Randomized Algorithms

Randomized Rounding

- **Brief Introduction to Linear Programming and Its Usage in Combinatorial Optimization**
- Randomized Rounding for Cut Problems
- Randomized Rounding for Satisfiability Problems
- Randomized Rounding for Covering Problems
- Randomized Rounding and Semi-definite Programming

Approximate Sampling and Counting

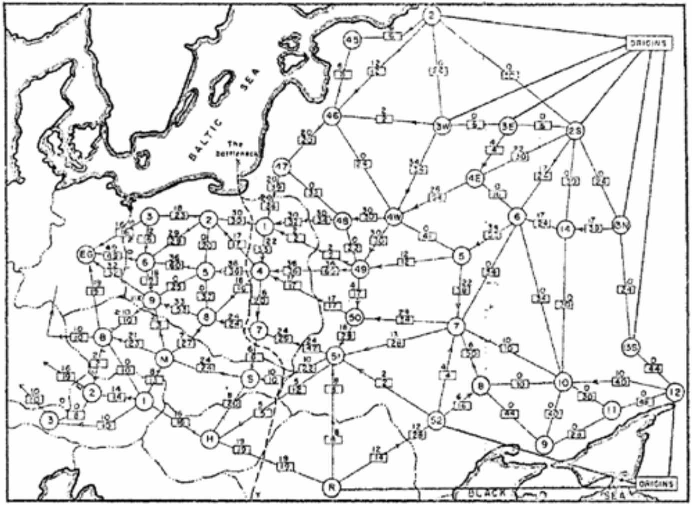
- ...

Some Combinatorial Optimization Problems

- MAXFLOW and MINCUT problems
- MULTIWAY CUT problem
- MAX-2SAT, MAX-E3SAT, MAX-SAT problems
- SET COVER, VERTEX COVER problems

They can all be formulated as (integer) linear programs

Soviet Rail Network, 1955



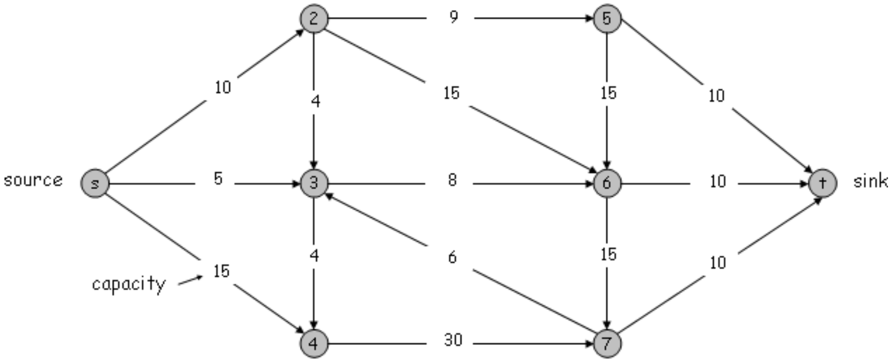
Reference: *On the history of the transportation and maximum flow problems.* Alexander Schrijver in Math Programming, 91: 3, 2002.

Maximum Flow and Minimum Cut Problems

- Cornerstone problems in combinatorial optimization
- Many non-trivial applications/reductions: airline scheduling, data mining, bipartite matching, image segmentation, network survivability, many many many more ...
- Simple Example: on the Internet with error-free transmission, what is the maximum data rate that a router s can send to a router t (assuming no network coding is allowed), given that each link has limited capacity
- More examples and applications to come

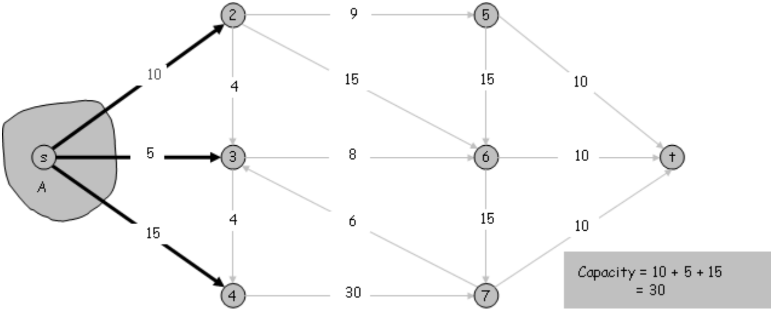
Flow Networks

- A flow network is a directed graph $G = (V, E)$ where each edge e has a capacity $c(e) > 0$
- Also, there are two distinguished nodes: the source s and the sink t



- An s,t-cut is a partition (A, B) of V where $s \in A$, $t \in B$
- Let $[A, B]$ = set of edges (u, v) with $u \in A, v \in B$
- The capacity of the cut (A, B) is defined by

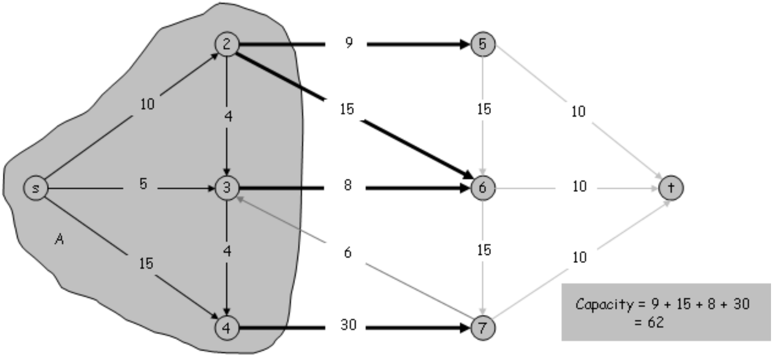
$$\text{cap}(A, B) = \sum_{e \in [A,B]} c(e)$$



Cuts

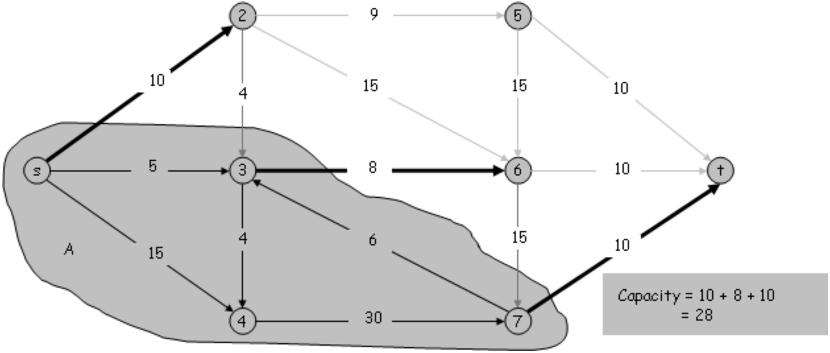
- An s, t-cut is a partition (A, B) of V where $s \in A$, $t \in B$
- Let $[A, B]$ = set of edges (u, v) with $u \in A, v \in B$
- The capacity of the cut (A, B) is defined by

$$\text{cap}(A, B) = \sum_{e \in [A,B]} c(e)$$



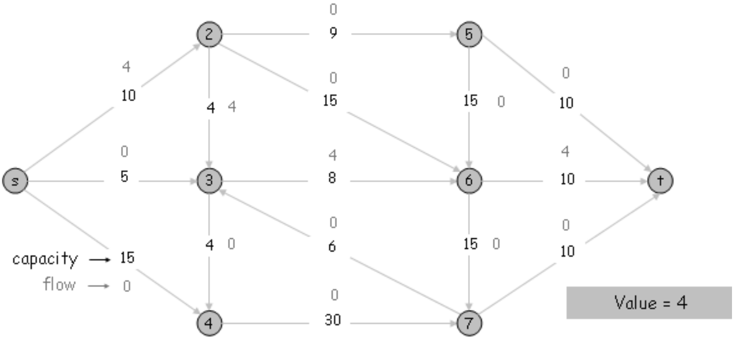
Minimum Cut - Problem Definition

Given a flow network, find an s, t-cut with minimum capacity



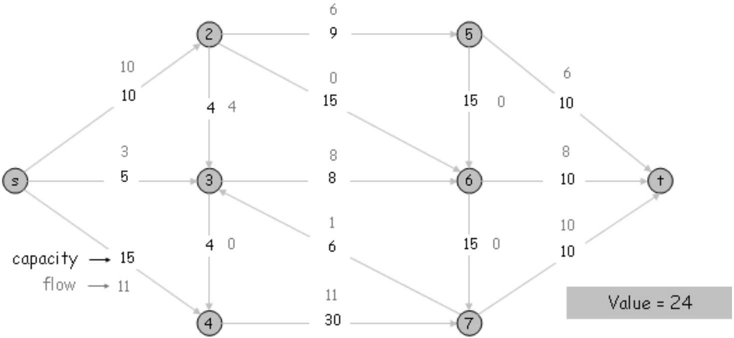
Flows

- An s, t-flow is a function $f : E \to \mathbb{R}$ satisfying
  - Capacity constraint: $0 \leq f(e) \leq c(e)$, $\forall e \in E$
  - Flow Conservation constraint: $\displaystyle\sum_{e=(u,v)\in E} f(e) = \sum_{e=(v,w)\in E} f(e)$
- The value of f: $\text{val}(f) = \displaystyle\sum_{e=(s,v)\in E} f(e)$

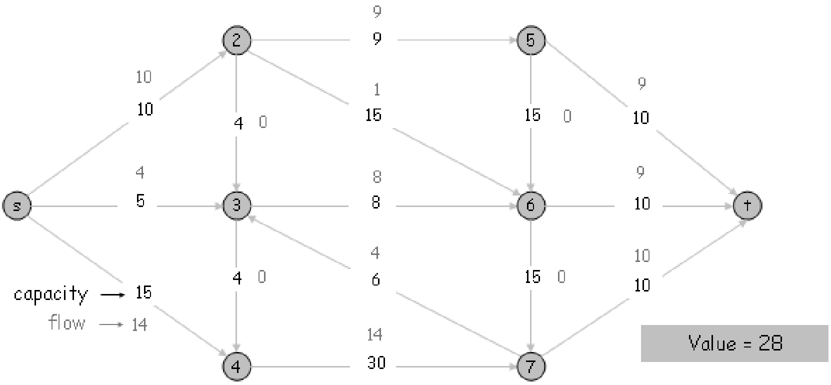


Flows

- An s,t-flow is a function $f : E \to \mathbb{R}$ satisfying
  - Capacity constraint: $0 \leq f(e) \leq c(e)$, $\forall e \in E$
  - Flow Conservation constraint: $\displaystyle\sum_{e=(u,v)\in E} f(e) = \sum_{e=(v,w)\in E} f(e)$
- The value of f: $\text{val}(f) = \displaystyle\sum_{e=(s,v)\in E} f(e)$



Given a flow network, find a flow f with maximum capacity



First Linear Program for Maximum Flow

$$
\begin{array}{lrcll}
\max & \displaystyle\sum_{e \in E} f_e & & & \\
\text{subject to} & f_e & \leq & c_e, & \forall e \in E, \\
& \displaystyle\sum_{uv \in E} f_{uv} - \sum_{vw \in E} f_{vw} & = & 0, & \forall v \neq s, t \\
& f_e & \geq & 0, & \forall e \in E
\end{array} \tag{1}
$$

Second Linear Program for Maximum Flow

- Let $\mathcal{P}$ be the set of all s, t-paths.
- f_P denote the flow amount sent along P

$$
\begin{aligned}
\max \quad & \sum_{P \in \mathcal{P}} f_P \\
\text{subject to} \quad & \sum_{P : e \in P} f_P \leq c_e, \quad \forall e \in E, \\
& \qquad f_P \geq 0, \quad \forall P \in \mathcal{P}.
\end{aligned} \tag{2}
$$

What are Linear Programs?

Optimize linear objective subject to linear equalities/inequalities

Example 1:

$$
\begin{array}{llllllllll}
\min & c_1x_1 + c_2x_2 + \cdots + c_nx_n & & & & & & & & \\
\text{subject to} & a_{11}x_1 & + & a_{12}x_2 & + & \dots & + & a_{1n}x_n & = & b_1 \\
 & a_{21}x_1 & + & a_{22}x_2 & + & \dots & + & a_{2n}x_n & = & b_2 \\
 & \vdots & & \vdots & & \dots & & \vdots & = & \vdots \\
 & a_{m1}x_1 & + & a_{m2}x_2 & + & \dots & + & a_{mn}x_n & = & b_m \\
 & & & & & & & \multicolumn{3}{r}{x_i \geq 0, \forall i = 1, \dots, n,}
\end{array}
$$

Or simply: $\min\{\mathbf{c}^T\mathbf{x} \mid \mathbf{A}\mathbf{x} = \mathbf{b}, \mathbf{x} \geq \mathbf{0}\}$

What are Linear Programs?

Optimize linear objective subject to linear equalities/inequalities

Example 2:

$$
\begin{array}{lccccccccc}
\max & c_1x_1 + c_2x_2 + \cdots + c_nx_n & & & & & & & & \\
\text{subject to} & a_{11}x_1 & + & a_{12}x_2 & + & \dots & + & a_{1n}x_n & \leq & b_1 \\
 & a_{21}x_1 & + & a_{22}x_2 & + & \dots & + & a_{2n}x_n & \leq & b_2 \\
 & \vdots & & \vdots & & \dots & & \vdots & \leq & \vdots \\
 & a_{m1}x_1 & + & a_{m2}x_2 & + & \dots & + & a_{mn}x_n & \leq & b_m \\
 & & & & & & & \multicolumn{3}{r}{x_i \geq 0, \forall i = 1, \dots, n,}
\end{array}
$$

Or simply: $\max\{\mathbf{c}^T\mathbf{x} \mid \mathbf{A}\mathbf{x} \leq \mathbf{b}, \mathbf{x} \geq \mathbf{0}\}$

Standard and Canonical Forms

Certainly, constraints may be mixed: $=, \leq, \geq$, some variables may not need to be non-negative, etc.

Example 3:

$$
\begin{array}{llllllll}
\min / \max & \mathbf{a}^{\mathbf{T}}\mathbf{x} + \mathbf{b}^{\mathbf{T}}\mathbf{y} + \mathbf{c}^{\mathbf{T}}\mathbf{z} & & & & & & \\
\text{subject to} & \mathbf{A}_{11}\mathbf{x} & + & \mathbf{A}_{12}\mathbf{y} & + & \mathbf{A}_{13}\mathbf{z} & = & \mathbf{d} \\
& \mathbf{A}_{21}\mathbf{x} & + & \mathbf{A}_{22}\mathbf{y} & + & \mathbf{A}_{23}\mathbf{z} & \leq & \mathbf{e} \\
& \mathbf{A}_{31}\mathbf{x} & + & \mathbf{A}_{32}\mathbf{y} & + & \mathbf{A}_{33}\mathbf{z} & \geq & \mathbf{f} \\
& \mathbf{x} \geq \mathbf{0}, \mathbf{y} \leq \mathbf{0}. & & & & & &
\end{array}
$$

Note that $\mathbf{A}_{ij}$ are matrices and $\mathbf{a}, \mathbf{b}, \mathbf{c}, \mathbf{d}, \mathbf{e}, \mathbf{f}, \mathbf{x}, \mathbf{y}, \mathbf{z}$ are vectors.

Fortunately, easy to “convert” any LP into any one of the following:

- The min and the max versions of the standard form:

$$\min \left\{ \mathbf{c}^T\mathbf{x} \mid \mathbf{A}\mathbf{x} = \mathbf{b}, \mathbf{x} \geq \mathbf{0} \right\}, \quad \text{and} \quad \max \left\{ \mathbf{c}^T\mathbf{x} \mid \mathbf{A}\mathbf{x} = \mathbf{b}, \mathbf{x} \geq \mathbf{0} \right\}.$$

- The min and the max versions of the canonical form:

$$\min \left\{ \mathbf{c}^T\mathbf{x} \mid \mathbf{A}\mathbf{x} \geq \mathbf{b}, \mathbf{x} \geq \mathbf{0} \right\}, \quad \text{and} \quad \max \left\{ \mathbf{c}^T\mathbf{x} \mid \mathbf{A}\mathbf{x} \leq \mathbf{b}, \mathbf{x} \geq \mathbf{0} \right\}.$$

- Simplex Method (Dantzig, 1948): worst-case exponential time, but runs very fast on most practical inputs
- Ellipsoid Method (Khachian, 1979): worst-case polynomial time, but quite slow in practice. Can even solve some LP with an exponential number of constraints if a *separation oracle* exists
- Interior Point Method (Karmarkar, 1984): worst-case polynomial time, quite fast in practice, not as popular as the simplex method

Linear Programming Duality

To each LP (called the primal LP) there corresponds another LP called the *dual LP* satisfying the following:

			Dual		
			Feasible		Infeasible
			Optimal	Unbounded	
Primal	Feasible	Optimal	X	O	O
		Unbounded	O	O	X
	Infeasible		O	X	X

(X = Possible, O = Impossible)

If the primal is a $\min\{\dots\}$, then the dual is a $\max\{\dots\}$ and vice versa

Theorem (Strong duality)

If both the primal and the dual LPs are feasible, then their optimal objective values are the same.

Rules for Writing Down the Dual LP

Maximization problem	**Minimization problem**
Constraints	Variables
ith constraint $\leq$	ith variable ≥ 0
ith constraint $\geq$	ith variable ≤ 0
ith constraint $=$	ith variable unrestricted
Variables	Constraints
jth variable ≥ 0	jth constraint $\geq$
jth variable ≤ 0	jth constraint $\leq$
jth variable unrestricted	jth constraint $=$

Table: Rules for converting between primals and duals.

Primal/Dual Pair - Standard Form

In **standard form**, the primal and dual LPs are

$$\begin{array}{rll} \min & \mathbf{c}^T\mathbf{x} & \text{(primal program)} \\ \text{subject to} & \mathbf{A}\mathbf{x} = \mathbf{b} & \\ & \mathbf{x} \geq \mathbf{0} & \end{array}$$

$$\begin{array}{rll} \max & \mathbf{b}^T\mathbf{y} & \text{(dual program)} \\ \text{subject to} & \mathbf{A}^T\mathbf{y} \leq \mathbf{c} & \text{no non-negativity restriction!.} \end{array}$$

Primal/Dual Pair - Canonical Form

In **canonical form**, the primal and dual LPs are

$$\begin{aligned} \min \quad & \mathbf{c}^T\mathbf{x} \qquad \text{(primal program)} \\ \text{subject to} \quad & \mathbf{A}\mathbf{x} \geq \mathbf{b} \\ & \mathbf{x} \geq \mathbf{0} \end{aligned}$$

$$\begin{aligned} \max \quad & \mathbf{b}^T\mathbf{y} \qquad \text{(dual program)} \\ \text{subject to} \quad & \mathbf{A}^T\mathbf{y} \leq \mathbf{c} \\ & \mathbf{y} \geq \mathbf{0}. \end{aligned}$$

Weak Duality and Strong Duality

Primal LP: $\min\{\mathbf{c}^T\mathbf{x} \mid \mathbf{A}\mathbf{x} \geq \mathbf{b}, \mathbf{x} \geq \mathbf{0}\}$
Dual LP: $\max\{\mathbf{b}^T\mathbf{y} \mid \mathbf{A}^T\mathbf{y} \leq \mathbf{c}, \mathbf{y} \geq \mathbf{0}\}$.

Theorem (Weak Duality)

Suppose $\mathbf{x}$ *is primal feasible, and* $\mathbf{y}$ *is dual feasible, then* $\mathbf{c}^T\mathbf{x} \geq \mathbf{b}^T\mathbf{y}$. *In particular, if* $\mathbf{x}^*$ *is primal-optimal and* $\mathbf{y}^*$ *is dual-optimal, then*

$$\mathbf{c}^T\mathbf{x}^* \geq \mathbf{b}^T\mathbf{y}^*.$$

Theorem (Strong Duality)

If the primal LP has an optimal solution $\mathbf{x}^*$, *then the dual LP has an optimal solution* $\mathbf{y}^*$ *such that*

$$\mathbf{c}^T\mathbf{x}^* = \mathbf{b}^T\mathbf{y}^*.$$

Complementary Slackness

Corollary (Complementary Slackness - canonical form)

Given the following programs

$$\text{Primal LP:} \quad \min\{\mathbf{c}^T\mathbf{x} \mid \mathbf{A}\mathbf{x} \geq \mathbf{b}, \mathbf{x} \geq \mathbf{0}\},$$
$$\text{Dual LP:} \quad \max\{\mathbf{b}^T\mathbf{y} \mid \mathbf{A}^T\mathbf{y} \leq \mathbf{c}, \mathbf{y} \geq \mathbf{0}\}.$$

Let $\mathbf{x}^*$ *and* $\mathbf{y}^*$ *be feasible for the primal and the dual programs, respectively. Then,* $\mathbf{x}^*$ *and* $\mathbf{y}^*$ *are optimal for their respective LPs if and only if*

$$\left(\mathbf{c} - \mathbf{A}^{\mathbf{T}}\mathbf{y}^*\right)^{\mathbf{T}} \mathbf{x}^* = \mathbf{0}, \quad \text{and} \quad (\mathbf{b} - \mathbf{A}\mathbf{x})^{\mathbf{T}} \mathbf{y}^* = \mathbf{0}. \tag{3}$$

First ILP for Mincut

Intuition: for a cut (A, B), set $x_v = 1$ if $v \in A$ and $x_v = 0$ otherwise.

$$\begin{array}{lll} \min & \displaystyle\sum_{e \in E} c_e z_e & \\ \text{subject to} & z_e \geq x_u - x_v & \forall e = uv \in E, \\ & z_e \geq x_v - x_u & \forall e = uv \in E, \\ & x_s = 1 & \\ & x_t = 0 & \\ & z_e, x_v \in \{0, 1\}, & \forall v \in V, e \in E \end{array}$$

Second ILP for Mincut

Let $\mathcal{P}$ be the collection of all s, t-paths

$$\begin{aligned} \min \quad & \sum_{e \in E} c_e y_e \\ \text{subject to} \quad & \sum_{e \in P} y_e \geq 1, \quad \forall P \in \mathcal{P}, \\ & y_e \in \{0,1\}, \quad \forall e \in E. \end{aligned} \tag{4}$$

Multiway Cut

MULTIWAY CUT:

Given an edge weighted graph $G = (V, E)$ ($w : E \to \mathbb{R}^+$) and k terminals $\{t_1, \dots, t_k\}$. Find a min-weight subset of edges whose removal disconnects the terminals from one another.

Let $\mathcal{P}$ be the collection of all s_i, s_j-paths

$$\begin{aligned} \min \quad & \sum_{e \in E} w_e x_e \\ \text{subject to} \quad & \sum_{e \in P} x_e \geq 1, \quad \forall P \in \mathcal{P}, \\ & x_e \in \{0, 1\}, \quad \forall e \in E. \end{aligned} \tag{5}$$

Vertex Cover

WEIGHTED VERTEX COVER

Given a graph $G = (V, E)$, $|V| = n$, $|E| = m$, a weight function $w : V \to \mathbb{R}$. Find a vertex cover $C \subseteq V$ for which $\sum_{i \in C} w(i)$ is minimized.

An equivalent integer linear program (ILP) is

$$\begin{array}{lll} \min & w_1x_1 + w_2x_2 + \cdots + w_nx_n & \\ \text{subject to} & x_i + x_j \geq 1, & \forall ij \in E, \\ & x_i \in \{0, 1\}, & \forall i \in V. \end{array}$$

WEIGHTED SET COVER

Given a collection $\mathcal{S} = \{S_1, \dots, S_n\}$ *of subsets of* $[m] = \{1, \dots, m\}$*, and a weight function* $w : \mathcal{S} \to \mathbb{R}$*. Find a cover* $\mathcal{C} = \{S_j \mid j \in J\}$ *with minimum total weight.*

Use a 01-variable x_j to indicate the inclusion of S_j in the cover. The corresponding ILP is thus

$$\begin{array}{rl} \min & w_1x_1 + \cdots + w_nx_n \\ \text{subject to} & \displaystyle\sum_{j:S_j \ni i} x_j \geq 1, \quad \forall i \in [m], \\ & x_j \in \{0,1\}, \quad \forall j \in [n]. \end{array}$$

Max-SAT

WEIGHTED MAX-SAT:

Given a CNF formula φ with m weighted clauses on n variables, find a truth assignment maximizing the total weight of satisfied clauses.

Say, clause C_j has weight $w_j \in \mathbb{R}^+$. Here's an ILP

$$\begin{array}{rll} \max & w_1 z_1 + \cdots + w_m z_n & \\ \text{subject to} & \displaystyle\sum_{i:x_i \in C_j} y_i + \sum_{i:\bar{x}_i \in C_j} (1 - y_i) \geq z_j, & \forall j \in [m], \\ & \qquad\qquad y_i, z_j \in \{0, 1\}, & \forall i \in [n], j \in [m] \end{array}$$