Solving Travelling Salesperson Problem using Dynamic Programming

TRAVELING SALESPERSON PROBLEM (TSP)

Let G=(V,E) be directed graph

Edge Cost Matrix C : C_{ij} representing cost of edge <i, j>

$C_{ij} = \infty$ if $< i,j >$ is not an edge of G

Tour of G is a directed cycle that includes every vertex in V.

Tour consists of all vertices but a subset of edges that connect the vertices

Cost of the tour is the sum of the cost of edges included in the tour

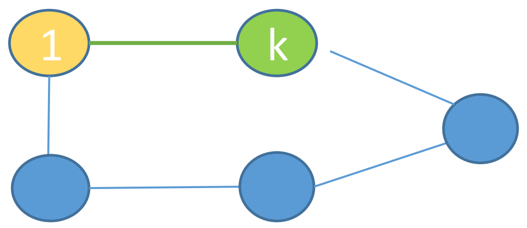
Optimal Tour is the one in which the sum of the weights of the edges included in the tour is minimum

- cost of visiting all the vertices of the graph is minimum

TRAVELING SALES PERSON PROBLEM (TSP)

Problem : To find a tour of minimal cost(optimal tour)

Assume every tour starts with vertex 1. Starts and stops at vertex 1



Every tour consists of an edge <1,k> for some k $\in$ V-{1} and a path from vertex k to 1

Path from k to 1 goes through each vertex in V-{1,k} exactly once.

This path should be optimal for the tour from 1 to be optimal.

Hence Principal of Optimality holds.

Let g(i,S) be the length of a shortest path starting at a vertex i, going through all the vertices in S, and terminating at vertex 1.

Solution : g(1, V-{1}), the length of an Optimal Tour

$$g(1, V-\{1\}) = \min_{2 \le k \le n}\{c_{1k} + g(k, V-\{1,k\})\}$$

Recurrence equation

$$g(i,S) = \min_{j \in S} \{C_{ij} + g(j, S-\{j\})\}$$

1. Set $g(i,\Phi) = c_{i,1}$; $1 \le i \le n$ (cost of the direct edge)

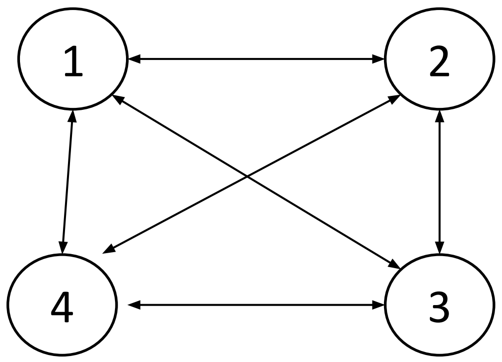
2. Obtain g (i,S) for $|S| = 1$, then obtain g(i,S) for $|S| = 2$ and so on till $|S| < n-1$

3. Then obtain g(1,v-{1})

Example : Consider a set of cities represented by the following edge cost matrix. Find an Optimal tour

C =

0	10	15	20
5	0	9	10
6	13	0	12
8	8	9	0



Step 1 : Find the cost of path starting from vertices 2 , 3, 4 to vertex 1 passing through Φ vertices (ie direct edge)

|S| = 0

$g(2, \Phi) = c_{21} = 5$

$g(3, \Phi) = c_{31} = 6$

$g(4, \Phi) = c_{41} = 8$

Step 2 : Find the cost of path starting from vertices 2 , 3, 4 to vertex 1 passing through any one vertex

|S| = 1 [Excluding vertex one, take all other vertices one by one]

g{2, }, g{3, }, g{4, }

Second term is one of the remaining vertices, excluding vertex 1 and the vertex in the first term

$g(2,\{3\})=C_{23}+g(3, \Phi) =9+6=15$ [Take the edge 2-3 and then path 3-1 passing through Φ vertices ie direct path

$g(2,\{4\})=C_{24}+g(4, \Phi) = 10 + 8 =18$

$g(3,\{2\})= C_{32}+g(2, \Phi) = 13 + 5 =18$

$g(3,\{4\})= C_{34}+g(4, \Phi) = 12 + 8 =20$

$g(4,\{2\})= C_{42}+g(2, \Phi) = 8 + 5 = 13$

$g(4,\{3\})= C_{43}+g(3, \Phi) = 9 + 6 = 15$

0	10	15	20
5	0	9	10
6	13	0	12
8	8	9	0

Step 3 : Find the cost of shortest path starting from vertices 2 , 3, 4 to one vertex 1 with **|S| = 2**

g{2, }, g{3, } and g{4, } [first term vertex other than 1]

Second term - any two from the remaining vertices excluding 1 and the vertex in first term

$g(2,\{3,4\}) = \min(C_{23} + g(3,\{4\}),\ C_{24} + g(4,\{3\}))$ [~~1~~, ~~2~~, 3, 4] , [~~1~~, ~~2~~, 3, 4]

$= \min(9+20, 10+15) = 25$ [min = 4]

$g(3,\{2,4\}) = \min(C_{32} + g(2,\{4\}), C_{34} + g(4,\{2\}))$

$= \min(13+18, 12+13) = 25$ [min = 4]

$g(4,\{2,3\}) = \min(C_{42} + g(2,\{3\}), C_{43} + g(3,\{2\}))$

$= \min(8+15, 9+8) = 23$ [min = 2]

Step 4: Continue till |S| < n-1 ; Here n = 4 ; |S| = 1,2

Since we have completed |S| = 2, proceed to next step

Step 5: Calculate g (1,{2,3,4)) - This is the goal : g(1, V-{1})

[~~1~~, 2, 3, 4], [~~1~~, 2, ~~3~~, 4], [~~1~~, 2, 3, ~~4~~]

g (1,{2,3,4}) = min ((C_{12}+g(2,{3,4}), (C_{13}+g(3,{2,4}), (C_{14}+g(4,{2,3}))

=min(10+25, 15+25, 20+23) = 35 [min =2]

Optimal tour length =35

Optimal tour -> constructed using minimum value recorded in every step.

P(1,{2,3,4})=2 Tour starts from 1 then to 2 [min in g (1,{2,3,4}) = 2]

P(2,{3,4})=4 then to 4 [min in g (2,{3,4}) =4]

P(4,{3})=3 then to 3 and return back to 1

Optimal tour : 1-2-4-3-1

Dynamic Programming Algorithm to solve TSP

1. For i = 2 to n

   set g(i, Φ) = c_{i1}

2. For k = 1 to n-2

   for all subset $S \subseteq V - \{1\}$ containing k vertices

   for all $i \notin S, i \neq 1$

$$g(1, S\}) = \min_{j \in S}\{c_{ij} + g(j, S - \{j\})\}$$

   p(i,S) = value of j that gave minimum (successor of vertex i in Set j)

3. Length of optimum TSP tour

   f = g(1, V-{1}) = $g(1, V - \{1\}) = \min_{j=2 \text{ to n}}\{c_{1j} + g(j, V - \{1, j\})\}$

   p{1, V-{1}) = value of j that gave the minimum (successor of vertex 1)

Algorithm Complexity

Let N be the number of $g(i, S)$'s that have to be computed before computation of $g(1, V - \{1\})$. For each value of $|S|$ there are $n - 1$ choices for i. The number of distinct sets S of size k not including 1 and i is $\binom{n-2}{k}$. Hence

$$N = \sum_{k=0}^{n-2} (n-1)\binom{n-2}{k} = (n-1)2^{n-2}$$

An algorithm that proceeds to find an optimal tour will require $\Theta(n^2 2^n)$ time as the computation of $g(i, S)$ with $|S| = k$ requires $k - 1$ comparisons

This is better than enumerating all $n!$ different tours to find the best one.

space needed, $O(n2^n)$

Exercise : Determine optimal tour and its length for the following instances of TSP

0	20	42	35
20	0	30	34
42	30	0	12
35	34	12	0

0	2	0	6	1
1	0	4	4	2
5	3	0	1	5
4	7	2	0	1
2	6	3	6	0