CS F364: Design & Analysis of Algorithm

11 Traveling Salesman Problem (TSP)

Dr. Kamlesh Tiwari
Assistant Professor, Department of CSIS,
BITS Pilani, Pilani Campus, Rajasthan-333031 INDIA

Feb 10, 2021 ONLINE (Campus @ BITS-Pilani Jan-May 2021)

http://ktiwari.in/algo

Traveling Salesman Problem (TSP)

Consider a map.
You have to visit all the cities and come back at the starting one.

- What should be the order so that cost of tour is minimum.

- It is an optimization problem involving permutation
- Harder then the problem of subset sum nature $n! > 2^n$

Let $G = (V, E)$ be graph with $c_{ij} > 0$ weight between node i and node j

- Choose the start place in n ways, next in $n-1$ and so on. Total number of ways $n!$ [1]
- NP-Complete

[1] Stirling's approximation $n! = \sqrt{2\pi n}(\frac{n}{e})^n(1+\theta(\frac{1}{n}))$

Solution

1. You can start from the node 1
2. Go to immediate node k
3. From node k come back to 1

$$tour_{1\to 1} = edge_{1\to k} + path_{k\to 1}$$

- $k \in V - \{1\}$
- It is logical to pick the k with minimum c_{1k}
- Let $g(i, S)$ be the length of shortest path starting at vertex i, going through all the vertices in S and terminating at **1**.

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

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

Example: consider graph

$$\begin{bmatrix} 0 & 10 & 15 & 20 \\ 5 & 0 & 9 & 10 \\ 6 & 13 & 0 & 12 \\ 8 & 8 & 9 & 0 \end{bmatrix}$$

$$\begin{aligned} g(1, \{2,3,4\}) &= \min\{c_{12} + g(2, S-\{2\}), \\ &= c_{13} + g(3, S-\{3\}), \\ &= c_{14} + g(4, S-\{4\})\} \end{aligned}$$

$$\begin{aligned} g(2, \{3,4\}) &= \min\{c_{23} + g(3, \{4\}), \\ &= c_{24} + g(4, \{3\})\} \end{aligned}$$

$$g(3, \{4\}) = c_{34} + g(4, \phi) = c_{34} + c_{41}$$

TSP Algorithm

Algorithm 1: TSP (V, c_{ij})

1 **for** $i = 2$ *to* n **do**
2 $\quad g(i, \phi) = c_{i1}$
3 **for** $k = 1$ *to* $n-2$ **do**
4 $\quad$ **for** $i = 2$ *to* n **do**
5 $\quad\quad$ **forall** $S \subseteq V - \{i, 1\}$ with $|S| = k$
6 $\quad\quad\quad g(i, S) = \min_{j\in S}\{c_{ij} + g(j, S-\{j\})\}$
7 $g(1, V-\{1\}) = \min_{2\le i\le n}\{c_{1i} + g(i, V-\{1,i\})\}$
8 **return** $g(1, V-\{1\})$

Time Complexity

Line	Complexity
1	$n-1$
2	$n-1$
3	loop
4	$n-1$
5	${}^{n-2}C_k$
6	$2\|S\| = 2k$
7	$2(n-1)$
8	1

$$T(n) = (n-1) + \sum_{k=1}^{n-2}(n-1)({}^{n-2}C_k)2k + 2(n-1) + 1$$

$$T(n) = \theta(n^2.2^n)$$

Space Complexity

$$
\begin{aligned}
S(n) &= \theta(n^2.2^n) \\
&= \sum_{k=0}^{n-2} (n-1)({}^{n-2}C_k) \\
&= (n-1)\sum_{k=0}^{n-2} ({}^{n-2}C_k) \\
&= (n-1)2^{n-2} \\
&= \theta(n2^n)
\end{aligned}
$$

Thank You!

Thank you very much for your attention! (Reference[2])

Queries ?

[2] [1] Book - *Introduction to Algorithm*, By THOMAS H. CORMEN, CHARLES E. LEISERSON, RONALD L. RIVEST, CLIFFORD STEIN