A Short History of the Traveling Salesman Problem

by

Gilbert Laporte

Canada Research Chair in Distribution Management,
Centre for Research on Transportation (CRT) and GERAD
HEC Montréal, Canada

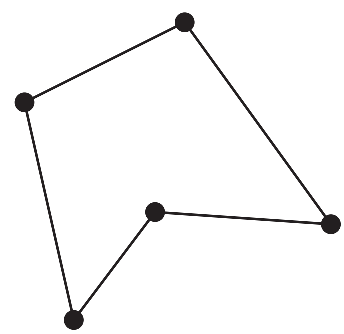
Traveling Salesman Problem (TSP)

Undirected graph $G = (V, E)$
or directed graph $G = (V, A)$
where

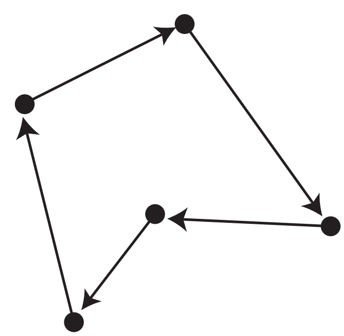
$$V = \{i, \ldots, n\} : \text{vertex set}$$
$$E = \{(i, j) : i, j \in V, i < j\} : \text{edge set}$$
$$A = \{(i, j) : i, j \in V, i \neq j\} : \text{arc set}$$
$$C = (c_{ij}) : \text{cost matrix defined on } E \text{ or } A.$$

TSP: determine a least cost Hamiltonian cycle or circuit (tour) on G.

Hamiltonian cycle
(symmetric TSP)



Hamiltonian circuit
(asymmetric TSP)



TSP history

BC

1950

AD

TSP history

BC **Before Computers**

1950 ———————————————

AD **After Dantzig**

Ancient history

- Euler ("Solution d'une question curieuse qui ne paraît soumise à aucune analyse"), *Mémoire de l'Académie des Sciences de Berlin* 15 (1759) 310–337 published in *Opera Omnia* (1) 7 (1766) 26–56.

  Studied the Knight's tour problem

- Kirkman, rector of the parish of Craft with Southworth, Lancashire, from 1840 to 1892. *Philosophical Transactions of the Royal Society*, London, 146 (1856) 413–418.

  "Given the graph of a polyhedron, can one always final a circuit which passes through each vertex one and only once?"

- Hamilton (1856)

  Icosian game (marketed in 1859): find paths and circuits on the dodecahedral graph, satisfying certain conditions (e.g., adjacency conditions, etc.).

  The rights were sold for £25 to a wholesaler dealer in games and puzzles.

- Menger (1930)

  Hamiltonian path: "We call this the messenger problem (because in practice the problem has to be solved by every postman, and also by many travelers): finding the shortest path joining all of a finite set of points, whose pairwise distances are known".

  Book published in 1931–32 in German: "The Traveling Salesman Problem, how he should be and what he should do to be successful in his business. By a veteran traveling salesman". Last chapter: "By a proper choice and scheduling of the tour, one can often gain so much time that we have to make some suggestions . . . The most important aspect is to cover as many location as possible without visiting a location twice".

- Tucker (1937 ?)

  Introduced the problem to Flood in (1937) according to Flood, and in 1931–32 according to Tucker, in relation with a school-bus routing study in New Jersey.

Recent history

1. **Symmetric TSP**

   The seminal article of Dantzig, Fulkerson and Johnson (1954) and the origin of branch-and-cut.

2. **Asymmetric TSP**

   Eastman's thesis (1958), the seminal articles of Land and Doig (1960) and of Little, Murty, Sweeney and Karel (1963), and the origin of branch-and-bound.

3. **Heuristics**

   The seminal articles of Croes (1958) and of Lin (1965) and the origin of local search.

References

- Biggs, N.L., Lloyd, K.E., Wilson, R., *Graph Theory, 1736–1936*, Clarendon Press, Oxford, 1977.
- Lawler, E.L., Lenstra, J.K., Rinnooy Kan, A.H.G., Shmoys, D.B. (eds.), *The Traveling Salesman Problem: A Guided Tour of Combinatorial Optimization*, Wiley, Chichester, 1985.
- Halskau, Ø., Sr., *Decompositions of Traveling Salesman Problems*, Doctoral Thesis, Norwegian School of Economics and Business Administration, 1999.
- Gutin, G., Punnen, A.P. (eds.), *The Traveling Salesman Problem and Its Variations*, Kluwer, Boston, 2002.

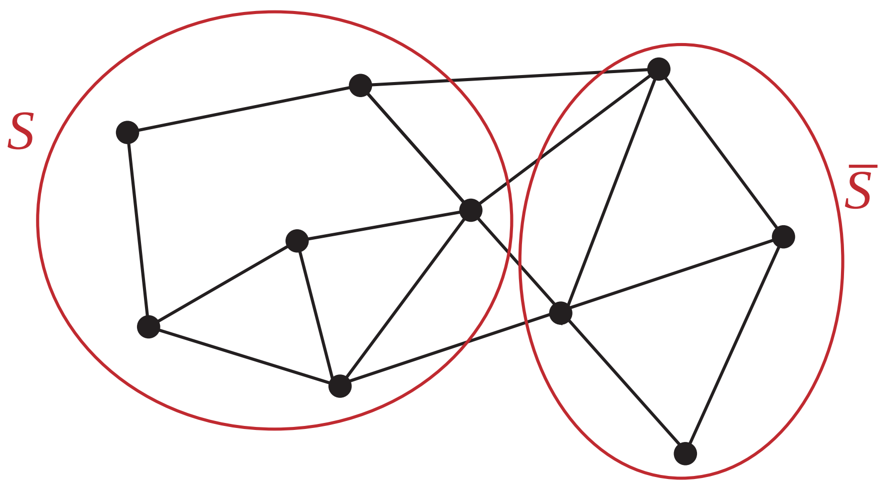
1) The Symmetric TSP: Branch-and-Cut Methods

For $S \subseteq V$, let

$$\begin{aligned} E(S) &= \{(i,j) \in E : i,j \in S\} \text{ and} \\ \delta(S) &= \{(i,j) \in E : i \in S, j \notin S \text{ or } i \notin S, j \in S\} \\ \delta(i) &= \delta(\{i\})(i \in S) \\ D(S) &= \sum_{i \in S} |\delta(i)| \end{aligned}$$

Useful "degree identity":

$$\boxed{2E(S) + \delta(S) = D(S)}$$



$$D(S) = 20, \qquad E(S) = 8, \qquad \delta(S) = 4$$

First integer linear programming formulation: Cutting planes

(Dantzig, Fulkerson and Johnson, 1954)

$$x_e = \begin{cases} 1 & \text{edge } e \in E \text{ appears on the tour} \\ 0 & \text{otherwise} \end{cases}$$

$x(F) = \sum_{e \in F} x_e \qquad (F \subseteq E)$

Minimize $\sum_{e \in E} c_e x_e$		
subject to		
(Degree constraints)	$x(\delta(i)) = 2$	$(i \in V)$
(Connectivity constraints)	$x(\delta(S)) \geq 2$	$(S \subset V, 3 \leq \lvert S \rvert \leq n-3)$
(Integrality constraints)	$x_e = 0, 1$	$e \in E$

Connectivity constraints $\quad x(\delta(S)) \geq 2$

$\Updownarrow$

Subtour elimination constraints $\quad x(E(S)) \leq |S| - 1$

because $2E(S) + \delta(S) = D(S) = 2|S|$.

Some historical notes: "49-city problem"

- Dantzig, Fulkerson, Johnson initially relaxed the integrality conditions. Feasibility regained by imposing a mix of subtour elimination constraints, connectivity contraints and "complicated types of restraints" to regain integrality. These are based on combinatorial arguments. Several computer runs are required.
- On the 49-city problem, these are unnecessary if the right subtour elimination constraints are imposed first.
- 49 or 42 cities? Seven cities (Baltimore, Wilmington, Philadelphia, Newark, New York, Hartford, Providence) appear in this order on the (Washington, Boston) edge of the 42-city problem.

Martin (1966) (CEIR, New York)

- Relaxes integrality requirements and $x_e \leq 1$!
- Imposes n degree constraints and n subtour elimination constraints: $S = \{i, \text{closest neighbour of } i\}$
- Obtains integrality first by means of the "Accelerated Euclidian algorithm" (Martin, 1963), an extension of the "Method of Integer Forms" (Gomory, 1963).
- Identifies subtour elimination constraints visually, and reoptimizes.

Results on the "42-city" problem

Pass	Total number of constraints	Integrality cuts	Iterations	Objective	Subtour eliminations constraints
1	84	10	99	646	8
2	92	7	129	686	1
3	93	10	135	699	0
		27	363		9

- Each pass requires a new computer run.

The London School of Economics years

Miliotis (1975, 1976, 1978): Branch-and-cut

1975: "Combining cutting-plane and branch-and-bound methods to solve integer programming problems: Applications to the travelling salesman problem and the 1-matching problem", Ph.D. thesis, London School of Economics.

1976: "Integer programming approaches to the travelling salesman problem", *Mathematical Programming* 10, 367–378.

1978: "Using cutting planes to solve the symmetric travelling salesman problem", *Mathematical Programming* 15, 177–188.

- Fully automated algorithms:

1976 paper: integrality reached first by branch and bound; subtour elimination constraints then introduced. $(42 \leq n \leq 64)$.

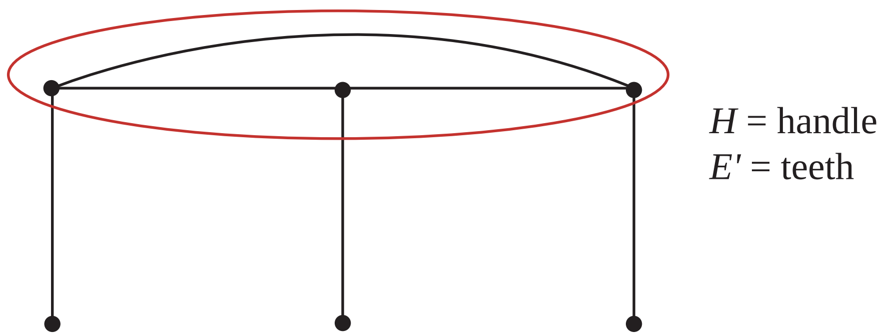
1978 paper: integrality reached by Gomory cutting planes and subtour elimination constraints then introduced (straight algorithm) or generate subtour elimination constraints before Gomory cuts (reverse algorithm). $(42 \leq n \leq 64)$.

The reverse algorithm is more efficient than the straight algorithm on larger instances and requires the generation of fewer Gomory cuts.

Land (1979): Cut-and-price

"The solution of some 100-city Travelling Salesman Problem", Working paper, London School of Economics, 1979.

- Dynamic generation of subtour elimination constraints, 2-matching constraints (Edmonds, 1965) and Gomory cuts ("as a last resort"). Most calculations carried out on a subset of the variables. A pricing scheme is used to determine whether new variables should be introduced.

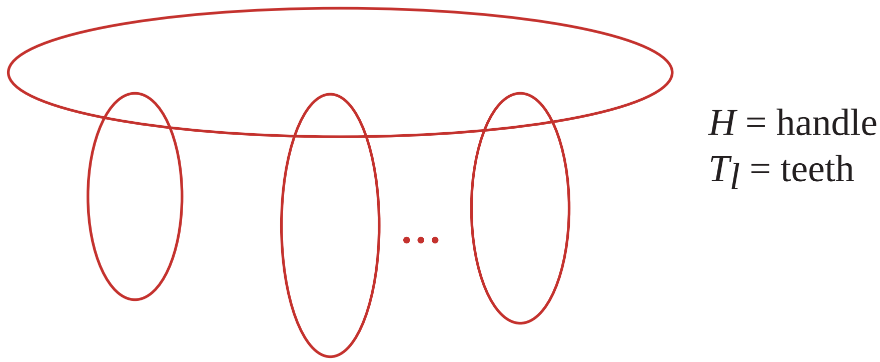


$$\sum_{e \in E(H)} x_e + \sum_{e \in E'} x_e \leq |H| + (|E'| - 1)/2$$

$H \subset V, E' \subset E, |E'| \geq 3$ and odd, $|\{i,j\} \cap H| = 1$ for all $(i,j) \in E', \{i,j\} \cap \{h,l\} = \emptyset$ for all $(i,j),(h,l) \in E', (i,j) \neq (h,l)$.

- Twelve 100-city instances out thirteen solved optimally.

Comb inequalities (Chvátal, 1973)



$$\sum_{e \in E(H)} x_e + \sum_{l=1}^{s} \sum_{e \in E(T_l)} x_e \leq |H| + \sum_{l=1}^{s} (|T_l| - 1) - (s+1)/2$$

$H, T_1, \ldots, T_l \ \subset V, s \geq 3$ and odd, $H \cap T_l \neq \emptyset$
and $T_e \backslash H \neq \emptyset (l = 1, \ldots, s), T_h \cap T_e = \emptyset$
$(h, l = 1, \ldots, s, h \neq l)$.

Other valid inequalities (see Naddef, 2002)

Clique tree inequalities
Crown inequalities
Ladder inequalities
Bicycle inequalities
Star inequalities
Bipartition inequalities
Path inequalities
Domino inequalities
Hypohamiltonian inequalities, etc.

Polyhedral theory (1970s, 1980s, 1990s)

- Study of polytopes associated with the TSP
- Facets
- Separation procedures (exact and approximate)
- Facet identification procedures
- Strong branch-and-cut algorithms capable of solving instances with several 100s vertices

(Fonlupt, Grötschel, Holland, Hong, Naddef, Padberg, Pulleyblank, Rinaldi, Thienel, etc.) ($n = 2392$)

Recent efficient implementations (Applegate, Bixby, Chvátal, Cook) – Concorde

- Branch-and-cut-and-price
- 2-matching and comb inequalities
- Certain path inequalities

1998: *Documenta Mathematica*, 645–656

n	Nodes in search tree	Computation time
120	1	3.3 seconds
318	1	24.6 seconds
1002	1	94.7 seconds
666	1	260.0 seconds
532	3	294.3 seconds
2392	1	342.2 seconds
225	1	438.9 seconds
3038	193	1.5 days
4461	159	1.7 days
7397	129	49.5 days
13509	9539	$\sim$ 10 years (48 workstations)

Concorde available at www.caam.rice.edu/$\sim$keck/concorde.html

2003: Mathematical Programming Series B

1,904,711-city "world TSP": solved within 0.112% of optimality in 256.1 days.

2) The Asymmetric TSP: Branch-and-Bound Methods

(Dantzig, Fulkerson and Johnson, 1954)

$$x_{ij} = \begin{cases} 1 & \text{arc } (i,j) \in A \text{ appears on the tour} \\ 0 & \text{otherwise} \end{cases}$$

Minimize

$$\sum_{(i,j)\in A} c_{ij}\, x_{ij}$$

subject to

(Degree constraints)

$$\sum_{j\neq i} x_{ij} = 1 \quad (i \in V)$$

$$\sum_{i\neq j} x_{ij} = 1 \quad (j \in V)$$

(Subtour elimination constraints)

$$\sum_{j\neq i} x_{ij} \leq |S| - 1 \quad (S \subset V, |S| \geq 2)$$

(Integrality constraints)

$$x_{ij} = 0, 1 \quad ((i,j) \in A).$$

Relaxing the subtour elimination constraints yields an *Assignment Problem* (AP) which is easy to solve.

Early Branch-and-Bound Methods

- Eastman (1958) probably developed the first branch-and-bound algorithm.
- Croes (1958) also used a branching scheme for the TSP.
- Land and Doig (1960) proposed branch-and-bound as a generic algorithm for mixed integer linear programs.

Editor's note on the Croes paper (1958)

> "This paper was received from G. A. Croes shortly before he returned to Holland after an assignment at Shell Development Company. Subsequent attempts by the Editor to contact Croes have failed. As a result, the author has neither seen proof of this article nor has he taken advantage of the constructive comments of the referees. Minor corrections suggested by the referees have been made. One comment that should be called to the reader's attention is that the use of x_{ij} with i representing the *column* and j the *row* is the opposite of the usual usage."

Before Little et al. (1963)

Largest size solved consistently: $n = 13$ (exception: $n = 42$, Dantzig et al., 1954)

Little et al. (1963)

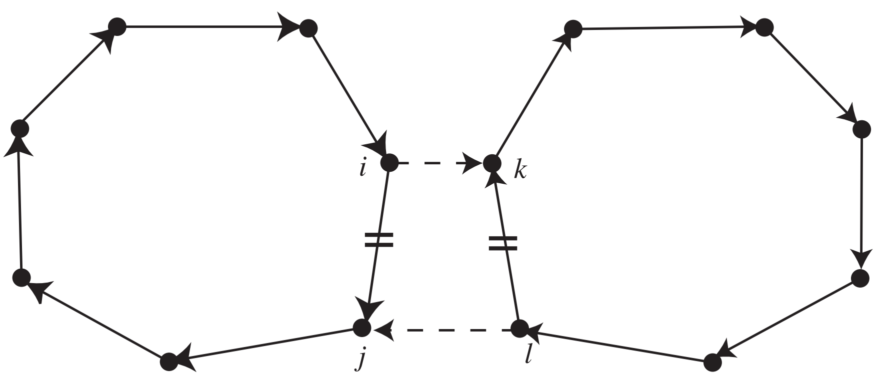
- Do not start from a formulation
- Compute a *lower bound* on the AP solution value (by matrix reduction)
- Branch on "included" and "excluded" arcs ($x_{ij} = 1$ or $x_{ij} = 0$)
- Coined the expression "branch-and-bound"
- Refer to earlier methods based on the same idea: Eastman (1958), Ph.D. thesis, Harvard University; Rossman, Twery and Stone (undated unpublished paper); Rossman and Twery (1958), presentation at the 6th ORSA meeting, mimeographed.
- Refer to Land and Doig (1960) [wrongly as Doig and Land] who introduced branch-and-bound as a generic algorithm for solving mixed or pure integer linear programs. ["For another example, see Doig and Land"].
- Solve instances with $n = 30$ (100 out of 100), $n = 40$ (5 out of 100), and $n = 64$ (Knight's tour, 0.178 minutes).

Algorithms Based on the AP Relaxation

- Eastman (1958)
- Little, Murty, Sweeney and Karel (1963)
- Shapiro (1966)
- Murty (1968)
- Bellmore and Malone (1971)
- Garfinkel (1973)
- Smith, Srinivasan and Thompson (1977)
- Carpaneto and Toth (1980)
- Balas and Christofides (1981)
- Carpaneto, Dell'Amico and Toth (1995)

Main ingredients of the Carpaneto and Toth Method (1980) improved by Carpaneto, Dell'Amico and Toth (1995).

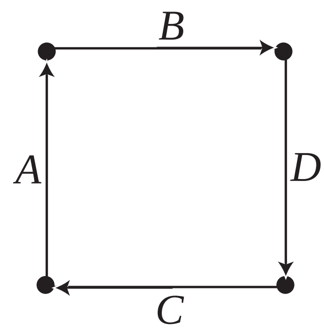
- Solution of an AP at the root node
- If subtours, generation of a feasible solution by means of a patching algorithm (Karp, 1979) (but only at the root node)

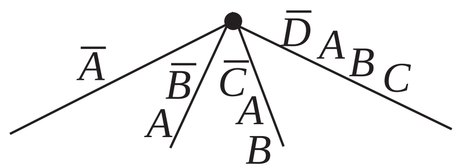


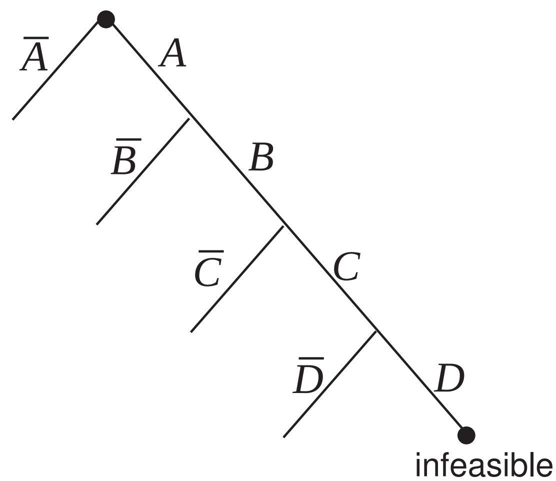
$$\min\{c_{ik} + c_{lj} - c_{ij} - c_{lk}\}$$

Excellent heuristic for the asymmetric TSP.

- Bounding: AP relaxation (as in Little et al.) or AP solution if necessary.
- Branching on the subtour with the smallest number of fixed arcs.



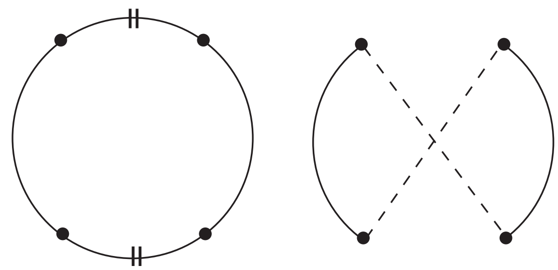




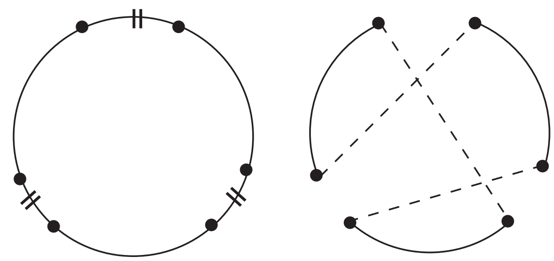
- When fixing $x_{ij} = 0$: set $c_{ij} = \infty$
- When fixing $x_{ij} = 1$: remove row i and column j from cost matrix.
- Lower bound at root node $\cong$ 99.2% optimal value for $50 \leq n \leq 250$. Increases with n.
- Large random instances $(n \leq 3000)$ can be solved optimally.

3) Local search heuristics

- Croes (1958): first 2-opt algorithm.



- Bock (1958): first 3-opt algorithm (unpublished manuscript, 14th ORSA National meeting).



- Lin (1965): generalization to r-opt.
- Checking r-optimality requires $O(n^r)$ operations.
- Christofides and Eilon (1972):

3-opt	vs	2-opt	: major improvement
4-opt	vs	3-opt	: small improvement
5-opt	vs	4-opt	: major improvement

- Very efficient 3-opt implementation: Johnson and McGeoch (2002).
- Or-opt (Ilhan Or, 1976): relocate chains of length 3, 2 and 1.

  Checking Or-optimality requires $O(n^2)$ iterations.
- Golden and Stewart (1985) applied Or-opt after their construction heuristic: "In general the Or-opt procedure yields solutions that are comparable to the 3-opt in terms of quality of solution in amount of time closer to that of the 2-opt procedure".
- Johnson and McGeoch (2002): "Or-opt no longer appears to be a serious competitor".

- Babin, Deneault and Laporte (2005) show that when starting from a random solution, 2-opt is faster and better than Or-opt.

  Or-opt can be improved to yield better results in a shorter time.

  Improved Or-opt + 2-opt is an excellent combination and is easy to implement.

- Lin and Kernighan (1973): r-opt with dynamically adapted values of r. Probably the best available heuristic for the symmetric TSP. See the implementations of Helsgaun (2000) and of Johnson and McGeoch (1997, 2002).

- Comparisons made with Held-Karp lower bounds show gaps of less than 0.1% on very large instances.

Conclusion

- The TSP is a fascinating problem.
- Many of the exact and approximate solution techniques used in combinatorial optimization originate from the study of the TSP.
- Any further progress will probably require the use of highly sophisticated programming techniques.