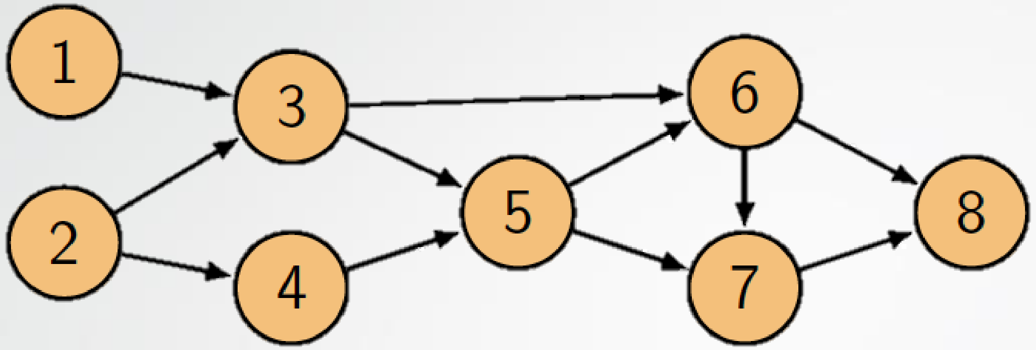
Lecture 4 – Duality in MCNF

Module 6 – Duality in Linear Programming

CS/ISyE/ECE 524

The general MCNF model



$$\min_{x \in \mathbb{R}^{|E|}} \; c^T x$$
$$\text{s.t. } \; Ax = b$$
$$p \leq x \leq q$$

Question:
Without loss of generality, we can assume $p = 0$. Why?

The Dual of the MCNF problem

$$\min_x \; c^T x \quad \text{(min)}$$
$$\text{s.t. } Ax = b \quad \text{(constraint } =\text{)}$$
$$x \le q \quad \text{(constraint } \le\text{)}$$
$$x \ge 0 \quad \text{(variable } \ge\text{)}$$

- Balance constraints (each node)
- Capacity constraints (each edge)
- Flow variables (each edge)

$$\max_{\mu,\eta} \; b^T \mu + q^T \eta \quad \text{(max)}$$
$$\text{s.t. } \mu \text{ free} \quad \text{(variable free)}$$
$$\eta \le 0 \quad \text{(variable } \le\text{)}$$
$$A^T \mu + \eta \le c \quad \text{(constraint } \le\text{)}$$

- Dual variables (each node)
- Dual variables (each edge)
- Dual constraints (each edge)

Question:
How can we interpret the various types of dual MCNF problems in the "real world"? (Planning? Max-flow?)

Longest-path problem dual

$$\max_{x} \; c^T x \quad \text{(max)} \qquad\qquad \min_{\mu} \; b^T \mu \quad \text{(min)}$$

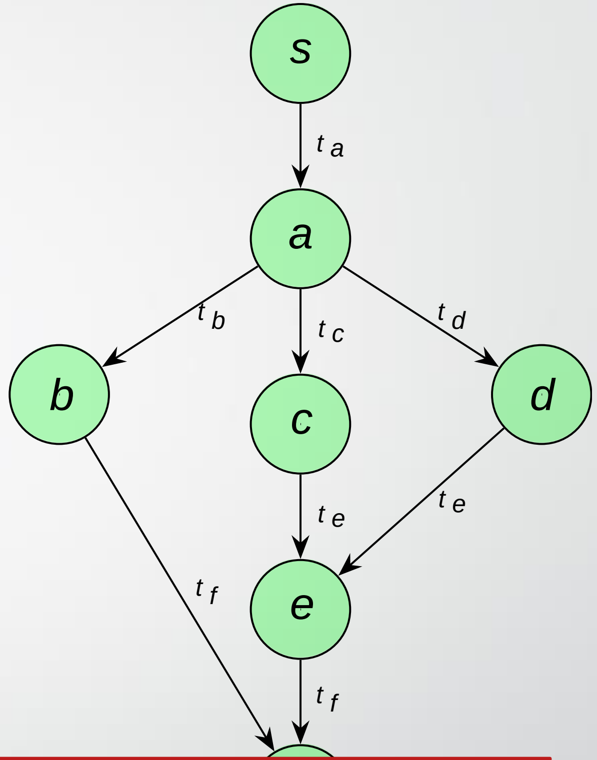
$$\text{s.t. } Ax = b \quad (\text{constraint } =) \qquad \text{s.t. } A^T \mu \geq c \quad (\text{constraint } \geq)$$

$$x \geq 0 \quad (\text{variable } \geq) \qquad\qquad \mu \text{ free} \quad (\text{variable free})$$

To interpret the dual, we'll use a transformation trick:

- Let $y_i = -\mu_i$. (the dual variable is free, so we make this substitution only for ease of understanding)
- The dual problem becomes:

$$\begin{aligned} \min_{\mu} \; & -\mu_{sink} + \mu_{source} \\ \text{s.t. } & \mu_{source} - \mu_a \geq t_a \\ & \mu_a - \mu_b \geq t_b \\ & \mu_a - \mu_c \geq t_c \\ & \ldots \end{aligned} \quad \Rightarrow \quad \begin{aligned} \min_{y} \; & y_{sink} - y_{source} \\ \text{s.t. } & y_a - y_{source} \geq t_a \\ & y_b - y_a \geq t_b \\ & y_c - y_a \geq t_c \\ & \ldots \end{aligned}$$



We can interpret this as minimizing the finishing time of the final task (sink), with constraints requiring the finishing time of subsequent tasks = previous task time + time to complete current task

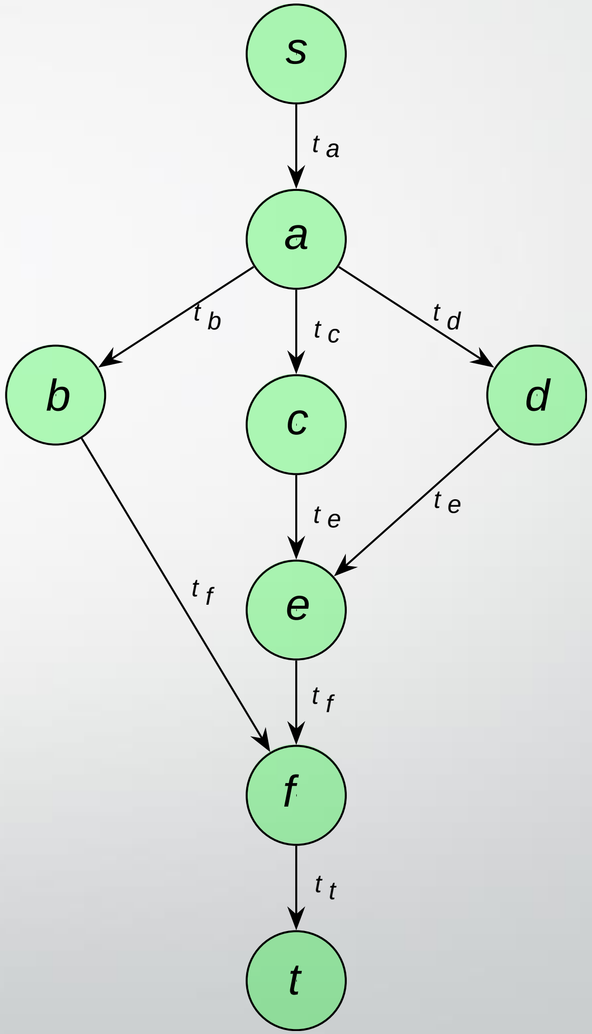
Complementary slackness in longest path

$$\max_x \; c^T x \quad \text{(max)} \qquad\qquad \min_\mu \; b^T \mu \quad \text{(min)}$$
$$\text{s.t. } Ax = b \quad \text{(constraint } =) \qquad \text{s.t. } A^T\mu \geq c \quad \text{(constraint } \geq)$$
$$x \geq 0 \quad \text{(variable } \geq) \qquad\qquad \mu \text{ free} \quad \text{(variable free)}$$

Recall the property of **complementary slackness**: at an optimal solution, either the primal (dual) variable is 0 or the dual (primal) constraint is active

- Primal constraints are *all* active in this problem, so only interesting to consider primal variables and dual constraints
- If $x_{ij} = 1$ then $y_j - y_i = c_{ij}$ (longest path corresponds to tight time constraints)
- If $y_j - y_i > c_{ij}$ then $x_{ij} = 0$ (this path has slack)

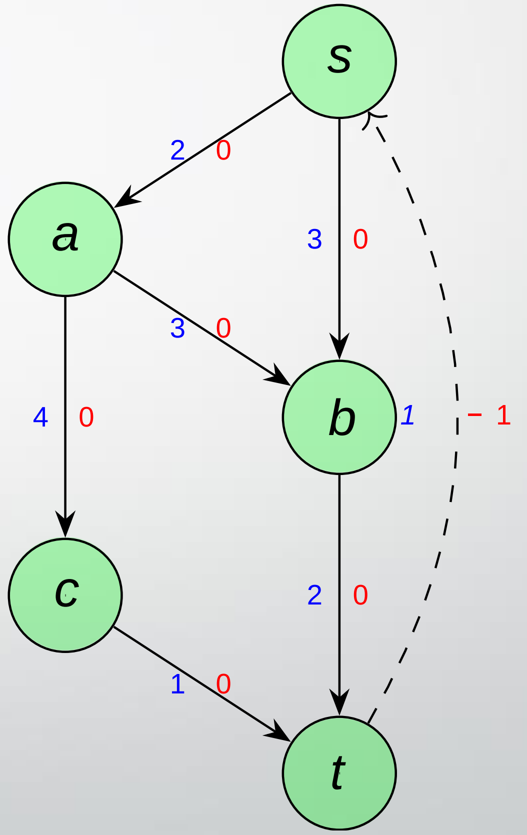
Let's revisit our longest path example



Example: Max-flow problem

$$\begin{aligned} \max_x \ & c^T x & \text{(max)} \\ \text{s.t. } & Ax = b & \text{(constraint } =) \\ & x \le q & \text{(constraint } \le) \\ & x \ge 0 & \text{(variable } \ge) \end{aligned} \qquad \begin{aligned} \min_{\mu,\eta} \ & b^T\mu + q^T\eta & \text{(min)} \\ \text{s.t. } & \mu \text{ free} & \text{(variable free)} \\ & \eta \ge 0 & \text{(variable } \ge) \\ & A^T\mu + \eta \ge c & \text{(constraint } \ge) \end{aligned}$$

(edge capacity)
(cost)



Recall the max-flow problem:

- Edges have max **capacities**
- Edges have zero **cost**, except the feedback edge which has cost -1
- Finding the max flow is equivalent to finding the minimum cost flow
- All nodes have 0 supply/demand

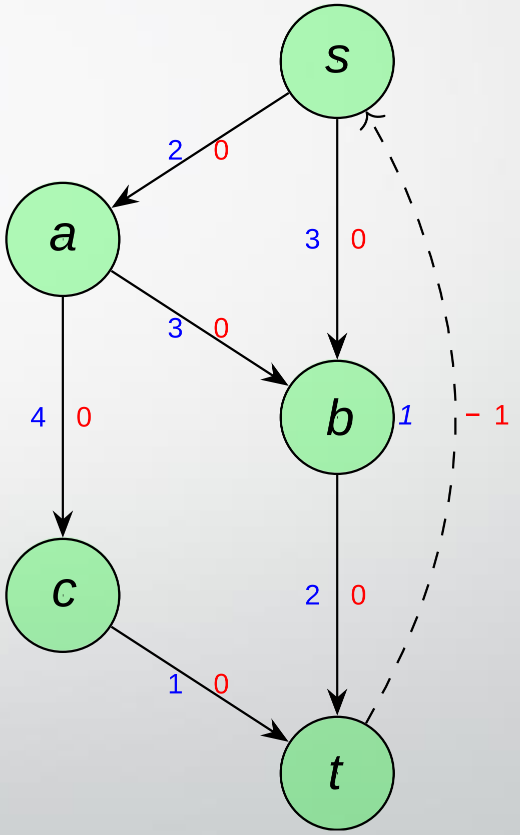
Primal of the max-flow problem

$$\max_{x} \; x_{ts}$$

$$\text{s.t.} \begin{bmatrix} 1 & 1 & 0 & 0 & 0 & 0 & -1 \\ -1 & 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & -1 & -1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & -1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & -1 & -1 & 1 \end{bmatrix} \begin{bmatrix} x_{sa} \\ x_{sb} \\ x_{ab} \\ x_{ac} \\ x_{bt} \\ x_{ct} \\ x_{ts} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

$$\begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \leq \begin{bmatrix} x_{sa} \\ x_{sb} \\ x_{ab} \\ x_{ac} \\ x_{bt} \\ x_{ct} \end{bmatrix} \leq \begin{bmatrix} 2 \\ 3 \\ 3 \\ 4 \\ 2 \\ 1 \end{bmatrix}$$

(edge capacity)
(cost)

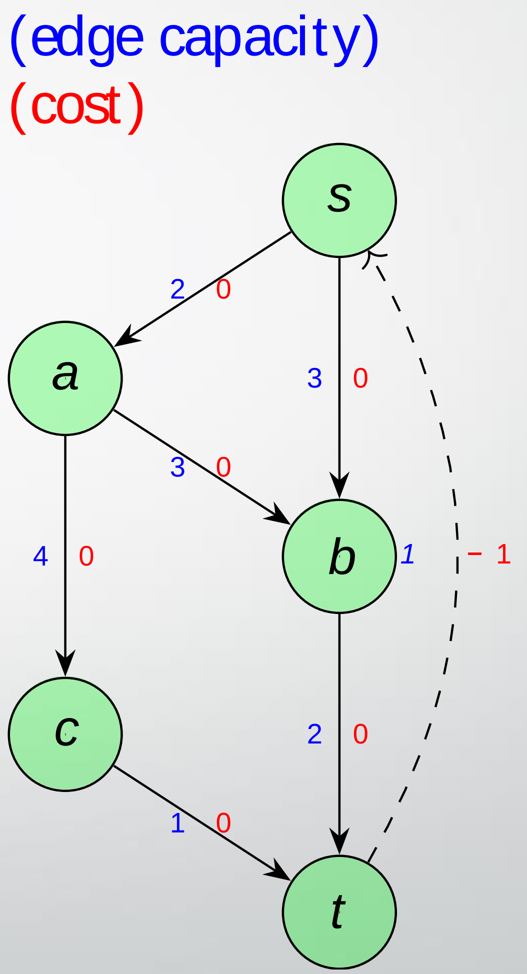


Dual of the max-flow problem

$$\min_{\lambda,\mu} \; 2\lambda_{sa} + 3\lambda_{sb} + 3\lambda_{ab} + 4\lambda_{ac} + 2\lambda_{bt} + \lambda_{ct}$$

$$\text{s.t.} \quad \begin{bmatrix} 1 & -1 & 0 & 0 & 0 \\ 1 & 0 & -1 & 0 & 0 \\ 0 & 1 & -1 & 0 & 0 \\ 0 & 1 & 0 & -1 & 0 \\ 0 & 0 & 1 & 0 & -1 \\ 0 & 0 & 0 & 1 & -1 \\ -1 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \mu_s \\ \mu_a \\ \mu_b \\ \mu_c \\ \mu_t \end{bmatrix} + \begin{bmatrix} \lambda_{sa} \\ \lambda_{sb} \\ \lambda_{ab} \\ \lambda_{ac} \\ \lambda_{bt} \\ \lambda_{ct} \\ 0 \end{bmatrix} \geq \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 1 \end{bmatrix}$$

$$\lambda_{ij} \geq 0, \qquad \mu_i \text{ free}$$



- Rearrange the constraints to isolate λ_{ij}
- Our goal is to get the λ_{ij} variables small

Reformulated dual problem

$$
\begin{aligned}
\min_{\lambda,\mu}\ & 2\lambda_{sa} + 3\lambda_{sb} + 3\lambda_{ab} + 4\lambda_{ac} + 2\lambda_{bt} + \lambda_{ct} \\
\text{s.t. } & \mu_s \geq 0 \\
& \mu_a - \mu_s \leq \lambda_{sa} \\
& \mu_b - \mu_s \leq \lambda_{sb} \\
& \mu_b - \mu_a \leq \lambda_{ab} \\
& \mu_c - \mu_a \leq \lambda_{ac} \\
& \mu_t - \mu_b \leq \lambda_{bt} \\
& \mu_t - \mu_c \leq \lambda_{ct} \\
-\mu_s + {} & \mu_t \geq 1 \\
& \lambda_{ij} \geq 0, \quad \mu_i \text{ free}
\end{aligned}
$$

Key:
If A is TU, then A^T is TU!

Challenge:
Show that *every* path from *s* to *t* (e.g. s – a – c – t) has $0 = \mu_s \leq \mu_a \leq \mu_c \leq \mu_t = 1$.

Claim: $\mu_i \in \{0,1\} \ \forall i \in N$

This means all $\lambda_{ij} \in \{0,1\} \ \forall i,j \in \mathrm{E}$ too!

- Since we're minimizing, as many λ_{ij} as possible will be 0

A really important duality result

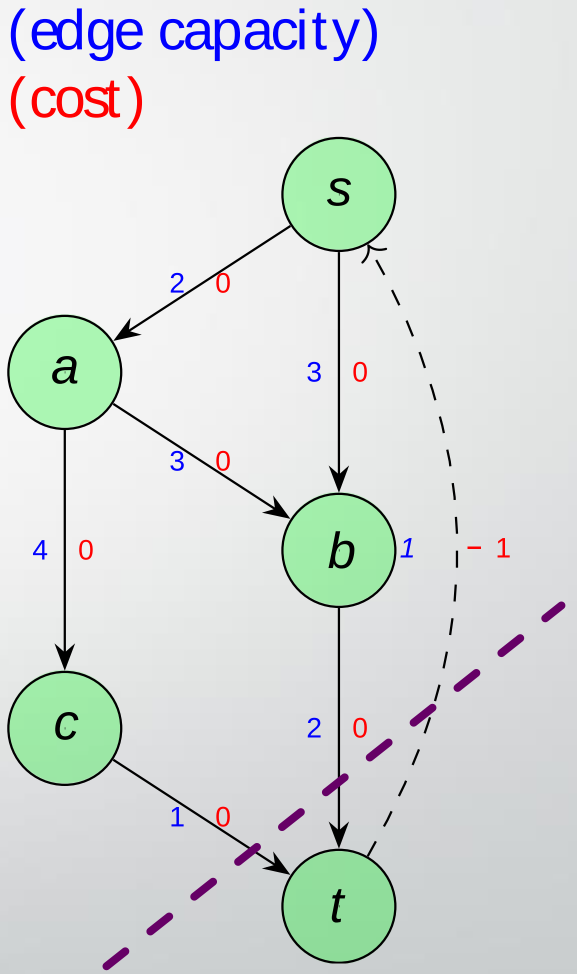
With these observations, we can re-write the dual as:

$$\min_{\lambda,\mu} \; 2\lambda_{sa} + 3\lambda_{sb} + 3\lambda_{ab} + 4\lambda_{ac} + 2\lambda_{bt} + \lambda_{ct}$$

$$\text{s.t.} \quad \text{Along each path } s \to i \to \cdots \to j \to t$$

$$\text{exactly one edge } p \to q \text{ is chosen.}$$

$$\lambda_{pq} = 1 \text{ and } \lambda_{ij} = 0 \text{ for all other edges on that path}$$



- Each path is broken by selecting edges to remove (*cut*) from the path. We choose the cut with the lowest total capacity (lowest cost).

Important (and cool) Theorem:
Max flow = Min cut

Summary of max flow

Primal problem:

- Each edge of the network has a maximum capacity
- Pick how much of the commodity flows along each edge to maximize the total amount transported from the source to the sink while obeying flow balance. The total amount is called the **max flow**.

Dual problem:

- Find a **partition** of the nodes into two subsets where the first subset includes the source and the second subset includes the sink
- Choose the partition that minimizes the sum of the capacities of all edges connecting the subsets. This total capacity is called the **min cut**.

LP Duality Module
Learning Outcomes

Now, you should be able to…

- Find lower and upper bounds on any linear program's optimal objective
- Define duality in linear programs
  - Understand the relationship between primal and dual linear programs
  - State the strong duality theorem
- Perform sensitivity analysis on a linear program using duality
  - Understand relationship between duality and sensitivity analysis
- Define complementary slackness
  - Use complementary slackness to find primal and/or dual solutions

LP Duality Module
Learning Outcomes

Now, you should be able to…

- Write down the dual for the general form of the general min-cost network flow problem as well as specifically for
  - Longest path
  - Max flow
- Describe the real-world interpretation of the dual problem for each problem type
- Give examples of the dual problem for each type of min-cost network flow problem