Lecture 2: Linear Programming and Duality

Lecture Outline

- Part I: Linear Programming and Examples
- Part II: Von Neumann's Minimax Theorem and Linear Programming Duality
- Part III: Linear Programming as a Problem Relaxation

Part I: Linear Programming, Examples, and Canonical Form

Linear Programming

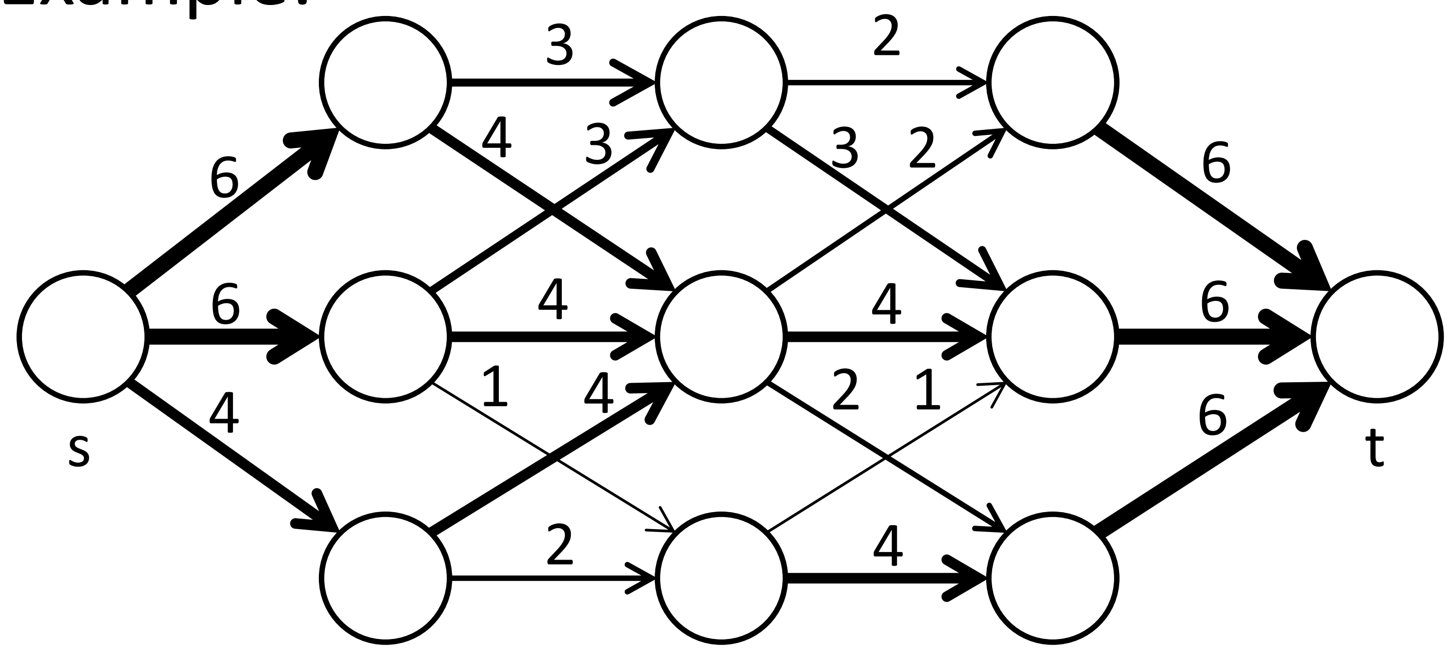
- Linear Programming: Want to optimize a linear function over linear equalities and inequalities.
- Example: Maximize $\mathrm{f(x, y, z)} = 3x + 4y + 5z$ when
  1. $x + y + z = 1$
  2. $x \geq 0$
  3. $y \geq 0$
  4. $z \geq 0$
- Answer: $x = y = 0, z = 1, f(x, y, z) = 5$

Example: Directed Connectivity

- Directed connectivity: Is there a path from $s = x_1$ to $t = x_n$ in a directed graph G?
- Linear program: Minimize x_n subject to
  1. $x_1 = 1$
  2. $x_j \geq x_i$ whenever $\mathrm{x}_i \to x_j \in E(G)$
  3. $\forall i, x_i \geq 0$
- Answer is 1 if there is a path from s to t in G and 0 otherwise.

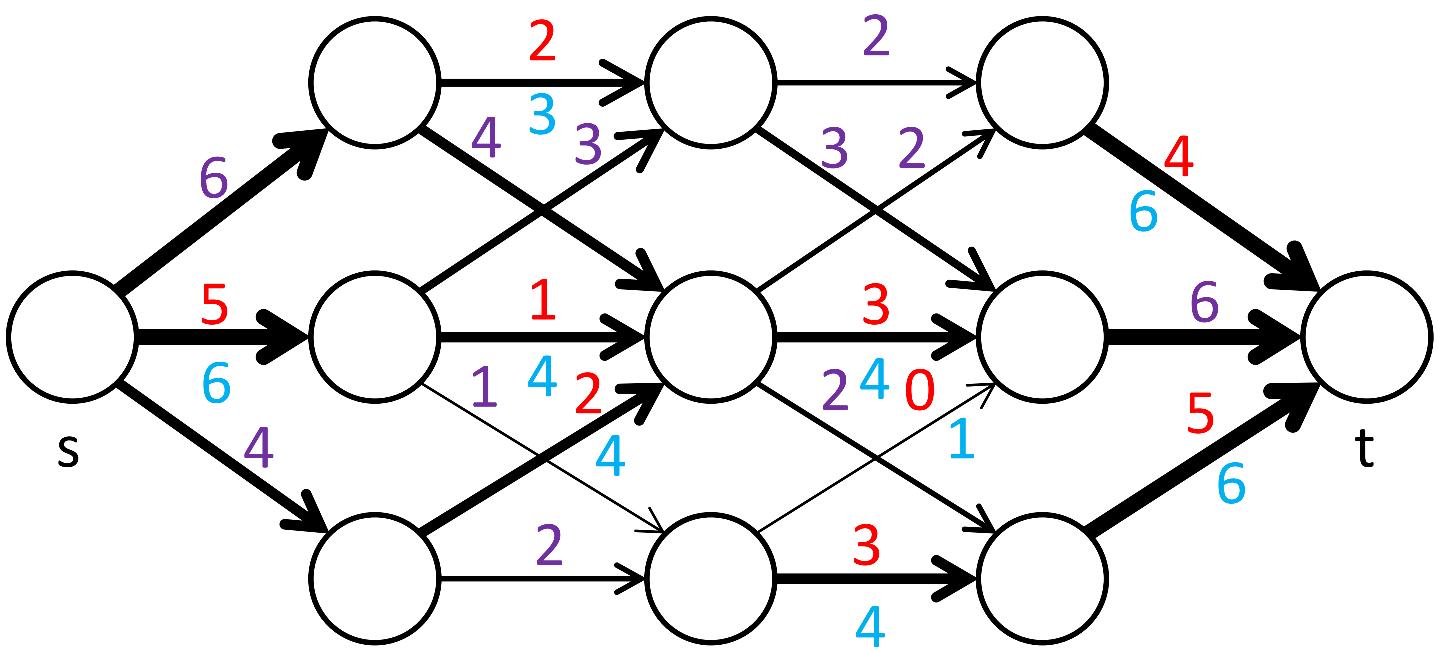
Example: Maximum Flow

- Max flow: Given edge capacities c_{ij} for each edge $x_i \to x_j$ in G, what is the maximum flow from $s = x_1$ to $t = x_n$?
- Example:



Example Answer: 15

- Answer: 15. Actual flow is red/purple, capacity is blue/purple.



Max Flow Equations

- Take x_{ij} = flow from i to j
- Recall: c_{ij} is the capacity from i to j
- Program: Maximize x_{n1} subject to
  1. $\forall i, j, 0 \leq x_{ij} \leq c_{ij}$ (no capacity is exceeded, no negative flow)
  2. $\forall i, \sum_{j=1}^{n} x_{ji} = \sum_{j=1}^{n} x_{ij}$ (flow in = flow out)

In-class Exercise

- Shortest path problem: Given a directed graph G with lengths l_{ij} on the edges, what is the shortest path from $s = x_1$ to $t = x_n$ in G?
- Exercise: Express the shortest path problem as a linear program.

In-class Exercise Answer

- Shortest path problem: How long is the shortest path from $s = x_1$ to $t = x_n$ in a directed graph G?
- Linear Program: Have variables d_i representing the distance of vertex x_i from vertex $s = x_1$. Maximize d_n subject to
  1. $d_1 = 0$
  2. $\forall i, j, d_j \leq d_i + l_{ij}$ where l_{ij} is the length of the edge from x_i to x_j

Canonical Form

- Canonical form: Maximize $\mathrm{c}^{\mathrm{T}}x$ subject to
  1. $Ax \leq b$
  2. $x \geq 0$

Putting Things Into Canonical Form

- Canonical form: Maximize $\mathrm{c}^{\mathrm{T}}x$ subject to
  1. $Ax \leq b$
  2. $x \geq 0$
- To put a linear program into canonical form:
  1. Replace each equality $a_i^T x = b_i$ with two inequalities $a_i^T x \leq b_i$ and $-a_i^T x \leq -b_i$
  2. In each expression, replace x_j with $(x_j^+ - x_j^-)$ where x_j^+, x_j^- are two new variables.

Slack Form

- Slack form: Maximize $\mathrm{c}^{\mathrm{T}}x$ subject to
  1. $Ax = b$
  2. $x \geq 0$

Putting Things Into Slack Form

- Slack form: Maximize $\mathrm{c}^{\mathrm{T}}x$ subject to
  1. $Ax = b$
  2. $x \geq 0$
- To put a linear program into slack form from canonical form, simply add a slack variable for each inequality.

$$\sum_{j=1}^{n} a_{ij} x_j \leq b_i \Leftrightarrow \left(\sum_{j=1}^{n} a_{ij} x_j\right) + s_i = b_i, s_i \geq 0$$

Part II: Von Neumann's Minimax Theorem and Linear Programming Duality

Linear Programming Duality

- Primal: Maximize $\mathrm{c}^{\mathrm{T}}x$ subject to
  1. $Ax \leq b$
  2. $x \geq 0$
- Dual: Minimize $b^T y$ subject to
  1. $A^T y \geq c$
  2. $y \geq 0$
- Observation: For any feasible x, y, $c^T x \leq b^T y$ because

$$c^T x \leq y^T A x = y^T (Ax - b) + y^T b \leq b^T y$$

- Strong duality: $c^T x = b^T y$ at optimal x, y

Heart of Duality

- Game: Have a function $f: X \times Y \to R$.
- X player wants to minimize $f(x, y)$, Y player wants to maximize $f(x, y)$
- Obvious: Better to go second, i.e

$$\max_{y \in Y} \min_{x \in X} f(x, y) \leq \min_{x \in X} \max_{y \in Y} f(x, y)$$

- Minimax theorems: Under certain conditions,

$$\max_{y \in Y} \min_{x \in X} f(x, y) = \min_{x \in X} \max_{y \in Y} f(x, y) \ !$$

Von Neumann's Minimax Theorem

- Von Neumann [1928]: If X and Y are convex compact subsets of R^m and R^n and $f: X \times Y \rightarrow R$ is a continuous function which is convex in X and concave in Y then

$$\max_{y \in Y} \min_{x \in X} f(x, y) = \min_{x \in X} \max_{y \in Y} f(x, y)$$

- These conditions are necessary (see problem set)

Example

- Let $X = Y = [-1,1]$ and consider the function $f(x, y) = xy$.
- If the x player goes first and plays $x = .5$, the y player should play $y = 1$, obtaining $f(x, y) = .5$
- If the x player goes first and plays $x = -.5$, the y player should play $y = -1$, obtaining $f(x, y) = .5$
- The best play for the x player is $x = 0$ as then $f(x, y) = 0$ regardless of what y is.

Connection to Nash Equilibria

- Recall: X player wants to minimize $f(x,y)$, Y player wants to maximize $f(x,y)$.
- If (x^*, y^*) is a Nash equilibrium then

$$f(x^*, y^*) \leq \max_{y \in Y} \min_{x \in X} f(x,y)$$

$$\leq \min_{x \in X} \max_{y \in Y} f(x,y) \leq f(x^*, y^*)$$

- Note: Since f is convex in x and concave in y, pure strategies are always optimal.
- However, this is circular: proof that Nash equilibria exist ≈ proof of minimax theorem

Minimax Theorem Proof Sketch

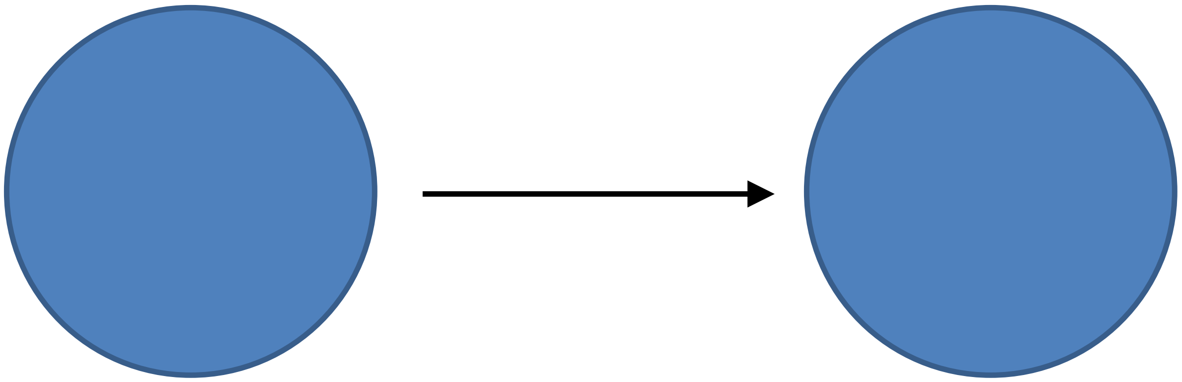
- Proof idea:
  1. Define a function $T: X \times Y \rightarrow X \times Y$ so that $T(x, y) = (x, y)$ if and only if (x, y) is a Nash equilibrium.
  2. Use Brouwer's fixed point theorem to argue that T must have a fixed point.

Attempt #1

- We could try to define T as follows
  1. Starting from (x, y), take x' to be the closest point to x which minimizes $f(x', y)$.
  2. Now take y' to be the closest point to y which maximizes $f(x', y')$
  3. Take $T(x, y) = (x', y')$
- $T(x, y) = (x, y) \Leftrightarrow (x, y)$ is a Nash equilibrium.

Brouwer's Fixed Point Theorem

- Brouwer's fixed point theorem: If X is a convex, compact subset of R^n then any continuous map $f: X \to X$ has a fixed point
- Example: Any continuous function $f: D^2 \to D^2$ has a fixed point.



Correct function T

- Problem: Previous T may not be continuous!
- Correct T: Starting from (x, y):

1. Define $\Delta(x_2) = f(x, y) - f(x_2, y)$ if $f(x_2, y) < f(x, y)$ and $\Delta(x_2) = 0$ otherwise.
2. Take $x' = \frac{x + \int_{x_2 \in X} \Delta(x_2) x_2}{1 + \int_{x_2 \in X} \Delta(x_2)}$
3. Define $\Delta(y_2) = f(x', y_2) - f(x', y)$ if $f(x, y_2) > f(x, y)$ and $\Delta(y_2) = 0$ otherwise.
4. Take $y' = \frac{y + \int_{y_2 \in Y} \Delta(y_2) y_2}{1 + \int_{y_2 \in Y} \Delta(y_2)}$ and $T(x, y) = (x', y')$

Duality Via Minimax Theorem

- Idea: Instead of trying to enforce some of the constraints, make the program into a two player game where the new player can punish any violated constraints.
- Example: Maximize $\mathrm{c}^{\mathrm{T}}x$ subject to
  1. $Ax \leq b$
  2. $x \geq 0$
- Game: Take $f(x, y) = c^T x + y^T(b - Ax)$ where we have the constraint that $y \geq 0$ (here y wants to minimize $f(x, y)$).
- If $(Ax)_i > b_i$, y can take $y_i \to \infty$ to punish this.

Strong Duality Intuition

- Canonical primal form: Maximize $\mathrm{c}^{\mathrm{T}}x$ subject to
  1. $Ax \leq b$
  2. $x \geq 0$
- $= \max\limits_{x \geq 0} \min\limits_{y \geq 0} c^T x + y^T (b - Ax)$
- $= \min\limits_{y \geq 0} \max\limits_{x \geq 0} y^T b + (c^T - y^T A)x$
- Canonical dual form: Minimize $b^T y$ subject to
  1. $A^T y \geq c^T$
  2. $y \geq 0$
- Not quite a proof, domains of x, y aren't compact!

Slack Form Duality Intuition

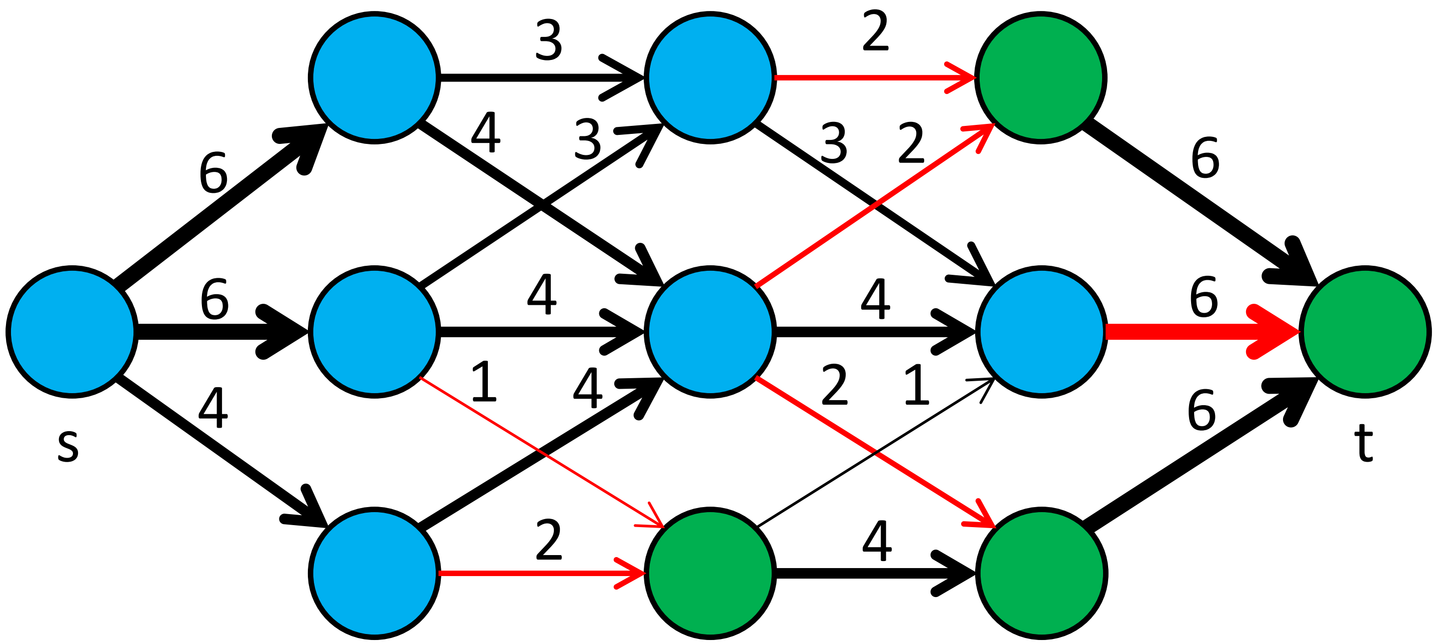
- Slack primal form: Maximize $\mathrm{c}^{\mathrm{T}}x$ subject to
  1. $Ax = b$
  2. $x \geq 0$
- $= \max_{x \geq 0} \min_{y} c^T x + y^T (b - Ax)$
- $= \min_{y} \max_{x \geq 0} y^T b + (c^T - y^T A)x$
- Slack dual form: Minimize $b^T y$ subject to
  1. $A^T y \geq c^T$
- See problem set for a true proof of strong duality.

Max-flow/Min-cut Theorem

- Classical duality example: max-flow/min-cut
- Max-flow/min-cut theorem: The maximum flow from s to t is equal to the minimum capacity across a cut separating s and t.
- Duality is a bit subtle (see problem set)

Max-flow/Min-cut Example

- Maximum flow was 15, this is matched by the minimal cut shown below:



Part III: Linear Programming as a Problem Relaxation

Convex Relaxations

- Often we want to optimize over a nonconvex set, which is very difficult.
- To obtain an approximation, we can take a convex relaxation of our set.
- Linear programming can give such convex relaxations.

Bad Example: 3-SAT solving

- Actual problem: Want each $x_i \in \{0,1\}$.
- A clause $x_i \vee x_j \vee_k$ can be re-expressed as
$$x_i + x_j + x_k \geq 1$$
- Negations can be handled with the equality $\neg x_i = 1 - x_i$
- Convex relaxation: Only require $0 \leq x_i \leq 1$
- Too relaxed: Could just take all $x_i = \frac{1}{2}$!
- Note: strengthening this gives cutting planes

Example: Maximum Matching

- Have a variable x_{ij} for each edge $(i, j) \in E(G)$
- Actual problem: Maximize $\sum_{i,j:(i,j)\in E(G)} x_{ij}$ subject to
  1. $\forall i < j: (i, j) \in E(G), x_{ij} \in \{0,1\}$
  2. $\forall i, \sum_{j<i:(j,i)\in E(G)} x_{ji} + \sum_{j>i:(i,j)\in E(G)} x_{ij} \leq 1$
- Convex relaxation: Only require $0 \leq x_{ij} \leq 1$
- Gives exact value for bipartite graphs, not in general (see problem set)