Lecture 7

October 14, 2024

Network Revenue Management

- Airline is planning operations for a specific day in the future
- Airline operates a set F of direct flights in its (hub-and-spoke) network
- For each flight leg $f \in F$, we know the capacity of the aircraft c_f
- The airline can offer a large number of "products" (i.e., itineraries) I:
  - each itinerary refers to an origin-destination-fare class combination
  - each itinerary i has a price r_i that is fixed
  - for each itinerary, the airline estimates the demand d_i
  - each itinerary requires a seat on several flight legs operated by the airline

Network Revenue Management

- Airline is planning operations for a specific day in the future
- Airline operates a set F of direct flights in its (hub-and-spoke) network
- For each flight leg $f \in F$, we know the capacity of the aircraft c_f
- The airline can offer a large number of "products" (i.e., itineraries) I:
  - each itinerary refers to an origin-destination-fare class combination
  - each itinerary i has a price r_i that is fixed
  - for each itinerary, the airline estimates the demand d_i
  - each itinerary requires a seat on several flight legs operated by the airline
- Requirements: $A \in \{0,1\}^{F \cdot I}$ with $A_{f,i} = 1 \Leftrightarrow$ itinerary i needs seat on flight leg f

Resource matrix A :

	Itinerary 1	Itinerary 2	...	Itinerary $\|I\|$
Flight leg 1	1	0	...	1
Flight leg 2	0	1	...	0
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
Flight leg $\|F\|$	1	1	...	0

Network Revenue Management

- Airline is planning operations for a specific day in the future
- Airline operates a set F of direct flights in its (hub-and-spoke) network
- For each flight leg $f \in F$, we know the capacity of the aircraft c_f
- The airline can offer a large number of "products" (i.e., itineraries) I:
  - each itinerary refers to an origin-destination-fare class combination
  - each itinerary i has a price r_i that is fixed
  - for each itinerary, the airline estimates the demand d_i
  - each itinerary requires a seat on several flight legs operated by the airline
- Requirements: $A \in \{0,1\}^{F \cdot I}$ with $A_{f,i} = 1 \Leftrightarrow$ itinerary i needs seat on flight leg f

Resource matrix A :

	Itinerary 1	Itinerary 2	...	Itinerary $\|I\|$
Flight leg 1	1	0	...	1
Flight leg 2	0	1	...	0
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
Flight leg $\|F\|$	1	1	...	0

- Goal: decide how many itineraries of each type to sell to maximize revenue

Network Revenue Management

- x_i: number of itineraries of type i that the airline plans to sell
- Airline Network RM problem:

$$\max_{x \in \mathbb{R}^I} \left\{ r^\mathsf{T} x : Ax \leq c,\ x \leq d \right\}$$

- $Ax \leq c$: constraints on plane capacity
- $x \leq d$: planned sales cannot exceed the demand
- In practice, would not include **all possible itineraries**

Network Revenue Management

- x_i: number of itineraries of type i that the airline plans to sell
- Airline Network RM problem:

$$\max_{x \in \mathbb{R}^I} \left\{ r^\intercal x : Ax \leq c,\ x \leq d \right\}$$

- $Ax \leq c$: constraints on plane capacity
- $x \leq d$: planned sales cannot exceed the demand
- In practice, would not include **all possible itineraries**
  - gargantuan LP
  - poor demand estimates for some itineraries

Network Revenue Management

- x_i: number of itineraries of type i that the airline plans to sell
- Airline Network RM problem:

$$\max_{x \in \mathbb{R}^I} \left\{ r^\top x : Ax \leq c, \ x \leq d \right\}$$

- $Ax \leq c$: constraints on plane capacity
- $x \leq d$: planned sales cannot exceed the demand
- In practice, would not include **all possible itineraries**
  - gargantuan LP
  - poor demand estimates for some itineraries
- To sell "exotic itineraries", use the **shadow prices for the capacity constraints**

Network Revenue Management

- x_i: number of itineraries of type i that the airline plans to sell
- Airline Network RM problem:

$$\max_{x \in \mathbb{R}^I} \left\{ r^\intercal x : Ax \le c,\ x \le d \right\}$$

- $Ax \le c$: constraints on plane capacity
- $x \le d$: planned sales cannot exceed the demand
- In practice, would not include **all possible itineraries**
  - gargantuan LP
  - poor demand estimates for some itineraries
- To sell "exotic itineraries", use the **shadow prices for the capacity constraints**
  - $p \in \mathbb{R}^F$: dual variables for capacity constraints $Ax \le c$
  - At optimality, p_f is marginal revenue lost if airline loses one seat on flight f

Network Revenue Management

- x_i: number of itineraries of type i that the airline plans to sell
- Airline Network RM problem:

$$\max_{x \in \mathbb{R}^I} \left\{ r^\intercal x : Ax \leq c,\ x \leq d \right\}$$

- $Ax \leq c$: constraints on plane capacity
- $x \leq d$: planned sales cannot exceed the demand
- In practice, would not include **all possible itineraries**
  - gargantuan LP
  - poor demand estimates for some itineraries
- To sell "exotic itineraries", use the **shadow prices for the capacity constraints**
  - $p \in \mathbb{R}^F$: dual variables for capacity constraints $Ax \leq c$
  - At optimality, p_f is marginal revenue lost if airline loses one seat on flight f
  - For an "exotic" itinerary that requires seats on several flights $f \in E$, the **minimum price to charge** is given by the sum of the shadow prices, $\sum_{f \in E} p_f$

Network Revenue Management

- x_i: number of itineraries of type i that the airline plans to sell
- Airline Network RM problem:

$$\max_{x \in \mathbb{R}^I} \left\{ r^\intercal x : Ax \leq c,\ x \leq d \right\}$$

- $Ax \leq c$: constraints on plane capacity
- $x \leq d$: planned sales cannot exceed the demand
- In practice, would not include **all possible itineraries**
  - gargantuan LP
  - poor demand estimates for some itineraries
- To sell "exotic itineraries", use the **shadow prices for the capacity constraints**
  - $p \in \mathbb{R}^F$: dual variables for capacity constraints $Ax \leq c$
  - At optimality, p_f is marginal revenue lost if airline loses one seat on flight f
  - For an "exotic" itinerary that requires seats on several flights $f \in E$, the **minimum price to charge** is given by the sum of the shadow prices, $\sum_{f \in E} p_f$
- **Bid-price heuristic** in network revenue management
- Broader principle of how to **price "products" through resource usage/cost**

Discrete Optimization

Today, we consider optimization problems with discrete variables:

$$\begin{aligned} \min\ & c^T x + d^T y \\ & Ax + By = b \\ & x, y \geq 0 \\ & x \text{ integer} \end{aligned}$$

This is called a **mixed integer programming** (MIP) problem

Without continuous variables y, it is called an **integer program** (IP)

If instead of $x \in \mathbb{Z}^n$ we have $x \in \{0,1\}^n$: **binary optimization** problem

Very powerful modeling paradigm

Example: Knapsack

- n items
- Item j has weight w_j and reward r_j
- Have a bound K on the weight that can be carried in the knapsack
- Want to select items to maximize the total value

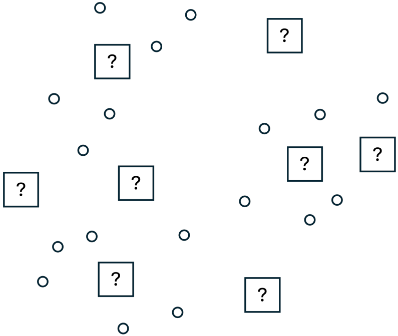
Example: Knapsack

- n items
- Item j has weight w_j and reward r_j
- Have a bound K on the weight that can be carried in the knapsack
- Want to select items to maximize the total value

$$\begin{aligned} \text{maximize} \; & \sum_{j=1}^{n} r_j x_j \\ \text{subject to} \; & \sum_{j=1}^{n} w_j x_j \leq K \\ & x_j \in \{0,1\}, \quad j = 1, \ldots, n. \end{aligned}$$

Example: Facility Location

- n potential locations to open facilities
- Cost c_j for opening a facility at location j
- m clients who need service
- Cost d_{ij} for serving client i from facility j
- Smallest cost for opening facilities while serving all clients



Example: Facility Location

- n potential locations to open facilities
- Cost c_j for opening a facility at location j
- m clients who need service
- Cost d_{ij} for serving client i from facility j
- Smallest cost for opening facilities while serving all clients

$$
\begin{aligned}
\min \; & \sum_{j=1}^{n} c_j y_j + \sum_{i=1}^{m} \sum_{j=1}^{n} d_{ij} x_{ij} \\
& \sum_{j=1}^{n} x_{ij} = 1, \quad \forall\, i \\
& x_{ij} \leq y_j, \qquad \forall i,\ \forall j \\
& x_{ij}, y_j \in \{0, 1\}
\end{aligned}
$$

Example: Facility Location

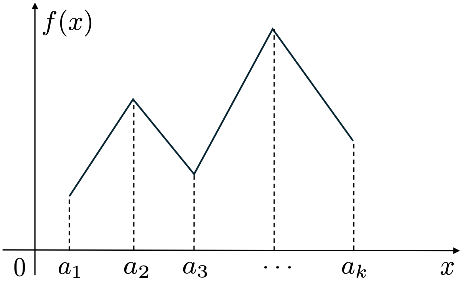
- n potential locations to open facilities
- Cost c_j for opening a facility at location j
- m clients who need service
- Cost d_{ij} for serving client i from facility j
- Smallest cost for opening facilities while serving all clients

$$
\begin{aligned}
\min \ & \sum_{j=1}^{n} c_j y_j + \sum_{i=1}^{m} \sum_{j=1}^{n} d_{ij} x_{ij} \\
& \sum_{j=1}^{n} x_{ij} = 1, \quad \forall\, i \\
& x_{ij} \le y_j, \qquad \forall i,\ \forall j \\
& x_{ij}, y_j \in \{0,1\}
\end{aligned}
$$

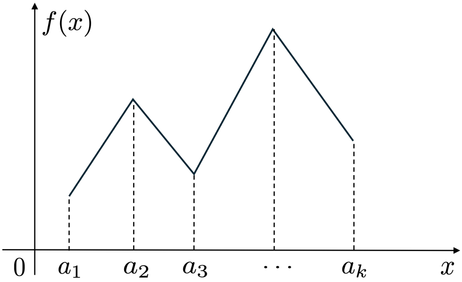
$$
\begin{aligned}
\min \ & \sum_{j=1}^{n} c_j y_j + \sum_{i=1}^{m} \sum_{j=1}^{n} d_{ij} x_{ij} \\
& \sum_{j=1}^{n} x_{ij} = 1, \quad \forall\, i \\
& \sum_{i=1}^{m} x_{ij} \le m y_j, \quad \forall j \\
& x_{ij}, y_j \in \{0,1\}.
\end{aligned}
$$

Which formulation is "better"?

Example: Piecewise Linear Cost



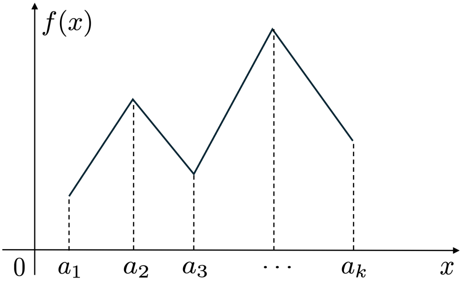
Example: Piecewise Linear Cost



- Idea: $x = \sum_{i=1}^{k} \lambda_i a_i$
- Cost: $\sum_{i=1}^{k} \lambda_i f(a_i)$
- How to impose adjacency?

$$x = \lambda_i a_i + \lambda_{i+1} a_{i+1}$$

Example: Piecewise Linear Cost



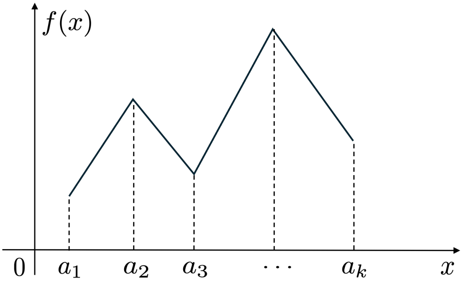
- Idea: $x = \sum_{i=1}^{k} \lambda_i a_i$
- Cost: $\sum_{i=1}^{k} \lambda_i f(a_i)$
- How to impose adjacency?

$$x = \lambda_i a_i + \lambda_{i+1} a_{i+1}$$

- New binary variables y_i to impose:

$$y_j = 1 \;\Rightarrow\; \lambda_i = 0 \text{ for } i \notin \{j, j+1\}$$

Example: Piecewise Linear Cost



- Idea: $x = \sum_{i=1}^{k} \lambda_i a_i$
- Cost: $\sum_{i=1}^{k} \lambda_i f(a_i)$
- How to impose adjacency?

$$x = \lambda_i a_i + \lambda_{i+1} a_{i+1}$$

- New binary variables y_i to impose:

$$y_j = 1 \;\Rightarrow\; \lambda_i = 0 \text{ for } i \notin \{j, j+1\}$$

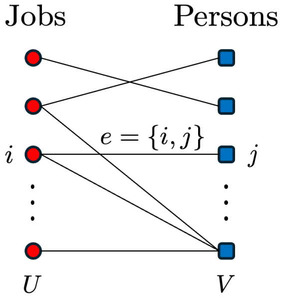
$$\begin{aligned}
\sum_{i=1}^{k} \lambda_i &= 1, \\
\lambda_1 &\leq y_1, \\
\lambda_i &\leq y_{i-1} + y_i, \;\; i = 2, \ldots, k-1, \\
\lambda_k &\leq y_{k-1}, \\
\sum_{i=1}^{k-1} y_i &= 1, \\
\lambda_i &\geq 0, \\
y_i &\in \{0, 1\}, \;\; \forall i.
\end{aligned}$$

Example: Matching Problems

- Set U of jobs/tasks to complete; set V of persons available to work
- Each task assigned to at most one person; a person can only complete some tasks
- Reward w_{ij} if task $i \in U$ completed by person $j \in V$

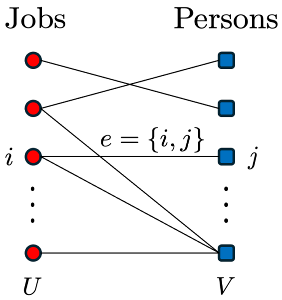
Example: Matching Problems

- Set U of jobs/tasks to complete; set V of persons available to work
- Each task assigned to at most one person; a person can only complete some tasks
- Reward w_{ij} if task $i \in U$ completed by person $j \in V$
- Graph representation $G = (\mathcal{N}, \mathcal{E})$
- $e \equiv \{i, j\} \in \mathcal{E}$ indicates $j \in V$ can complete task $i \in U$



Example: Matching Problems

- Set U of jobs/tasks to complete; set V of persons available to work
- Each task assigned to at most one person; a person can only complete some tasks
- Reward w_{ij} if task $i \in U$ completed by person $j \in V$
- Graph representation $G = (\mathcal{N}, \mathcal{E})$
- $e \equiv \{i, j\} \in \mathcal{E}$ indicates $j \in V$ can complete task $i \in U$



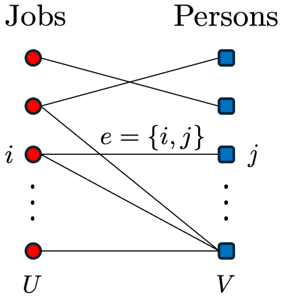
$x_e \in \{0, 1\}$: whether edge selected

$$\begin{aligned} \text{maximize} \ & \sum_{e \in E} w_e x_e \\ & \sum_{e \in \delta(i)} x_e \leq 1, \quad \forall\, i \in N, \\ & x_e \in \{0, 1\}, \end{aligned}$$

$\delta(i) := \{j : \{i, j\} \in \mathcal{E}\}$: all neighbors of i

Example: Matching Problems

- Set U of jobs/tasks to complete; set V of persons available to work
- Each task assigned to at most one person; a person can only complete some tasks
- Reward w_{ij} if task $i \in U$ completed by person $j \in V$
- Graph representation $G = (\mathcal{N}, \mathcal{E})$
- $e \equiv \{i, j\} \in \mathcal{E}$ indicates $j \in V$ can complete task $i \in U$



$x_e \in \{0, 1\}$: whether edge selected

$$\text{maximize} \sum_{e \in E} w_e x_e$$

$$\sum_{e \in \delta(i)} x_e \leq 1, \quad \forall\, i \in N,$$

$$x_e \in \{0, 1\},$$

$\delta(i) := \{j : \{i, j\} \in \mathcal{E}\}$: all neighbors of i

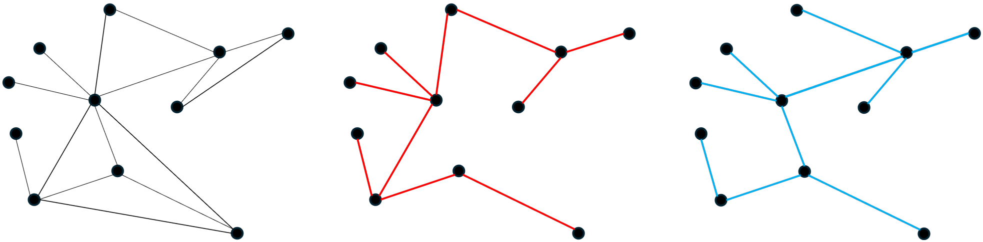
Many variations: minimize cost, require jobs completed, perfect matching, ...

Example: Minimum Spanning Tree

- Given an undirected graph $G = (\mathcal{N}, \mathcal{E})$; $|\mathcal{N}| = n$, $|\mathcal{E}| = m$
- Edge $e \in \mathcal{E}$ has associated cost c_e
- Find **minimum spanning tree** (MST)
  (subset of edges that connect all nodes in $\mathcal{N}$ at minimum cost)

Example: Minimum Spanning Tree

- Given an undirected graph $G = (\mathcal{N}, \mathcal{E})$; $|\mathcal{N}| = n$, $|\mathcal{E}| = m$
- Edge $e \in \mathcal{E}$ has associated cost c_e
- Find **minimum spanning tree** (MST)
  (subset of edges that connect all nodes in $\mathcal{N}$ at minimum cost)



Example: Minimum Spanning Tree

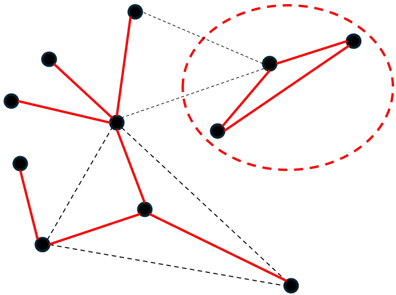
- Given an undirected graph $G = (\mathcal{N}, \mathcal{E})$; $|\mathcal{N}| = n$, $|\mathcal{E}| = m$
- Edge $e \in \mathcal{E}$ has associated cost c_e
- Find **minimum spanning tree** (MST)
  (subset of edges that connect all nodes in $\mathcal{N}$ at minimum cost)

$$
\begin{aligned}
\text{min} \quad & \sum_{e \in \mathcal{E}} c_e x_e \\
& x_e \in \{0, 1\} \\
\textbf{(Connectivity)} \quad & \sum_{e \in \mathcal{E}} x_e = n - 1 \\
\textbf{(Cutset)} \quad & \sum_{e \in \delta(S)} x_e \geq 1, \ S \subset \mathcal{N}, S \neq \emptyset
\end{aligned}
$$

Example: Minimum Spanning Tree

- Given an undirected graph $G = (\mathcal{N}, \mathcal{E})$; $|\mathcal{N}| = n$, $|\mathcal{E}| = m$
- Edge $e \in \mathcal{E}$ has associated cost c_e
- Find **minimum spanning tree** (MST)
  (subset of edges that connect all nodes in $\mathcal{N}$ at minimum cost)

$$\begin{aligned} \min \quad & \sum_{e \in \mathcal{E}} c_e x_e \\ & x_e \in \{0, 1\} \\ \textbf{(Connectivity)} \quad & \sum_{e \in \mathcal{E}} x_e = n - 1 \\ \textbf{(Cutset)} \quad & \sum_{e \in \delta(S)} x_e \geq 1, \;\; S \subset \mathcal{N}, S \neq \emptyset \end{aligned}$$



Example: Minimum Spanning Tree

- Given an undirected graph $G = (\mathcal{N}, \mathcal{E})$; $|\mathcal{N}| = n$, $|\mathcal{E}| = m$
- Edge $e \in \mathcal{E}$ has associated cost c_e
- Find **minimum spanning tree** (MST)
  (subset of edges that connect all nodes in $\mathcal{N}$ at minimum cost)

$$\min \sum_{e \in \mathcal{E}} c_e x_e$$

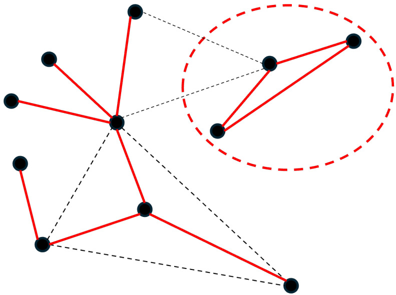
$$x_e \in \{0, 1\}$$

$$\textbf{(Connectivity)} \quad \sum_{e \in \mathcal{E}} x_e = n - 1$$

$$\textbf{(Cutset)} \quad \sum_{e \in \delta(S)} x_e \geq 1, \ S \subset \mathcal{N}, S \neq \emptyset$$

... or ...

$$\textbf{(Subtour-elimination)} \quad \sum_{e \in \mathcal{E}(S)} x_e \leq |S| - 1, \ S \subset \mathcal{N}, S \neq \emptyset$$



Example: Minimum Spanning Tree

- Given an undirected graph $G = (\mathcal{N}, \mathcal{E})$; $|\mathcal{N}| = n$, $|\mathcal{E}| = m$
- Edge $e \in \mathcal{E}$ has associated cost c_e
- Find **minimum spanning tree** (MST)
  (subset of edges that connect all nodes in $\mathcal{N}$ at minimum cost)

$$\min \sum_{e \in \mathcal{E}} c_e x_e$$

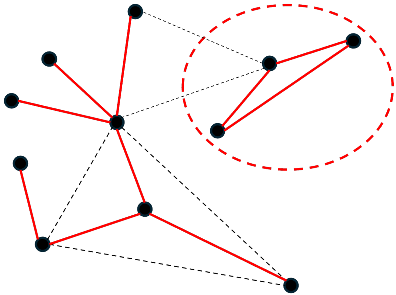
$$x_e \in \{0, 1\}$$

(Connectivity) $$\sum_{e \in \mathcal{E}} x_e = n - 1$$

(Cutset) $$\sum_{e \in \delta(S)} x_e \geq 1, \ S \subset \mathcal{N}, S \neq \emptyset$$

... or ...

(Subtour-elimination) $$\sum_{e \in \mathcal{E}(S)} x_e \leq |S| - 1, \ S \subset \mathcal{N}, S \neq \emptyset$$



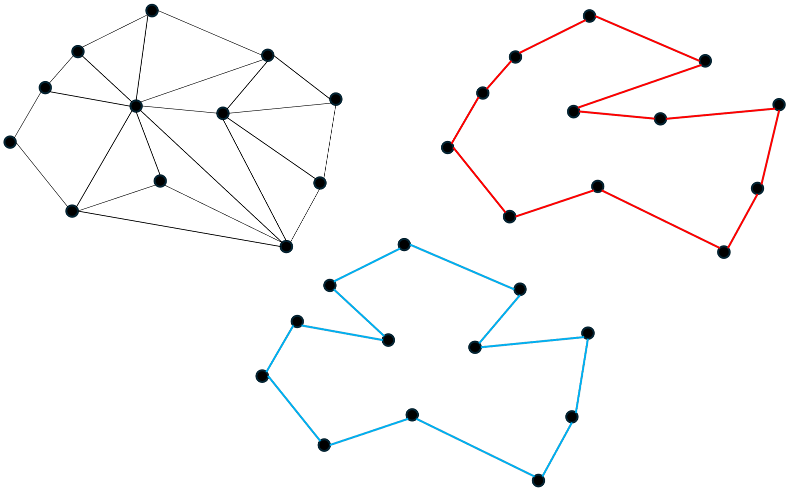
Again exponentially-sized formulations! *Any preference between them?*

Example: Traveling Salesperson Problem (TSP)

- Given an undirected graph $G = (\mathcal{N}, \mathcal{E})$; $|\mathcal{N}| = n$, $|\mathcal{E}| = m$
- Edge $e \in \mathcal{E}$ has associated cost c_e
- Find a **tour** *(cycle that visits each node exactly once)* with minimum cost

Example: Traveling Salesperson Problem (TSP)

- Given an undirected graph $G = (\mathcal{N}, \mathcal{E})$; $|\mathcal{N}| = n$, $|\mathcal{E}| = m$
- Edge $e \in \mathcal{E}$ has associated cost c_e
- Find a **tour** *(cycle that visits each node exactly once)* with minimum cost



Example: Traveling Salesperson Problem (TSP)

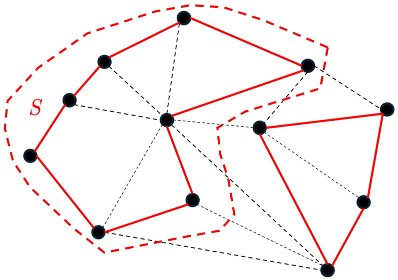
- Given an undirected graph $G = (\mathcal{N}, \mathcal{E})$; $|\mathcal{N}| = n$, $|\mathcal{E}| = m$
- Edge $e \in \mathcal{E}$ has associated cost c_e
- Find a **tour** *(cycle that visits each node exactly once)* with minimum cost

$$
\begin{aligned}
& \min \sum_{e \in \mathcal{E}} c_e x_e \\
& x_e \in \{0, 1\} \\
\textbf{(Connectivity)} \quad & \sum_{e \in \delta(\{i\})} x_e = 2, \forall i \in N \\
\textbf{(Cutset)} \quad & \sum_{e \in \delta(S)} x_e \geq 2, \forall S \subset N, S \neq \emptyset
\end{aligned}
$$

Example: Traveling Salesperson Problem (TSP)

- Given an undirected graph $G = (\mathcal{N}, \mathcal{E})$; $|\mathcal{N}| = n$, $|\mathcal{E}| = m$
- Edge $e \in \mathcal{E}$ has associated cost c_e
- Find a **tour** *(cycle that visits each node exactly once)* with minimum cost

$$
\begin{aligned}
\min \quad & \sum_{e \in \mathcal{E}} c_e x_e \\
& x_e \in \{0, 1\} \\
\textbf{(Connectivity)} \quad & \sum_{e \in \delta(\{i\})} x_e = 2, \forall i \in N \\
\textbf{(Cutset)} \quad & \sum_{e \in \delta(S)} x_e \geq 2, \forall S \subset N, S \neq \emptyset
\end{aligned}
$$



Example: Traveling Salesperson Problem (TSP)

- Given an undirected graph $G = (\mathcal{N}, \mathcal{E})$; $|\mathcal{N}| = n$, $|\mathcal{E}| = m$
- Edge $e \in \mathcal{E}$ has associated cost c_e
- Find a **tour** *(cycle that visits each node exactly once)* with minimum cost

$$\min \sum_{e \in \mathcal{E}} c_e x_e$$

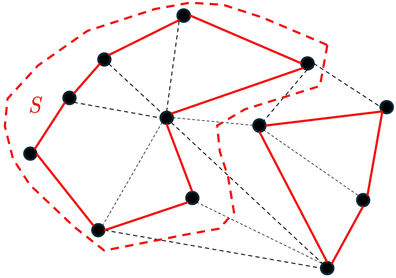
$$x_e \in \{0, 1\}$$

(Connectivity) $$\sum_{e \in \delta(\{i\})} x_e = 2, \forall i \in N$$

(Cutset) $$\sum_{e \in \delta(S)} x_e \geq 2, \forall S \subset N, S \neq \emptyset$$

... or ...

(Subtour-elimination) $$\sum_{e \in \mathcal{E}(S)} x_e \leq |S| - 1, \forall S \subset N, S \neq \emptyset$$



Example: Traveling Salesperson Problem (TSP)

- Given an undirected graph $G = (\mathcal{N}, \mathcal{E})$; $|\mathcal{N}| = n$, $|\mathcal{E}| = m$
- Edge $e \in \mathcal{E}$ has associated cost c_e
- Find a **tour** *(cycle that visits each node exactly once)* with minimum cost

$$\min \sum_{e \in \mathcal{E}} c_e x_e$$

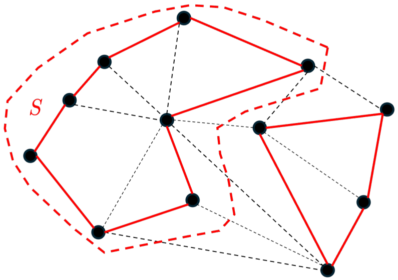
$$x_e \in \{0, 1\}$$

(Connectivity) $$\sum_{e \in \delta(\{i\})} x_e = 2, \forall i \in N$$

(Cutset) $$\sum_{e \in \delta(S)} x_e \geq 2, \forall S \subset N, S \neq \emptyset$$

... or ...

(Subtour-elimination) $$\sum_{e \in \mathcal{E}(S)} x_e \leq |S| - 1, \forall S \subset N, S \neq \emptyset$$



Again exponentially-sized formulations! *Any preference between them?*

Bad News First

Bad News First

Example. The optimal solution is the following IP **does not exist**:

$$
\begin{aligned}
\sup_{x,y}\; & x + \sqrt{2}y \\
& x + \sqrt{2}y \leq \frac{1}{2} \\
& x, y \in \mathbb{Z}.
\end{aligned}
$$

Bad News First

Example. The optimal solution is the following IP **does not exist**:

$$\begin{aligned} \sup_{x,y} \ & x + \sqrt{2}y \\ & x + \sqrt{2}y \leq \frac{1}{2} \\ & x, y \in \mathbb{Z}. \end{aligned}$$

Example. Consider the following pair of optimization programs:

$$(\mathscr{P}) \ \min_{x \geq 0} \ x \qquad\qquad (\mathscr{D}) \ \max_{p} \ p$$

$$2x = 1 \qquad\qquad 2p \leq 1$$

Bad News First

Example. The optimal solution is the following IP **does not exist**:

$$\sup_{x,y} x + \sqrt{2}y$$
$$x + \sqrt{2}y \leq \frac{1}{2}$$
$$x, y \in \mathbb{Z}.$$

Example. Consider the following pair of optimization programs:

$$(\mathscr{P})\ \min_{x \geq 0} x \qquad\qquad (\mathscr{D})\ \max_{p} p$$
$$2x = 1 \qquad\qquad 2p \leq 1$$

- $x, p \in \mathbb{R} \Rightarrow$ this is a primal-dual pair; optimal value $\frac{1}{2}$ by strong duality

Bad News First

Example. The optimal solution is the following IP **does not exist**:

$$\sup_{x,y} \; x + \sqrt{2}y$$
$$x + \sqrt{2}y \leq \frac{1}{2}$$
$$x, y \in \mathbb{Z}.$$

Example. Consider the following pair of optimization programs:

$$(\mathscr{P}) \;\; \min_{x \geq 0} \; x \qquad\qquad (\mathscr{D}) \;\; \max_{p} \; p$$
$$2x = 1 \qquad\qquad 2p \leq 1$$

- $x, p \in \mathbb{R} \Rightarrow$ this is a primal-dual pair; optimal value $\frac{1}{2}$ by strong duality
- $x, p \in \mathbb{Z} \Rightarrow (\mathscr{P})$ infeasible, $(\mathscr{D})$ has optimal value 0.

  Strong duality does not hold in IPs

Bad News First

Unfortunately, (M)IPs are **significantly harder** than LPs

Theorem

Given a matrix $A \in \mathbb{Q}^{m \times n}$ and a vector $b \in \mathbb{Q}^m$, the problem: "does $Ax \leq b$ have an integral solution x" is **NP-complete**.

- IP "feasibility problem" is already in the hardest class of problems in NP

Bad News First

Unfortunately, (M)IPs are **significantly harder** than LPs

Theorem

Given a matrix $A \in \mathbb{Q}^{m \times n}$ and a vector $b \in \mathbb{Q}^m$, the problem: "does $Ax \leq b$ have an integral solution x" is **NP-complete**.

- IP "feasibility problem" is already in the hardest class of problems in NP
- Despite this, substantial body of theory and scalable algorithms exist for IPs
- We will focus on optimization problems with **rational entries**:
  $A \in \mathbb{Q}^{m \times n}, b \in \mathbb{Q}^m, c \in \mathbb{Q}^n$ (in fact, often **integer**)
- We assume that the feasible set is bounded

Lower Bounds Again

Same question as in LP: *how can we find a good lower bound?*

Lower Bounds Again

Same question as in LP: *how can we find a good lower bound?*

LP Relaxation for Facility Location IP

Recall the **two** formulations of the Facility Location Problem

(FL)

$$\sum_{j=1}^{n} x_{ij} = 1, \quad i = 1, \ldots, m$$

$$x_{ij} \leq y_j, \quad i = 1, \ldots, m, \;\; j = 1, \ldots, n$$

$$x_{ij}, y_j \in \{0, 1\}$$

(AFL)

$$\sum_{j=1}^{n} x_{ij} = 1, \quad i = 1, \ldots, m$$

$$\sum_{i=1}^{m} x_{ij} \leq m y_j, \quad j = 1, \ldots, n$$

$$x_{ij}, y_j \in \{0, 1\}.$$

- $P_{\mathbf{FL}}, P_{\mathbf{AFL}}$: feasible sets for LP relaxations

LP Relaxation for Facility Location IP

Recall the **two** formulations of the Facility Location Problem

(FL)

$$\sum_{j=1}^{n} x_{ij} = 1, \quad i = 1, \dots, m$$

$$x_{ij} \le y_j, \quad i = 1, \dots, m, \ \ j = 1, \dots, n$$

$$x_{ij}, y_j \in \{0, 1\}$$

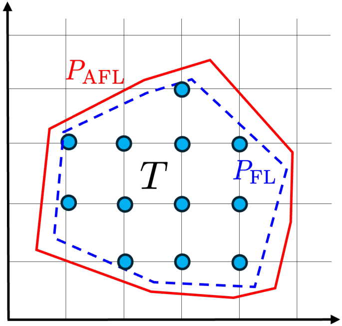
(AFL)

$$\sum_{j=1}^{n} x_{ij} = 1, \quad i = 1, \dots, m$$

$$\sum_{i=1}^{m} x_{ij} \le m y_j, \quad j = 1, \dots, n$$

$$x_{ij}, y_j \in \{0, 1\}.$$

- $P_{\mathbf{FL}}, P_{\mathbf{AFL}}$: feasible sets for LP relaxations
- $P_{\mathrm{FL}} \subseteq P_{\mathrm{AFL}}$ and can have **strict** inclusion
- **(FL) provides better lower bound than (AFL)**
- **Same IP feasible set, different LP feasible set!**



LP Relaxation for Minimum Spanning Tree Problem

(Cutset MST)

$$
\begin{aligned}
&\sum_{e \in \mathcal{E}} x_e = n - 1, \\
&\sum_{e \in \delta(S)} x_e \geq 1, \quad S \subset \mathcal{N}, S \neq \emptyset \\
&x_e \in \{0, 1\}
\end{aligned}
$$

(Subtour-elimination MST)

$$
\begin{aligned}
&\sum_{e \in \mathcal{E}} x_e = n - 1, \\
&\sum_{e \in \mathcal{E}(S)} x_e \leq |S| - 1, \quad S \subset \mathcal{N}, S \neq \emptyset, \\
&x_e \in \{0, 1\}.
\end{aligned}
$$

- $P_{\mathbf{cut}}, P_{\mathbf{sub}}$: feasible sets for LP relaxations

LP Relaxation for Minimum Spanning Tree Problem

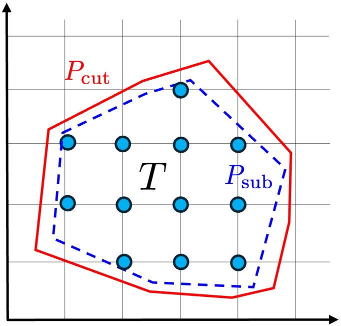
(Cutset MST)

$$\sum_{e \in \mathcal{E}} x_e = n - 1,$$
$$\sum_{e \in \delta(S)} x_e \geq 1, \quad S \subset \mathcal{N}, S \neq \emptyset$$
$$x_e \in \{0, 1\}$$

(Subtour-elimination MST)

$$\sum_{e \in \mathcal{E}} x_e = n - 1,$$
$$\sum_{e \in \mathcal{E}(S)} x_e \leq |S| - 1, \quad S \subset \mathcal{N}, S \neq \emptyset,$$
$$x_e \in \{0, 1\}.$$

- $P_{\mathbf{cut}}, P_{\mathbf{sub}}$: feasible sets for LP relaxations
- $P_{\text{sub}} \subseteq P_{\text{cut}}$ and can have **strict** inclusion *(Proof in the notes)*
- **(SUB) provides better lower bound than (CUT)**
- **Same IP feasible set, different LP feasible set!**



LP Relaxation for Traveling Salesperson Problem (TSP)

(Cutset TSP)

$$\sum_{e \in \delta(\{i\})} x_e = 2, \forall i \in N$$
$$\sum_{e \in \delta(S)} x_e \geq 2, \forall S \subset N, S \neq \emptyset$$

(Subtour-elimination TSP)

$$\sum_{e \in \delta(\{i\})} x_e = 2, \forall i \in N$$
$$\sum_{e \in \mathcal{E}(S)} x_e \leq |S| - 1, \forall S \subset N, S \neq \emptyset.$$

- $P_{\mathbf{cut}}, P_{\mathbf{sub}}$: feasible sets for LP relaxations

LP Relaxation for Traveling Salesperson Problem (TSP)

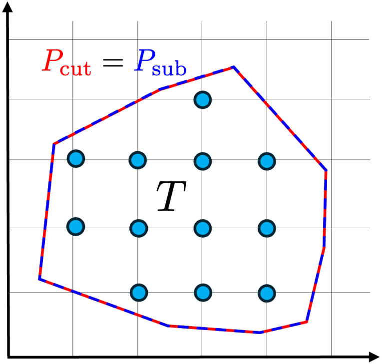
(Cutset TSP)

$$\sum_{e \in \delta(\{i\})} x_e = 2, \forall i \in N$$
$$\sum_{e \in \delta(S)} x_e \geq 2, \forall S \subset N, S \neq \emptyset$$

(Subtour-elimination TSP)

$$\sum_{e \in \delta(\{i\})} x_e = 2, \forall i \in N$$
$$\sum_{e \in \mathcal{E}(S)} x_e \leq |S| - 1, \forall S \subset N, S \neq \emptyset.$$

- $P_{\mathbf{cut}}, P_{\mathbf{sub}}$: feasible sets for LP relaxations
- $P_{\text{sub}} = P_{\text{cut}}$



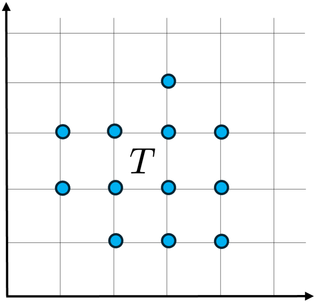
Strength of IP Formulation

- Different formulations of the same IP can result in **different LP relaxations**

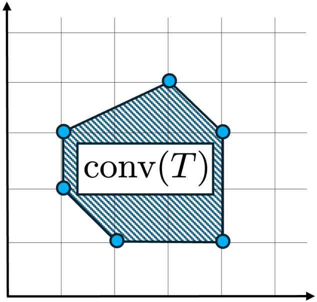
- *What is an "ideal" formulation?*

Strength of IP Formulation

- T : all feasible points to an IP and $\mathrm{conv}(T)$ is their convex hull
  - T finite because we assumed bounded feasible set
  - $\mathrm{conv}(T)$ is a polyhedron



(a)



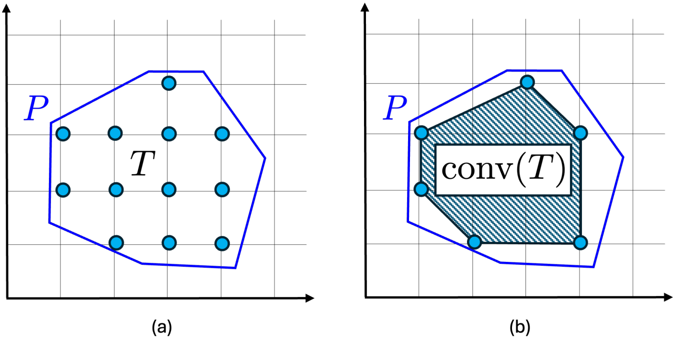
(b)

Strength of IP Formulation

- T : all feasible points to an IP and $\mathrm{conv}\,(T)$ is their convex hull
  - T finite because we assumed bounded feasible set
  - $\mathrm{conv}\,(T)$ is a polyhedron

- If P is the feasible region of an LP relaxation to our IP, then

$$T \subseteq \mathrm{conv}\,(T) \subseteq P.$$



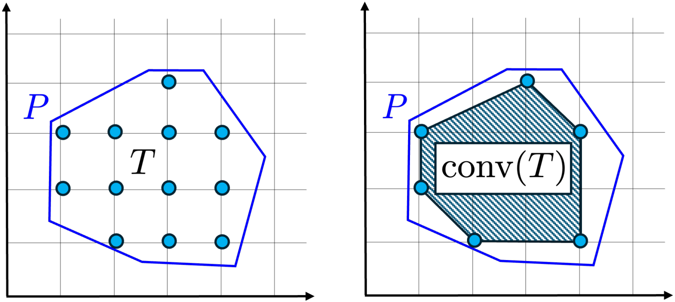
Strength of IP Formulation

- T : all feasible points to an IP and $\text{conv}(T)$ is their convex hull
  - T finite because we assumed bounded feasible set
  - $\text{conv}(T)$ is a polyhedron

- If P is the feasible region of an LP relaxation to our IP, then

$$T \subseteq \text{conv}(T) \subseteq P.$$

- **Ideal** LP relaxation would have $P = \text{conv}(T)$



(a) (b)

Strength of IP Formulation

- T : all feasible points to an IP and $\mathrm{conv}\,(T)$ is their convex hull
  - T finite because we assumed bounded feasible set
  - $\mathrm{conv}\,(T)$ is a polyhedron

- If P is the feasible region of an LP relaxation to our IP, then

$$T \subseteq \mathrm{conv}\,(T) \subseteq P.$$

- **Ideal** LP relaxation would have $P = \mathrm{conv}\,(T)$

Key take-aways:

- Quality of IP formulation : how closely its LP relaxation approximates $\mathrm{conv}\,(T)$
- Formulation A is better than formulation B for some IP if $P_A \subset P_B$
- **Constraints** play a more subtle role in IPs than in LPs
  - Adding valid constraints for T that cut off fractional points from P is very useful!
  - More constraints not necessarily worse in IP!

Ideal Formulations

Setup:

- $T = \{x \in \mathbb{Z}^n_+ \mid Ax \leq b\}$: feasible set for an IP with $A \in \mathbb{Z}^{m \times n}$ and $b \in \mathbb{Z}^m$
- P : feasible set of its LP relaxation, $P = \{x \in \mathbb{R}^n_+ \mid Ax \leq b\}$.
- **Goal:** conditions on A so that P **is integral**, i.e., $P = \mathrm{conv}(T)$

Can anyone recall Cramer's rule?

(Total) Unimodularity

Definition

1. Matrix $A \in \mathbb{Z}^{m \times n}$ of full row rank is **unimodular** if the $\det(A_B) \in \{1, -1\}$ for every basis B.
2. Matrix $A \in \mathbb{Z}^{m \times n}$ is **totally unimodular** if the determinant of each square submatrix of A is 0, 1, or -1.

- **Unimodularity** allows handling standard form $\{x \in \mathbb{Z}^n_+ \mid Ax = b\}$
- **Total Unimodularity (TU)** allows handling inequality form $\{x \in \mathbb{Z}^n_+ \mid Ax \leq b\}$

(Total) Unimodularity

Definition

1. Matrix $A \in \mathbb{Z}^{m \times n}$ of full row rank is **unimodular** if the $\det(A_B) \in \{1, -1\}$ for every basis B.
2. Matrix $A \in \mathbb{Z}^{m \times n}$ is **totally unimodular** if the determinant of each square submatrix of A is 0, 1, or -1.

- **Unimodularity** allows handling standard form $\{x \in \mathbb{Z}^n_+ \mid Ax = b\}$
- **Total Unimodularity (TU)** allows handling inequality form $\{x \in \mathbb{Z}^n_+ \mid Ax \leq b\}$
- **Note:** a TU matrix must belong to $\{0, 1, -1\}^{m \times n}$, but not a unimodular matrix:

$$\text{e.g.} \quad A = \begin{bmatrix} 3 & 2 \\ 1 & 1 \end{bmatrix}$$

(Total) Unimodularity

Definition

1. Matrix $A \in \mathbb{Z}^{m \times n}$ of full row rank is **unimodular** if the $\det(A_B) \in \{1, -1\}$ for every basis B.
2. Matrix $A \in \mathbb{Z}^{m \times n}$ is **totally unimodular** if the determinant of each square submatrix of A is 0, 1, or -1.

- **Unimodularity** allows handling standard form $\{x \in \mathbb{Z}^n_+ \mid Ax = b\}$
- **Total Unimodularity (TU)** allows handling inequality form $\{x \in \mathbb{Z}^n_+ \mid Ax \leq b\}$
- **Note:** a TU matrix must belong to $\{0, 1, -1\}^{m \times n}$, but not a unimodular matrix:

$$\text{e.g.} \quad A = \begin{bmatrix} 3 & 2 \\ 1 & 1 \end{bmatrix}$$

- Will provide easier ways to test for U and TU, but first let's see why we care...

(Total) Unimodularity Yields Integral LP Relaxations

Theorem

1. *The matrix $A \in \mathbb{Z}^{m \times n}$ of full row rank* **is unimodular if and only if** *the polyhedron $P(b) = \{x \in \mathbb{R}^n_+ \mid Ax = b\}$* **is integral** *for all $b \in \mathbb{Z}^m$ with $P(b) \neq \emptyset$.*
2. *The matrix A* **is totally unimodular if and only if** *the polyhedron $P(b) = \{x \in \mathbb{R}^n_+ \mid Ax \leq b\}$* **is integral** *for all $b \in \mathbb{Z}^m$ with $P(b) \neq \emptyset$.*

(Total) Unimodularity Yields Integral LP Relaxations

Theorem

1. *The matrix $A \in \mathbb{Z}^{m \times n}$ of full row rank* **is unimodular if and only if** *the polyhedron $P(b) = \{x \in \mathbb{R}^n_+ \mid Ax = b\}$* **is integral** *for all $b \in \mathbb{Z}^m$ with $P(b) \neq \emptyset$.*
2. *The matrix A* **is totally unimodular if and only if** *the polyhedron $P(b) = \{x \in \mathbb{R}^n_+ \mid Ax \leq b\}$* **is integral** *for all $b \in \mathbb{Z}^m$ with $P(b) \neq \emptyset$.*

Proof. (a) "$\Rightarrow$" Because A unimodular, for any $b \in \mathbb{Z}^m$ with $P(b) \neq \emptyset$, any basic feasible solution $x = (x_B, x_N) \in P(b)$ must satisfy $x_B = A_B^{-1} b \in \mathbb{Z}^{|B|}$.

(Total) Unimodularity Yields Integral LP Relaxations

Theorem

1. *The matrix $A \in \mathbb{Z}^{m \times n}$ of full row rank* **is unimodular if and only if** *the polyhedron $P(b) = \{x \in \mathbb{R}^n_+ \mid Ax = b\}$* **is integral** *for all $b \in \mathbb{Z}^m$ with $P(b) \neq \emptyset$.*
2. *The matrix A* **is totally unimodular if and only if** *the polyhedron $P(b) = \{x \in \mathbb{R}^n_+ \mid Ax \leq b\}$* **is integral** *for all $b \in \mathbb{Z}^m$ with $P(b) \neq \emptyset$.*

Proof. (a) "$\Rightarrow$" Because A unimodular, for any $b \in \mathbb{Z}^m$ with $P(b) \neq \emptyset$, any basic feasible solution $x = (x_B, x_N) \in P(b)$ must satisfy $x_B = A_B^{-1} b \in \mathbb{Z}^{|B|}$.

"$\Leftarrow$" We have that $P(b) \neq \emptyset$ is integral $b \in \mathbb{Z}^m$. Let B be any basis of A.

- Sufficient to prove that A_B^{-1} is integral; (A_B integral and $\det(A_B) \cdot \det(A_B^{-1}) = 1$ would imply that $\det(A_B) \in \{1, -1\}$ and thus A is unimodular)

(Total) Unimodularity Yields Integral LP Relaxations

Theorem

1. *The matrix $A \in \mathbb{Z}^{m \times n}$ of full row rank* **is unimodular if and only if** *the polyhedron $P(b) = \{x \in \mathbb{R}^n_+ \mid Ax = b\}$* **is integral** *for all $b \in \mathbb{Z}^m$ with $P(b) \neq \emptyset$.*
2. *The matrix A* **is totally unimodular if and only if** *the polyhedron $P(b) = \{x \in \mathbb{R}^n_+ \mid Ax \leq b\}$* **is integral** *for all $b \in \mathbb{Z}^m$ with $P(b) \neq \emptyset$.*

Proof. (a) "$\Rightarrow$" Because A unimodular, for any $b \in \mathbb{Z}^m$ with $P(b) \neq \emptyset$, any basic feasible solution $x = (x_B, x_N) \in P(b)$ must satisfy $x_B = A_B^{-1} b \in \mathbb{Z}^{|B|}$.

"$\Leftarrow$" We have that $P(b) \neq \emptyset$ is integral $b \in \mathbb{Z}^m$. Let B be any basis of A.

- Sufficient to prove that A_B^{-1} is integral; (A_B integral and $\det(A_B) \cdot \det(A_B^{-1}) = 1$ would imply that $\det(A_B) \in \{1, -1\}$ and thus A is unimodular)
- To prove A_B^{-1} integral, consider $b = A_B \cdot z + e_i$ where z is an integral vector
- Then $A_B^{-1} \cdot b = z + A_B^{-1} e_i$

(Total) Unimodularity Yields Integral LP Relaxations

Theorem

1. *The matrix $A \in \mathbb{Z}^{m \times n}$ of full row rank* **is unimodular if and only if** *the polyhedron $P(b) = \{x \in \mathbb{R}^n_+ \mid Ax = b\}$* **is integral** *for all $b \in \mathbb{Z}^m$ with $P(b) \neq \emptyset$.*
2. *The matrix A* **is totally unimodular if and only if** *the polyhedron $P(b) = \{x \in \mathbb{R}^n_+ \mid Ax \leq b\}$* **is integral** *for all $b \in \mathbb{Z}^m$ with $P(b) \neq \emptyset$.*

Proof. (a) "$\Rightarrow$" Because A unimodular, for any $b \in \mathbb{Z}^m$ with $P(b) \neq \emptyset$, any basic feasible solution $x = (x_B, x_N) \in P(b)$ must satisfy $x_B = A_B^{-1} b \in \mathbb{Z}^{|B|}$.

"$\Leftarrow$" We have that $P(b) \neq \emptyset$ is integral $b \in \mathbb{Z}^m$. Let B be any basis of A.

- Sufficient to prove that A_B^{-1} is integral; (A_B integral and $\det(A_B) \cdot \det(A_B^{-1}) = 1$ would imply that $\det(A_B) \in \{1, -1\}$ and thus A is unimodular)
- To prove A_B^{-1} integral, consider $b = A_B \cdot z + e_i$ where z is an integral vector
- Then $A_B^{-1} \cdot b = z + A_B^{-1} e_i$
- By choosing z large so $z + A_B^{-1} e_i \geq 0$, we obtain a b.f.s. for $P(b)$
- Because $P(b)$ integral, $A_B^{-1} e_i$ must be integral
- Repeat argument for all e_i to proves that A_B^{-1} is integral.

(b) Similar logic, omitted (see notes)

Checking for Total Unimodularity

Proposition

Consider a matrix $A \in \{0, 1, -1\}^{m \times n}$. The following are equivalent:

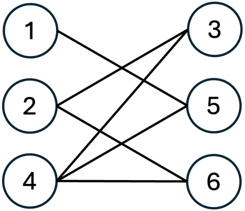
1. *A is totally unimodular.*
2. *$A^\intercal$ is totally unimodular.*
3. *$[A^\intercal \;\; -A^\intercal \;\; I \;\; -I]$ is totally unimodular.*
4. *$\{x \in \mathbb{R}^n_+ \mid Ax = b, 0 \leq x \leq u\}$ is integral for all integral b, u.*
5. *$\{x \mid a \leq Ax \leq b, \ell \leq x \leq u\}$ is integral for all integral a, b, ℓ, u.*
6. *Each collection of columns of A can be partitioned into two parts so that the sum of the columns in one part minus the sum of the columns in the other part is a vector with entries $0, +1$, and -1. (By part 2, a similar result also holds for the rows of A.)*
7. *Each nonsingular submatrix of A has a row with an odd number of non-zero components.*
8. *The sum of entries in any square submatrix with even row and column sums is divisible by four.*
9. *No square submatrix of A has determinant $+2$ or -2.*

#6 perhaps most useful in practice...

Examples of TU Matrices #1

- $G = (\mathcal{N}, \mathcal{E})$ **undirected graph**

- $A \in \{0,1\}^{|\mathcal{N}| \times |\mathcal{E}|}$ is the node-edge incidence matrix of G
  $A_{i,e} = 1$ *if and only if* $i \in e$

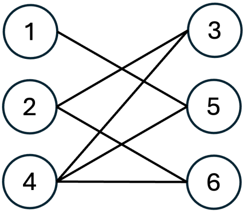


	$\{1,5\}$	$\{2,3\}$	$\{2,6\}$	$\{4,3\}$	$\{4,5\}$	$\{4,6\}$
1	1	0	0	0	0	0
2	0	1	1	0	0	0
3	0	1	0	1	0	0
4	0	0	0	1	1	1
5	1	0	0	0	1	0
6	0	0	1	0	0	1

Examples of TU Matrices #1

- $G = (\mathcal{N}, \mathcal{E})$ **undirected graph**

- $A \in \{0,1\}^{|\mathcal{N}| \times |\mathcal{E}|}$ is the node-edge incidence matrix of G
  $A_{i,e} = 1$ *if and only if* $i \in e$



	$\{1,5\}$	$\{2,3\}$	$\{2,6\}$	$\{4,3\}$	$\{4,5\}$	$\{4,6\}$
1	1	0	0	0	0	0
2	0	1	1	0	0	0
3	0	1	0	1	0	0
4	0	0	0	1	1	1
5	1	0	0	0	1	0
6	0	0	1	0	0	1

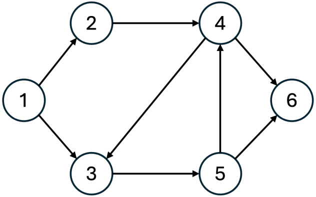
- A **is TU if and only if** G **is bipartite**

- Bipartite matching problems have integral LP relaxations...

Examples of TU Matrices #2

- $D = (V, A)$ is a **directed graph**
- M is the $V \times A$ incidence matrix of D

$$M_{v,a} = \begin{cases} 1 & \text{if and only if } a = (\cdot, v) \text{ (arc } a \text{ enters node } v) \\ -1 & \text{if and only if } a = (v, \cdot) \text{ (arc } a \text{ leaves node } v) \\ 0 & \text{otherwise.} \end{cases}$$

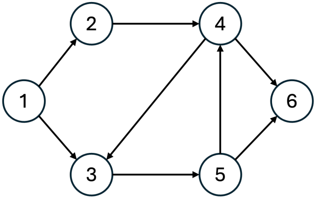


	(1, 2)	(1, 3)	(2, 4)	(4, 3)	(3, 5)	(5, 4)	(4, 6)	(5, 6)
1	−1	−1	0	0	0	0	0	0
2	1	0	−1	0	0	0	0	0
3	0	1	0	1	−1	0	0	0
4	0	0	1	−1	0	1	−1	0
5	0	0	0	0	1	−1	0	−1
6	0	0	0	0	0	0	1	1

Examples of TU Matrices #2

- $D = (V, A)$ is a **directed graph**
- M is the $V \times A$ incidence matrix of D

$$M_{v,a} = \begin{cases} 1 & \text{if and only if } a = (\cdot, v) \text{ (arc } a \text{ enters node } v) \\ -1 & \text{if and only if } a = (v, \cdot) \text{ (arc } a \text{ leaves node } v) \\ 0 & \text{otherwise.} \end{cases}$$

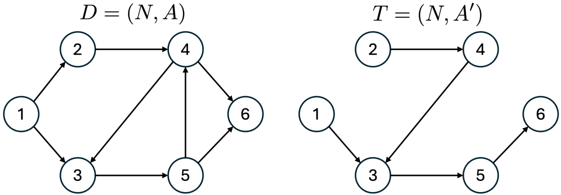


	$(1,2)$	$(1,3)$	$(2,4)$	$(4,3)$	$(3,5)$	$(5,4)$	$(4,6)$	$(5,6)$
1	-1	-1	0	0	0	0	0	0
2	1	0	-1	0	0	0	0	0
3	0	1	0	1	-1	0	0	0
4	0	0	1	-1	0	1	-1	0
5	0	0	0	0	1	-1	0	-1
6	0	0	0	0	0	0	1	1

- Then M **is TU**
- **Network flow problems** (e.g., Prosche Motors) with integral arc capacities and integral supply/demand have integral LP relaxations

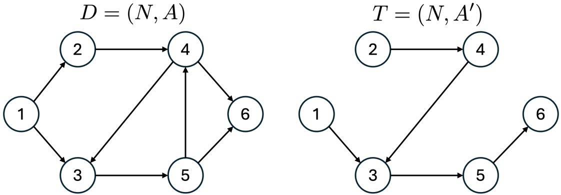
Examples of TU Matrices #3

- $D = (V, A)$ is a **directed graph**, $T = (V, A_0)$ is a directed tree on V



Examples of TU Matrices #3

- $D = (V, A)$ is a **directed graph**, $T = (V, A_0)$ is a directed tree on V

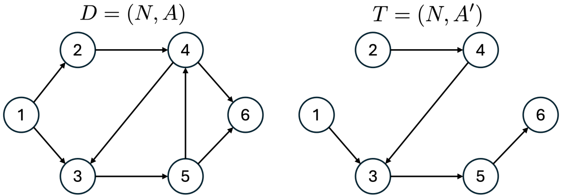


- M is the $A_0 \times A$ matrix defined as follows: for $a = (v, w) \in A$ and $a' \in A_0$,

$$M_{a',a} = \begin{cases} +1 & \text{if the unique } v - w \text{ path in } T \text{ passes through } a' \text{ forwardly} \\ -1 & \text{if the unique } v - w \text{ path in } T \text{ passes through } a' \text{ backwardly} \\ 0 & \text{if the unique } v - w \text{ path in } T \text{ does not pass through } a'. \end{cases}$$

Examples of TU Matrices #3

- $D = (V, A)$ is a **directed graph**, $T = (V, A_0)$ is a directed tree on V



- M is the $A_0 \times A$ matrix defined as follows: for $a = (v, w) \in A$ and $a' \in A_0$,

$$M_{a',a} = \begin{cases} +1 & \text{if the unique } v - w \text{ path in } T \text{ passes through } a' \text{ forwardly} \\ -1 & \text{if the unique } v - w \text{ path in } T \text{ passes through } a' \text{ backwardly} \\ 0 & \text{if the unique } v - w \text{ path in } T \text{ does not pass through } a'. \end{cases}$$

- Then M **is TU**

- All previous examples were **special cases** of this

- Paul Seymour: **all TU matrices** generated from network matrices and **two** other matrices

Dual Integrality and Submodular Functions

- Alternative way to show integrality of polyhedra based on **LP** duality
- Simple observation: to show that LP relaxation is integral, it suffices to check that the optimal value of any LP with integer cost vector c is an integer

Dual Integrality and Submodular Functions

- Alternative way to show integrality of polyhedra based on **LP** duality

- Simple observation: to show that LP relaxation is integral, it suffices to check that the optimal value of any LP with integer cost vector c is an integer

Proposition

P polyhedron with at least one extreme point. Then P is integral **if and only if** *the optimal value $Z_{LP} := \min\{c^\intercal x \mid x \in P\}$ is an integer for all $c \in \mathbb{Z}^n$.*

Proof. Straightforward; omitted.

- To show integrality of P, we **construct an integral dual-optimal solution** (for any $c \in \mathbb{Z}^n$)

Dual Integrality and Submodular Functions

- Alternative way to show integrality of polyhedra based on **LP** duality
- Simple observation: to show that LP relaxation is integral, it suffices to check that the optimal value of any LP with integer cost vector c is an integer

Proposition

P polyhedron with at least one extreme point. Then P is integral **if and only if** *the optimal value $Z_{LP} := \min\{c^\intercal x \mid x \in P\}$ is an integer for all $c \in \mathbb{Z}^n$.*

Proof. Straightforward; omitted.

- To show integrality of P, we **construct an integral dual-optimal solution** (for any $c \in \mathbb{Z}^n$)
- Our discussion here is quite specific
  - broader concepts possible related to Totally Dual Integrality
  - if interested, see notes for references

Submodular Functions

Definition

A function $f(S)$ defined on subsets S of a finite set $N = \{1, \ldots, n\}$ is **submodular** if

$$f(S) + f(T) \geq f(S \cap T) + f(S \cup T), \quad \forall S, T \subset N \tag{1}$$

and it is **supermodular** if the reverse inequality holds.

Submodular Functions

Definition

A function $f(S)$ defined on subsets S of a finite set $N = \{1, \ldots, n\}$ is **submodular** if

$$f(S) + f(T) \geq f(S \cap T) + f(S \cup T), \quad \forall S, T \subset N \tag{1}$$

and it is **supermodular** if the reverse inequality holds.

- For a more intuitive take, note that (1) is equivalent to:

$$(1) \;\Leftrightarrow f(S) - f(S \cap T) \geq f(S \cup T) - f(T)$$

Submodular Functions

Definition

A function $f(S)$ defined on subsets S of a finite set $N = \{1, \ldots, n\}$ is **submodular** if

$$f(S) + f(T) \geq f(S \cap T) + f(S \cup T), \quad \forall S, T \subset N \tag{1}$$

and it is **supermodular** if the reverse inequality holds.

- For a more intuitive take, note that (1) is equivalent to:

$$\begin{aligned}
(1) \;\; &\Leftrightarrow f(S) - f(S \cap T) \geq f(S \cup T) - f(T) \\
&\Leftrightarrow f\big((S \cap T) \cup (S \setminus T)\big) - f(S \cap T) \geq f\big(T \cup (S \cap T)\big) - f(T)
\end{aligned}$$

Submodular Functions

Definition

A function $f(S)$ defined on subsets S of a finite set $N = \{1, \ldots, n\}$ is **submodular** if

$$f(S) + f(T) \geq f(S \cap T) + f(S \cup T), \quad \forall S, T \subset N \tag{1}$$

and it is **supermodular** if the reverse inequality holds.

- For a more intuitive take, note that (1) is equivalent to:

$$\begin{aligned}
(1) \ &\Leftrightarrow f(S) - f(S \cap T) \geq f(S \cup T) - f(T) \\
&\Leftrightarrow f\big((S \cap T) \cup (S \setminus T)\big) - f(S \cap T) \geq f\big(T \cup (S \cap T)\big) - f(T)
\end{aligned}$$

- Set difference between arguments on the left is $S \setminus (S \cap T) = S \setminus T$
- Set difference between arguments on the right is $(S \cup T) \setminus T = S \setminus T$

Submodular Functions

Definition

A function $f(S)$ defined on subsets S of a finite set $N = \{1, \ldots, n\}$ is **submodular** if

$$f(S) + f(T) \geq f(S \cap T) + f(S \cup T), \quad \forall S, T \subset N \tag{1}$$

and it is **supermodular** if the reverse inequality holds.

- For a more intuitive take, note that (1) is equivalent to:

$$\begin{aligned}(1) \;\; &\Leftrightarrow f(S) - f(S \cap T) \geq f(S \cup T) - f(T) \\ &\Leftrightarrow f\big((S \cap T) \cup (S \setminus T)\big) - f(S \cap T) \geq f\big(T \cup (S \cap T)\big) - f(T)\end{aligned}$$

- Set difference between arguments on the left is $S \setminus (S \cap T) = S \setminus T$
- Set difference between arguments on the right is $(S \cup T) \setminus T = S \setminus T$
- (1): **gains when adding something** $(S \cap T)$ **to a smaller set** $(S \setminus T)$ **are larger than when adding it to a larger set** (T)

Submodular Functions

Definition

A function $f(S)$ defined on subsets S of a finite set $N = \{1, \dots, n\}$ is **submodular** if

$$f(S) + f(T) \geq f(S \cap T) + f(S \cup T), \quad \forall S, T \subset N \tag{1}$$

and it is **supermodular** if the reverse inequality holds.

- For a more intuitive take, note that (1) is equivalent to:

$$\begin{aligned}(1) \;\; &\Leftrightarrow f(S) - f(S \cap T) \geq f(S \cup T) - f(T) \\ &\Leftrightarrow f\big((S \cap T) \cup (S \setminus T)\big) - f(S \cap T) \geq f\big(T \cup (S \cap T)\big) - f(T)\end{aligned}$$

- Set difference between arguments on the left is $S \setminus (S \cap T) = S \setminus T$
- Set difference between arguments on the right is $(S \cup T) \setminus T = S \setminus T$
- (1): **gains when adding something** $(S \cap T)$ **to a smaller set** $(S \setminus T)$ **are larger than when adding it to a larger set** (T)
  - Submodular functions exhibit **"diminishing returns"** or **"decreasing differences"**
  - Might resemble concavity in economic intuition, but **not** computationally! (submodular functions are more like **convex** functions!)

Submodular Functions - Equivalent Definitions

Proposition

A set function $f : 2^N \to \mathbb{R}$ *is* **submodular if and only if***:*

(a) *For any* $S, T \subseteq N$ *such that* $S \subseteq T$ *and* $k \notin T$*:*

$$f(S \cup \{k\}) - f(S) \geq f(T \cup \{k\}) - f(T).$$

(b) *For any* $S \subseteq N$ *and any* j, k *with* $j, k \notin S$ *and* $j \neq k$*:*

$$f(S \cup \{j\}) - f(S) \geq f(S \cup \{j, k\}) - f(S \cup \{k\}). \tag{3.2}$$

Submodular Functions - Equivalent Definitions

Proposition

A set function $f : 2^N \to \mathbb{R}$ is **submodular if and only if***:*

(a) *For any $S, T \subseteq N$ such that $S \subseteq T$ and $k \notin T$:*

$$f(S \cup \{k\}) - f(S) \geq f(T \cup \{k\}) - f(T).$$

(b) *For any $S \subseteq N$ and any j, k with $j, k \notin S$ and $j \neq k$:*

$$f(S \cup \{j\}) - f(S) \geq f(S \cup \{j, k\}) - f(S \cup \{k\}). \tag{3.2}$$

- **Submodular**: **"diminishing returns"** or **"decreasing differences"**
  - cost: economies of scale/scope
  - profit: substitutability

- **Supermodular** is the opposite

Submodular Functions - Equivalent Definitions

Proposition

A set function $f : 2^N \to \mathbb{R}$ *is* **submodular if and only if**:

(a) *For any* $S, T \subseteq N$ *such that* $S \subseteq T$ *and* $k \notin T$*:*

$$f(S \cup \{k\}) - f(S) \geq f(T \cup \{k\}) - f(T).$$

(b) *For any* $S \subseteq N$ *and any* j, k *with* $j, k \notin S$ *and* $j \neq k$*:*

$$f(S \cup \{j\}) - f(S) \geq f(S \cup \{j, k\}) - f(S \cup \{k\}). \tag{3.2}$$

- **Submodular**: **"diminishing returns"** or **"decreasing differences"**
  - cost: economies of scale/scope
  - profit: substitutability
- **Supermodular** is the opposite
- Subsequently, interested in non-negative and **increasing** submodular functions

$$f(S) \leq f(T), \quad \forall S \subset T \subseteq N.$$

Submodular Functions - Equivalent Definitions

- **Linear functions.** For $w \in \mathbb{R}^n$, $f(A) = \sum_{i \in A} w_i$ is both sub- and super-modular.

- **Composition 2.** If $w \geq 0$ and g concave, then $f(S) = g\left(\sum_{i \in S} w_i\right)$ is submodular.

- **Optimal TSP cost on tree graphs.** Consider **undirected tree graph** $G = (N, E)$ with a positive cost for traversing the edges ($c_e \geq 0$ for every edge $e \in E$). For every $S \subseteq N$, define $f(S)$ as the optimal (i.e., smallest) cost for a TSP that goes through all the nodes in S. Then, $f(S)$ is submodular.

- **Network optimization:** consider directed graph with capacities on edges that constrain how much flow can be transported; if $f(S)$ is the maximum flow that can be received at a set of sink nodes S, $f(S)$ is submodular.

- **Inventory and supply chain management:** perishable inventory systems, dual sourcing, and inventory control problems with trans-shipment.

Main Result

- For a submodular function f, consider the problem:

$$\begin{aligned} \text{maximize } & \sum_{j=1}^{n} r_j \cdot x_j \\ & \sum_{j \in S} x_j \leq f(S), \ \forall S \subseteq N \\ & x \in \mathbb{Z}_+^n. \end{aligned}$$

- T: set of feasible integer solutions
- $P(f)$ the feasible set of the LP relaxation:

$$P(f) = \left\{ x \in \mathbb{R}_+^n \;\middle|\; \sum_{j \in S} x_j \leq f(S), \ \forall S \subset N \right\}$$

Main Result

- For a submodular function f, consider the problem:

$$\begin{aligned} \text{maximize } & \sum_{j=1}^{n} r_j \cdot x_j \\ & \sum_{j \in S} x_j \le f(S), \ \forall S \subseteq N \\ & x \in \mathbb{Z}_+^n. \end{aligned}$$

- T: set of feasible integer solutions
- $P(f)$ the feasible set of the LP relaxation:

$$P(f) = \left\{ x \in \mathbb{R}_+^n \,\middle|\, \sum_{j \in S} x_j \le f(S), \ \forall S \subset N \right\}$$

Theorem

If f is submodular, increasing, integer valued, and $f(\emptyset) = 0$, then

$$P(f) = \text{conv}(T).$$

Main Result - Proof

To show: f is submodular, increasing, integer-valued, $f(\emptyset) = 0$, then $P(f) = \text{conv}(T)$.

Proof. Consider the linear relaxation and its dual:

$$\begin{aligned} \text{maximize } & \sum_{j=1}^{n} r_j x_j \\ & \sum_{j \in S} x_j \leq f(S), \quad S \subset N, \\ & x_j \geq 0, \ j \in N \end{aligned}$$

- Key idea: construct feasible solutions for both, with equal value
- Key intuition: use a **greedy** construction in the primal!

Main Result - Proof

To show: f is submodular, increasing, integer-valued, $f(\emptyset) = 0$, then $P(f) = \text{conv}(T)$.

Proof. Consider the linear relaxation and its dual:

$$\begin{aligned} \text{maximize } & \sum_{j=1}^{n} r_j x_j \\ & \sum_{j \in S} x_j \leq f(S), \quad S \subset N, \\ & x_j \geq 0, \; j \in N \end{aligned}$$

- Key idea: construct feasible solutions for both, with equal value
- Key intuition: use a **greedy** construction in the primal!
- Suppose $r_1 \geq r_2 \geq \ldots \geq r_k > 0 \geq r_{k+1} \geq \ldots \geq r_n$.
- Let $S^0 = \emptyset$ and $S^j = \{1, \ldots, j\}$ for $j \in N$.

Main Result - Proof

To show: f is submodular, increasing, integer-valued, $f(\emptyset) = 0$, then $P(f) = \text{conv}(T)$.

Proof. Consider the linear relaxation and its dual:

$$\begin{aligned} \text{maximize } & \sum_{j=1}^{n} r_j x_j \\ & \sum_{j \in S} x_j \le f(S), \quad S \subset N, \\ & x_j \ge 0, \ j \in N \end{aligned}$$

- Key idea: construct feasible solutions for both, with equal value
- Key intuition: use a **greedy** construction in the primal!
- Suppose $r_1 \ge r_2 \ge \ldots \ge r_k > 0 \ge r_{k+1} \ge \ldots \ge r_n$.
- Let $S^0 = \emptyset$ and $S^j = \{1, \ldots, j\}$ for $j \in N$.
- We prove that the following x and y are optimal for the primal and dual, respectively.

$$x_j = \begin{cases} f(S^j) - f(S^{j-1}), & 1 \le j \le k, \\ 0, & j > k. \end{cases} \qquad y_S = \begin{cases} r_j - r_{j+1}, & S = S^j, \quad 1 \le j < k, \\ r_k, & S = S^k, \\ 0, & \text{otherwise.} \end{cases}$$

Main Result - Proof

To show: f is submodular, increasing, integer-valued, $f(\emptyset) = 0$, then $P(f) = \text{conv}(T)$.

Proof. Consider the linear relaxation and its dual:

$$\begin{aligned} \underset{x \geq 0}{\text{maximize}} \; & \sum_{j=1}^{n} r_j x_j \\ & \sum_{j \in S} x_j \leq f(S), \quad S \subset N \end{aligned} \qquad\qquad \begin{aligned} \underset{y \geq 0}{\text{minimize}} \; & \sum_{S \subset N} f(S) y_S \\ & \sum_{S : j \in S} y_S \geq r_j, \; j \in N. \end{aligned}$$

- We prove that the following x and y are optimal for the primal and dual, respectively.

$$x_j = \begin{cases} f(S^j) - f(S^{j-1}), & 1 \leq j \leq k, \\ 0, & j > k. \end{cases} \qquad y_S = \begin{cases} r_j - r_{j+1}, & S = S^j, \quad 1 \leq j < k, \\ r_k, & S = S^k, \\ 0, & \text{otherwise.} \end{cases}$$

- f is integer-valued $\Rightarrow x \in \mathbb{Z}^n$. f increasing $\Rightarrow x_j \geq 0$. For all $T \subset N$, we have:

Main Result - Proof

To show: f is submodular, increasing, integer-valued, $f(\emptyset) = 0$, then $P(f) = \text{conv}(T)$.

Proof. Consider the linear relaxation and its dual:

$$\max_{x \geq 0} \sum_{j=1}^{n} r_j x_j \qquad \sum_{j \in S} x_j \leq f(S), \quad S \subset N$$

$$\min_{y \geq 0} \sum_{S \subset N} f(S) y_S \qquad \sum_{S : j \in S} y_S \geq r_j, \; j \in N.$$

- We prove that the following x and y are optimal for the primal and dual, respectively.

$$x_j = \begin{cases} f(S^j) - f(S^{j-1}), & 1 \leq j \leq k, \\ 0, & j > k. \end{cases} \qquad y_S = \begin{cases} r_j - r_{j+1}, & S = S^j, \quad 1 \leq j < k, \\ r_k, & S = S^k, \\ 0, & \text{otherwise.} \end{cases}$$

- f is integer-valued $\Rightarrow x \in \mathbb{Z}^n$. f increasing $\Rightarrow x_j \geq 0$. For all $T \subset N$, we have:

$$\sum_{j \in T} x_j = \sum_{j \in T, j \leq k} \Big(f(S^j) - f(S^{j-1}) \Big)$$

Main Result - Proof

To show: f is submodular, increasing, integer-valued, $f(\emptyset) = 0$, then $P(f) = \text{conv}(T)$.

Proof. Consider the linear relaxation and its dual:

$$\max_{x \geq 0} \sum_{j=1}^{n} r_j x_j \qquad \qquad \min_{y \geq 0} \sum_{S \subset N} f(S) y_S$$

$$\sum_{j \in S} x_j \leq f(S), \quad S \subset N \qquad \qquad \sum_{S : j \in S} y_S \geq r_j, \ j \in N.$$

- We prove that the following x and y are optimal for the primal and dual, respectively.

$$x_j = \begin{cases} f(S^j) - f(S^{j-1}), & 1 \leq j \leq k, \\ 0, & j > k. \end{cases} \qquad y_S = \begin{cases} r_j - r_{j+1}, & S = S^j, \quad 1 \leq j < k, \\ r_k, & S = S^k, \\ 0, & \text{otherwise.} \end{cases}$$

- f is integer-valued $\Rightarrow x \in \mathbb{Z}^n$. f increasing $\Rightarrow x_j \geq 0$. For all $T \subset N$, we have:

$$\sum_{j \in T} x_j = \sum_{j \in T, j \leq k} \Big(f(S^j) - f(S^{j-1}) \Big)$$

$$\text{(because } f \text{ submodular)} \quad \leq \sum_{j \in T, j \leq k} \Big(f(S^j \cap T) - f(S^{j-1} \cap T) \Big) =$$

Main Result - Proof

To show: f is submodular, increasing, integer-valued, $f(\emptyset) = 0$, then $P(f) = \text{conv}(T)$.

Proof. Consider the linear relaxation and its dual:

$$\underset{x \geq 0}{\text{maximize}} \sum_{j=1}^{n} r_j x_j \qquad\qquad \underset{y \geq 0}{\text{minimize}} \sum_{S \subset N} f(S) y_S$$

$$\sum_{j \in S} x_j \leq f(S), \quad S \subset N \qquad\qquad \sum_{S : j \in S} y_S \geq r_j, \; j \in N.$$

- We prove that the following x and y are optimal for the primal and dual, respectively.

$$x_j = \begin{cases} f(S^j) - f(S^{j-1}), & 1 \leq j \leq k, \\ 0, & j > k. \end{cases} \qquad y_S = \begin{cases} r_j - r_{j+1}, & S = S^j, \quad 1 \leq j < k, \\ r_k, & S = S^k, \\ 0, & \text{otherwise.} \end{cases}$$

- f is integer-valued $\Rightarrow x \in \mathbb{Z}^n$. f increasing $\Rightarrow x_j \geq 0$. For all $T \subset N$, we have:

$$\begin{aligned} \sum_{j \in T} x_j &= \sum_{j \in T, j \leq k} \Big(f(S^j) - f(S^{j-1}) \Big) \\ \text{(because } f \text{ submodular)} \quad &\leq \sum_{j \in T, j \leq k} \Big(f(S^j \cap T) - f(S^{j-1} \cap T) \Big) = \\ &= f(S^k \cap T) - f(\emptyset) \end{aligned}$$

Main Result - Proof

To show: f is submodular, increasing, integer-valued, $f(\emptyset) = 0$, then $P(f) = \text{conv}(T)$.

Proof. Consider the linear relaxation and its dual:

$$\max_{x \geq 0} \sum_{j=1}^{n} r_j x_j \qquad\qquad \min_{y \geq 0} \sum_{S \subset N} f(S) y_S$$

$$\sum_{j \in S} x_j \leq f(S), \quad S \subset N \qquad\qquad \sum_{S : j \in S} y_S \geq r_j, \ j \in N.$$

- We prove that the following x and y are optimal for the primal and dual, respectively.

$$x_j = \begin{cases} f(S^j) - f(S^{j-1}), & 1 \leq j \leq k, \\ 0, & j > k. \end{cases} \qquad y_S = \begin{cases} r_j - r_{j+1}, & S = S^j, \quad 1 \leq j < k, \\ r_k, & S = S^k, \\ 0, & \text{otherwise.} \end{cases}$$

- f is integer-valued $\Rightarrow x \in \mathbb{Z}^n$. f increasing $\Rightarrow x_j \geq 0$. For all $T \subset N$, we have:

$$\begin{aligned} \sum_{j \in T} x_j &= \sum_{j \in T, j \leq k} \Big(f(S^j) - f(S^{j-1}) \Big) \\ \text{(because } f \text{ submodular)} \quad &\leq \sum_{j \in T, j \leq k} \Big(f(S^j \cap T) - f(S^{j-1} \cap T) \Big) = \\ &= f(S^k \cap T) - f(\emptyset) \\ \text{(because } f \text{ monotone)} \quad &\leq f(T) - f(\emptyset) \end{aligned}$$

Main Result - Proof

To show: f is submodular, increasing, integer-valued, $f(\emptyset) = 0$, then $P(f) = \mathrm{conv}(T)$.

Proof. Consider the linear relaxation and its dual:

$$\underset{x\geq 0}{\text{maximize}} \sum_{j=1}^{n} r_j x_j \qquad\qquad \underset{y\geq 0}{\text{minimize}} \sum_{S\subset N} f(S) y_S$$

$$\sum_{j\in S} x_j \leq f(S), \quad S \subset N \qquad\qquad \sum_{S:j\in S} y_S \geq r_j,\ j \in N.$$

- We prove that the following x and y are optimal for the primal and dual, respectively.

$$x_j = \begin{cases} f(S^j) - f(S^{j-1}), & 1 \leq j \leq k, \\ 0, & j > k. \end{cases} \qquad y_S = \begin{cases} r_j - r_{j+1}, & S = S^j, \quad 1 \leq j < k, \\ r_k, & S = S^k, \\ 0, & \text{otherwise.} \end{cases}$$

- f is integer-valued $\Rightarrow x \in \mathbb{Z}^n$. f increasing $\Rightarrow x_j \geq 0$. For all $T \subset N$, we have:

$$\begin{aligned} \sum_{j\in T} x_j &= \sum_{j\in T, j\leq k} \left(f(S^j) - f(S^{j-1}) \right) \\ \text{(because } f \text{ submodular)} \quad &\leq \sum_{j\in T, j\leq k} \left(f(S^j \cap T) - f(S^{j-1} \cap T) \right) = \\ &= f(S^k \cap T) - f(\emptyset) \\ \text{(because } f \text{ monotone)} \quad &\leq f(T) - f(\emptyset) \\ \text{(because } f(\emptyset) = 0) \quad &= f(T). \end{aligned}$$

Main Result - Proof

To show: f is submodular, increasing, integer-valued, $f(\emptyset) = 0$, then $P(f) = \text{conv}(T)$.

Proof. Consider the linear relaxation and its dual:

$$\begin{aligned} \underset{x \geq 0}{\text{maximize}} \; & \sum_{j=1}^{n} r_j x_j \\ & \sum_{j \in S} x_j \leq f(S), \quad S \subset N \end{aligned} \qquad \begin{aligned} \underset{y \geq 0}{\text{minimize}} \; & \sum_{S \subset N} f(S) y_S \\ & \sum_{S : j \in S} y_S \geq r_j, \; j \in N. \end{aligned}$$

- We prove that the following x and y are optimal for the primal and dual, respectively.

$$x_j = \begin{cases} f(S^j) - f(S^{j-1}), & 1 \leq j \leq k, \\ 0, & j > k. \end{cases} \qquad y_S = \begin{cases} r_j - r_{j+1}, & S = S^j, \quad 1 \leq j < k, \\ r_k, & S = S^k, \\ 0, & \text{otherwise.} \end{cases}$$

- To show y is dual feasible, note that $y_S \geq 0$ and:

Main Result - Proof

To show: f is submodular, increasing, integer-valued, $f(\emptyset) = 0$, then $P(f) = \text{conv}(T)$.

Proof. Consider the linear relaxation and its dual:

$$\begin{aligned} &\underset{x \geq 0}{\text{maximize}} \sum_{j=1}^{n} r_j x_j \\ &\sum_{j \in S} x_j \leq f(S), \quad S \subset N \end{aligned} \qquad \begin{aligned} &\underset{y \geq 0}{\text{minimize}} \sum_{S \subset N} f(S) y_S \\ &\sum_{S : j \in S} y_S \geq r_j, \; j \in N. \end{aligned}$$

- We prove that the following x and y are optimal for the primal and dual, respectively.

$$x_j = \begin{cases} f(S^j) - f(S^{j-1}), & 1 \leq j \leq k, \\ 0, & j > k. \end{cases} \qquad y_S = \begin{cases} r_j - r_{j+1}, & S = S^j, \quad 1 \leq j < k, \\ r_k, & S = S^k, \\ 0, & \text{otherwise.} \end{cases}$$

- To show y is dual feasible, note that $y_S \geq 0$ and:

$$\sum_{S : j \in S} y_S = y_{S^j} + \ldots + y_{S^k} = r_j, \text{ if } j \leq k \quad \text{and} \quad \sum_{S : j \in S} y_S = 0 \geq r_j, \text{ if } j > k.$$

Main Result - Proof

To show: f is submodular, increasing, integer-valued, $f(\emptyset) = 0$, then $P(f) = \text{conv}(T)$.

Proof. Consider the linear relaxation and its dual:

$$\max_{x \geq 0} \sum_{j=1}^{n} r_j x_j \qquad\qquad \min_{y \geq 0} \sum_{S \subset N} f(S) y_S$$

$$\sum_{j \in S} x_j \leq f(S), \quad S \subset N \qquad\qquad \sum_{S : j \in S} y_S \geq r_j,\ j \in N.$$

- We prove that the following x and y are optimal for the primal and dual, respectively.

$$x_j = \begin{cases} f(S^j) - f(S^{j-1}), & 1 \leq j \leq k, \\ 0, & j > k. \end{cases} \qquad y_S = \begin{cases} r_j - r_{j+1}, & S = S^j, \quad 1 \leq j < k, \\ r_k, & S = S^k, \\ 0, & \text{otherwise.} \end{cases}$$

- To show y is dual feasible, note that $y_S \geq 0$ and:

$$\sum_{S : j \in S} y_S = y_{S^j} + \ldots + y_{S^k} = r_j,\ \text{if } j \leq k \quad \text{and} \quad \sum_{S : j \in S} y_S = 0 \geq r_j,\ \text{if } j > k.$$

- The primal objective: $\sum_{j=1}^{k} r_j \left(f(S^j) - f(S^{j-1}) \right)$

Main Result - Proof

To show: f is submodular, increasing, integer-valued, $f(\emptyset) = 0$, then $P(f) = \text{conv}(T)$.

Proof. Consider the linear relaxation and its dual:

$$\underset{x \geq 0}{\text{maximize}} \sum_{j=1}^{n} r_j x_j \qquad\qquad \underset{y \geq 0}{\text{minimize}} \sum_{S \subset N} f(S) y_S$$

$$\sum_{j \in S} x_j \leq f(S), \quad S \subset N \qquad\qquad \sum_{S : j \in S} y_S \geq r_j, \ j \in N.$$

- We prove that the following x and y are optimal for the primal and dual, respectively.

$$x_j = \begin{cases} f(S^j) - f(S^{j-1}), & 1 \leq j \leq k, \\ 0, & j > k. \end{cases} \qquad y_S = \begin{cases} r_j - r_{j+1}, & S = S^j, \quad 1 \leq j < k, \\ r_k, & S = S^k, \\ 0, & \text{otherwise.} \end{cases}$$

- To show y is dual feasible, note that $y_S \geq 0$ and:

$$\sum_{S : j \in S} y_S = y_{S^j} + \ldots + y_{S^k} = r_j, \text{ if } j \leq k \quad \text{and} \quad \sum_{S : j \in S} y_S = 0 \geq r_j, \text{ if } j > k.$$

- The primal objective: $\displaystyle\sum_{j=1}^{k} r_j \left(f(S^j) - f(S^{j-1}) \right)$
- The dual objective $\displaystyle\sum_{j=1}^{k-1} (r_j - r_{j+1}) f(S^j) + r_k f(S^k) = \sum_{j=1}^{k} r_j \left(f(S^j) - f(S^{j-1}) \right).$