Linear arithmetic: Geometry, algorithms, and logic

Tuesday

Dmitry Chistikov Christoph Haase

Centre for Discrete Mathematics and its Applications (DIMAP) &
Department of Computer Science, University of Warwick, UK

Department of Computer Science
University of Oxford, UK

ESSLLI 2018

Linear programming

Given:

- Polyhedron in $\mathbb{R}^d$ defined by conjunction of linear constraints:

$$A \cdot \boldsymbol{x} \geq \boldsymbol{c}$$

- Linear objective function $\boldsymbol{b} \cdot \boldsymbol{x}$

Linear programming

Given:

- Polyhedron in $\mathbb{R}^d$ defined by conjunction of linear constraints:

$$A \cdot \boldsymbol{x} \geq \boldsymbol{c}$$

- Linear objective function $\boldsymbol{b} \cdot \boldsymbol{x}$

Goal:

- Maximize objective function while respecting all constraints

Linear programming

Given:

- Polyhedron in $\mathbb{R}^d$ defined by conjunction of linear constraints:

$$A \cdot \boldsymbol{x} \geq \boldsymbol{c}$$

- Linear objective function $\boldsymbol{b} \cdot \boldsymbol{x}$

Goal:

- Maximize objective function while respecting all constraints

Requirements:

- Decide feasibility
- Decide boundedness of objective function
- Understand where to find optima

Today's lecture: Linear programming (LP)

- Algorithm for LP: Fourier-Motzkin variable elimination
- Farkas' lemma and faces of polyhedra
- Algorithm for LP: Simplex algorithm
- Algorithm for LP: Ellipsoid method

Fourier-Motzkin variable elimination

Fourier-Motzkin variable elimination

- Simple!
- Decision procedure for feasibility
- Shows closure of polyhedra under projection
- Proves one direction of Minkowski-Weyl theorem
- Useful technical tool

Variable elimination

- Given quantifier-free formula $\Phi(x, \boldsymbol{y})$, find quantifier-free $\Psi(\boldsymbol{y})$ such that $\exists x \exists \boldsymbol{y} : \Phi(x, \boldsymbol{y})$ holds iff $\exists y : \Psi(\boldsymbol{y})$ holds
- Even more useful if solutions are preserved:

$$[\![\exists x : \Phi(x, \boldsymbol{y})]\!] = [\![\Psi(\boldsymbol{y})]\!]$$

Fourier-Motzkin variable elimination

Fourier-Motzkin eliminates a variable x while retaining all solutions of $\boldsymbol{y}$ from system of linear inequalties

$$\bigwedge_{1 \leq i \leq n} b_i \cdot x + \boldsymbol{a}_i \cdot \boldsymbol{y} \leq c_i$$

Fourier-Motzkin variable elimination

Fourier-Motzkin eliminates a variable x while retaining all solutions of $\boldsymbol{y}$ from system of linear inequalties

$$\bigwedge_{1 \leq i \leq n} b_i \cdot x + \boldsymbol{a}_i \cdot \boldsymbol{y} \leq c_i$$

- Group b_i according to sign:

$$P = \{i : b_i > 0\} \qquad N = \{i : b_i < 0\}$$

- Isolate x:

$$\bigwedge_{i \in N} \boldsymbol{a}_i \cdot \boldsymbol{y} - c_i \leq -b_i \cdot x \qquad \bigwedge_{i \in P} b_i \cdot x \leq -\boldsymbol{a}_i \cdot \boldsymbol{y} + c_i$$

- Normalize:

$$\bigwedge_{i \in N} -\frac{1}{b_i}(\boldsymbol{a}_i \cdot \boldsymbol{y} - c_i) \leq x \qquad \bigwedge_{i \in P} x \leq \frac{1}{b_i}(-\boldsymbol{a}_i \cdot \boldsymbol{y} + c_i)$$

- Take all combinations of lower and upper bounds

Fourier-Motzkin variable elimination

Remarks:

- Elimination of all variables yields conjunction of inequalties between rational numbers
- After every iteration obtain projected polyhedron
- Constant terms play “no role”

Application to Minkowski-Weyl

Theorem

For any $P \subseteq \mathbb{R}^d$, the following are equivalent:

1. $P = \{\boldsymbol{x} \colon A \cdot \boldsymbol{x} \geq \boldsymbol{c}\}$ for some matrix A and vector $\boldsymbol{c}$; and
2. $P = \mathrm{conv}(E) + \mathrm{cone}(F)$ for some finite sets E, F.

Application to Minkowski-Weyl

Theorem

For any $P \subseteq \mathbb{R}^d$, the following are equivalent:

1. $P = \{\boldsymbol{x} \colon A \cdot \boldsymbol{x} \geq \boldsymbol{c}\}$ for some matrix A and vector $\boldsymbol{c}$; and
2. $P = \mathrm{conv}(E) + \mathrm{cone}(F)$ for some finite sets E, F.

Proof of $2 \Rightarrow 1$.

Let $E = \{\boldsymbol{e}_1, \ldots, \boldsymbol{e}_m\}$ and $F = \{\boldsymbol{f}_1, \ldots, \boldsymbol{f}_n\}$. Then $\boldsymbol{x} \in P$ iff

$$\exists\gamma_1 \cdots \exists\gamma_m \exists\lambda_1 \cdots \exists\lambda_n : \boldsymbol{x} = \sum_{i=1}^{m} \gamma_i \cdot \boldsymbol{e}_i + \sum_{i=1}^{n} \lambda_i \cdot \boldsymbol{f}_i \wedge$$
$$\wedge \bigwedge_{i=1}^{m} \gamma_i \geq 0 \wedge \sum_{i=1}^{m} \gamma_i = 1 \wedge \bigwedge_{i=1}^{n} \lambda_i \geq 0.$$

Apply Fourier-Motzkin to eliminate γ_i and λ_i, resulting formula has desired properties. ∎

Farkas' lemma and faces of polyhedra

Farkas' lemma (1894)

Lemma (form 1)

$A \cdot \boldsymbol{x} \geq \boldsymbol{c}$ has no solution in $\mathbb{R}^d$ if and only if
there exists a $\boldsymbol{y}^\mathsf{T} \geq \mathbf{0}^\mathsf{T}$ such that $\boldsymbol{y}^\mathsf{T} \cdot A = \mathbf{0}^\mathsf{T}$ and $\boldsymbol{y}^\mathsf{T} \cdot \boldsymbol{c} > 0$.

Farkas' lemma (1894)

Lemma (form 1)

$A \cdot \boldsymbol{x} \geq \boldsymbol{c}$ has no solution in $\mathbb{R}^d$ if and only if
there exists a $\boldsymbol{y}^\intercal \geq \mathbf{0}^\intercal$ such that $\boldsymbol{y}^\intercal \cdot A = \mathbf{0}^\intercal$ and $\boldsymbol{y}^\intercal \cdot \boldsymbol{c} > 0$.

Remark

The choice of $\boldsymbol{y}^\intercal$ depends on A, but not on $\boldsymbol{c}$.

Farkas' lemma (1894)

Lemma (form 1)

$A \cdot \boldsymbol{x} \geq \boldsymbol{c}$ has no solution in $\mathbb{R}^d$ if and only if there exists a $\boldsymbol{y}^\mathsf{T} \geq \mathbf{0}^\mathsf{T}$ such that $\boldsymbol{y}^\mathsf{T} \cdot A = \mathbf{0}^\mathsf{T}$ and $\boldsymbol{y}^\mathsf{T} \cdot \boldsymbol{c} > 0$.

Remark

The choice of $\boldsymbol{y}^\mathsf{T}$ depends on A, but not on $\boldsymbol{c}$.

Lemma (form 2)

Either there exists an $\boldsymbol{x} \geq \mathbf{0}$ such that $A \cdot \boldsymbol{x} = \boldsymbol{c}$, or there exists a $\boldsymbol{z}^\mathsf{T} \geq \mathbf{0}^\mathsf{T}$ such that $\boldsymbol{z}^\mathsf{T} \cdot A \leq \mathbf{0}^\mathsf{T}$ and $\boldsymbol{z}^\mathsf{T} \cdot \boldsymbol{c} > 0$ (but not both).

Faces of polyhedra (recap)

Let P be a polyhedron.

Definition

$F \subseteq P$ is a face of P if

- either $F = P$,
- or $F = P \cap H$ for some valid hyperplane H.

Faces are sets of optimal solutions to **linear programs**.
Faces form a partial order with respect to set inclusion.

Characterization for faces

Theorem

Let $P = \{\boldsymbol{x} \colon A \cdot \boldsymbol{x} \geq \boldsymbol{c}\}$ be a polyhedron.
A non-empty subset $F \subseteq P$ is a face of P if and only if

$$F = P \cap \{\boldsymbol{x} \colon A' \cdot \boldsymbol{x} = \boldsymbol{c}'\},$$

that is, F is the set of solutions to a system of inequalities and equalities obtained from $A \cdot \boldsymbol{x} \geq \boldsymbol{c}$ by changing some of the inequalities to equalities.

Dimension of polyhedra (recap)

Definition
Let $P \subseteq \mathbb{R}^d$ be a polyhedron.
The dimension of P is the dimension of $\operatorname{aff}(P)$.

Faces of dimension 0, 1, and $\dim P - 1$ are called vertices, edges, and facets, respectively.

Characterization for faces

Theorem

Let $P = \{\boldsymbol{x} \colon A \cdot \boldsymbol{x} \geq \boldsymbol{c}\}$ be a polyhedron.
A non-empty subset $F \subseteq P$ is a face of P if and only if

$$F = P \cap \{\boldsymbol{x} \colon A' \cdot \boldsymbol{x} = \boldsymbol{c}'\},$$

that is, F is the set of solutions to a system of inequalities and equalities obtained from $A \cdot \boldsymbol{x} \geq \boldsymbol{c}$ by changing some of the inequalities to equalities.

Faces of polyhedra come from equalities

Let $P = \{\boldsymbol{x} \colon A \cdot \boldsymbol{x} \geq \boldsymbol{c}\}$ be a **pointed** polyhedron in $\mathbb{R}^d$.

- Each facet of P is defined by **implied equalities** and 1 additional inequality changed to equality
- Each vertex of P is defined by d equalities
- Each edge of P corresponds to a nonzero vector satisfying $d-1$ equalities

Solutions to systems of linear equations

Cramer's rule (1750):

Suppose $M \cdot \boldsymbol{x} = \boldsymbol{f}$ has a unique solution, where $M \in \mathbb{R}^{d \times d}$. Then

$$x_i = \frac{\det M}{\det M_i}$$

where M_i is M with ith column replaced by $\boldsymbol{f}$.

Representations of numbers, vectors, and matrices

Integer $n = \pm\big(a_k \cdot 2^k + \ldots + a_1 \cdot 2^1 + a_0\big) \in \mathbb{Z}$ with all $a_i \in \{0, 1\}$:

size (coding length) $\quad \langle n \rangle = 2 + \lceil \log_2(|n| + 1) \rceil$

Representations of numbers, vectors, and matrices

Integer $n = \pm\big(a_k \cdot 2^k + \ldots + a_1 \cdot 2^1 + a_0\big) \in \mathbb{Z}$ with all $a_i \in \{0, 1\}$:

size (coding length) $\quad \langle n \rangle = 2 + \lceil \log_2(|n| + 1) \rceil$

Rational $r = a/b \in \mathbb{Q}$ with $a \in \mathbb{Z}$, $b \in \mathbb{Z}_{>0}$, $\gcd(a, b) = 1$:

$$\langle r \rangle = \langle a \rangle + \langle b \rangle$$

Vector $\boldsymbol{v} \in \mathbb{Q}^m$:

$$\langle \boldsymbol{v} \rangle = \sum_{i=1}^{m} \langle v_i \rangle$$

Matrix $M = (m_{ij}) \in \mathbb{Q}^{k \times \ell}$:

$$\langle M \rangle = \sum_{i=1}^{k} \sum_{j=1}^{\ell} \langle m_{ij} \rangle$$

The Minkowski–Weyl theorem, revisited

Theorem

For any $P \subseteq \mathbb{R}^d$, the following are equivalent:

1. $P = \{\boldsymbol{x} \colon A \cdot \boldsymbol{x} \geq \boldsymbol{c}\}$ for some matrix A and vector $\boldsymbol{c}$; and
2. $P = \mathrm{conv}(E) + \mathrm{cone}(F)$ for some finite sets E, F.

The Minkowski–Weyl theorem, revisited

Theorem

For any $P \subseteq \mathbb{R}^d$, the following are equivalent:

1. $P = \{\boldsymbol{x} \colon A \cdot \boldsymbol{x} \geq \boldsymbol{c}\}$ for some matrix A and vector $\boldsymbol{c}$; and
2. $P = \mathrm{conv}(E) + \mathrm{cone}(F)$ for some finite sets E, F.

In both translations $1 \Rightarrow 2$ and $2 \Rightarrow 1$:

- The blowup in size can be exponential.
- The size of all numbers stays polynomial.

The Minkowski–Weyl theorem, revisited

Theorem

For any $P \subseteq \mathbb{R}^d$, the following are equivalent:

1. $P = \{\boldsymbol{x} \colon A \cdot \boldsymbol{x} \geq \boldsymbol{c}\}$ for some matrix A and vector $\boldsymbol{c}$; and
2. $P = \mathrm{conv}(E) + \mathrm{cone}(F)$ for some finite sets E, F.

In both translations $1 \Rightarrow 2$ and $2 \Rightarrow 1$:

- The blowup in size can be exponential.
- The size of all numbers stays polynomial.

The Minkowski–Weyl theorem, revisited

Theorem

For any $P \subseteq \mathbb{R}^d$, the following are equivalent:

1. $P = \{\boldsymbol{x} \colon A \cdot \boldsymbol{x} \geq \boldsymbol{c}\}$ for some matrix A and vector $\boldsymbol{c}$; and
2. $P = \mathrm{conv}(E) + \mathrm{cone}(F)$ for some finite sets E, F.

In both translations $1 \Rightarrow 2$ and $2 \Rightarrow 1$:

- The blowup in size can be exponential.
- The size of all numbers stays polynomial.

Simplex algorithm

Representations of linear programs

Conflicting goals:

- Want greatest level of flexibility for modeling problems
- Sometimes want (strong) structural properties on linear programs for algorithms

Representations of linear programs

Conflicting goals:

- Want greatest level of flexibility for modeling problems
- Sometimes want (strong) structural properties on linear programs for algorithms

Modeling:

$$\begin{aligned} \max_{\boldsymbol{x} \in \mathbb{R}^d} \; & \boldsymbol{c} \cdot \boldsymbol{x} \\ \text{s.t.} \; & A \cdot \boldsymbol{x} \geq \boldsymbol{b} \end{aligned}$$

Representations of linear programs

Conflicting goals:

- Want greatest level of flexibility for modeling problems
- Sometimes want (strong) structural properties on linear programs for algorithms

Modeling:

$$\begin{aligned} \underset{\boldsymbol{x}\in\mathbb{R}^d}{\text{maximize}}\, & \boldsymbol{c}\cdot\boldsymbol{x} \\ \text{s.t. } & A\cdot\boldsymbol{x}\geq\boldsymbol{b} \end{aligned}$$

Algorithms:

$$\begin{aligned} \underset{\boldsymbol{x}\in\mathbb{R}^d}{\text{minimize}}\, & \boldsymbol{c}\cdot\boldsymbol{x} \\ \text{s.t. } & A\cdot\boldsymbol{x}=\boldsymbol{b} \\ & \boldsymbol{x}\geq\boldsymbol{0} \end{aligned}$$

Solving linear programs

Simplex algorithm on a high level:

- Start at some initial vertex of **polytope**
- Move to neighbor vertex improving objective function if it exists
- Otherwise return current vertex

Solving linear programs

Simplex algorithm on a high level:

- Start at some initial vertex of **polytope**
- Move to neighbor vertex improving objective function if it exists
- Otherwise return current vertex

Historical remarks:

- Developed by George Dantzig in 1947
- One of the ten most important algorithms of the 20th century

Simplex in detail

Vertices:

- Point $\boldsymbol{v} \in \mathbb{R}^d$ is vertex of polytope only if it satisfies d defining inequality constraints with equality
- Point $\boldsymbol{w}$ is neighbor of $\boldsymbol{v}$ if $\boldsymbol{v}$ and $\boldsymbol{w}$ share $d - 1$ defining inequalities

Simplex is simple when starting at origin:

- If objective function is non-positive we are done
- Otherwise increase some variable with positive coefficient until a constraint becomes tight (pivot rule)

Example

$$\begin{aligned} \text{maximize } & x_1 + 6x_2 \\ \text{s.t. } & x_1 \leq 2 \\ & x_2 \leq 3 \\ & x_1 + x_2 \leq 4 \\ & x_1, x_2 \geq 0 \end{aligned}$$

Example

$$\begin{aligned} \text{maximize } & x_1 + 6x_2 \\ \text{s.t. } & x_1 \leq 2 \\ & x_2 \leq 3 \\ & x_1 + x_2 \leq 4 \\ & x_1 \geq 0 \\ & x_2 \geq 0 \end{aligned}$$

Example

$$
\begin{aligned}
\text{maximize } & x_1 + 6x_2 \\
\text{s.t. } & x_1 \le 2 \\
& x_2 \le 3 \\
& x_1 + x_2 \le 4 \\
& x_1 \ge 0 \\
& x_2 \ge 0
\end{aligned}
$$

Example

$$\begin{aligned} \text{maximize } & x_1 + 6x_2 \\ \text{s.t. } & x_1 \leq 2 \\ & x_2 \leq 3 \\ & x_1 + x_2 \leq 4 \\ & x_1 \geq 0 \\ & x_2 \geq 0 \end{aligned}$$

Increasing x_2 from 0 to 3:

- Increases objective function to 18
- Leads to new neighbor vertex $\boldsymbol{w} = (0, 3)$

Example

$$\begin{aligned} \text{maximize } & x_1 + 6x_2 \\ \text{s.t. } & x_1 \leq 2 \\ & x_2 \leq 3 \\ & x_1 + x_2 \leq 4 \\ & x_1 \geq 0 \\ & x_2 \geq 0 \end{aligned}$$

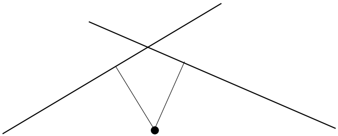
Increasing x_2 from 0 to 3:

- Increases objective function to 18
- Leads to new neighbor vertex $\boldsymbol{w} = (0, 3)$

Translating new vertex to origin

Turn new vertex into new origin:

- Any point inside polytope uniquely definable in terms of distances from defining hyperplanes



- Slack (distance) of point in polytope to i-th hyperplane $\boldsymbol{a}_i \cdot \boldsymbol{x} \leq b_i$ given by $y_i = b_i - \boldsymbol{a}_i \cdot \boldsymbol{x}$ and $y_i \geq 0$
- Yields d equations in d unknowns
- Express every x_i in terms of $y_1, \ldots, y_d$ and substitute

Example

$$
\begin{aligned}
\text{maximize } & x_1 + 6x_2 \\
\text{s.t. } & x_1 \le 2 \\
& x_2 \le 3 \\
& x_1 + x_2 \le 4 \\
& x_1 \ge 0 \\
& x_2 \ge 0
\end{aligned}
$$

Example

$$\begin{aligned} \text{maximize } & x_1 + 6x_2 \\ \text{s.t. } & x_1 \le 2 \\ & x_2 \le 3 \\ & x_1 + x_2 \le 4 \\ & x_1 \ge 0 \\ & x_2 \ge 0 \end{aligned}$$

$$\begin{aligned} y_1 &= x_1 & \leadsto && x_1 &= y_1 \\ y_2 &= 3 - x_2 & \leadsto && x_2 &= 3 - y_2 \end{aligned}$$

Example

$$\begin{aligned} \text{maximize } & x_1 + 6x_2 \\ \text{s.t. } & x_1 \le 2 \\ & x_2 \le 3 \\ & x_1 + x_2 \le 4 \\ & x_1 \ge 0 \\ & x_2 \ge 0 \end{aligned}$$

$$\begin{aligned} \text{maximize } & 18 + y_1 - 6y_2 \\ \text{s.t. } & y_1 \le 2 \\ & y_2 \ge 0 \\ & y_1 - y_2 \le 1 \\ & y_1 \ge 0 \\ & y_2 \le 3 \end{aligned}$$

$$y_1 = x_1 \quad \rightsquigarrow \quad x_1 = y_1$$

$$y_2 = 3 - x_2 \quad \rightsquigarrow \quad x_2 = 3 - y_2$$

Example

$$\begin{aligned} \text{maximize } & x_1 + 6x_2 \\ \text{s.t. } & x_1 \leq 2 \\ & {\color{red} x_2 \leq 3} \\ & x_1 + x_2 \leq 4 \\ & {\color{red} x_1 \geq 0} \\ & x_2 \geq 0 \end{aligned}$$

$$\begin{aligned} \text{maximize } & 18 + y_1 - 6y_2 \\ \text{s.t. } & y_1 \leq 2 \\ & y_2 \geq 0 \\ & y_1 - y_2 \leq 1 \\ & y_1 \geq 0 \\ & y_2 \leq 3 \\ & y_1 \geq 0 \\ & y_2 \geq 0 \end{aligned}$$

$$y_1 = x_1 \quad \rightsquigarrow \quad x_1 = y_1$$

$$y_2 = 3 - x_2 \quad \rightsquigarrow \quad x_2 = 3 - y_2$$

Finding a vertex to start from

Simplex can be used to find initial vertex. Given

$$\begin{aligned} \underset{\boldsymbol{x}\in\mathbb{R}^d}{\text{minimize}}\, & \boldsymbol{c}\cdot\boldsymbol{x} \\ \text{s.t. } & A\cdot\boldsymbol{x}=\boldsymbol{b} \\ & \boldsymbol{x}\geq\boldsymbol{0} \end{aligned}$$

Finding a vertex to start from

Simplex can be used to find initial vertex. Given

$$\begin{aligned} \underset{\boldsymbol{x} \in \mathbb{R}^d}{\text{minimize}}\, & \boldsymbol{c} \cdot \boldsymbol{x} \\ \text{s.t. } & A \cdot \boldsymbol{x} = \boldsymbol{b} \\ & \boldsymbol{x} \geq \boldsymbol{0} \end{aligned}$$

To find initial vertex:

- Augment i-th row of system with fresh variable z_i, $z_i \geq 0$
- Change objective function to $z_1 + \cdots + z_m$
- New system has initial vertex $z_i := b_i$, $x_j := 0$
- Run simplex, two possible outcomes
  - Solution with objective 0 $\rightsquigarrow$ gives vertex of original system
  - Otherwise original system infeasible

Ellipsoid method

Computational complexity: Decision problems and algorithms

Decision problem:
$L \subseteq \{0,1\}^*$ (encodings of yes-instances)

Algorithm for L:
says "yes" on every $x \in L$, "no" on every $x \in \{0,1\}^* \setminus L$

Time complexity

- of algorithm $\mathcal{A}$ on input x
- of algorithm $\mathcal{A}$ on inputs of length n (worst-case)
- of decision problem L

Complexity class $\mathbf{P}$ and efficient algorithms

Cobham–Edmonds thesis (1965):

Efficiently computable in a reasonable computational model
=
Computable in polynomial time on a Turing machine

$$\mathbf{P} = \bigcup_{d \geq 1} \bigcup_{c \geq 1} \mathrm{DTIME}(c \cdot n^d)$$

Ellipsoid method for LP (Khachiyan, 1979)

Ellipsoid method for LP (Khachiyan, 1979)

1. Reduce optimality to feasibility
2. Find a feasible solution if one exists

Ellipsoid method for LP (Khachiyan, 1979)

1. Reduce optimality to feasibility

2. Find a feasible solution if one exists

Insight: Polyhedra live in a discrete world!

Ellipsoid method for LP (Khachiyan, 1979)

1. Reduce optimality to feasibility
2. Find a feasible solution if one exists

Insight: Polyhedra live in a discrete world!

How to find a feasible solution?

1. Choose a $\rho > 0$ s.t. all vertices of P are in $E = \{\boldsymbol{x}^2 \leq \rho^2\}$.

2. `while(true):`

   Let z be the center of E. If z is feasible, stop.
   Otherwise find a violated constraint (use separation oracle).

   $E' \leftarrow$ smallest ellipsoid containing $E \cap \{\boldsymbol{x} : \boldsymbol{a}_i \cdot \boldsymbol{x} \geq c_i\}$,
   where $\boldsymbol{a}_i \cdot \boldsymbol{x} \geq c_i$ is the violated constraint.
   $E \leftarrow E'$.

   If $\mathrm{vol}(E) \leq$ magic number, stop with "infeasible".

Is this efficient?

Theorem

The number of iterations of the ellipsoid method is polynomial in n and s, the maximum size of numbers in the system $A \cdot \boldsymbol{x} \geq \boldsymbol{c}$.

Is this efficient?

Theorem

The number of iterations of the ellipsoid method is polynomial in n and s, the maximum size of numbers in the system $A \cdot \boldsymbol{x} \geq \boldsymbol{c}$.

Lemma 1: $\mathrm{vol}(E') \leq \mathrm{vol}(E) \cdot \left(1 - \frac{1}{\mathrm{poly}(n,s)}\right)$.

Lemma 2: If P is full-dimensional, then $\mathrm{vol}(P) \geq 2^{-\mathrm{poly}(n,s)}$.

Lemma 3: ρ can be chosen as $2^{\mathrm{poly}(n,s)}$.

Lemma 4: "Magic number" can be chosen as $2^{-\mathrm{poly}(n,s)}$.

Caveats

1. We assumed that $\mathrm{vol}(P) > 0$ if $P \neq \varnothing$.

2. We assumed unit-cost arithmetic.

Caveats

1. We assumed that $\mathrm{vol}(P) > 0$ if $P \neq \varnothing$.
2. We assumed unit-cost arithmetic.

Neither assumption is necessary.

Conclusion

Linear programming is in $\mathbf{P}$.

Summary of today's lecture

We have seen 3 algorithms for linear programming (LP):

- Fourier-Motzkin variable elimination
- Simplex algorithm
- Ellipsoid method

We have also characterized faces of polyhedra
using Farkas' lemma (a form of duality argument):

- Translations between $\{\boldsymbol{x} \colon A \cdot \boldsymbol{x} \geq \boldsymbol{c}\}$ and $\mathrm{conv}(E) + \mathrm{cone}(F)$ keep size of numbers small

Agenda

Tuesday	Linear programming
Wednesday	Integer programming
Thursday	Decision procedures for arithmetic theories
Friday	Expressive power of arithmetic theories