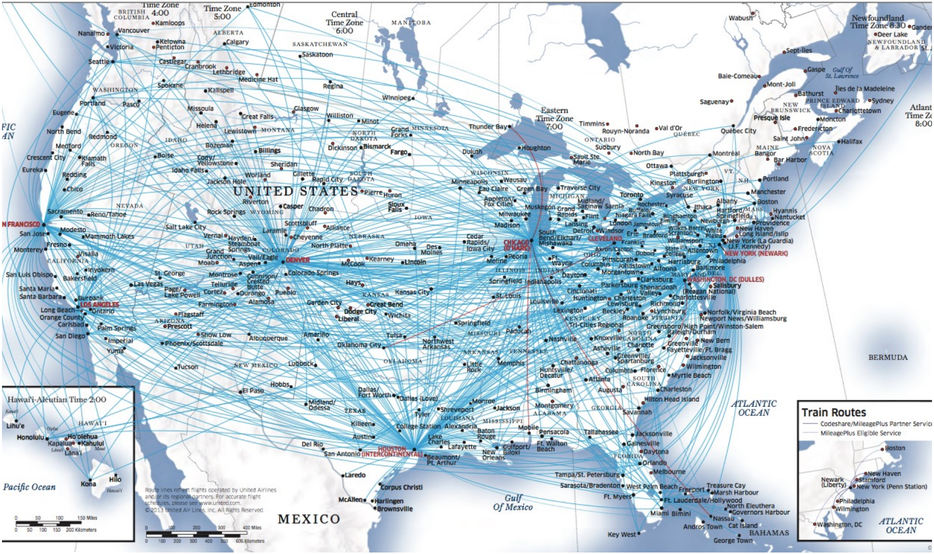
Lecture 7

October 14, 2024

Real-World Hub and Spoke Airline Network



Source: www.united.com

Airline Revenue Management (RM)

- **Strategic RM**
  - Determine several price points for various itineraries
  - “Product” or “itinerary”: origin, destination, day, time, various restrictions, …
    - E.g., JFK – ORD – SFO, 10:30am on Oct 12, 2024, Economy class Y fare
  - Typically done by (or in conjunction with) marketing department
    - Market segmentation; competition

- **Tactical RM** (“yield management”) decides **booking limits**
  - A *booking limit* determines how many seats to reserve for each “product”
  - RM not based on setting prices, but rather changing availability of fare classes
  - Legacy due to original IT systems used (e.g., SABRE)

Airline RM

Hub: Chicago ORD

Two planes

Westbound flights for some day in the future

SFO

BOS

ORD

LAX

JFK

Airline RM

Flight segments (legs)

SFO

BOS

ORD

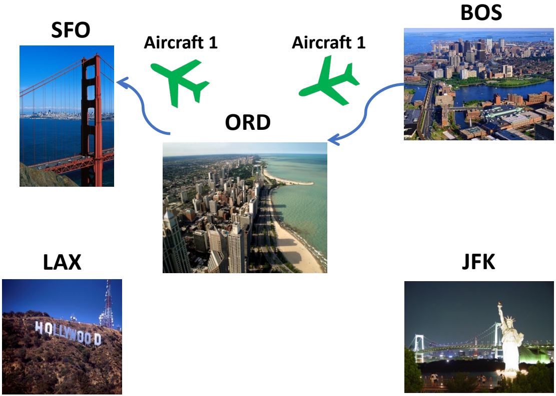
LAX

JFK

Airline RM

Flight segments (legs)

- **Aircraft 1:**
  - BOS-ORD in the morning
  - ORD-SFO in the afternoon

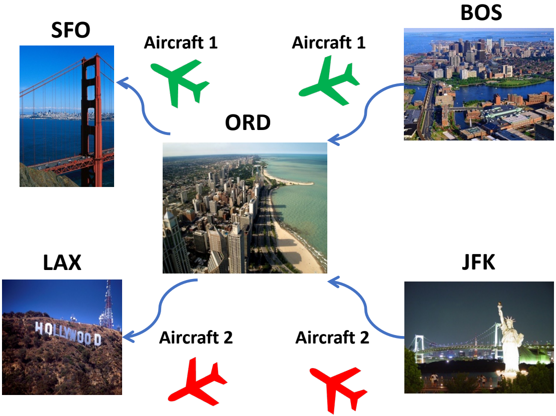


Airline RM

Flight segments (legs)

- **Aircraft 1:**
  - BOS-ORD in the morning
  - ORD-SFO in the afternoon

- **Aircraft 2:**
  - JFK-ORD in the morning
  - ORD-LAX in the afternoon

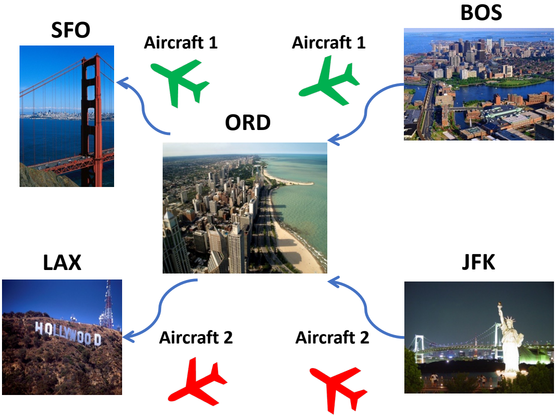


Airline RM

Flight segments (legs)

- **Aircraft 1:**
  - BOS-ORD in the morning
  - ORD-SFO in the afternoon

- **Aircraft 2:**
  - JFK-ORD in the morning
  - ORD-LAX in the afternoon



Itineraries

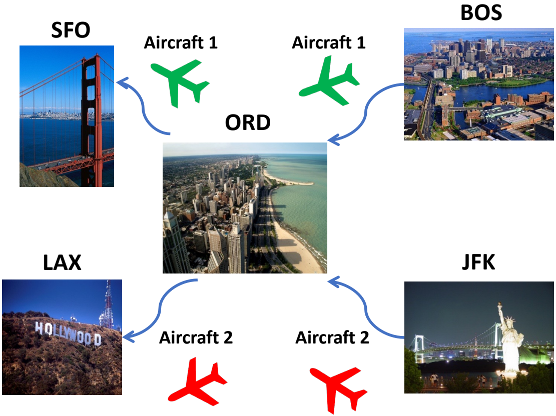
Origin-Destination	Q_Fare	Y_Fare
BOS_ORD	$200	$220
BOS_SFO	$320	$420
BOS_LAX	$400	$490
JFK_ORD	$250	$290
JFK_SFO	$410	$540
JFK_LAX	$450	$550
ORD_SFO	$210	$230
ORD_LAX	$260	$300

Airline RM

Flight segments (legs)

- **Aircraft 1:**
  - BOS-ORD in the morning
  - ORD-SFO in the afternoon

- **Aircraft 2:**
  - JFK-ORD in the morning
  - ORD-LAX in the afternoon



Itineraries

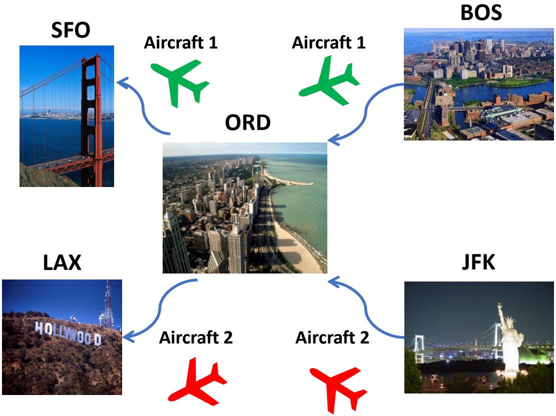
Origin-Destination	Q_Fare	Y_Fare	Q_Demand	Y_Demand
BOS_ORD	$200	$220	25	20
BOS_SFO	$320	$420	55	40
BOS_LAX	$400	$490	65	25
JFK_ORD	$250	$290	24	16
JFK_SFO	$410	$540	65	50
JFK_LAX	$450	$550	40	35
ORD_SFO	$210	$230	21	50
ORD_LAX	$260	$300	25	14

Airline RM

Flight segments (legs)

- **Aircraft 1:**
  - BOS-ORD in the morning
  - ORD-SFO in the afternoon

- **Aircraft 2:**
  - JFK-ORD in the morning
  - ORD-LAX in the afternoon



Resources needed

	BOS_ORD	BOS_SFO	BOS_LAX	JFK_ORD	JFK_SFO	JFK_LAX	ORD_SFO	ORD_LAX
Flight leg								
BOS_ORD_Leg	1	1	1	0	0	0	0	0
JFK_ORD_Leg	0	0	0	1	1	1	0	0
ORD_SFO_Leg	0	1	0	0	1	0	1	0
ORD_LAX_Leg	0	0	1	0	0	1	0	1

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
⋮	⋮	⋮	⋮	⋮
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
⋮	⋮	⋮	⋮	⋮
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:

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

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

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:

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

- $Ax \leq c$: 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**

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

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**
  - $p \in \mathbb{R}^F$: 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

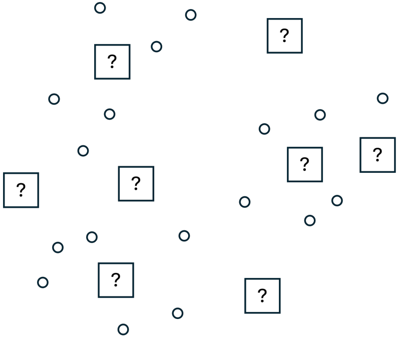
Example: Knapsack

- 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

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

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

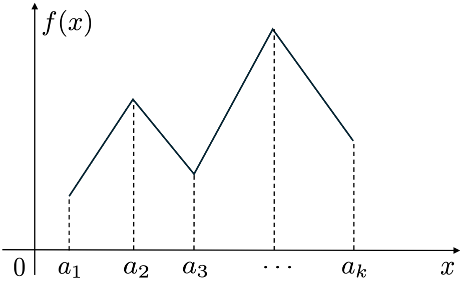
- n potential locations to open facilities
- Cost c_j for opening a facility at location j
- m clients who need service
- Cost d_{ij} for serving client i from facility j
- Smallest cost for opening facilities while serving all clients

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

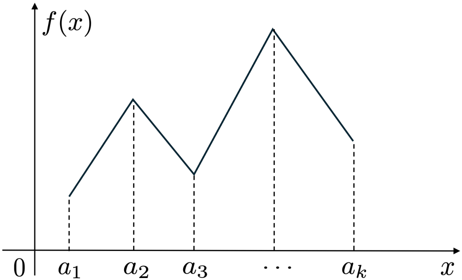
$$
\begin{aligned}
\min \ & \sum_{j=1}^{n} c_j y_j + \sum_{i=1}^{m} \sum_{j=1}^{n} d_{ij} x_{ij} \\
& \sum_{j=1}^{n} x_{ij} = 1, \quad \forall\, i \\
& \sum_{i=1}^{m} x_{ij} \le m y_j, \quad \forall j \\
& x_{ij}, y_j \in \{0, 1\}.
\end{aligned}
$$

Which formulation is "better"?

Example: Piecewise Linear Cost



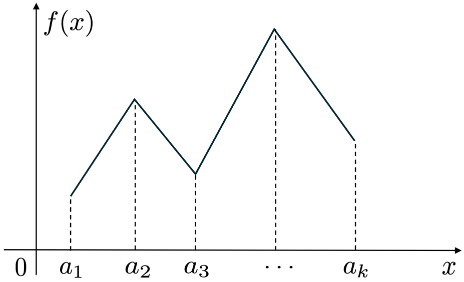
Example: Piecewise Linear Cost



- 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



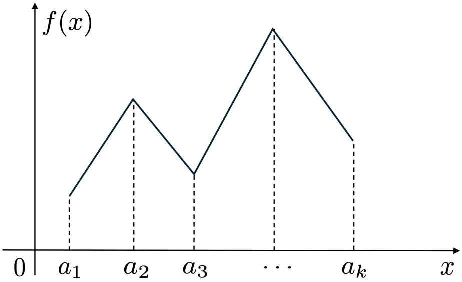
- Idea: $x = \sum_{i=1}^{k} \lambda_i a_i$
- Cost: $\sum_{i=1}^{k} \lambda_i f(a_i)$
- How to impose adjacency?

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

- 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:

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

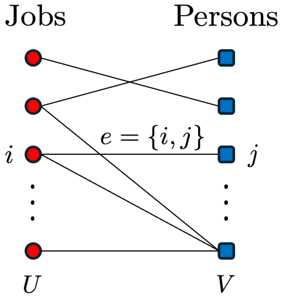
$$\begin{aligned}
\textstyle\sum_{i=1}^{k} \lambda_i &= 1,\\
\lambda_1 &\leq y_1,\\
\lambda_i &\leq y_{i-1} + y_i, \; i = 2, \ldots, k-1,\\
\lambda_k &\leq y_{k-1},\\
\textstyle\sum_{i=1}^{k-1} y_i &= 1,\\
\lambda_i &\geq 0,\\
y_i &\in \{0, 1\}, \; \forall i.
\end{aligned}$$

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 $j \in V$

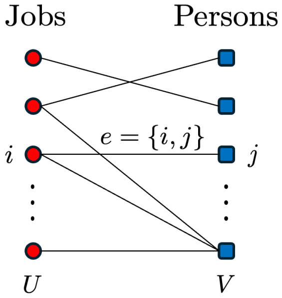
Example: Matching Problems

- 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 $i \in U$



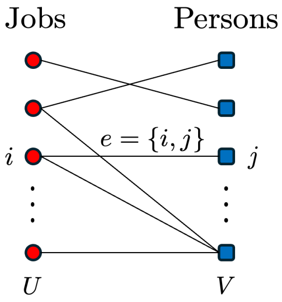
$x_e \in \{0, 1\}$: whether edge selected

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

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

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

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

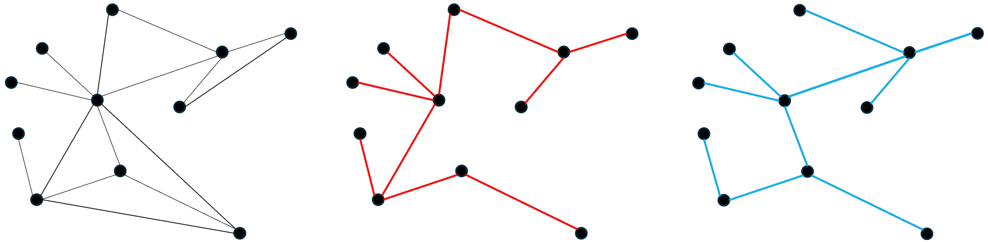
Many variations: minimize cost, require jobs completed, perfect matching, ...

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)

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

Example: Minimum Spanning Tree

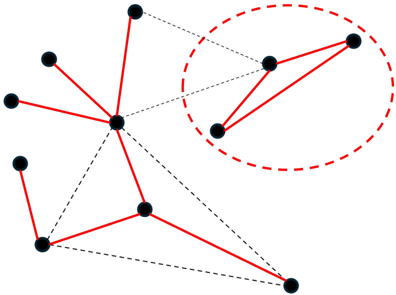
- Given an undirected graph $G = (\mathcal{N}, \mathcal{E})$; $|\mathcal{N}| = n$, $|\mathcal{E}| = m$
- Edge $e \in \mathcal{E}$ has associated cost c_e
- Find **minimum spanning tree** (MST)
  (subset of edges that connect all nodes in $\mathcal{N}$ at minimum cost)

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

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

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

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



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)

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

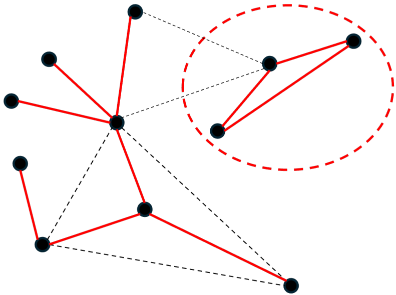
$$x_e \in \{0, 1\}$$

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



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)

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

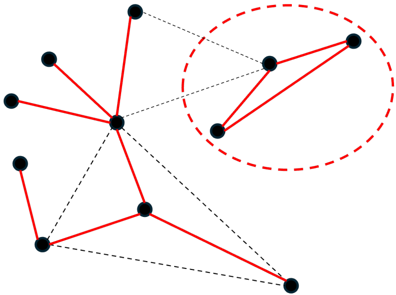
$$x_e \in \{0, 1\}$$

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



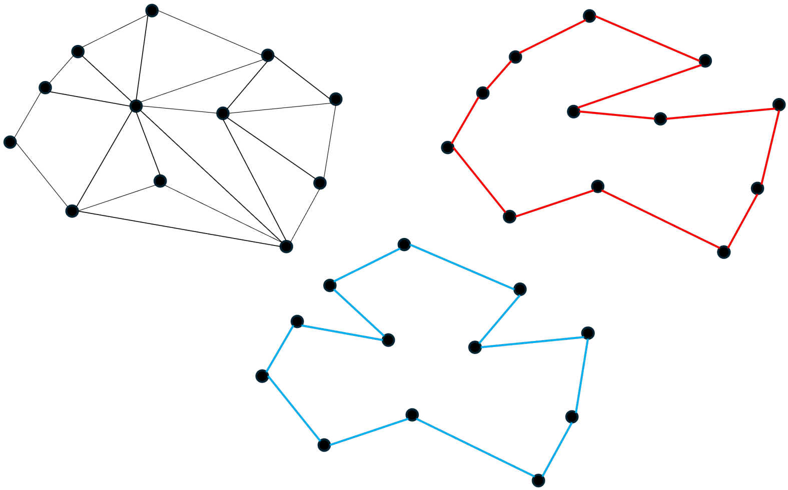
Again exponentially-sized formulations! *Any preference between them?*

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

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

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

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

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

Example: Traveling Salesperson Problem (TSP)

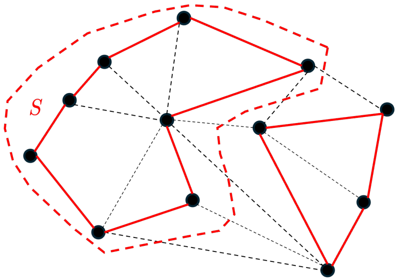
- Given an undirected graph $G = (\mathcal{N}, \mathcal{E})$; $|\mathcal{N}| = n$, $|\mathcal{E}| = m$
- Edge $e \in \mathcal{E}$ has associated cost c_e
- Find a **tour** *(cycle that visits each node exactly once)* with minimum cost

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

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

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

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



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

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

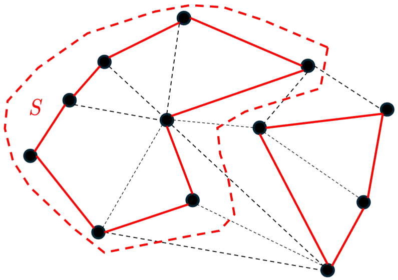
$$x_e \in \{0, 1\}$$

(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

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

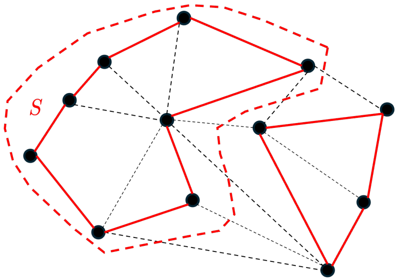
$$x_e \in \{0, 1\}$$

(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**:

$$\sup_{x,y} x + \sqrt{2}y$$

$$x + \sqrt{2}y \leq \frac{1}{2}$$

$$x, y \in \mathbb{Z}.$$

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:

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

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?*

If we relaxed integrality requirements, we would get at LP!

Lower Bounds Again

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

If we relaxed integrality requirements, we would get at LP!

Definition (LP relaxation)

The **linear programming relaxation** for the integer program

$$\begin{aligned} \min\ & c^\intercal x + d^\intercal y \\ & Ax + By = b \\ & x, y \geq 0 \\ & x \in \{0,1\}^{n_1}, y \in \mathbb{Z}^{n_2}, \end{aligned}$$

is obtained by replacing $x \in \{0,1\}^{n_1}$ with $x \in [0,1]^{n_1}$ and $y \in \mathbb{Z}^{n_2}$ with $y \in \mathbb{R}^{n_2}$.

Lower Bounds Again

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

If we relaxed integrality requirements, we would get at LP!

Definition (LP relaxation)

The **linear programming relaxation** for the integer program

$$\begin{aligned} \min\ & c^\intercal x + d^\intercal y \\ & Ax + By = b \\ & x, y \geq 0 \\ & x \in \{0,1\}^{n_1}, y \in \mathbb{Z}^{n_2}, \end{aligned}$$

is obtained by replacing $x \in \{0,1\}^{n_1}$ with $x \in [0,1]^{n_1}$ and $y \in \mathbb{Z}^{n_2}$ with $y \in \mathbb{R}^{n_2}$.

Observation

1) The LP relaxation's optimal value is a lower bound on the IP's optimal value.
2) If the LP relaxation's optimal solution is feasible for the IP, it is optimal for the IP.

Lower Bounds Again

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

If we relaxed integrality requirements, we would get at LP!

Lower Bounds Again

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

If we relaxed integrality requirements, we would get at LP!

Definition (LP relaxation)

The **linear programming relaxation** for the integer program

$$\begin{aligned} \min\ & c^\intercal x + d^\intercal y \\ & Ax + By = b \\ & x, y \geq 0 \\ & x \in \{0,1\}^{n_1}, y \in \mathbb{Z}^{n_2}, \end{aligned}$$

is obtained by replacing $x \in \{0,1\}^{n_1}$ with $x \in [0,1]^{n_1}$ and $y \in \mathbb{Z}^{n_2}$ with $y \in \mathbb{R}^{n_2}$.

Lower Bounds Again

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

If we relaxed integrality requirements, we would get at LP!

Definition (LP relaxation)

The **linear programming relaxation** for the integer program

$$\begin{aligned} \min\ & c^\intercal x + d^\intercal y \\ & Ax + By = b \\ & x, y \geq 0 \\ & x \in \{0,1\}^{n_1}, y \in \mathbb{Z}^{n_2}, \end{aligned}$$

is obtained by replacing $x \in \{0,1\}^{n_1}$ with $x \in [0,1]^{n_1}$ and $y \in \mathbb{Z}^{n_2}$ with $y \in \mathbb{R}^{n_2}$.

Observation

1) The LP relaxation's optimal value is a lower bound on the IP's optimal value.
2) If the LP relaxation's optimal solution is feasible for the IP, it is optimal for the IP.

Lower Bounds Again

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

If we relaxed integrality requirements, we would get at LP!

Definition (LP relaxation)

The **linear programming relaxation** for the integer program

$$\begin{aligned} \min\ & c^\intercal x + d^\intercal y \\ & Ax + By = b \\ & x, y \geq 0 \\ & x \in \{0,1\}^{n_1}, y \in \mathbb{Z}^{n_2}, \end{aligned}$$

is obtained by replacing $x \in \{0,1\}^{n_1}$ with $x \in [0,1]^{n_1}$ and $y \in \mathbb{Z}^{n_2}$ with $y \in \mathbb{R}^{n_2}$.

Observation

1) The LP relaxation's optimal value is a lower bound on the IP's optimal value.
2) If the LP relaxation's optimal solution is feasible for the IP, it is optimal for the IP.

Key Q: How good is this 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)

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

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

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

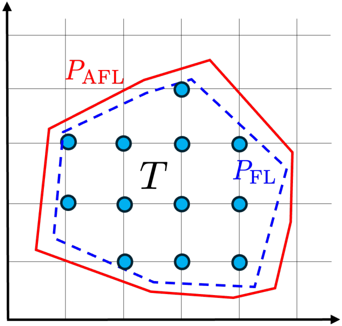
(AFL)

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

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

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

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

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

(Subtour-elimination MST)

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

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

LP Relaxation for Minimum Spanning Tree Problem

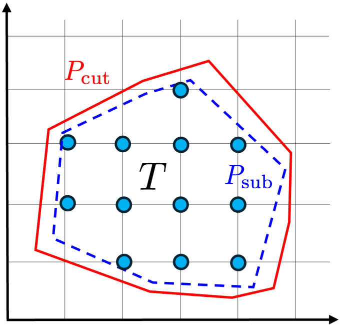
(Cutset MST)

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

(Subtour-elimination MST)

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

- $P_{\textbf{cut}}, P_{\textbf{sub}}$: 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)

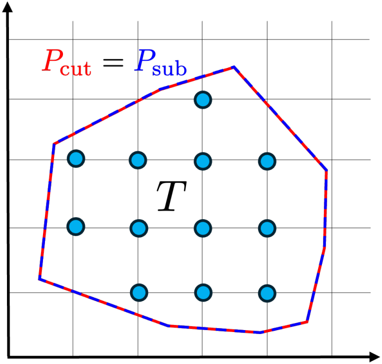
(Cutset TSP)

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

(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
- $P_{\text{sub}} = P_{\text{cut}}$



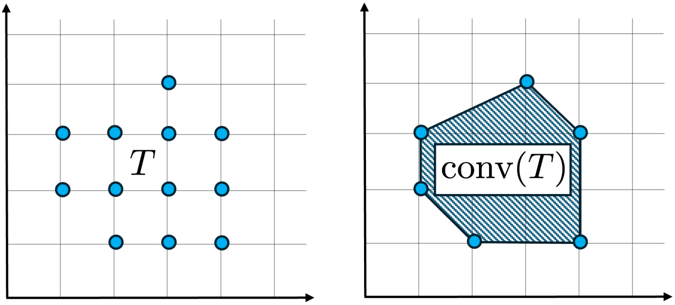
Strength of IP Formulation

- 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 $\text{conv}(T)$ is their convex hull
  - T finite because we assumed bounded feasible set
  - $\text{conv}(T)$ is a polyhedron



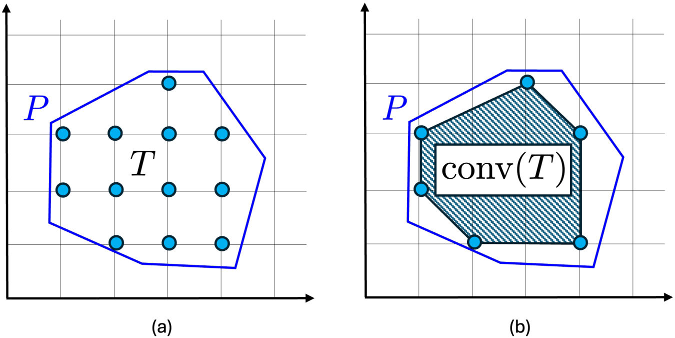
(a) (b)

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

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

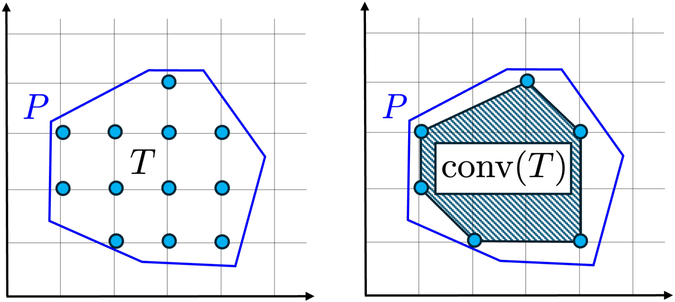


Strength of IP Formulation

- 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 $\text{conv}(T)$ 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 $P = \text{conv}(T)$

Key take-aways:

- Quality of IP formulation : how closely its LP relaxation approximates $\text{conv}(T)$
- 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!