Mathematical Programs

- Linear Program (LP)

$$\begin{aligned} \min \quad & c^\mathsf{T} x \\ \text{s.t.} \quad & a_i^\mathsf{T} x \le b_i \qquad \forall i = 1, ..., m \end{aligned}$$

Can be efficiently solved e.g., by Ellipsoid Method

- Integer Program (IP)

$$\begin{aligned} \min \quad & c^\mathsf{T} x \\ \text{s.t.} \quad & a_i^\mathsf{T} x \le b_i \qquad \forall i = 1, ..., m \\ & x \in \mathbb{Z}^n \end{aligned}$$

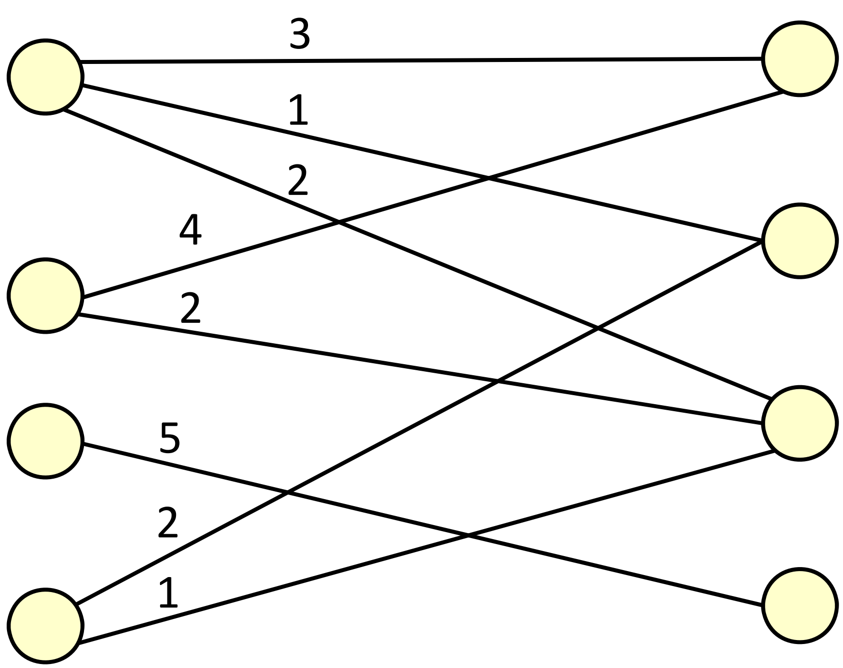
Cannot be efficiently solved assuming P $\neq$ NP

Combinatorial Optimization

- Study of optimization problems that have discrete solutions and some combinatorial flavor (e.g., involving graphs)
- Why are we interested in this?
  - Applications: OR (planning, scheduling, supply chain), Computer networks (shortest paths, low-cost trees), Compilers (coloring), Online advertising (matching)...
  - Rich theory of what can be solved **efficiently** and what cannot
  - Underlying math can be very interesting

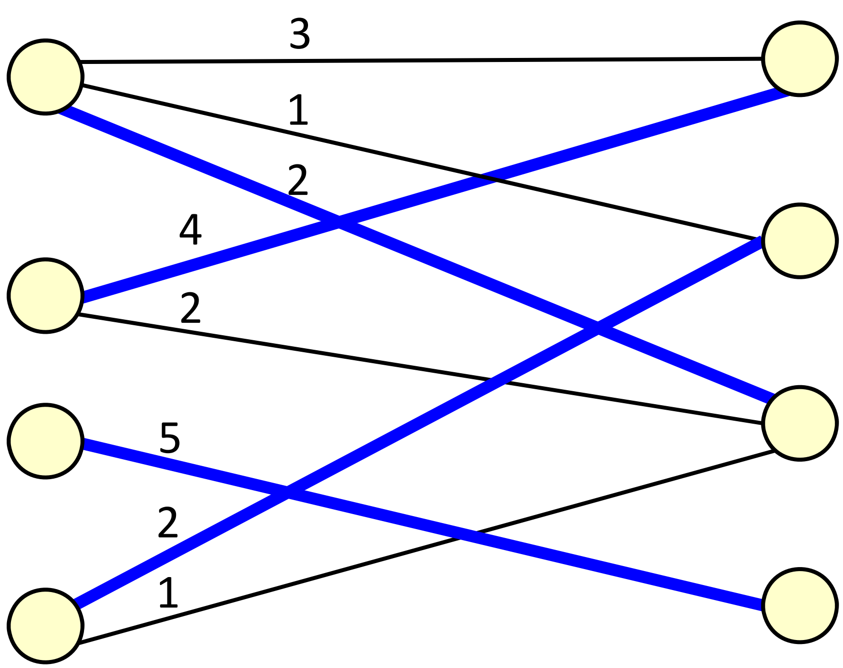
Combinatorial IPs are often nice

- **Max-Weight Perfect Matching**
- Given bipartite graph G=(V, E). Every edge e has a weight w_e.
- Find a maximum-weight perfect matching
  - A set M $\subseteq$ E s.t. every vertex has **exactly** one incident edge in M



Combinatorial IPs are often nice

- **Max-Weight Perfect Matching**
- Given bipartite graph G=(V, E). Every edge e has a weight w_e.
- Find a maximum-weight perfect matching
  - A set M $\subseteq$ E s.t. every vertex has **exactly** one incident edge in M



The blue edges are a max-weight perfect matching M

Combinatorial IPs are often nice

- **Max-Weight Perfect Matching**
- Given bipartite graph G=(V, E). Every edge e has a weight w_e.
- Find a maximum-weight perfect matching
  - A set M $\subseteq$ E s.t. every vertex has **exactly** one incident edge in M
- The natural integer program

$$
\begin{array}{lllll}
\max & \sum_{e \in E} w_e \cdot x_e & & \\
\text{s.t.} & \sum_{e \text{ incident to } v} x_e & = 1 & \forall v \in V \\
 & x_e & \in \{0,1\} & \forall e \in E
\end{array}
$$

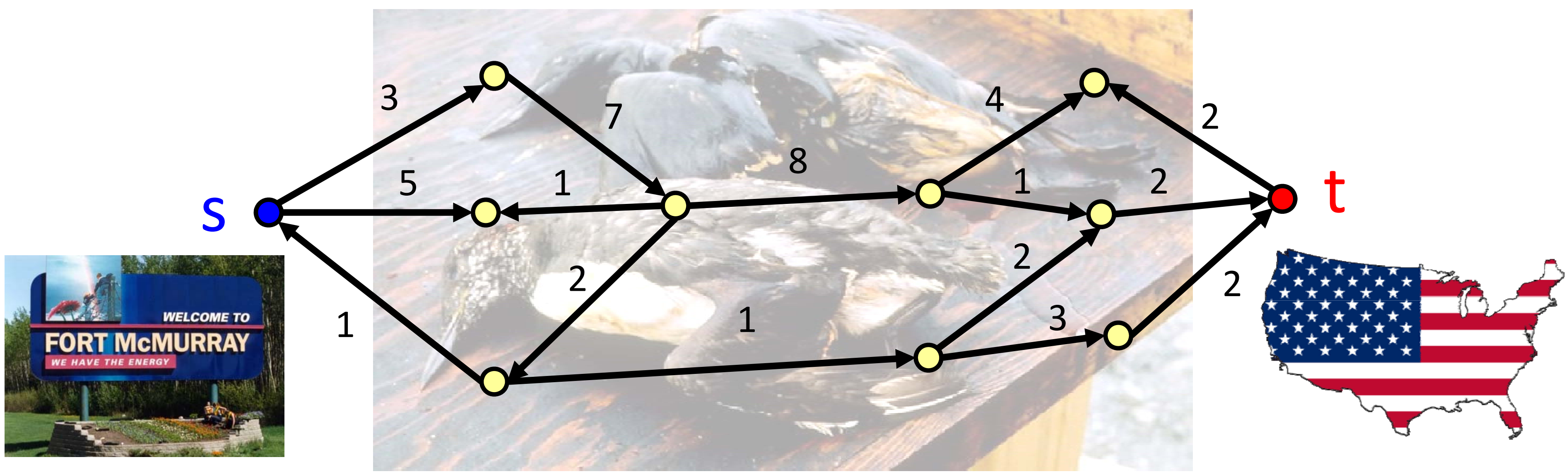
- This IP **can** be efficiently solved, in many different ways

How to solve combinatorial IPs?

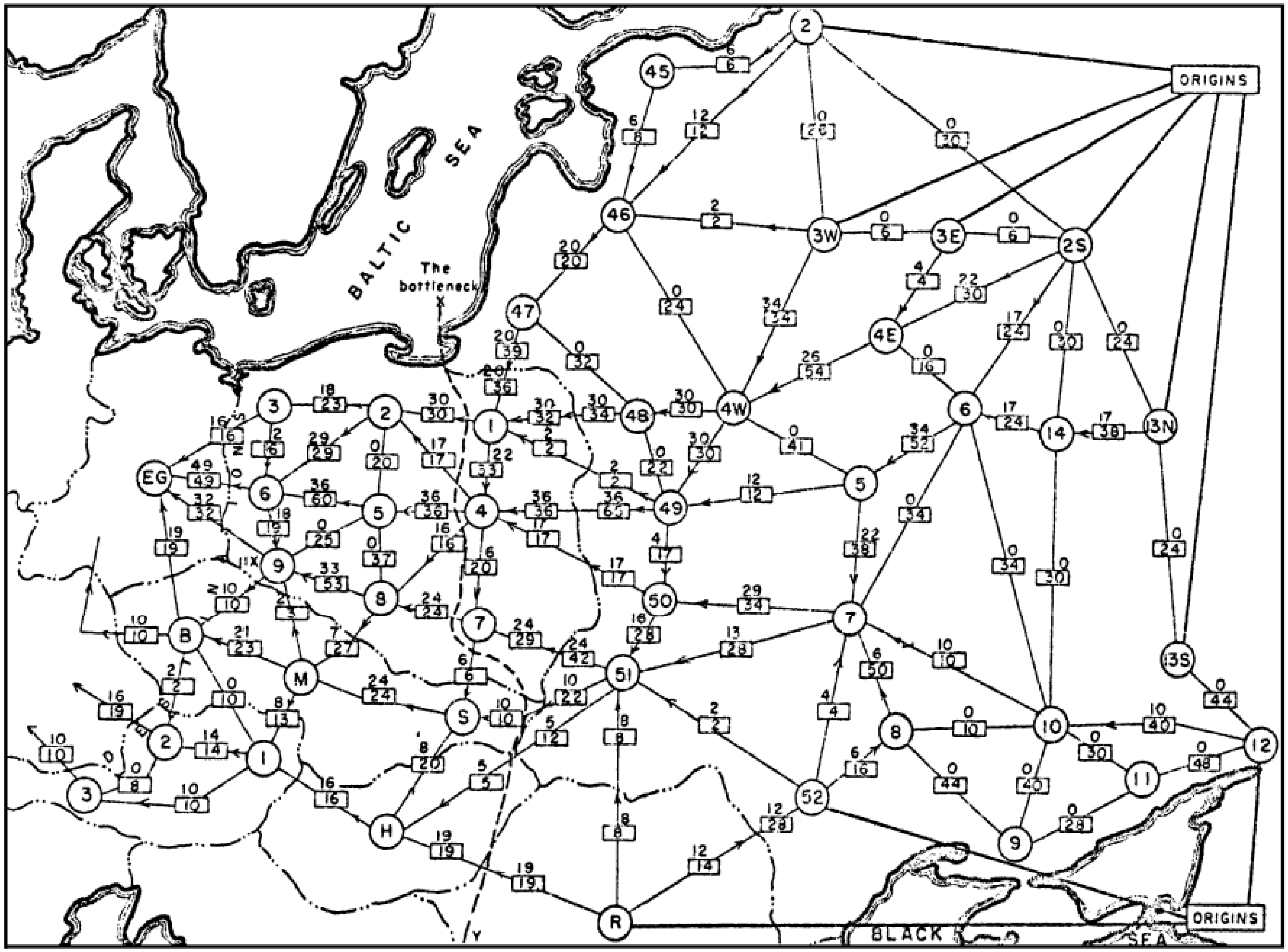
- Two common approaches
  1. Design combinatorial algorithm that directly solves IP
     - Often such algorithms have a nice LP interpretation
  2. Relax IP to an LP; prove that they give same solution; solve LP by the ellipsoid method
     - Need to show special structure of the LP’s extreme points
     - Sometimes we can analyze the extreme points **combinatorially**
     - Sometimes we can use **algebraic** structure of the constraints. For example, if constraint matrix is **Totally Unimodular** then IP and LP are equivalent
- We’ll see examples of these approaches

Network Flow

- Let D=(N,A) be a directed graph.
- Every arc a has a "capacity" $c_a \geq 0$. (Think of it as an oil pipeline)
- Want to send oil from node s to node t through pipelines
- Oil must not leak at any node, except s and t:
  flow in = flow out.
- How much oil can we send?
- For simplicity, assume no arc enters s and no arc leaves t.



Max Flow & Min Cut



Harris and Ross [1955]

Schematic diagram of the railway network of the Western Soviet Union and Eastern European countries, with a maximum flow of value 163,000 tons from Russia to Eastern Europe, and a cut of capacity 163,000 tons indicated as ‘The bottleneck’. [Schrijver, 2005]

Max Flow & Min Cut

- Let D=(N,A) be a digraph, where arc a has capacity c_a.
- **Definition:** For any $U \subseteq N$, the **cut** $\delta^+(U)$ is:

$$\delta^+(U) = \{\, uv \;:\; u \in U,\ v \notin U,\ uv \in A \,\}$$

The **capacity** of the cut is:

$$c(\delta^+(U)) = \sum_{a \in \delta^+(U)} c_a$$

Delbert Ray Fulkerson

- **Theorem:** [Ford & Fulkerson 1956]
  The maximum amount of flow from s to t equals
  the minimum capacity of a cut $\delta^+(U)$, where $s \in U$ and $t \notin U$
- Furthermore, if c is integral then there is an integral flow that achieves the maximum flow.

LP Formulation of Max Flow

- **Variables:** x_a = amount of flow to send on arc a
- **Constraints:**
  For every node except s & t, flow in = flow out.
  Flow through each arc can not exceed its capacity.
- **Objective value:** Total amount of flow sent by s.
- **Notation:** $\delta^+(v)$ = arcs with tail at v
  $\delta^-(v)$ = arcs with head at v
- The LP is:

$$
\begin{array}{lll}
\max & \sum_{a \in \delta^+(s)} x_a & \\
\text{s.t.} & \sum_{a \in \delta^-(v)} x_a - \sum_{a \in \delta^+(v)} x_a \quad = 0 & \forall v \in N \setminus \{s,t\} \\
 & 0 \leq x_a \leq c_a & \forall a \in A
\end{array}
$$

Max Flow & Min Cut

- Let D=(N,A) be a digraph, where arc a has capacity c_a.
- **Definition:** For any $U \subseteq N$, the **cut** $\delta^+(U)$ is:

$$\delta^+(U) = \{\, uv \,:\, u \in U,\, v \notin U,\, uv \in A \,\}$$

The **capacity** of the cut is:

$$c(\delta^+(U)) = \sum_{a \in \delta^+(U)} c_a$$

- **"Weak Duality":** For any flow and any U with $s \in U$, $t \notin U$, the amount of flow from s to t is at most $c(\delta^+(U))$.

- **Proof:** The net amount of flow crossing U is

$$\sum_{a \in \delta^+(U)} x_a - \sum_{a \in \delta^+(V \setminus U)} x_a \;\leq\; \sum_{a \in \delta^+(U)} c_a$$

since $0 \leq x \leq c$.

Incidence Matrix of a Directed Graph

$$
\begin{array}{llll}
\max & \sum_{a\in\delta^+(s)} x_a & & \\
\text{s.t.} & \sum_{a\in\delta^-(v)} x_a - \sum_{a\in\delta^+(v)} x_a & = 0 & \forall v \in N \setminus \{s,t\} \\
 & 0 \le x_a \le c_a & & \forall a \in A
\end{array}
$$

- What is the matrix M defining the constraints of this LP?
  - Row for every node (except s or t)
  - Column for every arc

$$
M_{v,a} = \begin{cases} +1 & \text{if node v is the head of arc a} \\ -1 & \text{if node v is the tail of arc a} \\ 0 & \text{otherwise} \end{cases}
$$

- **Goal:** Analyze extreme points of this LP.

Total Unimodularity

- Let M be a real m×n matrix
- **Definition:** Suppose that every square submatrix of M has determinant in {0, +1, -1}. Then M is **totally unimodular (TUM)**.
  - In particular, every entry of M must be in {0, +1, -1}
- **Key point:** Polytopes defined by TUM matrices have integral extreme points.

Lemma: Suppose M is TUM. Let b, c be integer vectors.
Then every extreme point of $P = \{ x : Mx \leq b \}$ is integral.
And every extreme point of $P = \{ x : Mx=b,\ 0 \leq x \leq c \}$ is integral.

Total Unimodularity

- Let A be a real $m \times n$ matrix
- **Definition:** Suppose that every square submatrix of A has determinant in {0, +1, -1}. Then A is **totally unimodular (TUM)**.
  - In particular, every entry of A must be in {0, +1, -1}
- **Lemma:** Suppose A is TUM. Let b be any integer vector. Then every basic feasible solution of $P = \{ x : Ax \leq b \}$ is integral.
- **Proof:** Let x be a basic feasible solution.

  Then the constraints that are tight at x have rank n.

  Let A' be a submatrix of A and b' a subvector of b corresponding to n linearly independent constraints that are tight at x.

  Then x is the unique solution to $A' x = b'$, i.e., $x = (A')^{-1} b'$.

  Cramer's Rule: If M is a square, non-singular matrix then $(M^{-1})_{i,j} = (-1)^{i+j} \det \underbrace{M_{del(j,i)}}_{\text{Submatrix of M obtained by deleting row j and column i}} / \det M$.

Total Unimodularity

- Let A be a real m×n matrix
- **Definition:** Suppose that every square submatrix of A has determinant in {0, +1, -1}. Then A is **totally unimodular (TUM)**.
- **Lemma:** Suppose A is TUM. Let b be any integer vector. Then every basic feasible solution of $P = \{ x : Ax \leq b \}$ is integral.
- **Proof:** Let x be a basic feasible solution.

  Then the constraints that are tight at x have rank n.

  Let A' be the submatrix of A and b' the subvector of b containing n linearly independent constraints that are tight at x.

  Then x is the unique solution to $A' x = b'$, i.e., $x = (A')^{-1} b'$.

  Cramer's Rule: If M is a square, non-singular matrix then $(M^{-1})_{i,j} = (-1)^{i+j} \det M_{del(j,i)} / \det M$.

  Thus all entries of $(A')^{-1}$ are in {0, +1, -1}.

  Since b' is integral, x is also integral. ■

Incidence Matrices are TUM

- Let D=(N, A) be a directed graph. Define M by:

$$M_{u,a} = \begin{cases} +1 & \text{if node u is the head of arc a} \\ -1 & \text{if node u is the tail of arc a} \\ 0 & \text{otherwise} \end{cases}$$

- **Lemma:** M is TUM.
- **Proof:** Let Q be a k$\times$k submatrix of M. Argue by induction on k. If k=1 then Q is a single entry of M, so det(Q) is either 0 or $\pm$1. So assume k>1.

- **Lemma:** M is TUM.
- **Proof:** Let Q be a k×k submatrix of M. Assume k>1.

  Case 1:

  If some column of Q has **no** non-zero entries, then det(Q)=0.

  Case 2:

  Suppose j^{th} column of Q has **exactly one** non-zero entry, say $Q_{t,j} \neq 0$

  Use "Column Expansion" of determinant:

  $$\det Q \;=\; \sum_i (-1)^{i+j} Q_{i,j} \cdot \det Q_{\mathrm{del}(i,j)} \;=\; (-1)^{t+j} Q_{t,j} \cdot \det Q_{\mathrm{del}(t,j)},$$

  where t is the unique non-zero entry in column j.

  By induction, det $Q_{del(t,j)}$ in {0,+1,-1} $\Rightarrow$ det Q in {0,+1,-1}.

  Case 3:

  Suppose **every** column of Q has **exactly two** non-zero entries.

  – For each column, one non-zero is a +1 and the other is a -1.

  So summing all rows in Q gives the vector [0,0,…,0].

  Thus Q is singular, and det Q = 0. ■

The Max Flow LP

$$\begin{array}{lll} \max & \sum_{a\in\delta^+(s)} x_a & \\ \text{s.t.} & \sum_{a\in\delta^-(v)} x_a - \sum_{a\in\delta^+(v)} x_a \quad = 0 & \forall v \in N \setminus \{s,t\} \\ & 0 \le x_a \le c_a & \forall a \in A \end{array}$$

- **Observations:**
  - The LP is feasible (assume the capacities are all non-negative)
  - The LP is bounded (because the feasible region is bounded)
  - It has an optimal solution, i.e., a maximum flow. (by FTLP)
- The feasible region is P = { x : Mx=b, 0≤x≤c } where M is TUM.

- **Corollary:** If c is integral, then every extreme point is integral, and so there is a maximum flow that is integral.

- **Q:** Why does P have any extreme points? **A:** It contains no line.

Max Flow LP & Its Dual

$$\begin{array}{lll} \max & \sum_{a\in\delta^+(s)} x_a & \\ \text{s.t.} & \sum_{a\in\delta^-(v)} x_a - \sum_{a\in\delta^+(v)} x_a \quad = 0 & \forall v \in N \setminus \{s,t\} \\ & 0 \le x_a \le c_a & \forall a \in A \end{array}$$

- **Dual variables:**
  - A variable y_v for every v ∈ N \ {s,t}
  - A variable z_{uv} for every arc uv
- The dual is

$$\begin{array}{llll} \min & \sum_{a\in A} c_a z_a & & \\ \text{s.t.} & -y_u + y_v + z_{uv} & \ge 0 & \forall uv \in A, v, w \in N \setminus \{s,t\} \\ & y_v + z_{sv} & \ge 1 & \forall sv \in A \\ & -y_u + z_{ut} & \ge 0 & \forall ut \in A \\ & z & \ge 0 & \end{array}$$

- **Let's simplify:** Set y_s = 1 and y_t = 0

The Dual

$$\begin{array}{llll} \min & \sum_{a \in A} c_a \, z_a & & \\ \text{s.t.} & -y_u + y_v + z_{uv} & \geq 0 & \forall uv \in A \\ & z & \geq 0 & \end{array}$$

where y_s and y_t are **not** variables: $y_s = 1$ and $y_t = 0$

- **We will show:** Given an optimal solution (y,z), we can construct a cut $\delta^+(U)$ such that

$$c(\delta^+(U)) = \sum_{a \in A} c_a z_a$$

- In other words, the capacity of the cut $\delta^+(U)$ equals the optimal value of the dual LP.
- By strong LP duality, this equals the optimal value of the primal LP, which is the maximum flow value.
- **Weak duality:** Every cut has capacity at least the max flow value, so this must be a minimum cut.

- Primal: max { $d^T x$: $Mx=0$, $0 \leq x \leq c$ }

  Dual: = min { $c^T z$: $\begin{pmatrix} M^T & I \end{pmatrix} \binom{y}{z} \geq 0$, $z \geq 0$, $y_s=1$, $y_t=0$ }

$$= \min \quad \sum_{a \in A} c_a \, z_a$$

$$\text{s.t.} \quad -y_u + y_v + z_{uv} \geq 0 \qquad \forall uv \in A$$

$$z \geq 0 \qquad y_s=1,\ y_t=0$$

- **Claim:** [M^T I] is also TUM

$\Rightarrow$ Any extreme point solution of Dual has y and z integral

- Since we're minimizing, can assume $z_{uv} = \max\{y_u - y_v, 0\}$
- Define U = { v : $y_v \geq 1$ }. Then $s \in U$, $t \notin U$.
- Note $z_{uv} \geq 1$ for all $uv \in \delta^+(U)$.
- Max Flow Value = $\Sigma_a z_a c_a \geq c(\delta^+(U)) \geq$ Max Flow Value

  Strong Duality　　　　Weak Duality

- $\delta^+(U)$ is a cut separating s&t with capacity = max flow

Summary

- We have proven:

- **Theorem:** [Ford & Fulkerson 1956]
  The maximum amount of flow from s to t equals
  the minimum capacity of a cut $\delta^+(U)$, where $s \in U$ and $t \notin U$
  Furthermore, if c is integral then there is an integral flow
  that achieves the maximum flow.

- We also get an algorithm for finding max flow & min cut
  - Solve Max Flow LP by the ellipsoid method.
  - Get an extreme point solution. It is an integral max flow.
  - Solve Dual LP by the ellipsoid method.
  - Get an extreme point solution. $U = \{ v : y_v \geq 1 \}$ is a min cut.
- This algorithm runs in polynomial time