CO370, Winter 2022, Week 6A

MIP Models in Discrete Opt, Part 2

Set-partitioning problem

$\min c^T x$

$Ax = \mathbb{1}$

$0 \le x \le 1$, integer

0-1 matrix A has m rows and n cols

$$\begin{array}{c|cccccc|}
 & 1 & 2 & 3 & \cdots & & n \\
\hline
1 & & & 1 & & & \\
2 & & & & & & \\
3 & & & 1 & & & \\
4 & & & 1 & & & \\
\hline
\end{array}$$

Column j corresponds to a set $S_j \subseteq \{1, \ldots, m\}$

column corresponds to set $S_3 = \{1, 3, 4\}$

Variables $x_1, \ldots, x_n$ meaning $x_j \begin{cases} 1 & \text{select } S_j \\ 0 & \text{otherwise} \end{cases}$

Costs $c = (c_1, \ldots, c_n)$ $\quad c_j$ = cost of selecting set S_j

Another way to express the problem is

$$\min \sum (c_j x_j : j = 1, \ldots, n)$$

s.t.

$$\sum (x_j : i \in S_j) = 1 \quad \forall\, i = 1, \ldots, m$$

$$x_j \in \{0, 1\} \quad \forall\, j = 1, \ldots, n$$

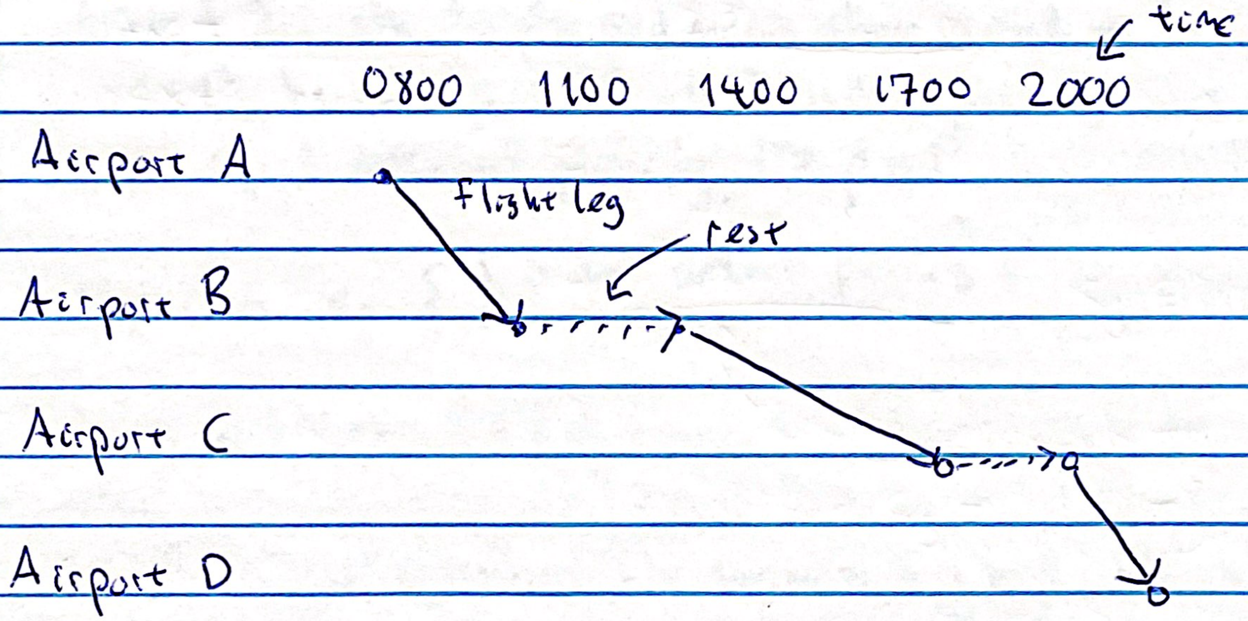
②

Week 6A

Ex: Airline crew pairing

Monthly schedule – list of city-to-city flight segments
– start time, estimated flight time

A "duty period" is a sequence of flight legs that together make a day's work for a crew



duty period: (A,B), (B,C), & (C,D)

⇒ cost of a duty is a combination of flight time, elapsed time, and pay structure

A "crew pairing" is a sequence of duties and layovers, ending with a crew back at home airport

⇒ many rules on feasible pairings (both legal and from union demands)

③

Week 6A

- a pairing can be between 1 and 5 days

Models construct a cost for a pairing, summing the duty costs and layover time and pay structure.

Typically solved as a set-partitioning problem

Let F be the flight segments that need to be covered

P be the set of all feasible pairings

c_p = cost of pairing $p \in P$

Variables $x_p \begin{cases} 1 & \text{if we select pairing } p \\ 0 & \text{otherwise} \end{cases}$

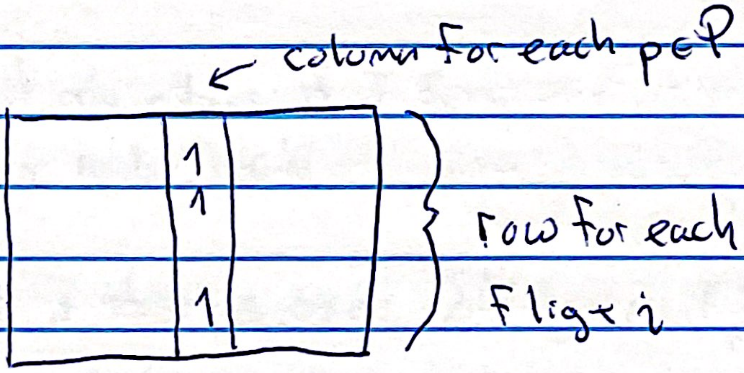
$\min \sum (c_p x_p : p \in P)$

s.t.

$\sum (x_p : p \text{ contains flight segment } i) = 1 \quad \text{for } i \in F$

$x_p \in \{0, 1\} \quad \forall p \in P$

Matrix A



$\min c^T x$

$Ax = \mathbb{1}$

$0 \leq x \leq 1$ integer

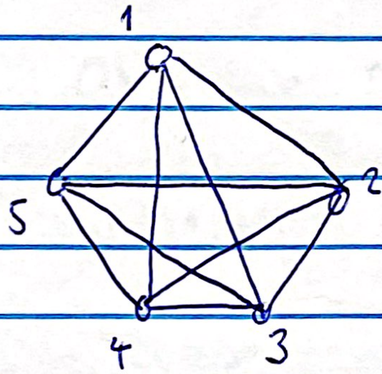
Note: Final step is to assign crew members to pairings. Airlines may use a bidding system following union seniority

4

Week 6A

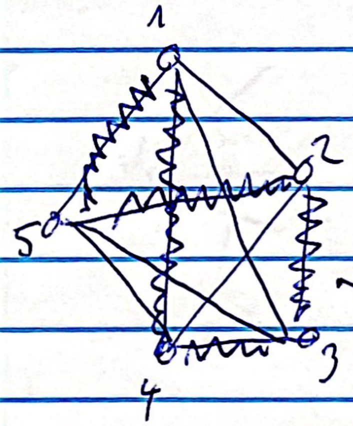
Traveling Salesman Problem (TSP)

Can be described as a problem on a complete graph $G=(V,E)$ (so $G=K_n$ for some positive $n \geq 3$)



- visit points 1, 2, 3, 4 & 5 and return to the starting point
- for each edge $e=(i,j) \in E$, let c_e be the travel cost (distance) from i to j

$$c_e \equiv \text{distance}(i,j)$$



- describe a tour by the edges it uses

Tour $T = \{(1,4), (4,3), (3,2), (2,5), (5,1)\}$

$T \subseteq E$ is a <u>tour</u> if the edges of T form a circuit that visits every node $v \in V$

TSP: Find a tour T that minizes $\sum(c_e : e \in T)$

<u>Note:</u> We assume in our discussion that ~~dist(i,j) = dist(j,i)~~ distance(i,j) = distance(j,i). If this is not the case, the problem is called the ATSP, where A = "asymmetric"

⑤

Week 6A

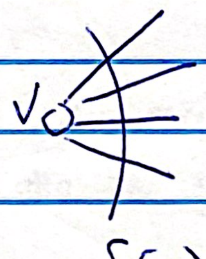
MIP Model

Variables: $x = (x_e : e \in E)$ $\quad x_e = \begin{cases} 1 & \text{if } e \in T \\ 0 & \text{if } e \notin T \end{cases}$

Objective: $\min \sum (c_e x_e : e \in E)$

Constraints: Must force a solution $\bar{x}$ to represent a tour

Consider a node $v \in V$. A tour T must include exactly two edges that meet v. (That is, the edges have v as an end node.)



$\delta(v)$

Let $\delta(v) \equiv \{e \in E : e \text{ meets node } v\}$

Then for any tour T we have

$$|T \cap \delta(v)| = 2$$

$\Rightarrow$ Every tour T satisfies the "degree constraint"

$$\sum (x_e : e \in \delta(v)) = 2$$

we write this as $x(\delta(v)) = 2$

* In general, for $F \subseteq E$ we write $x(F) \equiv \sum (x_e : e \in F)$.

Week6A

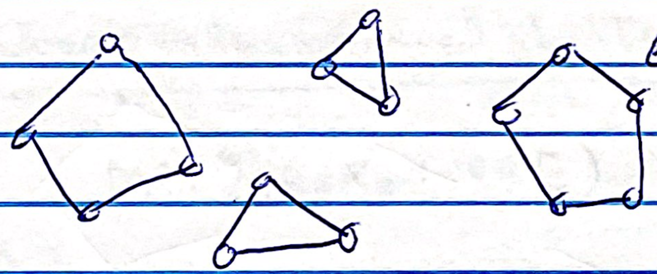
Degree LP Relaxation

$$\min \sum (c_e x_e : e \in E)$$

$$\text{s.t.} \quad x(\delta(v)) = 2 \quad \forall\, v \in V$$

$$0 \le x_e \le 1 \quad \forall\, e \in E$$

BUT an integer solution $\bar{x}$ to the degree LP may not be a tour.



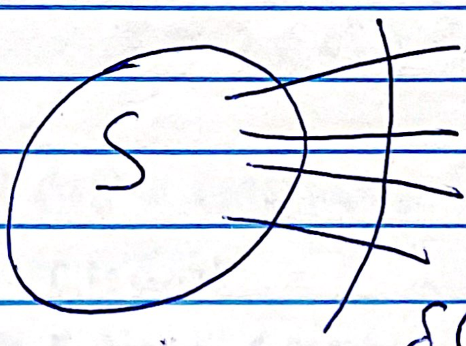
edges with $\bar{x}_e = 1$ may not give a connected graph

$\Rightarrow$ we need constraints to force the graph $G^* = (V, E^*)$ to be connected, where

$$E^* \equiv \{e \in E : \text{such that } \bar{x}_e > 0\}$$

For a set $S \subseteq V$, let

$$\delta(S) \equiv \{e \in E : e \text{ has exactly one end in } S\}$$



$\delta(S)$

Week 6A

<u>Note</u>: If $\emptyset \neq S \neq V$, then $\delta(S)$ must include at <u>least</u> two edges from any tour T.

$$\Rightarrow |T \cap \delta(S)| \geq 2$$

$$\Rightarrow \sum (x_e : e \in \delta(S)) \geq 2$$

$x(\delta(S)) \geq 2$ is called a <u>subtour constraint</u>.

<u>Subtour relaxation of the TSP</u>

$$\min \sum (c_e x_e : e \in E)$$

s.t.

$$x(\delta(v)) = 2 \quad \forall v \in V$$

$$x(\delta(S)) \geq 2 \quad \forall S \subseteq V, \ \emptyset \neq S \neq V$$

$$0 \leq x_e \leq 1 \quad \forall e \in E$$

<u>Note</u>: An integer solution $\bar{x}$ to the subtour relaxation is a TSP tour

$$T = \{e \in E : \bar{x}_e = 1\}$$

$\Rightarrow$ degree constraints force T to be the union of circuits

$\Rightarrow$ subtour constraints force T to be connected

Week 6A

So adding constraints x_e integer $\forall e$ we get a MIP model of the TSP.

$$\#\text{ variables} = \frac{n(n-1)}{2}$$

$\#$ constraints = exponential in n, roughly 2^n.

There are MIP models for the TSP that have polynomial # of variables and constraints (See Williams, 9.5)

BUT the subtour MIP is by far the most successful starting point for exactly solving large-scale instances of the TSP

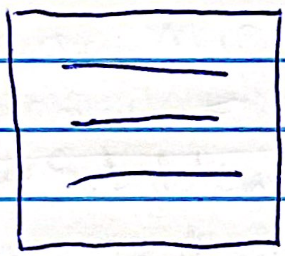
$\Rightarrow$ We will see how to handle the large # of constraints with the "cutting-plane method".

Week 6A

TSP Applications

Logistics – delivery vans (Amazon, etc.),
buses
⋮

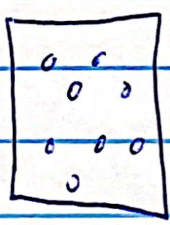
Example from 2015: jet.com



- warehouse of items (books, etc)
- given a collection of items for orders, a picker moves through the warehouse to retrieve the items
- pick path is a tour

jet.com: Optimal tours 20% shorter than S-shaped paths.
And 3% shorter than their own algorithms.

Manufacturing



- drill holes in boards

Points: $(x_1, y_1), (x_2, y_2), \ldots, (x_n, y_n)$

Distance(i,j) =

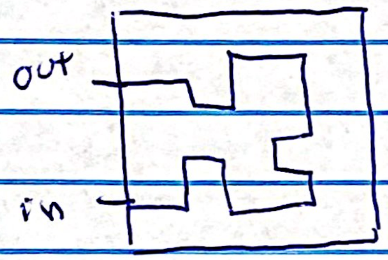
$$\max(|x_i - x_j|, |y_i - y_j|)$$

– L_∞ norm

(10)

Week 6A

VLSI - Scan Chains



- used for testing chips (post-production)

$$distance(i,j) = |x_i - x_j| + |y_i - y_j|$$

- L_1-norm

Machine Scheduling

Jobs to process on machines $j_1, \ldots, j_n$
Known time C_{ij} to re-configure machine from job i to job j. Find order of jobs to minimize the total time.

- typically an ATSP

Genetics

Problems related to DNA sequencing