**Course Title:** Computation & Complexity

**Course code:** CSC 4101

**Topic:** Metric Traveling Salesman Problem

## Group Members

| Student Name | Matric No. |
|---|---|
| Abid Ebna Saif Utsha | 1433527 |
| Mahfuzealahi Noman | 1515803 |
| Mohammad Nafees Bin Zaman | 1616357 |
| Atif Ahnaf Chowdhury | 1518335 |
| Abdoulraouf Naseh | 1216509 |
| Sheikh Rifath Riaz Biku | 1521147 |

**Section** : 02

**Lecturer** : Dr. Nurul Liyana Mohamad Zulkifli

**Date of Submission:** 20/12/2018

**Table of Contents:**


| No. | Contents | Page no. |
| --- | --- | --- |
| 1.0 | Introduction | 03 |
| 2.0 | Traveling Salesman Problem | 03 |
| 2.1 | History | 03 |
| 3.0 | Metric Traveling Salesman Problem | 04 |
| 3.1 | Condition of metric TSP | 04 |
| 4.0 | Real-life Application | 04 |
| 5.0 | Complexity | 05 |
| 5.1 | Approximation Algorithm | 05 |
| 6.0 | Alternative Algorithm | 08 |
| | Reference | 08 |

# 1.0 Introduction

Metric traveling Salesman is used in operations research and theoretical computer science. There are many applications where metric traveling salesman problem is implemented. Metric traveling salesman is an optimization problem where the target is to visit every city at a minimum cost. This concept can be used in many other problems and sub-problems are being created. Instead of city, the nodes can be many things and the edges can be represented as the connection between nodes.

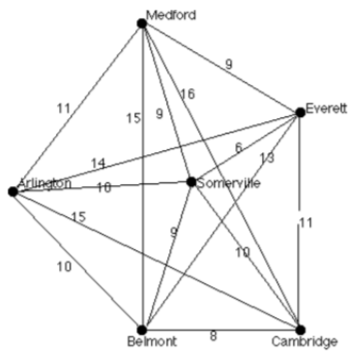
# 2.0 Traveling Salesman Problem

A salesman wishes to visit every city on his route exactly once and return home, at minimal cost.

**Formal definition**: A complete undirected graph G = (V, E) with nonnegative integer cost C(u, v) for each edge (u,v) ∈ E.

**Goal**: Find a Hamiltonian cycle of G with minimum cost.

Solution space consist of at most n! possible tours.




# 2.1 History

The origin of this problem is unclear. A handbook for traveling salesman from 1832 mentions the problem and example tours through Germany and Switzerland, but contains no mathematical treatment. The traveling salesman problem was first formulated by Irish mathematician W.R. Hamilton and British mathematician Thomas Kirkman in the 1800s.

The problem was first formulated in 1930 to solve a school bus routing problem, and it is one of the most intensively studied problems in optimization. This problem was used as a benchmark for many optimization methods.

Dantzig, Fulkerson and Johnson found an optimal tour through 42 cities. They expressed the problem as integer linear program. Then in 1960s another approach was created which produces the bound that the length of an optimal tour is at most twice the weight of a minimum spanning tree. Later Christofides made a big progress in 1976.

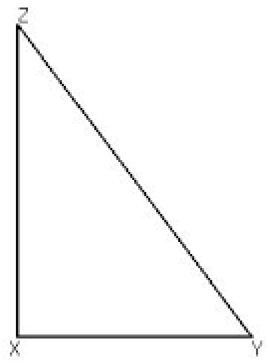
# 3.0 Metric Traveling Salesman Problem

Problem Statement: A salesman wishes to visit every city on his route exactly once and return home, at minimal cost.

Metric TSP is a subset of TSP.

# 3.1 Condition of Metric TSP

The condition of Metric TSP is that cost satisfy triangle inequality.




$C(Z, Y) \leq C(Z, X) + C(X, Y)$

$C(Z, X), C(X, Y) \geq 0$

# 4.0 Real-life Application

Metric TSP can be used in general – complex optimization problem. Metric TSP has several applications like -

1) Planning, logistics and manufacturing of microchips
2) DNA sequencing
3) Astronomy (minimize the time spent moving the telescope between sources)
4) Shift scheduling
5) Air flight system
6) Bus route scheduling system
7) GPS satellite system

## 5.0 Complexity

The general class of questions for which some algorithm can provide an answer in polynomial time is called ‘class P’ or just ‘P’. For some question there is no known way to find an answer quickly, but if one is provided with information showing what the answer is, it is possible to verify the answer quickly. The class of questions for which an answer can be verified in polynomial time is called NP, which stands for “nondeterministic polynomial time”.

For example – if one is given with a multiplication problem, one can easily solve the problem. That is an example of P problem. On the other hand, Sudoku, a game where the player is given a partially filled-in grid of numbers and attempts to complete the grid following certain rules. Any proposed solution is easily verified, and the time to check a solution grows polynomial as the grid gets bigger. So, P problems are quickly solvable whether NP problems are quickly checkable.

A problem is NP hard if the problem is at least as hard as the hardest problems NP.

The complexity of Metric Traveling Salesman is NP hard. There are a lot of possible answers in this problem, It is really difficult to tell the exact route salesman will take to visit every city, but we can check with the answer easily whether this is the minimal route or not.

Out of four cities, there are three possible ways that a salesman can travel visiting each city only once.

$$\frac{(n-1)!}{2} = \frac{3!}{2} = 3$$

Now, imagine the salesman has to travel 50 cities. How many possible ways can be there?

$$\frac{(n-1)!}{2} = \frac{49!}{2} = 3.0414093\text{e}+62$$

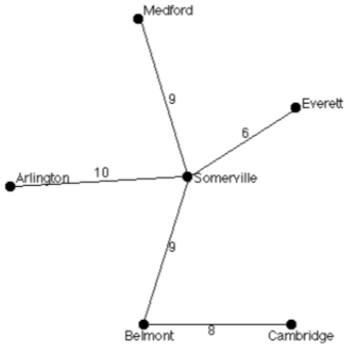
That is why it is too difficult to find the exact way that costs minimal the salesman can travel. Therefore, Approximation algorithm is used to solve the problem.

## 5.1 Approximation Algorithm

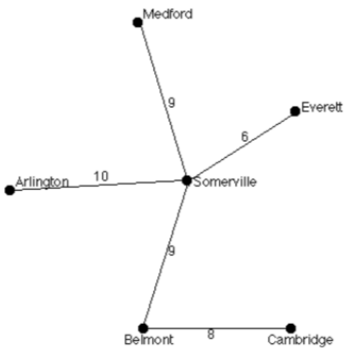
Approximation Algorithm can be used to get within a certain factor of the optimal answer. Let OPT denote the cost of minimum weight tour:

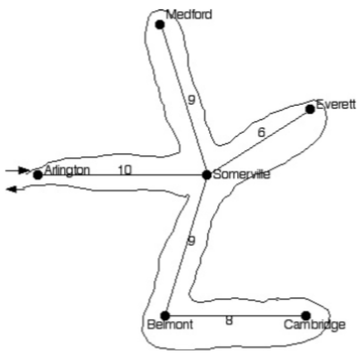
**Goal 1:** A polynomial-time algorithm that outputs a tour of cost C, where C ≤ 2 x OPT

1) Compute MST
   Take the minimum-weight spanning tree (MST) of the TSP graph.
      Weight of MST ≤ OPT

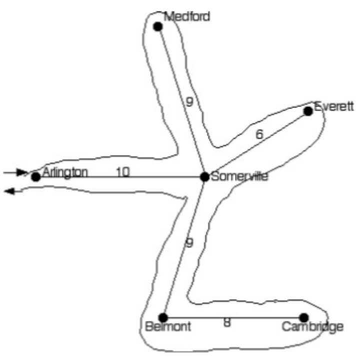



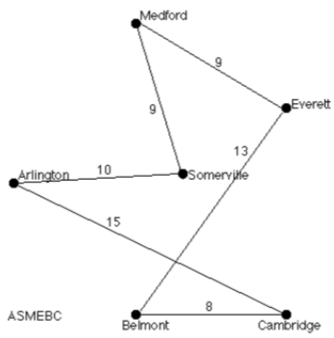

2) Perform preorder walk on MST
Do a depth-first search (DFS) of the MST, hitting every edge exactly twice. This "pseudo tour" PT has cost=2 × MST ≤ 2 × OPT. Write down each vertex as you visit it.
PT = ASMSESBCBSA




3) Return list of vertices according to the preorder tree walk.
To go from PT to T*: Rewrite the list of vertices, writing each vertex only the first time it appears in PT
T* = ASMEBCA

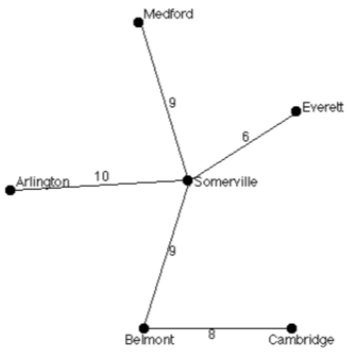



Cost of T* ≤ PT, therefore cost of T* ≤ 2 x OPT. Follows from the triangle inequality, going from A to B must be cheaper than A to C to B.

**Goal 2:** Christofides' Algorithm: A polynomial-time algorithm that outputs a tour of cost C, where C ≤ 1.5 x OPT.

1) Compute MST


In any graph, the number of vertices of odd degrees must be even. The sum of the degrees of all the vertices in a graph is equal to twice the number of edges, therefore it is an even number. Since an even and an odd number added together make an odd, it follows that the number of the degrees of the vertices of odd degree and the sum of the degrees of even degree must both also be even. Therefore, since the sum of odd degrees is an even number, there must be an even number of vertices of odd degree.

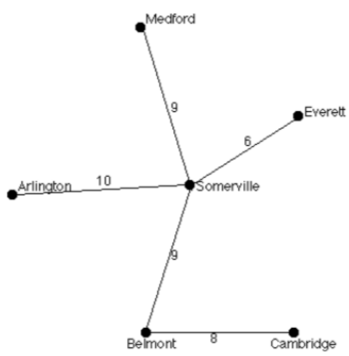
$$\sum_{v \in V} deg(v) = 2 \times E$$

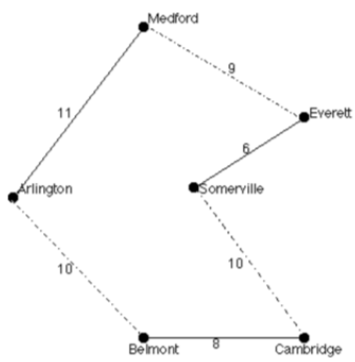
$$\sum_{v \in V} deg(v) = \sum_{v \in V_{odd}} deg(v) + \sum_{v \in V_{even}} deg(v)$$

2) Add a minimum-weight perfect matching of the odd vertices and find an Eulerian Circuit.
Find a minimum-weight perfect matching M* in the original graph between the vertices that have odd degree IN THE MST.
Any tour can be decomposed into two matchings, $M_1$ and $M_2$. By alternating matched and unmatched edges. Therefore,

OPT = $M_1 + M_2 \geq M^* + M^*$
Considering the subgraph MST + M*; Every vertex in this subgraph has even degree, so there exists some Eulerian tour E of this subgraph has even degree, so there exists some Eulerian tour E of this subgraph with cost exactly equal to MST + M* since it uses each edge exactly once. Therefore, we have:

MST + M* ≤ 0.5 x OPT + OPT ≤ 1.5 x OPT



3) Transform the circuit into a Hamiltonian cycle
   Repeat as above, write down each vertex the first time is appears in the Eulerian tour E, creating a salesman tour $E*$ with cost $\leq 1.5 \times OPT$.

## 6.0 Alternative Algorithm

As Metric TSP is a subcase for TSP problem and complexity of both problems is NP hard, it can be thought that branch and bound algorithm can be used to solve this Metric TSP problem. Branch and bound algorithm is used to solve traveling salesman problem. But the problem appears on the condition of Metric TSP. Because of the triangle inequality, it is really difficult to implement branch and bound algorithm in Metric TSP. So, there is no alternative algorithm.

## Reference


1) https://en.wikipedia.org/wiki/P_versus_NP_problem
2) https://www.youtube.com/watch?v=YX40hbAHx3s
3) https://en.wikipedia.org/wiki/NP-hardness