

算法设计与分析(2025年春季学期)

Linear Programming

授课老师: 栗师

南京大学计算机学院

Outline

1 Linear Programming

- Introduction
- Preliminaries
- Methods for Solving Linear Programs

2 Linear Programming Duality

3 Integral Polytopes: Exact Algorithms Using LP

- Bipartite Matching Polytope
- s - t Flow Polytope
- Weighted Interval Scheduling Problem and Totally Unimodular Matrices

Outline

1 Linear Programming

- Introduction
- Preliminaries
- Methods for Solving Linear Programs

2 Linear Programming Duality

3 Integral Polytopes: Exact Algorithms Using LP

- Bipartite Matching Polytope
- s - t Flow Polytope
- Weighted Interval Scheduling Problem and Totally Unimodular Matrices

Example of Linear Programming

$$\min \quad 7x_1 + 4x_2$$

$$x_1 + x_2 \geq 5$$

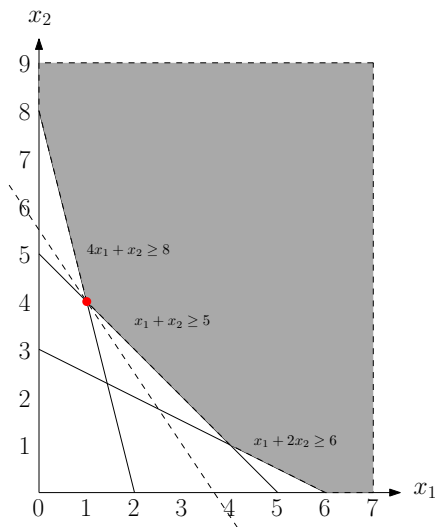
$$x_1 + 2x_2 \geq 6$$

$$4x_1 + x_2 \geq 8$$

$$x_1, x_2 \geq 0$$

- optimum point: $x_1 = 1, x_2 = 4$

- value = $7 \times 1 + 4 \times 4 = 23$



Standard Form of Linear Programming

$$\min \quad c_1x_1 + c_2x_2 + \cdots + c_nx_n \quad \text{s.t.}$$

$$\sum A_{1,1}x_1 + A_{1,2}x_2 + \cdots + A_{1,n}x_n \geq b_1$$

$$\sum A_{2,1}x_1 + A_{2,2}x_2 + \cdots + A_{2,n}x_n \geq b_2$$

$$\vdots \quad \vdots \quad \vdots \quad \vdots$$

$$\sum A_{m,1}x_1 + A_{m,2}x_2 + \cdots + A_{m,n}x_n \geq b_m$$

$$x_1, x_2, \cdots, x_n \geq 0$$

Standard Form of Linear Programming

$$\text{Let } x = \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix}, \quad c = \begin{pmatrix} c_1 \\ c_2 \\ \vdots \\ c_n \end{pmatrix},$$
$$A = \begin{pmatrix} A_{1,1} & A_{1,2} & \cdots & A_{1,n} \\ A_{2,1} & A_{2,2} & \cdots & A_{2,n} \\ \vdots & \vdots & \vdots & \vdots \\ A_{m,1} & A_{m,2} & \cdots & A_{m,n} \end{pmatrix}, \quad b = \begin{pmatrix} b_1 \\ b_2 \\ \vdots \\ b_m \end{pmatrix}.$$

Then, LP becomes $\min c^T x$ s.t.

$$Ax \geq b$$
$$x \geq 0$$

- \geq means coordinate-wise greater than or equal to

Standard Form of Linear Programming

$$\min \quad c^T x \quad \text{s.t.}$$

$$Ax \geq b$$

$$x \geq 0$$

- Linear programmings can be solved in polynomial time

Algorithm	Theory	Practice
Simplex Method	Exponential Time	Works Well
Ellipsoid Method	Polynomial Time	Slow
Internal Point Methods	Polynomial Time	Works Well

History

- [Fourier, 1827]: Fourier-Motzkin elimination method
- [Kantorovich, Koopmans 1939]: formulated the general linear programming problem
- [Dantzig 1946]: simplex method
- [Khachiyan 1979]: ellipsoid method, polynomial time, proved linear programming is in P
- [Karmarkar, 1984]: interior-point method, polynomial time, algorithm is practical

Outline

1 Linear Programming

- Introduction
- Preliminaries
- Methods for Solving Linear Programs

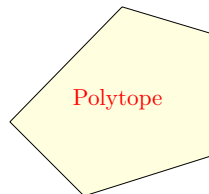
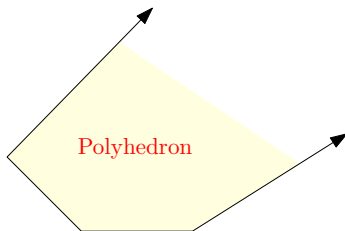
2 Linear Programming Duality

3 Integral Polytopes: Exact Algorithms Using LP

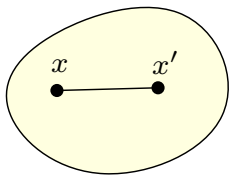
- Bipartite Matching Polytope
- s - t Flow Polytope
- Weighted Interval Scheduling Problem and Totally Unimodular Matrices

Preliminaries

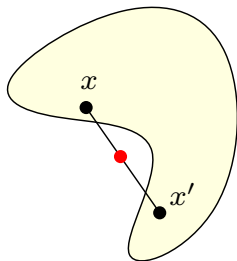
- **feasible region**: the set of x 's satisfying $Ax \geq b, x \geq 0$
- feasible region is a **polyhedron**
- if every coordinate has an upper and lower bound in the polyhedron, then the polyhedron is a **polytope**



Def. A set of points $P \subseteq \mathbb{R}^n$ is said to be convex if for every $x, x' \in P$ and two reals $\alpha, \beta \in [0, 1]$ with $\alpha + \beta = 1$, we have $\alpha x + \beta x' \in P$.



convex



not convex

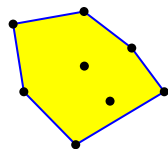
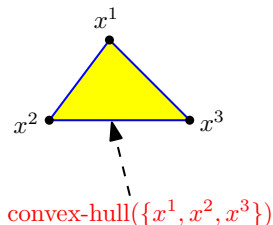
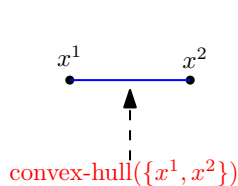
Obs. A polyhedron is convex.

Preliminaries

- We say x is a **convex combination** of $x^{(1)}, x^{(2)}, \dots, x^{(t)}$ if the following condition holds: there exist $\lambda_1, \lambda_2, \dots, \lambda_t \in [0, 1]$ such that

$$\lambda_1 + \lambda_2 + \dots + \lambda_t = 1, \quad \lambda_1 x^{(1)} + \lambda_2 x^{(2)} + \dots + \lambda_t x^{(t)} = x$$

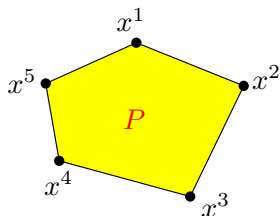
- The set of convex combinations of $x^{(1)}, x^{(2)}, \dots, x^{(t)}$ is called the **convex hull** of these points



Preliminaries

- let P be polytope, $x \in P$. If there are no other points $x', x'' \in P$ such that x is a convex combination of x' and x'' , then x is called a **vertex/extreme point** of P

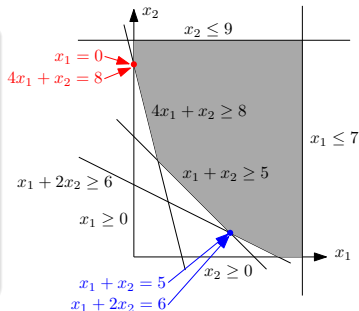
Lemma A polytope has finite number of vertices, and it is the convex hull of the vertices.



$$P = \text{convex-hull}(\{x^1, x^2, x^3, x^4, x^5\})$$

Preliminaries

Lemma Let $x \in \mathbb{R}^n$ be an extreme point in a n -dimensional polytope. Then, there are n constraints in the definition of the polytope, such that x is the unique solution to the linear system obtained from the n constraints by replacing inequalities to equalities.



Lemma If the feasible region of a linear program is a polytope, then the optimum value can be attained at some vertex of the polytope.

Special cases (for minimization linear programs):

- if feasible region is empty, then its value is ∞
- if the feasible region is unbounded, then its value can be $-\infty$

Outline

1 Linear Programming

- Introduction
- Preliminaries
- **Methods for Solving Linear Programs**

2 Linear Programming Duality

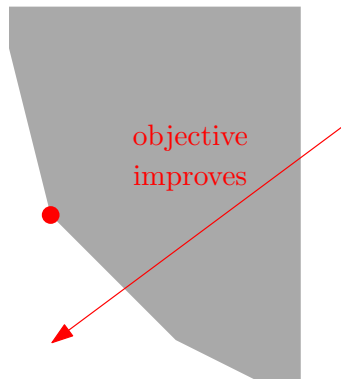
3 Integral Polytopes: Exact Algorithms Using LP

- Bipartite Matching Polytope
- s - t Flow Polytope
- Weighted Interval Scheduling Problem and Totally Unimodular Matrices

Simplex Method

- [Dantzig, 1946]

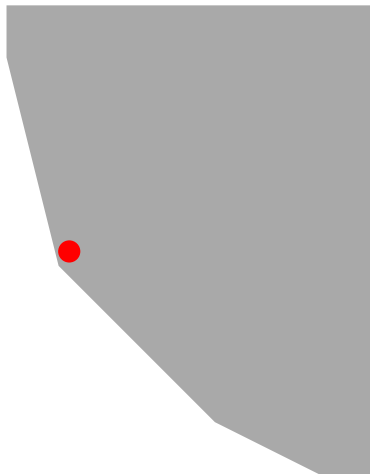
- move from one vertex to another, so as to improve the objective
- repeat until we reach an optimum vertex



- the number of iterations might be exponentially large; but algorithm runs fast in practice
- [Spielman-Teng,2002]: smoothed analysis

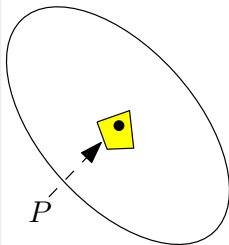
Interior Point Method

- [Karmarkar, 1984]
 - keep the solution inside the polytope
 - design penalty function so that the solution is not too close to the boundary
 - the final solution will be arbitrarily close to the optimum solution
-
- polynomial time



Ellipsoid Method

- [Khachiyan, 1979]
 - used to decide if the feasible region is empty or not
- maintain an ellipsoid that contains the feasible region
 - query a **separation oracle** if the center of ellipsoid is in the feasible region:
 - yes: then the feasible region is not empty
 - no: cut the ellipsoid in half, find smaller ellipsoid to enclose the half-ellipsoid, and repeat
- polynomial time, but impractical



Q: The exact running time of these algorithms?

- it depends on many parameters: #variables, #constraints, #(non-zero coefficients), magnitude of integers
- precision issue

Open Problem

Can linear programming be solved in strongly polynomial time algorithm?

Applications of Linear Programming

- domain: computer science, mathematics, operations research, economics
- types of problems: transportation, scheduling, clustering, network routing, resource allocation, facility location

Research Directions

- polynomial time exact algorithm
- polynomial time approximation algorithm
- sub-routines for the branch-and-bound method for integer programming
- other algorithmic models: online algorithm, distributed algorithms, dynamic algorithms, fast algorithms

Outline

1 Linear Programming

- Introduction
- Preliminaries
- Methods for Solving Linear Programs

2 Linear Programming Duality

3 Integral Polytopes: Exact Algorithms Using LP

- Bipartite Matching Polytope
- s - t Flow Polytope
- Weighted Interval Scheduling Problem and Totally Unimodular Matrices

$$\min \quad 7x_1 + 4x_2$$

$$x_1 + x_2 \geq 5$$

$$x_1 + 2x_2 \geq 6$$

$$4x_1 + x_2 \geq 8$$

$$x_1, x_2 \geq 0$$

- optimum point: $x_1 = 1, x_2 = 4$
- value = $7 \times 1 + 4 \times 4 = 23$

Q: How can we prove a lower bound for the value?

- $7x_1 + 4x_2 \geq 2(x_1 + x_2) + (x_1 + 2x_2) \geq 2 \times 5 + 6 = 16$
- $7x_1 + 4x_2 \geq (x_1 + 2x_2) + 1.5(4x_1 + x_2) \geq 6 + 1.5 \times 8 = 18$
- $7x_1 + 4x_2 \geq (x_1 + x_2) + (x_1 + 2x_2) + (4x_1 + x_2) \geq 5 + 6 + 8 = 19$
- $7x_1 + 4x_2 \geq 4(x_1 + x_2) \geq 4 \times 5 = 20$
- $7x_1 + 4x_2 \geq 3(x_1 + x_2) + (4x_1 + x_2) \geq 3 \times 5 + 8 = 23$

Primal LP

$$\begin{aligned} \min \quad & 7x_1 + 4x_2 \\ & x_1 + x_2 \geq 5 \\ & x_1 + 2x_2 \geq 6 \\ & 4x_1 + x_2 \geq 8 \\ & x_1, x_2 \geq 0 \end{aligned}$$

Dual LP

$$\begin{aligned} \max \quad & 5y_1 + 6y_2 + 8y_3 \quad \text{s.t.} \\ & y_1 + y_2 + 4y_3 \leq 7 \\ & y_1 + 2y_2 + y_3 \leq 4 \\ & y_1, y_2 \geq 0 \end{aligned}$$

A way to prove lower bound on the value of primal LP

$$\begin{aligned} & 7x_1 + 4x_2 \quad (\text{if } 7 \geq y_1 + y_2 + 4y_3 \text{ and } 4 \geq y_1 + 2y_2 + y_3) \\ & \geq y_1(x_1 + x_2) + y_2(x_1 + 2x_2) + y_3(4x_1 + x_2) \quad (\text{if } y_1, y_2, y_3 \geq 0) \\ & \geq 5y_1 + 6y_2 + 8y_3. \end{aligned}$$

- Goal: need to maximize $5y_1 + 6y_2 + 8y_3$

Primal LP

$$\begin{aligned} \min \quad & 7x_1 + 4x_2 \\ & x_1 + x_2 \geq 5 \\ & x_1 + 2x_2 \geq 6 \\ & 4x_1 + x_2 \geq 8 \\ & x_1, x_2 \geq 0 \end{aligned}$$

Dual LP

$$\begin{aligned} \max \quad & 5y_1 + 6y_2 + 8y_3 \quad \text{s.t.} \\ & y_1 + y_2 + 4y_3 \leq 7 \\ & y_1 + 2y_2 + y_3 \leq 4 \\ & y_1, y_2 \geq 0 \end{aligned}$$

$$A = \begin{pmatrix} 1 & 1 \\ 1 & 2 \\ 4 & 1 \end{pmatrix} \quad b = \begin{pmatrix} 5 \\ 6 \\ 8 \end{pmatrix} \quad c = \begin{pmatrix} 7 \\ 4 \end{pmatrix}$$

$$\min \quad c^T x \quad \text{s.t.}$$

$$Ax \geq b$$

$$x \geq 0$$

$$\max \quad b^T y \quad \text{s.t.}$$

$$A^T y \leq c$$

$$y \geq 0$$

Primal LP

$$\min \quad c^T x \quad \text{s.t.}$$

$$Ax \geq b$$

$$x \geq 0$$

Dual LP

$$\max \quad b^T y \quad \text{s.t.}$$

$$A^T y \leq c$$

$$y \geq 0$$

- P = value of primal LP
- D = value of dual LP

Theorem (weak duality theorem) $D \leq P$.

Theorem (strong duality theorem) $D = P$.

- Can always prove the optimality of the primal solution, by adding up primal constraints.

Primal LP

$$\min \quad c^T x \quad \text{s.t.}$$

$$Ax \geq b$$

$$x \geq 0$$

Dual LP

$$\max \quad b^T y \quad \text{s.t.}$$

$$A^T y \leq c$$

$$y \geq 0$$

- P = value of primal LP
- D = value of dual LP

Theorem (weak duality theorem) $D \leq P$.

Proof.

- x^* : optimal primal solution
- y^* : optimal dual solution

$$D = b^T y^* \leq (Ax^*)^T y^* = (x^*)^T A^T y^* \leq (x^*)^T c = c^T x^* = P. \quad \square$$

Proof of Strong Duality Theorem

Lemma (Variant of Farkas Lemma) $Ax \leq b, x \geq 0$ is infeasible, if and only if $y^T A \geq 0, y^T b < 0, y \geq 0$ is feasible.

- $\forall \epsilon > 0, \begin{pmatrix} -A \\ c^T \end{pmatrix} x \leq \begin{pmatrix} -b \\ P - \epsilon \end{pmatrix}, x \geq 0$ is infeasible
- There exists $y \in \mathbb{R}_{\geq 0}^m, \alpha \geq 0$, such that $(y^T, \alpha) \begin{pmatrix} -A \\ c^T \end{pmatrix} \geq 0$,
 $(y^T, \alpha) \begin{pmatrix} -b \\ P - \epsilon \end{pmatrix} < 0$
- we can prove $\alpha > 0$; assume $\alpha = 1$
- $-y^T A + c^T \geq 0, -y^T b + P - \epsilon < 0 \iff A^T y \leq c, b^T y > P - \epsilon$
- $\forall \epsilon > 0, D > P - \epsilon \implies D = P$ (since $D \leq P$) □

Example

Primal LP

$$\min \quad 5x_1 + 6x_2 + x_3 \quad \text{s.t.}$$

$$2x_1 + 5x_2 - 3x_3 \geq 2$$

$$3x_1 - 2x_2 + x_3 \geq 5$$

$$x_1 + 2x_2 + 3x_3 \geq 7$$

$$x_1, x_2, x_3 \geq 0$$

Dual LP

$$\max \quad 2y_1 + 5y_2 + 7y_3 \quad \text{s.t.}$$

$$2y_1 + 3y_2 + y_3 \leq 5$$

$$5y_1 - 2y_2 + 2y_3 \leq 6$$

$$-3y_1 + y_2 + 3y_3 \geq 1$$

$$y_1, y_2, y_3 \geq 0$$

Primal Solution

$$x_1 = 1.6, x_2 = 0.6$$

$$x_3 = 1.4, \text{value} = 13$$

Dual Solution

$$y_1 = 1, y_2 = 5/8$$

$$y_3 = 9/8, \text{value} = 13$$

$$\begin{aligned} & 5x_1 + 6x_2 + x_3 \\ \geq & (2x_1 + 5x_2 - 3x_3) + \frac{5}{8}(3x_1 - 2x_2 + x_3) + \frac{9}{8}(x_1 + 2x_2 + 3x_3) \\ \geq & 2 + \frac{5}{8} \times 5 + \frac{9}{8} \times 7 \\ = & 13 \end{aligned}$$

Outline

1 Linear Programming

- Introduction
- Preliminaries
- Methods for Solving Linear Programs

2 Linear Programming Duality

3 Integral Polytopes: Exact Algorithms Using LP

- Bipartite Matching Polytope
- s - t Flow Polytope
- Weighted Interval Scheduling Problem and Totally Unimodular Matrices

Def. A polytope $P \subseteq \mathbb{R}^n$ is said to be **integral**, if all vertices of P are in \mathbb{Z}^n .

- For some combinatorial optimization problems, a polynomial-sized LP $Ax \leq b$ already defines an integral polytope, whose vertices correspond to valid integral solutions.
- Such a problem can be solved directly using the LP:

$$\max / \min \quad c^T x \quad Ax \leq b.$$

Outline

1 Linear Programming

- Introduction
- Preliminaries
- Methods for Solving Linear Programs

2 Linear Programming Duality

3 Integral Polytopes: Exact Algorithms Using LP

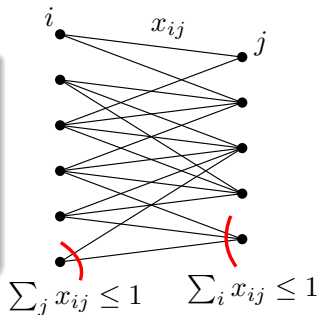
- Bipartite Matching Polytope
- s - t Flow Polytope
- Weighted Interval Scheduling Problem and Totally Unimodular Matrices

Maximum Weight Bipartite Matching

Input: bipartite graph $G = (L \uplus R, E)$

edge weights $w \in \mathbb{Z}_{>0}^E$

Output: a matching $M \subseteq E$ so as to maximize $\sum_{e \in M} w_e$



LP Relaxation

$$\max \sum_{e \in E} w_e x_e$$

$$\sum_{e \in \delta(v)} x_e \leq 1 \quad \forall v \in L \cup R$$

$$x_e \geq 0 \quad \forall e \in E$$

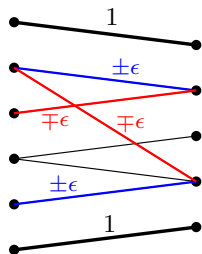
- In IP: $x_e \in \{0, 1\}$: $e \in M$?
- $\chi^M \in \{0, 1\}^E$: $\chi_e^M = 1$ iff $e \in M$

Theorem The LP polytope is integral: It is the convex hull of $\{\chi^M : M \text{ is a matching}\}$.

Theorem The LP polytope is integral: It is the convex hull of $\{\chi^M : M \text{ is a matching}\}$.

Proof.

- take x in the polytope P
- prove: x non integral $\implies x$ non-vertex
- find $x', x'' \in P: x' \neq x'', x = \frac{1}{2}(x' + x'')$
- case 1: fractional edges contain a cycle
 - color edges in cycle blue and red
 - x' : $+\epsilon$ for blue edges, $-\epsilon$ for red edges
 - x'' : $-\epsilon$ for blue edges, $+\epsilon$ for red edges
- case 2: fractional edges form a forest
 - color edges in a leaf-leaf path blue and red
 - x' : $+\epsilon$ for blue edges, $-\epsilon$ for red edges
 - x'' : $-\epsilon$ for blue edges, $+\epsilon$ for red edges \square



Outline

1 Linear Programming

- Introduction
- Preliminaries
- Methods for Solving Linear Programs

2 Linear Programming Duality

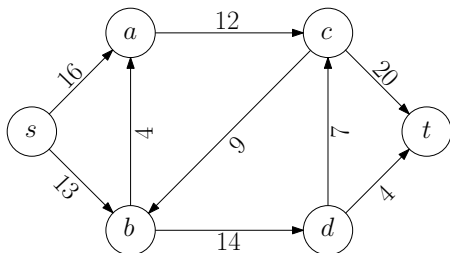
3 Integral Polytopes: Exact Algorithms Using LP

- Bipartite Matching Polytope
- s - t Flow Polytope
- Weighted Interval Scheduling Problem and Totally Unimodular Matrices

Example: s - t Flow Polytope

Flow Network

- directed graph $G = (V, E)$, **source** $s \in V$, **sink** $t \in V$, edge capacities $c_e \in \mathbb{Z}_{>0}, \forall e \in E$
- s has no incoming edges, t has no outgoing edges



Def. A s - t flow is a vector $f \in \mathbb{R}_{\geq 0}^E$ satisfying the following conditions:

- $\forall e \in E, 0 \leq f(e) \leq c_e$ (capacity constraints)
- $\forall v \in V \setminus \{s, t\},$

$$\sum_{e \in \delta^{\text{in}}(v)} f(e) = \sum_{e \in \delta^{\text{out}}(v)} f(e) \quad (\text{flow conservation})$$

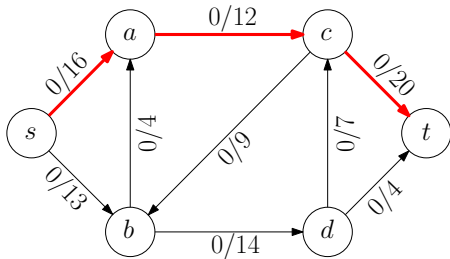
The value of flow f is defined as:

$$\text{val}(f) := \sum_{e \in \delta^{\text{out}}(s)} f(e) = \sum_{e \in \delta^{\text{in}}(t)} f(e)$$

Maximum Flow Problem

Input: flow network $(G = (V, E), c, s, t)$

Output: maximum value of a s - t flow f



- Ford-Fulkerson method
- **Maximum-Flow Min-Cut Theorem:** value of the maximum flow is equal to the value of the minimum s - t cut

LP for Maximum Flow

$$\begin{aligned} \max \quad & \sum_{e \in \delta_{\text{in}}(t)} x_e \\ & x_e \leq c_e \quad \forall e \in E \\ \sum_{e \in \delta_{\text{out}}(v)} x_e - \sum_{e \in \delta_{\text{in}}(v)} x_e &= 0 \quad \forall v \in V \setminus \{s, t\} \\ & x_e \geq 0 \quad \forall e \in E \end{aligned}$$

Theorem The LP polytope is integral.

Sketch of Proof.

- Take any s - t flow x ; consider fractional edges E'
- Every $v \notin \{s, t\}$ must be incident to 0 or ≥ 2 edges in E'
- Ignoring the directions of E' , it contains a cycle, or a s - t path
- We can increase/decrease flow values along cycle/path



Outline

1 Linear Programming

- Introduction
- Preliminaries
- Methods for Solving Linear Programs

2 Linear Programming Duality

3 Integral Polytopes: Exact Algorithms Using LP

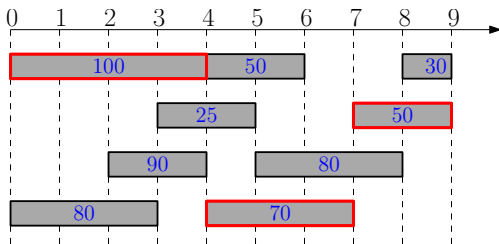
- Bipartite Matching Polytope
- s - t Flow Polytope
- Weighted Interval Scheduling Problem and Totally Unimodular Matrices

Weighted Interval Scheduling Problem

Input: n activities, activity i starts at time s_i , finishes at time f_i , and has weight $w_i > 0$

i and j can be scheduled together iff $[s_i, f_i)$ and $[s_j, f_j)$ are disjoint

Output: maximum weight subset of jobs that can be scheduled



- optimum value= 220
- Classic Problem for Dynamic Programming

Weighted Interval Scheduling Problem

Linear Program

$$\begin{aligned} \max \quad & \sum_{j \in [n]} x_j w_j \\ \sum_{j \in [n]: t \in [s_j, f_j]} x_j & \leq 1 \quad \forall t \in [T] \\ x_j & \geq 0 \quad \forall j \in [n] \end{aligned}$$

Theorem The LP polytope is integral.

Def. A matrix $A \in \mathbb{R}^{m \times n}$ is said to be **totally unimodular (TUM)**, if every sub-square of A has determinant in $\{-1, 0, 1\}$.

Theorem If a polytope P is defined by $Ax \geq b, x \geq 0$ with a totally unimodular matrix A and integral b , then P is integral.

Lemma A matrix $A \in \{0, 1\}^{m \times n}$ where the 1's on every column form an interval is TUM.

- So, the matrix for the LP is TUM, and the polytope is integral.

Theorem If a polytope P is defined by $Ax \geq b, x \geq 0$ with a totally unimodular matrix A and integral b , then P is integral.

Proof.

- Every vertex $x \in P$ is the unique solution to the linear system (after permuting coordinates): $\begin{pmatrix} A' & 0 \\ 0 & I \end{pmatrix} x = \begin{pmatrix} b' \\ 0 \end{pmatrix}$, where
 - A' is a square submatrix of A with $\det(A') = \pm 1$, b' is a sub-vector of b ,
 - and the rows for b' are the same as the rows for A' .
- Let $x = \begin{pmatrix} x^1 \\ x^2 \end{pmatrix}$, so that $A'x^1 = b'$ and $x^2 = 0$.
- Cramer's rule: $x_i^1 = \frac{\det(A'_i|b)}{\det(A')}$ for every $i \implies x_i^1$ is integer
 $A'_i|b$: the matrix of A' with the i -th column replaced by b □

Example for the Proof

$$\begin{pmatrix} a_{1,1} & a_{1,2} & a_{1,3} & a_{1,4} & a_{1,5} \\ a_{2,1} & a_{2,2} & a_{2,3} & a_{2,4} & a_{2,5} \\ a_{3,1} & a_{3,2} & a_{3,3} & a_{3,4} & a_{3,5} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \end{pmatrix} \geq \begin{pmatrix} b_1 \\ b_2 \\ b_3 \end{pmatrix}$$
$$x_1, x_2, x_3, x_4, x_5 \geq 0$$

The following equation system may give a vertex:

$$\begin{pmatrix} a_{1,1} & a_{1,2} & a_{1,3} & a_{1,4} & a_{1,5} \\ a_{3,1} & a_{3,2} & a_{3,3} & a_{3,4} & a_{3,5} \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \end{pmatrix} = \begin{pmatrix} b_1 \\ b_3 \\ 0 \\ 0 \\ 0 \end{pmatrix}$$

Example for the Proof

$$\begin{pmatrix} a_{1,1} & a_{1,2} & a_{1,3} & a_{1,4} & a_{1,5} \\ a_{3,1} & a_{3,2} & a_{3,3} & a_{3,4} & a_{3,5} \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \end{pmatrix} = \begin{pmatrix} b_1 \\ b_3 \\ 0 \\ 0 \\ 0 \end{pmatrix}$$

Equivalently, the vertex satisfies

$$\begin{pmatrix} a_{1,2} & a_{1,3} & 0 & 0 & 0 \\ a_{3,2} & a_{3,3} & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x_2 \\ x_3 \\ x_1 \\ x_4 \\ x_5 \end{pmatrix} = \begin{pmatrix} b_1 \\ b_3 \\ 0 \\ 0 \\ 0 \end{pmatrix}$$

Lemma Let $A' \in \{0, \pm 1\}^{n \times n}$ such that every row of A' contains at most one 1 and one -1 . Then $\det(A') \in \{0, \pm 1\}$.

Proof.

- wlog assume every row of A' contains one 1 and one -1
 - otherwise, we can reduce the matrix
- treat A' as a directed graph: columns \equiv vertices, rows \equiv arcs
- $\#edges = \#vertices \implies$ underlying undirected graph contains a cycle $\implies \det(A') = 0$ □

Lemma Let $A \in \{0, \pm 1\}^{m \times n}$ such that every row of A contains at most one 1 and one -1 . Then A is TUM.

Coro. The matrix for s - t flow polytope is TUM; thus, the polytope is integral.

Example for the Proof

$$\begin{pmatrix} 1 & -1 & 0 & 0 & 0 & 0 & 0 \\ 0 & -1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & -1 & 0 & 0 \\ 0 & 0 & -1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & -1 & 0 & 1 \\ 1 & 0 & 0 & 0 & -1 & 0 & 0 \end{pmatrix}$$

$$\begin{pmatrix} 1 & -1 & 0 & 0 & 2 & 0 & 0 \\ 0 & -1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & -1 & 0 & 1 & 0 \\ 1 & 0 & 0 & -1 & 0 & 0 & 0 \end{pmatrix}$$

$$\begin{pmatrix} 1 & -1 & 0 & 0 & 0 & 0 & 0 \\ 0 & -1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1 & 0 & 1 \\ 1 & 0 & 0 & -1 & 0 & 0 & 0 \end{pmatrix}$$

$$\begin{aligned} &+ \begin{pmatrix} 1 & -1 & 0 & 0 & 0 \\ 0 & -1 & 1 & 0 & 0 \\ 0 & 0 & 1 & -1 & 0 \\ 1 & 0 & 0 & -1 & 0 \end{pmatrix} \\ &- \begin{pmatrix} 0 & -1 & 1 & 0 & 0 \\ 0 & 0 & 1 & -1 & 0 \\ 1 & 0 & 0 & -1 & 0 \end{pmatrix} \\ &= \begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix} \end{aligned}$$

Lemma A matrix $A \in \{0, 1\}^{m \times n}$ where the 1's on every row form an interval is TUM.

Proof.

- take any square submatrix A' of A ,
- the 1's on every row of A' form an interval.
- $A'M$ is a matrix satisfying condition of first lemma, where

$$M = \begin{pmatrix} 1 & 0 & 0 & \cdots & 0 \\ -1 & 1 & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \cdots & 1 & 0 \\ 0 & 0 & \cdots & -1 & 1 \end{pmatrix}. \det(M) = 1.$$

- $\det(A'M) \in \{0, \pm 1\} \implies \det(A') \in \{0, \pm 1\}$. □

Example for the Proof

$$\begin{pmatrix} 0 & 1 & 1 & 1 & 0 & 0 \\ 1 & 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 1 & 1 & 1 & 1 & 0 \end{pmatrix} \begin{pmatrix} 0 & 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 \\ 0 & 1 & 1 & 1 & 1 \end{pmatrix} \implies \begin{pmatrix} 0 & 1 & 0 & 0 & -1 \\ 1 & 0 & 0 & 0 & -1 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 \end{pmatrix}$$

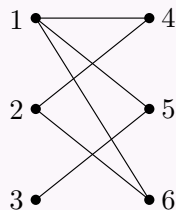
- (col 1, col 2 - col 1, col 3 - col 2, col 4 - col 3, col 5 - col 4)
- every row has at most one 1, at most one -1

Lemma The edge-vertex incidence matrix A of a bipartite graph is totally-unimodular.

Proof.

- $G = (L \uplus R, E)$: the bipartite graph
- A' : obtained from A by negating columns correspondent to R
- each row of A' has exactly one $+1$, and exactly one -1
- $\implies A'$ is TUM $\iff A$ is TUM □

Example



$$\begin{pmatrix} 1 & 0 & 0 & -1 & 0 & 0 \\ 1 & 0 & 0 & 0 & -1 & 0 \\ 1 & 0 & 0 & 0 & 0 & -1 \\ 0 & 1 & 0 & -1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & -1 \\ 0 & 0 & 1 & 0 & -1 & 0 \end{pmatrix}$$

- remark: bipartiteness is needed. The edge-vertex incidence matrix $\begin{pmatrix} 0 & 1 & 1 \\ 1 & 0 & 1 \\ 1 & 1 & 0 \end{pmatrix}$ of a triangle has determinant 2.

Coro. Bipartite matching polytope is integral.

In summary, given a matrix $A \in \{-1, 0, 1\}^{m \times n}$, A is TUM if one of the conditions hold:

- every row of A has at most one 1 and at most one -1
(network flow polytope)
- $A \in \{0, 1\}^{m \times n}$, and the 1's in every row form an interval
(interval scheduling polytope)
- A is edge-vertex incidence matrix of a bipartite graph
(bipartite matching polytope)

- $G = (L \uplus R, E)$: bipartite graph
- $\text{MM}(G)$: the size of the maximum matching of G
- $\text{MVC}(G)$: the size of the minimum vertex cover of G
- Using MFMC theorem, we know $\text{MM}(G) = \text{MVC}(G)$
- A new proof using LP duality:

LP for MM

$$\max \quad \sum_{e \in E} x_e$$

$$\sum_{e \in \delta(v)} x_e \leq 1 \quad \forall v \in L \uplus R$$

$$x_e \geq 0 \quad \forall e \in E$$

LP for MVC

$$\min \quad \sum_{v \in L \uplus R} y_v$$

$$y_u + y_v \geq 1 \quad \forall (u, v) \in E$$

$$y_u \geq 0 \quad \forall u \in L \uplus R$$

- Both LP polytopes are integral
- $\text{MM}(G) = \text{primal value} = \text{dual value} = \text{MVC}(G)$