Travelling Salesman Problem

Konstantin Pavlikov
kop@sam.sdu.dk

Department of Business and Management
University of Southern Denmark

February 15, 2025

Travelling Salesman Problem: Background

Every day companies like

- ▶ FedEx (UPS, DHL)
- ▶ School buses routing
  https://www.wbur.org/edify/2017/07/27/mit-quantum-boston-bus-routes
- ▶ Warehouse management
- ▶ Microprocessor production
- ▶ ...

have a set of customers that need to be visited in order to deliver (pick) goods (people). Some other problems can be interpreted as the TSP:

- ▶ Job scheduling (production planning)

Travelling Salesman Problem: Statement

There is a set of $\{1, \ldots, n\}$ customers, a matrix of travel distances (or costs, or travel times) c_{ij} that provides a cost of travelling directly from i to j, and we need to determine

- in which specific order to visit all n customers

such that

- every customer needs to be visited just once
- the total cost of the route is minimized.

If the cost matrix is symmetric:

$$c_{ij} = c_{ji} \qquad i, j \in \{1, \ldots, n\}, \ i \neq j,$$

then the problem is called the symmetric TSP (STSP), otherwise it is called the asymmetric TSP (ATSP).

Traveling Salesman Problem

The TSP is a fundamental problem in several domains simultaneously

- ▶ Mathematics
- ▶ Computer Science
- ▶ Management Science and Operations Research

which does not happen often.

Traveling Salesman Problem: State of the Art

Bill Cook at University of Waterloo leads the most advanced research group on the symmetric TSP problem:

- ▶ William Cook, Combinatorics and Optimization, University of Waterloo, Canada
- ▶ Daniel Espinoza, Gurobi Optimization, USA
- ▶ Marcos Goycoolea, School of Business, Universidad Adolfo Ibanez, Chile
- ▶ Keld Helsgaun, Computer Science, Roskilde University, Denmark

- ▶ Website http://www.math.uwaterloo.ca/tsp/
- ▶ Interview https://www.youtube.com/watch?v=q8nQTNvCrjE
- ▶ Apple Store App: Concord

Number of Possible Routes

There is a very large number of possible routes for the TSP. Specifically, for n locations and starting from location 1, there are $(n-1)!$ possible tours.

- for $n = 5$, $4! = 24$
- for $n = 6$, $5! = 120$
- for $n = 7$, $6! = 720$
- for $n = 8$, $7! = 5040$
- for $n = 9$, $8! = 40320$
- for $n = 10$, $9! = 362880$
- for $n = 11$, $10! = 3628800$

TSP

We consider the following very small instance of the asymmetric travelling salesman problem with 4 locations:

$$
\begin{array}{cccc}
 & 20 & 23 & 4 \\
30 & & 7 & 27 \\
25 & 5 & & 25 \\
3 & 21 & 26 &
\end{array}
$$

The possible tours can be expressed as

$$1 \to 3 \to 2 \to 4 \to 1$$

$$1 \to 4 \to 3 \to 2 \to 1$$

total number of possible tours is $3 \times 2 = 6$.

TSP: Formalizing a Tour

Let us try to represent the tour $1 \to 3 \to 2 \to 4 \to 1$ more formally:

$$
\begin{matrix}
 & 0 & 1 & 0 \\
0 & & 0 & 1 \\
0 & 1 & & 0 \\
1 & 0 & 0 &
\end{matrix}
$$

In other words, a tour is represented by a matrix, that should be read as follows

- ▶ from the origin location 1 go to 3
- ▶ from 3 go to 2
- ▶ from 2 go to 4
- ▶ from 4 go to 1

TSP: Tour Cost

The total distance of the below presented tour

$$
\begin{array}{cccc}
 & 0 & 1 & 0 \\
0 & & 0 & 1 \\
0 & 1 & & 0 \\
1 & 0 & 0 &
\end{array}
$$

is equal to

$$23 + 27 + 5 + 3 = 58.$$

TSP: Tour Cost

Let $x_{ij} \in \{0, 1\}$ define a tour, such that $x_{ij} = 1$ if a salesman goes from location i to location j. Then, a tour can be represented by a set of variables

$$
\begin{array}{cccc}
 & x_{12} & x_{13} & x_{14} \\
x_{21} & & x_{23} & x_{24} \\
x_{31} & x_{32} & & x_{34} \\
x_{41} & x_{42} & x_{43} &
\end{array}
$$

and the tour cost will be equal to

$$
\sum_{i,j \in \{1,2,3,4\},\, i \neq j} c_{ij} x_{ij}.
$$

What are the constraints on x_{ij}?

TSP: Tour Constraints

What we know about definition of the tour is that the salesman once visits any location i (arrives) and always departs from it, only once.

	x_{12}	x_{13}	x_{14}	$x_{12} + x_{13} + x_{14} = 1$
x_{21}		x_{23}	x_{24}	$x_{21} + x_{23} + x_{24} = 1$
x_{31}	x_{32}		x_{34}	$x_{31} + x_{32} + x_{34} = 1$
x_{41}	x_{42}	x_{43}		$x_{41} + x_{42} + x_{43} = 1$
$x_{21} + x_{31} + x_{41}$ $= 1$	$x_{12} + x_{32} + x_{42}$ $= 1$	$x_{13} + x_{23} + x_{43}$ $= 1$	$x_{14} + x_{24} + x_{34}$ $= 1$	

TSP: Problem Formulation

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

subject to

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

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

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

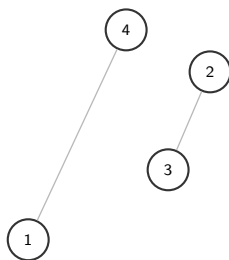If we solve the above problem, we might obtain a solution like this:

TSP: Problem Formulation

If we solve the above problem, we might obtain a solution like this:

$$
\begin{array}{cccc}
  & 0 & 0 & 1 \\
0 &   & 1 & 0 \\
0 & 1 &   & 0 \\
1 & 0 & 0 &
\end{array}
$$

The solution suggests that from 1 we go to 4, from 2 we go to 3, from 3 we go to 2, from 4 go to 1.

TSP: Problem Formulation

Graphical interpretation of the obtained solution. Clearly, the solution does not give us a proper tour.
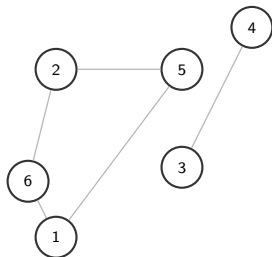
TSP: a Larger Problem Instance

	3	8	12	11	1
2		7	8	2	8
12	4		5	4	7
3	9	1		9	5
5	6	4	12		4
6	1	3	11	4	

with solution

	0	0	0	0	1
0		0	0	1	0
0	0		1	0	0
0	0	1		0	0
1	0	0	0		0
0	1	0	0	0	

TSP: Problem Formulation

In this solution we can clearly see the problem, the tour
$1 \rightarrow 5 \rightarrow 2 \rightarrow 6 \rightarrow 1$ does not include locations 3 and 4, and therefore is
called a subtour.

This observation clearly states that the above formulation does not
represent the problem correctly, and needs to be corrected. In
other words, subtours must be eliminated.

TSP: Subtour Elimination

Consider the subtour $1 \to 5 \to 2 \to 6 \to 1$ from the above picture:

- ▶ It contains 4 locations
- ▶ It contains 4 "arcs" between them

For a proper tour through all locations from 1 to 6, how many "arcs" there should be between 1, 2, 5, 6?

In a properly defined tour, there should be no more than $S - 1$ arcs between any S locations. This idea results in the additional set of constraints:

$$\sum_{(i,j) \in A(S)} x_{ij} \leq |S| - 1, \qquad \forall S \subset V, 2 \leq |S| \leq n - 2,$$

$$A(S) = \{(i,j) : i, j \in S, i \neq j\}.$$

Dantzig, George, Ray Fulkerson, and Selmer Johnson. "Solution of a Large-Scale Traveling-Salesman Problem." Journal of the Operations Research Society of America 2.4 (1954): 393-410.

TSP: Challenge of Subtour Elimination

The challenge is that for every subset S of locations, we need to incorporate a corresponding constraint

$$\sum_{(i,j)\in A(S)} x_{ij} \leq |S| - 1.$$

The number of all subsets S, $2 \leq |S| \leq n - 2$ of n locations is $2^n - 2n - 2$, i.e., grows exponentially fast.

0	0	1	1
0	1	1	0
1	1	0	0
1	0	1	0
1	0	0	1
0	1	0	1

A representation of 6 possible subsets (subtours) of 4 locations.

TSP: D−F−J Problem Formulation for the ATSP

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

subject to

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

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

$$\sum_{(i,j)\in A(S)} x_{ij} \leq |S| - 1, \qquad \forall S \subset V, |S| \geq 2,$$

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

If we solve the above problem, we might obtain a solution like this:

TSP: D−F−J Problem Formulation for the STSP

Let $x_{ij} = 1$ iff edge i, j, $i < j$ is traversed by the salesman in the unknown direction.

$$\min_{x_{ij}} \sum_{i,j=1,\, i<j}^{n} x_{ij} c_{ij}$$

subject to

$$\sum_{j=1,\, i<j}^{n} x_{ij} + \sum_{j=1,\, j<i}^{n} x_{ji} = 2, \qquad i = 1, \ldots, n,$$

$$\sum_{(i,j)\in E(S)} x_{ij} \leq |S| - 1, \qquad \forall S \subset V, |S| \geq 3,$$

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

$$E(S) = \{\{i, j\} : i, j \in S,\ i < j\}.$$

TSP: Another Approach to Subtour Elimination

We will consider another approach for elimination of subtours, or, to put it differently, separating "good" solutions like that

$$
\begin{array}{cccc}
 & 0 & 1 & 0 \\
0 & & 0 & 1 \\
0 & 1 & & 0 \\
1 & 0 & 0 &
\end{array}
$$

Tour: 1 - 3 - 2 - 4 - 1

from "bad" solutions like this:

$$
\begin{array}{cccc}
 & 0 & 0 & 1 \\
0 & & 1 & 0 \\
0 & 1 & & 0 \\
1 & 0 & 0 &
\end{array}
$$

Subtours: 1 - 4 - 1; 2 - 3

TSP: Another Approach for Subtour Elimination

Arguably, what makes the tour $1 \to 3 \to 2 \to 4 \to 1$ a valid tour for the TSP is the possibility to attach labels to locations that denote the order in which locations are travelled (in brackets):

$$1 \to 3 \to 2 \to 4 \to 1 \implies 1(1) \to 3(2) \to 2(3) \to 4(4) \to 1$$

In case of subtours $1 \to 4 \to 1; 2 - 3$ you can not attach such labels.

(One might try to assign labels like $1(1) \to 4(2) \to 1; 2(3) - 3(4)$, but the connection is missing between 4 and 2 in order for labels to be valid!)

TSP: Another Approach for Subtour Elimination

Formally, let $u_i \geq 0$ be the variables introducing the sequence structure. We need:

- ▶ $u_1 = 1$ (location 1 is the depot)
- ▶ $u_j = u_i + 1$ only if we travel directly from i to j
- ▶ Recall that x_{ij} defines whether or not we travel directly from i to j

The following constraints were proposed in Miller, Tucker, Zemlin (1960):

$$u_j \geq u_i - n(1 - x_{ij}) + 1, \quad j = 2, \ldots, n, \, i = 1, \ldots, n, \, i \neq j,$$
$$u_i \geq 0, \qquad\qquad\qquad\qquad\qquad\quad i = 2, \ldots, n.$$

Miller, Clair E., Albert W. Tucker, and Richard A. Zemlin. "Integer Programming Formulation of Traveling Salesman Problems." Journal of the ACM (JACM) 7.4 (1960): 326–329.

TSP Subtour Elimination: Problem Formulation (M−T−Z)

The valid ATSP formulation can be expressed as follows:

M−**T**−**Z** :

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

subject to

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

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

$$u_1 = 1,$$

$$u_j \geq u_i - n(1 - x_{ij}) + 1, \qquad i = 1, \ldots, n, \, j = 2, \ldots, n, \, i \neq j,$$

$$u_i \geq 0, \qquad\qquad i = 2, \ldots, n,$$

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

TSP M−T−Z Formulation: How to Verify

Proposition: a feasible integer solution to **M−T−Z** does not contain a subtour not connected to depot 1.

Proof. Suppose a feasible x^* implies a subtour over $S = \{i_1, \ldots, i_k\}$, $|S| = k$, $1 \notin S$:

$$x_{i_1} \to x_{i_2} \to x_{i_3} \to \ldots \to x_{i_k} \to x_{i_1}.$$

Let us aggregate constraints associated with active arcs in the subtour:

$$u_{i_{t+1}} \geq u_{i_t} - n(1 - x^*_{i_{t+1} i_t}) + 1, \qquad t = 1, \ldots, k-1, \quad (1)$$

$$u_{i_1} \geq u_{i_k} - n(1 - x^*_{i_k i_1}) + 1. \quad (2)$$

We will obtain

$$\sum_{j \in S} u_j \geq \sum_{i \in S} u_i - \sum_{t=1}^{k} n(1 - 1) + k \times 1,$$

which is an infeasible inequality. Hence, the assumption is not valid.□

Improved TSP M−T−Z Formulation (D−L)

M−T−Z subtour elimination approach was significantly improved by Desrochers and Laporte (1991):

- ▶ $2 \leq u_i \leq n$ for $i = 2, \ldots, n$ where
- ▶ $u_i = 2$ when $x_{1i} = 1$
- ▶ $u_i = n$ when $x_{i1} = 1$
- ▶ $u_j = u_i + 1$ when $x_{ij} = 1$ (instead of weaker $u_j \geq u_i + 1$ inequality in M−T−Z)

$$u_i - u_j + (n-1)x_{ij} + (n-3)x_{ji} \leq n-2, \quad i, j = 2, \ldots, n, i \neq j,$$
$$u_i \leq n - 1 - (n-3)x_{1i} + x_{i1} \qquad\qquad i = 2, \ldots, n,$$
$$u_i \geq 3 + (n-3)x_{i1} - x_{1i} \qquad\qquad i = 2, \ldots, n,$$
$$u_i \geq 0, \qquad\qquad i = 2, \ldots, n.$$

$O(n^2)$ sized formulation.

Desrochers, Martin, and Gilbert Laporte. "Improvements and extensions to the Miller-Tucker-Zemlin subtour elimination constraints." Operations Research Letters 10.1 (1991): 27–36.

D−L Subtour Elimination

M−T−Z: suppose that $x_{ij} = 1$, then

$$u_j \geq u_i - n(1 - x_{ij}) + 1 = u_i + 1,$$

$$u_i \geq u_j - n(1 - x_{ji}) + 1 = u_j - n + 1 \quad \text{(redundant)}.$$

and we obtain

$$u_j \geq u_j + 1.$$

Yet, the following more accurate relation should in fact hold

$$u_j = u_i + 1.$$

D−L: suppose that $x_{ij} = 1$, then

$$u_i - u_j + (n-1)x_{ij} + (n-3)x_{ji} = u_i - u_j + n - 1 \leq n - 2,$$

$$u_j - u_i + (n-1)x_{ji} + (n-3)x_{ij} = u_j - u_i + n - 3 \leq n - 2,$$

we obtain exactly $u_j = u_i + 1$.

TSP D−L Problem Formulation

The improved M−T−Z ATSP formulation is obtained:

D−L :

$$\min_{x_{ij},\, u_i} \sum_{i,\, j=1,\, i \neq j}^{n} x_{ij} c_{ij} \tag{3}$$

subject to (4)

$$\sum_{j=1,\, i \neq j}^{n} x_{ij} = 1, \quad \sum_{j=1,\, i \neq j}^{n} x_{ji} = 1, \qquad\qquad i = 1, \ldots, n, \tag{5}$$

$$u_i - u_j + (n-1)x_{ij} + (n-3)x_{ji} \leq n-2, \qquad i, j = 2, \ldots, n,\ i \neq j, \tag{6}$$

$$u_i \geq 3 + (n-3)x_{i1} - x_{1i}, \qquad\qquad\qquad i = 2, \ldots, n, \tag{7}$$

$$u_i \leq n - 1 - (n-3)x_{1i} + x_{i1}, \qquad\qquad\quad i = 2, \ldots, n, \tag{8}$$

$$x_{ij} \in \{0, 1\}, \qquad\qquad\qquad\qquad i, j = 1, \ldots, n,\ i \neq j. \tag{9}$$

M−T−Z in Practice

- In practice, the applicability of pure M−T−Z (as well as D−L) subtour elimination constraints remains limited, useful only for problems with size of 100 to 100, max 200 nodes.
- Nevertheless, the approach is conceptually extremely important:
  - It requires only n additional variables and $O(n^2)$ constraints in the extended solution space
  - It allows to incorporate side constraints on the optimal tour
  - It allows to model extended problems with several vehicles or side constraints on a tour or constraints on visiting time requirements (time windows constraints)
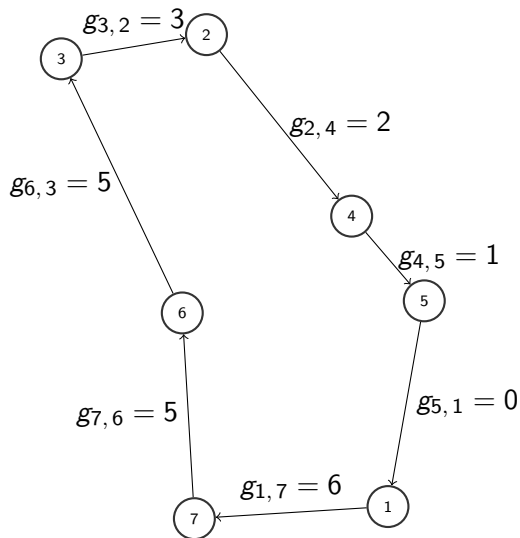
TSP Subtour Elimination (G−G)

A completely different idea for the ATSP was initiated by Gavish and Graves (1978): the network flow. Suppose that from the depot, node 1, we carry $n - 1$ units of a commodity via the arcs defined by binary x_{ij} variables. Every node should get 1 unit of that commodity.

If there is a tour, we can send a unit of a commodity to every location. If there is a subtour, we can not.

B. Gavish, S.C. Graves, "The Travelling Salesman Problem and Related Problems", Working Paper OR 078-78, Massachusetts Institute of Technology, Operations Research Center, Boston, 1978.

$g_{3,2} = 3$

$g_{2,4} = 2$

$g_{6,3} = 5$

$g_{4,5} = 1$

$g_{5,1} = 0$

$g_{7,6} = 5$

$g_{1,7} = 6$

TSP Subtour Elimination (G−G)

Let $g_{ij} \geq 0$ be the flow from node i to j. Then, the ATSP can be formulated as

G−G :

$$\min_{x_{ij}, g_{ij}} \sum_{i,j=1, i \neq j}^{n} x_{ij} c_{ij} \tag{10}$$

subject to $\tag{11}$

$$\sum_{j=1, i \neq j}^{n} x_{ij} = 1, \quad \sum_{j=1, i \neq j}^{n} x_{ji} = 1, \qquad\qquad i = 1, \dots, n, \tag{12}$$

$$\sum_{j=1}^{n} g_{ji} = \sum_{j=2}^{n} g_{ij} + 1, \qquad\qquad i = 2, \dots, n, \tag{13}$$

$$0 \leq g_{ij} \leq (n-2) x_{ij}, \qquad\qquad i, j = 2, \dots, n, \, i \neq j, \tag{14}$$

$$g_{1i} = (n-1) x_{1i}, \qquad\qquad i = 2, \dots, n, \tag{15}$$

$$x_{ij} \in \{0, 1\}, \qquad\qquad i, j = 1, \dots, n, \, i \neq j. \tag{16}$$

$O(n^2)$ sized formulation.

TSP G−G Formulation: How to Verify

Proposition: a feasible integer solution to **G−G** does not contain a subtour not connected to depot 1.

Proof. Suppose a feasible x^* implies a subtour over $S = \{i_1, \ldots, i_k\}$, $|S| = k$, $1 \notin S$:

$$x_{i_1} \to x_{i_2} \to x_{i_3} \to \ldots \to x_{i_k} \to x_{i_1}.$$

Let us aggregate (13) constraints associated with active arcs in the subtour:

$$g_{i_{t-1} i_t} = g_{i_t i_{t+1}} + 1, \qquad t = 2, \ldots, k-1, \qquad (17)$$

$$g_{i_k i_1} = g_{i_1 i_2} + 1, \qquad (18)$$

$$g_{i_{k-1} i_k} = g_{i_k i_1} + 1. \qquad (19)$$

We will obtain

$$0 = 0 + k \times 1,$$

which is an infeasible identity. Hence, the assumption is not valid. $\square$

TSP Subtour Elimination (G−G): Improved Idea

Note that the flow is never less than the value of x_{ij}. In other words, the flow is equal to $n-1$, $n-2$ and so on, but the flow is always at least 1 and along the arc where $x_{ij} = 1$. Therefore, the G−G formulation could be enhanced by the following set of valid, strengthening inequalities:

$$g_{ij} \geq x_{ij}, \qquad\qquad i, j = 2, \ldots, n, \qquad (20)$$

which, despite the strengthening property, complicates the LP relaxation problem.

We still can obtain the formulation of the enhanced strength without increasing the size of the formulation!

TSP Subtour Elimination (G−G): Improved Idea

Let the flow from i to j be defined as

$$g_{ij} = \tilde{g}_{ij} + x_{ij}$$

with $\tilde{g}_{ij} \geq 0$. Now, the flow satisfies condition (20) automatically and the remaining G−G formulation is as follows:

$$(n-1)x_{1i} + \sum_{j=2}^{n}(\tilde{g}_{ji} + x_{ji}) = \sum_{j=2}^{n}(\tilde{g}_{ij} + x_{ij}) + 1, \qquad i = 2, \ldots, n, \qquad (21)$$

$$\tilde{g}_{ij} + x_{ij} \leq (n-2)x_{ij}, \qquad\qquad i, j = 2, \ldots, n, \ i \neq j, \qquad (22)$$

$$\tilde{g}_{ij} \geq 0, \qquad\qquad i, j = 2, \ldots, n, \ i \neq j, \qquad (23)$$

$$x_{ij} \in \{0, 1\}, \qquad\qquad i, j = 1, \ldots, n, \ i \neq j. \qquad (24)$$

(22) is simplified as $\tilde{g}_{ij} \leq (n-3)x_{ij}$. (21) is simplified based on the following observations

$$\sum_{j=2}^{n} x_{ji} = 1 - x_{1i},$$

$$\sum_{j=2}^{n} x_{ij} = 1 - x_{i1}.$$

TSP Subtour Elimination (G−G m.)

G−G m. :

$$(n-2)x_{1i} + \sum_{j=2, j\neq i}^{n} \widetilde{g}_{ji} = \sum_{j=2, j\neq i}^{n} \widetilde{g}_{ij} + 1 - x_{i1}, \qquad i = 2, \ldots, n, \qquad (25)$$

$$\widetilde{g}_{ij} \leq (n-3)x_{ij}, \qquad\qquad i, j = 2, \ldots, n, \, i \neq j, \qquad (26)$$

$$\widetilde{g}_{ij} \geq 0, \qquad\qquad i, j = 2, \ldots, n, \, i \neq j. \qquad (27)$$

$O(n^2)$ sized formulation.

Remark. Notice the interesting interpretation of the above model. It turns out that the modified G−G model is such that every location requires 1 unit of commodity, except to the last location in the tour! That is why the salesman leaves with $n-2$ units of the commodity from the depot.

Compact Formulation for the STSP Case
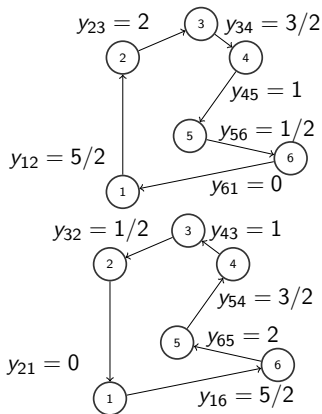
An illustration of the network flow model behind the compact formulation of the STSP with $n = 6$ locations. $5/2$ units of a commodity flow both into locations 2 and 6. The total amount of flow in both directions over any active edge is equal to $5/2$.

Compact Formulation for the STSP Case

NCP :

$$\min \sum_{i=1}^{n} \sum_{j=1,\, i<j}^{n} c_{ij} x_{ij} \tag{28}$$

subject to

$$\sum_{j=1,\, i<j}^{n} x_{ij} + \sum_{j=1,\, i>j}^{n} x_{ji} = 2, \qquad\qquad i = 1, \ldots, n, \tag{29}$$

$$\frac{n-1}{2} x_{1i} + \sum_{j=2,\, j\neq i}^{n} y_{ji} = \sum_{j=2,\, j\neq i}^{n} y_{ij} + 1, \qquad i = 2, \ldots, n, \tag{30}$$

$$y_{ij} + y_{ji} = \frac{n-1}{2} x_{ij}, \qquad\qquad i, j = 2, \ldots, n,\, i < j, \tag{31}$$

$$y_{ij} \geq 0, \qquad\qquad i, j = 2, \ldots, n,\, i \neq j, \tag{32}$$

$$x_{ij} \in \{0, 1\}, \qquad\qquad i, j = 1, \ldots, n,\, i < j. \tag{33}$$

Computational Results

Instance	**G–G**		**G–G m.**		**D–L**		**M–T–Z**		**D–F–J**		
	LP relax.	CPU	LP relax.	CPU	LP relax.	CPU	LP relax.	CPU	LP relax.	# cuts	CPU
d112	11,041.35	0.2	11,041.92	0.2	11,047.13	0.2		0.8	11,106.75	41	0.8
d126	118,716.15	0.5	118,719.00	0.4	121,129.5	0.3		1.6	123,199.00	129	1.2
d134	5,598.13	0.4	5,598.19	0.3	5,604.02	0.3		1.3	5,611.50	43	0.8
d176	8,533.92	0.7	8,533.95	0.7	8,565.51	0.8		2.7	8,585.00	82	2.8
d188	10,142.94	0.5	10,143.19	0.5	10,169.90	0.4		2.7	10,223.00	53	3.2
d563	25,870.74	8.5	25,870.75	8.9	25,923.71	12.3		50.6	25,949.06	122	18.4
d849	37,448.83	21.9	37,448.83	24.0	37,466.00	35.8		217.6	37,476.00	34	44.8
d895	106,963.10	28.2	106,963.14	31.7	107,397.22	25.2		158.5	107,669.61	401	75.0
d932	467,996.96	83.0	467,997.07	78.0	471,547.39	41.9		1622.6	479,837.04	257	32.7

Computational results of solving LP relaxations of integer and mixed integer linear ATSP formulations.

Instance	**G–G**			**G–G m.**			**D–L**			**M–T–Z**			**D–F–J**		
	Obj	Gap	CPU	Obj	Gap	CPU	Obj	Gap	CPU	Obj	Gap	CPU	Obj	Gap	CPU
dc112	11,109	0.00	155.6	11,109	0.00	461.4	11,110	0.26	lim			lim	11,109	0.00	11.8
dc126	123,235	0.00	51.5	123,235	0.00	201.3	123,243	0.21	lim			1506.1	123,235	0.00	20.9
dc134	5,612	0.00	46.9	5,612	0.00	72.5	5,612	0.04	lim			280.4	5,612	0.00	6.7
dc176	8,587	0.00	220.3	8,587	0.00	201.6	8,587	0.02	lim			1207.1	8,587	0.00	39.5
dc188	10,225	0.00	383.6	10,225	0.00	508.0	10,225	0.00	5192.0			4455.3	10,225	0.00	96.7
dc563	26,178	1.04	lim	26,067	0.61	lim	25,973	0.12	lim			lim	25,951	0.00	6231.0
dc849	39,977	6.32	lim	39,977	6.32	lim	37,481	0.03	lim			lim	37,476	0.00	301.0
dc895	119,216	10.28	lim	119,216	10.28	lim	108,136	0.54	lim			lim	107,715	0.03	lim
dc932	555,621	15.77	lim	555,621	15.77	lim	484,143	2.31	lim			lim	555,621	13.64	lim

Computational results of solving integer and mixed integer ATSP formulations, lim = 2 hours time limit.

- ▶ how could we solve the LP relaxation problem for the D−F−J formulation for $n > 200$?
- ▶ how do we solve the IP problem of the D−F−J formulation?

Solving the D−F−J Model 1

In the set of constraints

$$\sum_{(i,j)\in A(S)} x_{ij} \leq |S| - 1, \qquad \forall S \subset V, 2 \leq |S| \leq n - 2,$$

many constraints are not binding! If so, they are not needed to be present in the problem formulation. We do not know which ones are nonbinding, though. Hence, we omit them all and solve

$$\mathbf{P} : \min_{x_{ij}} \sum_{i,j=1,\, i\neq j}^{n} x_{ij} c_{ij}$$

subject to

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

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

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

Solving the D−F−J Model 2

Let $x^* \in arg \min \mathbf{P}$.

Question: construct directed graph $G = (V, A)$ where $A = \{(i, j) : x_{ij}^* > 0.5\}$, does there exist $S \subset V$:

$$\sum_{(i,j) \in A(S)} x_{ij}^* > |S| - 1?$$

If yes, then add

$$\sum_{(i,j) \in A(S)} x_{ij} \leq |S| - 1$$

violated inequality to $\mathbf{P}$.
If no, STOP, problem solved.
Continue until no violated S is found.

This approach implies solving IP problems iteratively that grow in size every iteration. That idea does not scale well.

Solving the D−F−J Model: Separation

Instead, we would like to solve this problem first

$$\mathbf{P}_{LP} : \min_{x_{ij}} \sum_{i,j=1,\, i \neq j}^{n} x_{ij} c_{ij}$$

subject to

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

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

$$x_{ij} \geq 0, \qquad\qquad i, j = 1, \ldots, n, \ i \neq j.$$

then take an optimal solution x^* and identify violated D−F−J constraints (cuts), i.e., one or more S, such that:

$$\sum_{(i,j) \in A(S)} x_{ij}^* > |S| - 1,$$

and add them to $\mathbf{P}_{LP}$. Iterate solving $\mathbf{P}_{LP}$ and adding cuts.
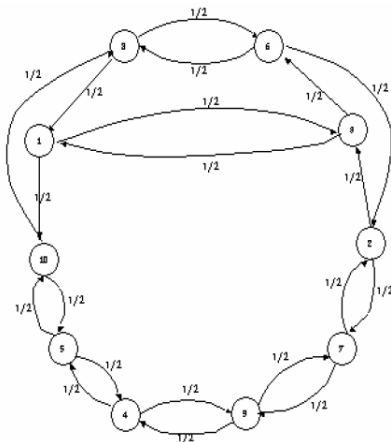
Solving the D−F−J Model

The desired result is formulation like that

$$\mathbf{P}_{LP} : \min_{x_{ij}} \sum_{i,j=1,\, i\neq j}^{n} x_{ij} c_{ij}$$

subject to

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

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

$$\sum_{(i,j)\in A(S)} x_{ij} \leq |S| - 1, \qquad\qquad S \in \mathcal{U},$$

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

where we prove that no constraint

$$\sum_{(i,j)\in A(S)} x_{ij} \leq |S| - 1,$$

for $S \notin \mathcal{U}$ is violated.

Cost = 878 (Optimal Cost = 881)

Observation: D−F−J constraints do not describe the convex hull of the
TSP. The above solution satisfies all D−F−J constraints but nevertheless
is fractional. We can proceed with branch-and-bound, though!

Equivalent Representation of D−F−J Constraints

Proposition:

$$\sum_{j=1,\, i\neq j}^{n} x_{ij} = 1, \qquad\qquad i = 1, \ldots, n, \qquad (34)$$

$$\sum_{j=1,\, i\neq j}^{n} x_{ji} = 1, \qquad\qquad i = 1, \ldots, n, \qquad (35)$$

$$\sum_{(i,j)\in A(S)} x_{ij} \leq |S| - 1, \qquad \forall S \subset V, 2 \leq |S| \leq n - 2, \qquad (36)$$

is equivalent to

$$\sum_{j=1,\, i\neq j}^{n} x_{ij} = 1, \qquad\qquad i = 1, \ldots, n, \qquad (37)$$

$$\sum_{j=1,\, i\neq j}^{n} x_{ji} = 1, \qquad\qquad i = 1, \ldots, n, \qquad (38)$$

$$\sum_{i\in S,\, j\in \bar{S}} x_{ij} \geq 1, \qquad \forall S \subset V, 2 \leq |S| \leq n - 2, \qquad (39)$$

where

$$\bar{S} = V \backslash S.$$

Equivalent Representation of D−F−J Constraints

Proof: Consider

$$\sum_{j=1,\, i\neq j}^{n} x_{ij} = 1, \qquad\qquad i \in S. \tag{40}$$

and aggregate the above constraints:

$$\sum_{i\in S}\sum_{j=1,\, i\neq j}^{n} x_{ij} = |S| \iff \tag{41}$$

$$\sum_{i\in S}\left(\sum_{j\in S,\, i\neq j} x_{ij} + \sum_{j\in \bar{S}} x_{ij}\right) = |S| \iff \tag{42}$$

$$\sum_{(i,j)\in A(S)} x_{ij} + \sum_{i\in S}\sum_{j\in \bar{S}} x_{ij} = |S|. \tag{43}$$

From the above, it is clear that

$$\sum_{(i,j)\in A(S)} x_{ij} \le |S|-1 \iff \sum_{i\in S}\sum_{j\in \bar{S}} x_{ij} \ge 1. \tag{44}$$

$\square$

TSP: Challenge of Subtour Elimination

Note that $(S, \bar{S})$ defines a cut-set and constraint

$$\sum_{i \in S} \sum_{j \in \bar{S}} x_{ij} \geq 1.$$

enforces that the capacity of this cut-set defined by x_{ij} is not less 1.

Consider $i \in V$, $i \neq 1$ and
$\mathcal{S}(i) = \{S \subset V : 2 \leq |S| \leq n - 2, i \in S\}$. Then, the set of constraints

$$\sum_{i \in S} \sum_{j \in \bar{S}} x_{ij} \geq 1, \qquad\qquad S \in \mathcal{S}(i), \qquad\qquad (45)$$

ensures that the capacity of any cut-set containing i is not less than 1.

Minimum s-t Cut Problem

Input: directed graph $G = (V, A)$ with arc weights $c_a \geq 0$, $e \in A$, vertices s and t.

Output: A minimum cut S that separates s and t, that is, a partition of nodes of G into S and $V \backslash S$ with $s \in S$ and $t \in V \backslash S$ that minimizes the weight of arcs going across the partition, i.e.,

$$\min_{S} \sum_{i \in S} \sum_{j \in \bar{S}} c_{ij} \tag{46}$$

FIGURE 9.10
A minimum cut for the Seervada Park maximum flow problem.
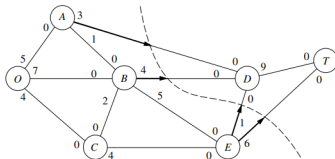
The minimum cut is a combinatorial problem, but....

Minimum s-t Cut Problem

The min cut problem is well connected by LP duality to the maximum flow problem
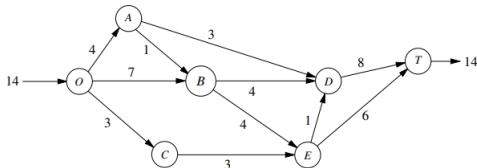

FIGURE 9.8
Optimal solution for the Seervada Park maximum flow problem.

which is extremely well solved with specialized network simplex algorithms. I refer you to the Hillier and Lieberman, Introduction to Operations Research textbook, https://github.com/mrmurilo75/Intro_to_Operations_Research.

Minimum $s - t$ Cut Problem: Connection to D$-$F$-$J

Input: Consider directed graph $G = (V, A)$ with arc weights $c_a = x_{ij}^*$, $(i, j) \in A$, vertices i and 1

Output: A minimum cut S that separates i and 1, that is, a partition of nodes of G into S and $V \backslash S$ with $i \in S$ and $1 \in V \backslash S$ that minimizes the weight of arcs going across the partition, i.e.,

$$\min \text{cut}(x^*, 1, i) = \min_S \sum_{i \in S} \sum_{j \in \bar{S}} x_{ij}^* \qquad (47)$$

Two cases:

1. $\min \text{cut}(x^*, 1, i) < 1$, which implies that the D$-$F$-$J inequality associated with set S^* in the argminimum of the min-cut problem is violated.

2. $\min \text{cut}(x^*, 1, i) \geq 1$ implies that any D$-$F$-$J inequality with $S \in \mathcal{S}(i)$ is satisfied.

Efficient Separation of violated D−F−J Cuts

Input: $\{x_{ij}^*\}$ = **opt. solution of the LP relaxation of** $\mathbf{P}_{LP}$
$G = (V, \emptyset)$
for $(i, j) \in \{V \times V, i \neq j\}$ **do**
    if $x_{ij}^* > 0$ **then**
        $G.add\_arc(i, j)$
        $G.arc(i, j).capacity = x_{ij}^*$
    end if
end for
for $i \in V \backslash \{1\}$ **do**
    if min cut $(G, 1, i) < 1$ **then**
        $S, \bar{S} = \arg\min \text{cut}(G, 1, i)$
        if $i \in S$ **then**
            $add\_cut \displaystyle\sum_{i,j \in S, i \neq j} x_{ij} \leq |S| - 1$
        else
            $add\_cut \displaystyle\sum_{i,j \in \bar{S}, i \neq j} x_{ij} \leq |\bar{S}| - 1$
        end if
    end if
end for

Separation of D−F−J Inequalities: Key Result

Proposition: The separation problem, i.e., finding a violated
D−F−J inequality or proving that none exists is polynomially
solvable.

Home Exercise

Proposition: If inequality

$$\sum_{i,\,j\in S,\,i\neq j} x_{ij} \leq |S| - 1$$

is violated, then

$$\sum_{i,\,j\in \bar{S},\,i\neq j} x_{ij} \leq |\bar{S}| - 1$$

is violated as well, $\bar{S} = V\backslash S$.

Implication: when the min cut problem is solved and cut-set $(S, \bar{S})$ is obtained, we might add constraint with smaller number of nodes, S vs $\bar{S}$ to make sure our LP stays sparse.

Improved Separation of violated D−F−J Cuts

```
Input: {x*ᵢⱼ} = opt. solution of the LP relaxation of P
G = (V, ∅)
for (i, j) ∈ {V × V, i ≠ j} do
    if x*ᵢⱼ > 0 then
        G.add_arc(i, j)
        G.arc(i, j).capacity = x*ᵢⱼ
    end if
end for
for i ∈ V\{1} do
    if min cut (G, 1, i) < 1 then
        S, S̄ = argmin cut(G, 1, i)
        if |S| ≤ |S̄| then
            add_cut  ∑        xᵢⱼ ≤ |S| − 1
                   i,j∈S, i≠j
        else
            add_cut  ∑        xᵢⱼ ≤ |S̄| − 1
                   i,j∈S̄, i≠j
        end if
    end if
end for
```

TSP: Symmetric Version

Proposition:

$$\sum_{j=1,\, i\neq j}^{n} x_{ji} + \sum_{j=1,\, i\neq j}^{n} x_{ji} = 2, \qquad\qquad i = 1, \ldots, n, \qquad (48)$$

$$\sum_{(i,j)\in E(S)} x_{ij} \leq |S| - 1, \qquad\qquad \forall S \subset V, 3 \leq |S| \leq n - 2, \qquad (49)$$

is equivalent to

$$\sum_{j=1,\, i\neq j}^{n} x_{ji} + \sum_{j=1,\, i\neq j}^{n} x_{ji} = 2, \qquad\qquad i = 1, \ldots, n, \qquad (50)$$

$$\sum_{i\in S,\, j\in \bar{S}} x_{ij} \geq 2, \qquad\qquad \forall S \subset V, 3 \leq |S| \leq n - 2, \qquad (51)$$

where

$$\bar{S} = V \backslash S.$$

Proof. Home Exercise □

Minimum $s - t$ Cut Problem: Connection to D$-$F$-$J Cuts

Input: Consider undirected graph $G = (V, E)$ with edge weights
$c_e = x_{ij}^*$, $\{i, j\} \in E$, vertices i and 1
Output: A minimum cut S that separates i and 1, that is, a
partition of nodes of G into S and $V \backslash S$ with $i \in S$ and $1 \in V \backslash S$
that minimizes the weight of edges going across the partition, i.e.,

$$\min \text{cut}(x^*, 1, i) = \min_S \sum_{i \in S, j \in \bar{S}, i < j} x_{ij}^* + \sum_{i \in S, j \in \bar{S}, j < i} x_{ji}^* \qquad (52)$$

Two cases:

1. $\min \text{cut}(x^*, 1, i) < 2$, which implies that the D$-$F$-$J
   inequality associated with set S^* in the argminimum of
   min-cut problem is violated.

2. $\min \text{cut}(x^*, 1, i) \geq 2$ implies that any D$-$F$-$J inequality with
   $S \in \mathcal{S}(i)$ is satisfied.

Efficient Separation of violated D−F−J Cuts

Input: $\{x_{ij}^*\} = $ **opt. solution of an LP relaxation**
$G = (V, \emptyset)$
for $\{i, j\} \in \{V \times V, i < j\}$ **do**
    if $x_{ij}^* > 0$ **then**
        $G.add\_edge(i, j)$
        $G.edge(i, j).capacity = x_{ij}^*$
    end if
end for
for $i \in V \setminus \{1\}$ **do**
    if min cut $(G, 1, i) < 2$ **then**
        $S, \bar{S} = $ argmin cut$(G, 1, i)$
        if $|S| \leq |\bar{S}|$ **then**
            $add\_cut \displaystyle\sum_{i,j \in S, i<j} x_{ij} \leq |S| - 1$
        else
            $add\_cut \displaystyle\sum_{i,j \in \bar{S}, i<j} x_{ij} \leq |\bar{S}| - 1$
        end if
    end if
end for

Summary

- ▶ The TSP has been defined
- ▶ Solution approaches based on mathematical programming have been described
- ▶ Exponentially sized and polynomially sized formulations have been described
- ▶ Polynomially sized formulations can be implemented with help of MILP solver right away
- ▶ Exponentially sized formulations have to be solved in a cutting planes framework
- ▶ Cutting planes can be incorporated into the solution process with help of advanced tools the solvers provide (i.e., callbacks)