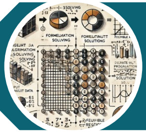
Integer programming & The Travelling Salesman Problem

Orr Zwebner
Baruch Hanya

Reichman University

Agenda



TSP and IP - Definitions

TSP Formulations – MTZ, DFJ

Implementation and results

Competitive quiz

Reichman University

Integer Programming

$\boldsymbol{IP} \subset LP$ s.t all variables are integer, Unlike **M**ixed **L**inear **I**nteger **P**rogramming

IP **standard** form:

$\mathbf{maximize}\ \boldsymbol{c^T x}$

Subject to:

$$\boldsymbol{Ax + s \leq b}$$

$$\boldsymbol{s \geq 0}$$

$$\boldsymbol{x \geq 0}$$

Where $\boldsymbol{x} \in \mathbb{Z}^{\boldsymbol{n}}, \boldsymbol{c} \in \mathbb{R}^{\boldsymbol{n}}, \boldsymbol{s}, \boldsymbol{b} \in \mathbb{R}^{\boldsymbol{m}}, \boldsymbol{a} \in \mathbb{R}^{\boldsymbol{m \times n}}$

Reichman University

Integer Programming

Example - **Maximum independent set:**

Given a graph $G = (V, E)$, find a maximum size $S \subseteq V$ such that no 2 vertices in S have an edge between

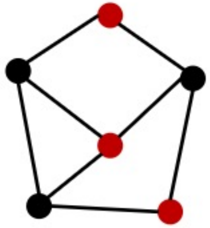
$(\forall v, u \in S \rightarrow (u, v) \notin E)$.

Maximize $\sum_{v \in V} \boldsymbol{x_v}$

Subject to:

$$\boldsymbol{x_v \in \{0, 1\}}$$

$$\boldsymbol{x_u + x_v \leq 1 \ \ \forall (u, v) \in E}$$



$\boldsymbol{MAX\ IS \leq_P IP} \rightarrow$ IP is NP-COMPLETE ($\boldsymbol{MAX\ IS}$ is NP-Hard)

Travelling Salesman Problem - TSP

- **Goal :** Find the **shortest route** visiting each city once and returning to the starting point
- **Challenge**: Number of possible routes grows fast with more cities
- **Approach**: Approximation methods offer quick and practical solutions
- **Important Note :** The starting point does not affect the optimality of the solution

Reichman University

Travelling Salesman Problem - TSP

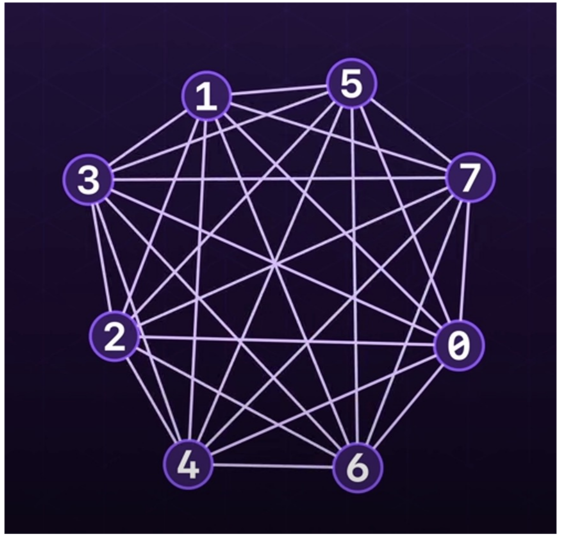
In our case the running time is approximately:

$\sim 10^{61}$

TSP is an NP-complete problem

Exponential time - it involves evaluating all possible permutations of the cities, which is

(n-1)! permutations



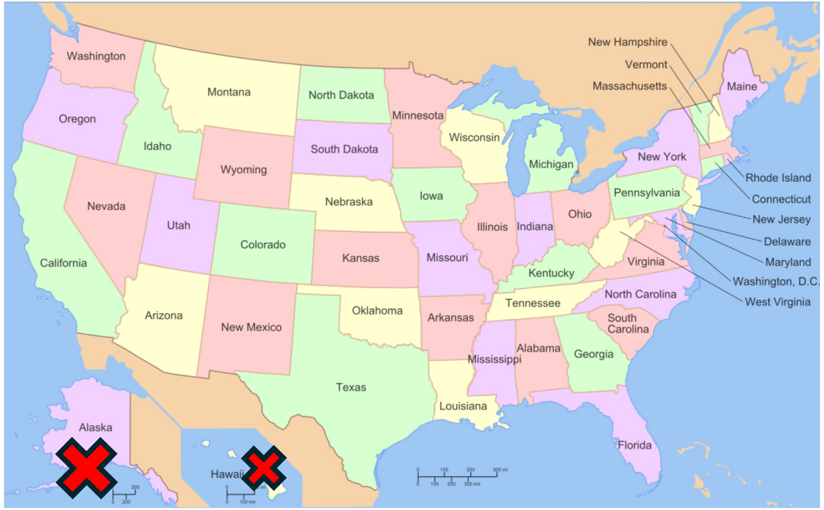
Naive Method

Number of Cities (n)	Possible Permutations ($(n-1)!$)	Approximate Running Time
3	2	Very fast
4	6	Very fast
5	24	Very fast
6	120	Very fast
7	720	Fast
8	5040	Moderate
9	40320	Moderate
10	362880	Slow
11	3628800	Very slow
12	39916800	Extremely slow

Reichman University

Our Dataset

- We took **48 Capitals of USA's states** Excluding Alaska and Hawaii to **simplify the dataset**
- We calculated the distances using **Euclidean distance by KMs**



Reichman University

TSP - Formulation in IP

Optimization problem:

Minimize $\sum_{i=1}^{n}\sum_{j\neq i,j=1}^{n} c_{ij}x_{ij}$

Where:

$$x_{ij} = \begin{cases} 1 & \text{The path goes from city } i \text{ to city } j \\ 0 & \text{Otherwise} \end{cases}$$

$c_{ij} > 0$ The distance from city i to city j

Reichman University

TSP - Formulation in IP

Minimize $\sum_{i=1}^{n} \sum_{j \neq i, j=1}^{n} c_{ij} x_{ij}$

Subject to:

$x_{ij} \in \{0,1\}$ $\qquad i, j = 1 \ldots n;$

$\sum_{j=1, j \neq i}^{n} x_{ij} = 1$ $\qquad i = 1 \ldots n;$

$\sum_{i=1, i \neq j}^{n} x_{ij} = 1$ $\qquad j = 1 \ldots n;$

$x_{ij} = \begin{cases} 1 & \text{The path goes from city } i \text{ to city } j \\ 0 & \text{Otherwise} \end{cases}$

$c_{ij} > 0$ The distance from city i to city j

Reichman University

TSP - Formulation in IP

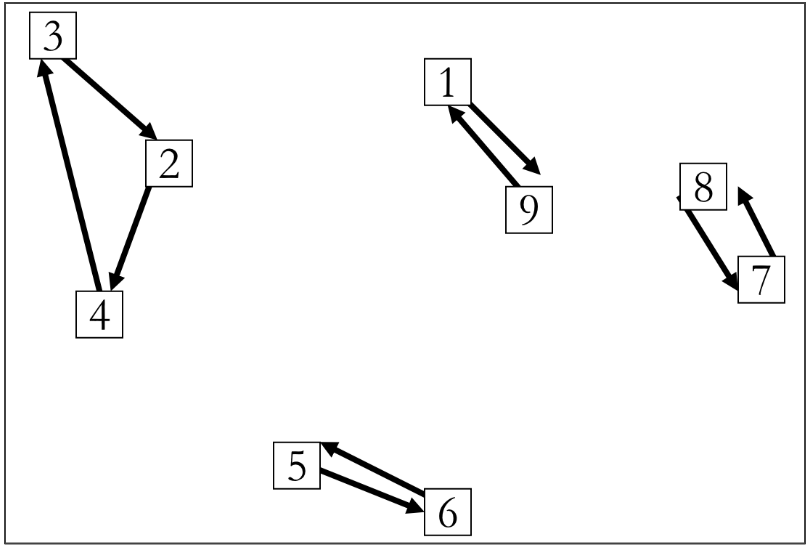
Minimize $\sum_{i=1}^{n} \sum_{j \neq i, j=1}^{n} c_{ij} x_{ij}$

Subject to:

$$x_{ij} \in \{0, 1\} \qquad i, j = 1 \ldots n;$$

$$\sum_{j=1, j \neq i}^{n} x_{ij} = 1 \qquad j = 1 \ldots n;$$

$$\sum_{i=1, i \neq j}^{n} x_{ij} = 1 \qquad i = 1 \ldots n;$$



$x_{ij} = \begin{cases} 1 & \text{The path goes from city } i \text{ to city } j \\ 0 & \text{Otherwise} \end{cases}$

$c_{ij} > 0$ The distance from city i to city j

TSP – Miller Tucker Zemlin formulation

Minimize $\sum_{i=1}^{n}\sum_{j\neq i,j=1}^{n} c_{ij}x_{ij}$

Subject to:

$x_{ij} \in \{0,1\},\ u_i \in \mathbb{N}$ $\quad\quad i,j = 1\ldots n;$

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

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

If $x_{ij} = 1$: $\quad\quad i,j = 2\ldots n;^{*}$

$$u_j = u_i + 1$$

$x_{ij} = \begin{cases} 1 & \text{The path goes from city } i \text{ to city } j \\ 0 & \text{Otherwise} \end{cases}$

$c_{ij} > 0$ The distance from city i to city j

Reichman University

TSP – Miller Tucker Zemlin formulation

Minimize $\sum_{i=1}^{n}\sum_{j\neq i,j=1}^{n} c_{ij}x_{ij}$

Subject to:

$$x_{ij} \in \{0,1\},\ u_i \in \mathbb{N} \qquad i,j = 1\ldots n;$$

$$\sum_{j=1,j\neq i}^{n} x_{ij} = 1 \qquad j = 1\ldots n;$$

$$\sum_{i=1,i\neq j}^{n} x_{ij} = 1 \qquad i = 1\ldots n;$$

$$u_i - u_j + 1 \leq (n-1)(1-x_{ij}) \quad 2 \leq i \neq j \leq n$$

$$2 \leq u_i \leq n \qquad 2 \leq i \leq n$$

$x_{ij} = \begin{cases} 1 & \text{The path goes from city } i \text{ to city } j \\ 0 & \text{Otherwise} \end{cases}$

$c_{ij} > 0$ The distance from city i to city j

Reichman University

TSP – Miller Tucker Zemlin formulation

$u_i - u_j + 1 \le (n-1)(1 - x_{ij})$ **s.t** $x_{ij} = 0$:

$u_i - u_j + 1 \le (n-1)(1-0) = n-1$ $\qquad 2 \le i \ne j \le n$

$^*2 \le u_i \le n$ $\qquad 2 \le i \le n$

$x_{ij} = \begin{cases} 1 & \text{The path goes from city } i \text{ to city } j \\ 0 & \text{Otherwise} \end{cases}$

$c_{ij} > 0$ The distance from city i to city j

Reichman University

TSP – Miller Tucker Zemlin formulation

$u_i - u_j + 1 \leq (n-1)(1 - x_{ij})$ **s.t** $x_{ij} = 0$:

$u_i - u_j + 1 \leq (n-1)(1-0) = n-1 \qquad 2 \leq i \neq j \leq n$

$^*2 \leq u_i \leq n \qquad 2 \leq i \leq n$

$u_i - u_j + 1 \leq (n-1)(1 - x_{ij})$ **s.t** $x_{ij} = 1$:

$u_i - u_j + 1 \leq (n-1)(1-1) = (n-1) * 0 = 0 \qquad 2 \leq i \neq j \leq n$

If $x_{ij} = 1$: $\qquad i, j = 2 \dots n;^*$

$u_j = u_i + 1$

$u_j = u_j + 1 \rightarrow u_i - u_j + 1 = 0 \rightarrow u_i - u_j + 1 \leq 0$

$x_{ij} = \begin{cases} 1 & \text{The path goes from city } i \text{ to city } j \\ 0 & \text{Otherwise} \end{cases}$

$c_{ij} > 0 \qquad$ The distance from city i to city j

Reichman University

TSP – Miller Tucker Zemlin formulation

Minimize $\sum_{i=1}^{n} \sum_{j \neq i, j=1}^{n} c_{ij} x_{ij}$

Subject to:

$$x_{ij} \in \{0,1\},\ u_i \in \mathbb{N} \qquad i, j = 1 \ldots n;$$

$$\sum_{j=1, j \neq i}^{n} x_{ij} = 1 \qquad j = 1 \ldots n;$$

$$\sum_{i=1, i \neq j}^{n} x_{ij} = 1 \qquad i = 1 \ldots n;$$

$$u_i - u_j + 1 \leq (n-1)(1 - x_{ij}) \quad 2 \leq i \neq j \leq n$$

$$2 \leq u_i \leq n \qquad 2 \leq i \leq n$$

$$x_{ij} = \begin{cases} 1 & \text{The path goes from city } i \text{ to city } j \\ 0 & \text{Otherwise} \end{cases}$$

$c_{ij} > 0$ The distance from city i to city j

Reichman University

Danzig Fulkerson Johnson formulation

Minimize $\sum_{i=1}^{n} \sum_{j \neq i, j=1}^{n} c_{ij} x_{ij}$

Subject to:

$$x_{ij} \in \{0,1\} \qquad i,j = 1 \ldots n;$$

$$\sum_{j=1, j \neq i}^{n} x_{ij} = 1 \qquad j = 1 \ldots n;$$

$$\sum_{i=1, i \neq j}^{n} x_{ij} = 1 \qquad j = 1 \ldots n;$$

$$\sum_{i,j \in S, i \neq j} x_{ij} \leq |S| - 1 \qquad \forall S \subsetneq \{1, 2, \ldots, n\}$$

Reichman University

$x_{ij} = \begin{cases} 1 & \text{The path goes from city } i \text{ to city } j \\ 0 & \text{Otherwise} \end{cases}$	
$c_{ij} > 0$	The distance from city i to city j

Danzig Fulkerson Johnson formulation

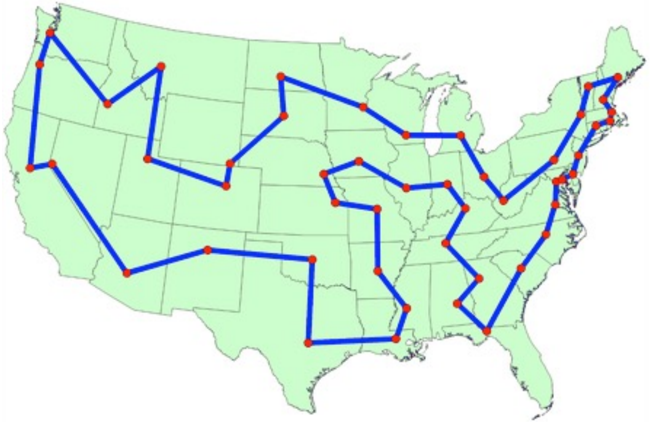
Minimize $\sum_{i=1}^{n} \sum_{j \neq i, j=1}^{n} c_{ij} x_{ij}$

Subject to:

$$x_{ij} \in \{0, 1\} \qquad i, j = 1 \ldots n;$$

$$\sum_{j=1, j \neq i}^{n} x_{ij} = 1 \qquad j = 1 \ldots n;$$

$$\sum_{i=1, i \neq j}^{n} x_{ij} = 1 \qquad j = 1 \ldots n;$$



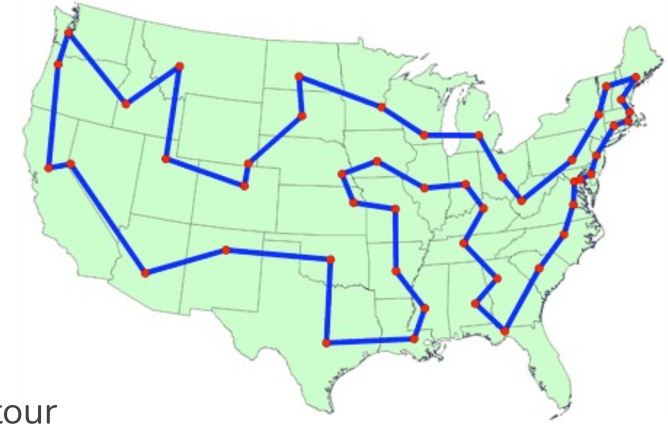
$$x_{ij} = \begin{cases} 1 & \text{The path goes from city } i \text{ to city } j \\ 0 & \text{Otherwise} \end{cases}$$

$c_{ij} > 0$ The distance from city i to city j

Reichman University

Danzig Fulkerson Johnson formulation

Minimize $\sum_{i=1}^{n} \sum_{j \neq i, j=1}^{n} c_{ij} x_{ij}$



Subject to:

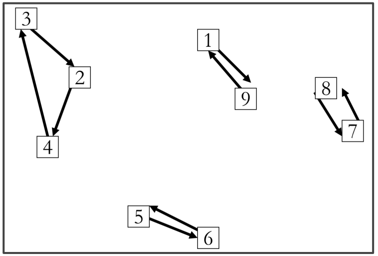
$x_{ij} \in \{0, 1\}$ $\qquad i, j = 1 \ldots n;$

$\sum_{j=1, j \neq i}^{n} x_{ij} = 1$ $\qquad j = 1 \ldots n;$

$\sum_{i=1, i \neq j}^{n} x_{ij} = 1$ $\qquad j = 1 \ldots n;$

There is no sub-tour - The solution is a single tour

(not a union of smaller tours)



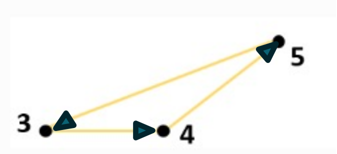
$x_{ij} = \begin{cases} 1 & \text{The path goes from city } i \text{ to city } j \\ 0 & \text{Otherwise} \end{cases}$

$c_{ij} > 0$ $\qquad$ The distance from city i to city j

Danzig Fulkerson Johnson formulation

There is no sub-tour - The solution is a single tour =

The number of arcs between nodes in the subset should be less than the number of nodes in that subset

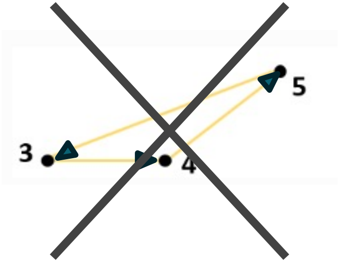


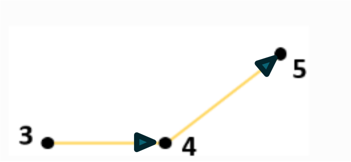
Reichman University

Danzig Fulkerson Johnson formulation

There is no sub-tour - The solution is a single tour =

The number of arcs between nodes in the subset should be less than the number of nodes in that subset



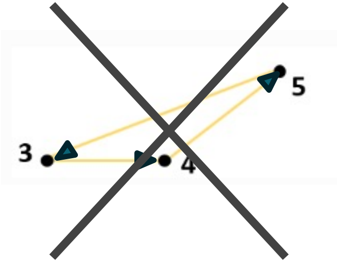


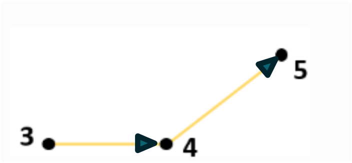
Reichman University

Danzig Fulkerson Johnson formulation

There is no sub-tour - The solution is a single tour =

The number of arcs between nodes in the subset should be less than the number of nodes in that subset





x_{ij} - An arc between nodes i and j

$$x_{34} + x_{45} + x_{53} \leq 2$$

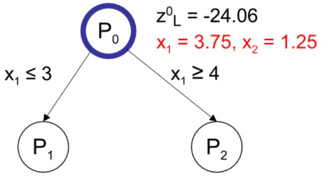
Reichman University

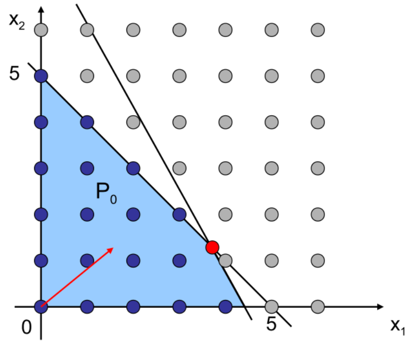
Branch and bound

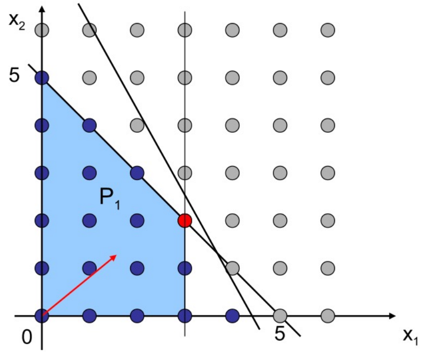
$$
\begin{array}{rrcl}
z_I^1 = \max & 5x_1 + \frac{17}{4}x_2 & & \\
 & x_1 + x_2 & \leq & 5 \\
 & 10x_1 + 6x_2 & \leq & 45 \\
 & x_1 & \leq & 3 \\
 & x_1, x_2 & \geq & 0 \\
 & x_1, x_2 & \in & \mathbb{Z}
\end{array}
\qquad (P_1)
$$

$$
\begin{array}{rrcl}
z_I^2 = \max & 5x_1 + \frac{17}{4}x_2 & & \\
 & x_1 + x_2 & \leq & 5 \\
 & 10x_1 + 6x_2 & \leq & 45 \\
 & x_1 & \geq & 4 \\
 & x_1, x_2 & \geq & 0 \\
 & x_1, x_2 & \in & \mathbb{Z}
\end{array}
\qquad (P_2)
$$

This operation is called a *branching on variable* x_1. Note that the solution $(3.75, 1.25)$ does not belong to the linear relaxation of (P_1) or (P_2). We can represent the subproblems and the corresponding bounds by means of a tree, called the Branch-and-Bound tree.







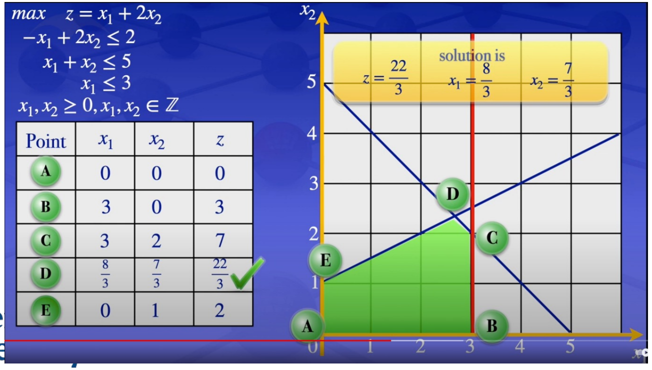
Reichman University

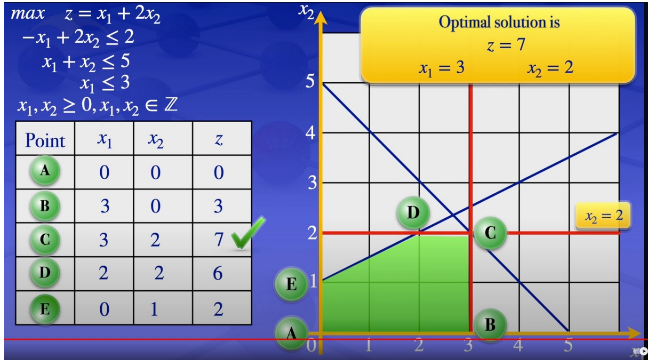
Cutting plane method

Cutting plane method

Start with the linear relaxation $\max\{c^T x \mid Ax \leq b,\ x \geq 0\}$.

1. Solve the current linear relaxation, and let x^* be a basic optimal solution;
2. If $x^* \in X$, then x^* is optimal for (P_I); STOP.
3. Otherwise, <u>find</u> an inequality $\alpha^T x \leq \beta$ that is valid for X and cuts off x^*;
4. Add the inequality $\alpha^T x \leq \beta$ to the current linear relaxation and go to 1.





Solutions

NP HARD - Approximated solutions

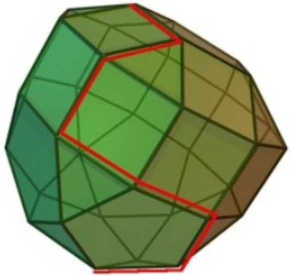
1. **Greedy randomized - Nearest neighbor**
   a. Generate a start city, always choose the closest city
   b. Try N times, choose the best tour
      i. Each iteration with a different start city

Reichman University

Solutions

NP HARD - Approximated solutions

1. **Greedy randomized - Nearest neighbor**
   a. Generate a start city, always choose the closest city
   b. Try N times, choose the best tour
2. **Solve LP and find a relaxed solution (simplex):**
   a. Rounded
   b. Branch and bound
   c. Cutting plane method



Solutions

NP HARD - Approximated solutions

1. Greedy randomized - Ne or

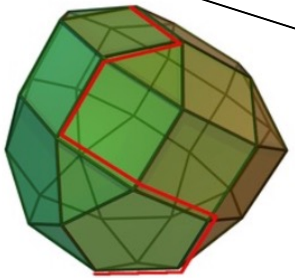
a. Generate the closest city

b.

randomized)

bound

utting plane method

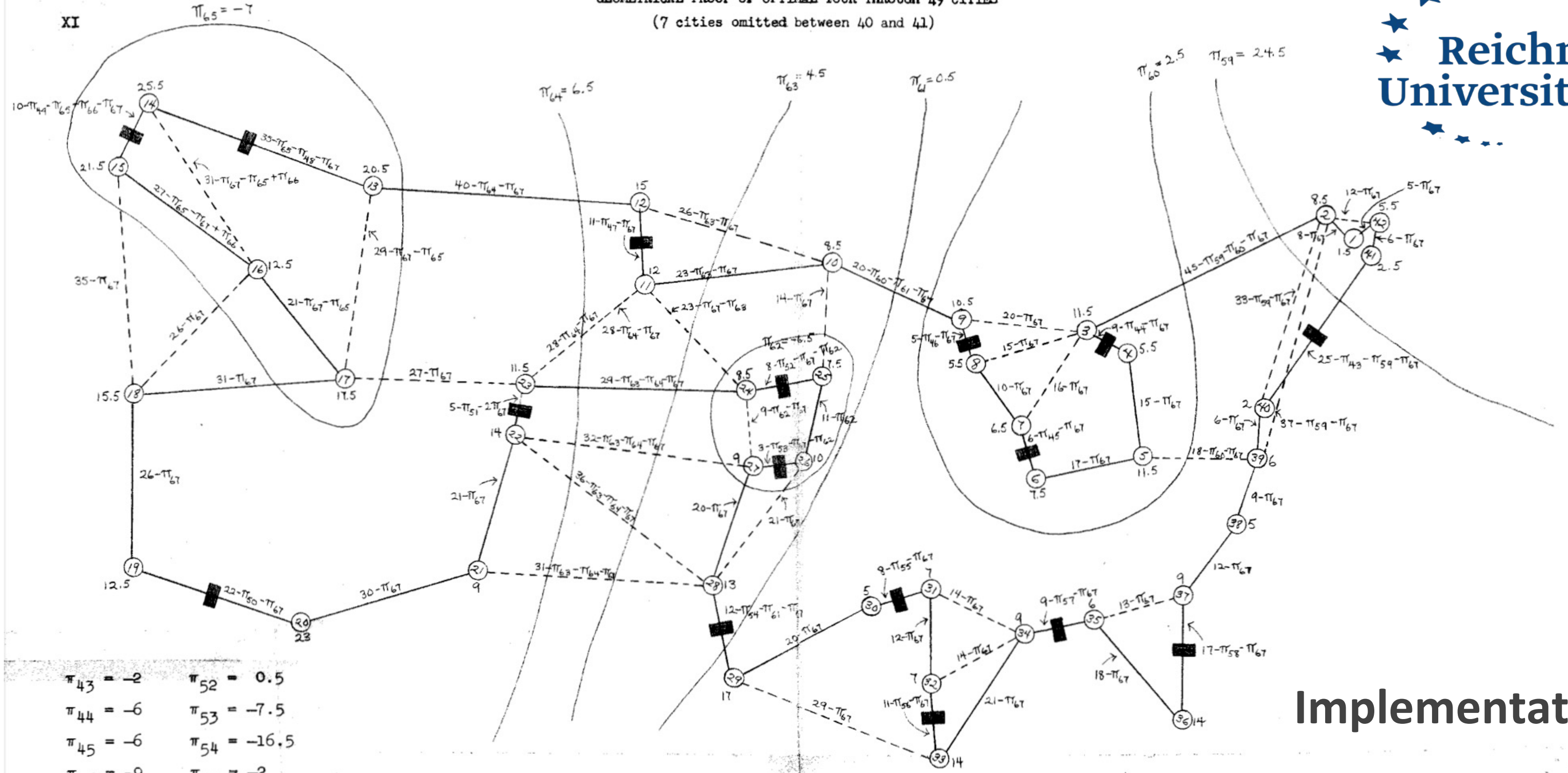


Implementation

GEOMETRICAL PROOF OF OPTIMAL TOUR THROUGH 49 CITIES

(7 cities omitted between 40 and 41)

XI



π_{43}	$= -2$	π_{52}	$= 0.5$
π_{44}	$= -6$	π_{53}	$= -7.5$
π_{45}	$= -6$	π_{54}	$= -16.5$
π_{46}	$= -9$	π_{55}	$= -2$
π_{47}	$= -14$	π_{56}	$= -8$
π_{48}	$= -2$	π_{57}	$= -4$
π_{49}	$= -26$	π_{58}	$= -4$
π_{50}	$= -11.5$	π_{66}	$= -2$
π_{51}	$= -16.5$	π_{67}	$= -2$

Note: In comparing πP_{ij} with d_{ij}, remember that the components of P_{ij} corresponding to the added relations must be considered. The added relations are pictured graphically above except for the last two, 66 and 67. A loop drawn about a subset of points corresponds to a relation of the kind $\Sigma_1 \leq n_1-1$; an arc separating a set of points from its complement corresponds to $\Sigma_{1,j} \geq 2$. For example, to compare $\pi P_{25,9}$ with $d_{25,9} = 21$, observe that the segment (25,9) crosses two arcs, i.e., $P_{25,9}$ has a 1 in components 60 and 61; it has a zero in components 66 and 67 since $x_{25,9}$ does not appear in those relations, and of course, has zeros elsewhere, except for components 25 and 9. Hence $\pi P_{25,9} = \pi_{25}+\pi_9+\pi_{60}+\pi_{61} = 7.5+10.5+2.5+0.5 = 21 \leq d_{25,9}$. One more example: $\pi P_{27,25}=\pi_{27}+\pi_{25}+\pi_{62}$, since (27,25) is contained within the loop representing the condition $\Sigma_{12} \leq 3$.

Results

Model	Formula	Objective Value (KM)	Running time
Nearest neighbor	-	20,003.87	0.1 sec
CBC*	MTZ	17,585.43	1 hr
	DFJ	17,585.43	1 hr
GUROBI	MTZ	17,083.89	4 min 25.2 sec
	DFJ**	**17,083.89**	**0.2 sec**
Optimum***	-	17,083.89	-

* Time limited

** Using GUROBI lazy constraints

*** According to 2 online blogs

Reichman University

Conclusions

1. IP is a tool for formulating complex optimization problems

2. The simple solution is a good and efficient approximation
   a. NN Objective value $<$ 1.18 OPT

3. Solve IP or TSP in polynomial time and you will be a millionaire

4. Until then – Use GUROBI (with academic license)

Question?

Reichman University

Reichman University