (V, E)$ a perfect matching is $M \subseteq E$ s.t. every vertex meets exactly one member of $M$
- $S \subseteq E,\ \chi_e^S = \{1 \text{ if } e \in S, 0 \text{ otherwise}\}$
- $B_n = conv\{\chi^M \mid M \text{ is a perfect matching of } K_{n,n}\}$

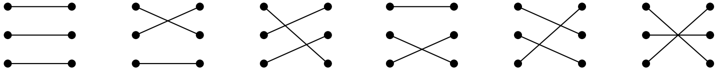
# Birkhoff polytopes

- Given the complete bipartite graph $K_{n,n} = (V, E)$ a perfect matching is $M \subseteq E$ s.t. every vertex meets exactly one member of $M$
- $S \subseteq E, \ \chi_e^S = \{1$ if $e \in S, 0$ otherwise$\}$
- $B_n = conv\{\chi^M \mid M$ is a perfect matching of $K_{n,n}\}$
- 

# Birkhoff polytopes

- Given the complete bipartite graph $K_{n,n} = (V, E)$ a perfect matching is $M \subseteq E$ s.t. every vertex meets exactly one member of $M$
- $S \subseteq E,\ \chi_e^S = \{1 \text{ if } e \in S, 0 \text{ otherwise}\}$
- $B_n = conv\{\chi^M \mid M \text{ is a perfect matching of } K_{n,n}\}$
- 
- # faces of $B_3$: 6, 15, 18, 9; $\mathrm{vol}(B_3) = 9/8$

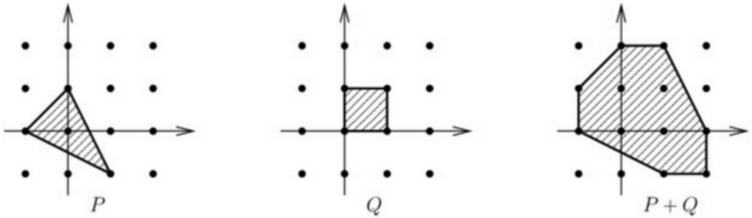
# Birkhoff polytopes

- Given the complete bipartite graph $K_{n,n} = (V, E)$ a perfect matching is $M \subseteq E$ s.t. every vertex meets exactly one member of $M$
- $S \subseteq E,\ \chi_e^S = \{1 \text{ if } e \in S, 0 \text{ otherwise}\}$
- $B_n = conv\{\chi^M \mid M \text{ is a perfect matching of } K_{n,n}\}$
- 
- # faces of $B_3$: 6, 15, 18, 9; $\text{vol}(B_3) = 9/8$
- there exist formulas for the volume [deLoera et al '07] but values only known for $n \leq 10$ after 1yr of parallel computing [Beck et al '03]

# Minkowski sum

The Minkowski sum of two convex sets $P$ and $Q$ is:

$$P + Q = \{p + q \mid p \in P, q \in Q\}$$




Volume of zonotopes (sums of segments) is used to test methods for order reduction which is important in several areas: autonomous driving, human-robot collaboration and smart grids

# Mixed volume

Let $P_1, P_2, \ldots, P_d$ be polytopes in $\mathbb{R}^d$ then the mixed volume is

$$M(P_1, \ldots, P_d) = \sum_{I \subseteq \{1,2,\ldots,d\}} (-1)^{(d-|I|)} \cdot \mathrm{Vol}(\sum_{i \in I} P_i)$$

where the sum is the Minkowski sum.

# Mixed volume

Let $P_1, P_2, \ldots, P_d$ be polytopes in $\mathbb{R}^d$ then the mixed volume is

$$M(P_1, \ldots, P_d) = \sum_{I \subseteq \{1,2,\ldots,d\}} (-1)^{(d-|I|)} \cdot \mathrm{Vol}(\sum_{i \in I} P_i)$$

where the sum is the Minkowski sum.

## Example

For $d = 2$: $M(P_1, P_2) = \mathrm{Vol}(P_1 + P_2) - \mathrm{Vol}(P_1) - \mathrm{Vol}(P_2)$