Homework Assignment - 13

Dhayalini Nagaraj
A20359686

1. Prove inequality (3) on page 1 of the notes with the analysis of the nearest neighbor heuristic for the TSP.

Let n be odd and $n = 2k+1$, $k \in N^+$.

Based on the same analysis in the notes

$$T_n = T_{2k+1} \geq \sum \{l_i, l_j\}$$

We can replace the first k edges l_i with members of the last k edges l_i, giving

$$T_n \geq 2\sum_{i=k+2}^{2k+1} l_i + l_{k+1}$$

$$> 2\sum_{i=k+2}^{2k+1} l_i$$

$$= 2\sum_{i=\left[\frac{n}{2}\right]+1}^{n} l_i$$

$$= 2\sum_{i=[n/2]+1}^{n} l_i$$

Hence Proved.

2) One criticism of the nearest neighbor TSP heuristic is that the last edge added (from the last city back to the starting city) can be very long. How much effect can that last edge have on the ratio HEUR/OPT?

If no path exists between two cities then an arbitarily long edge will complete the graph without affecting the optimal tour.

Nearest neighbor algorithm:

1. Start with an arbitrary node
2. Find the node not yet on the path which is closest to the node last added and add to the path the edge connecting these two nodes.
3. When all nodes have been added to the path add an edge from last city to the starting city.

The best case is if the last edge is not very long. ~~the~~ The long edge be the nearest edge from city 1 to city 2 and the other edges are small then it will not have any effect on the ratio.

Let length of the path be HEUR

$$\sum_{i=1}^{n} l_i = HEUR$$

If the cities are in a linear order then the length of the tour will be two time the length from city 1 to city n.

(ie) $$2 \sum_{i=1}^{n} l_i = HEUR$$

Hence the spanning tree is twice as the minimum spanning tree.

$OPT \geq 2l_1$ by triangle inequality

Since the first city has to be visited.

$$OPT \geq 2\sum_{i=k+1}^{2k} l_i$$

Triangle inequality tells us $OPT \geq |T_{2k}|$

Each l_i appears in this summation twice as there is no city in between city A and the last city.

$$OPT \geq 2 \sum_{i=\lceil n/2 \rceil +1}^{n} l_i$$

$$(\lceil \lg n \rceil + 1)\, OPT \geq 2 \sum_{i=1}^{n} l_i$$

$$= 2\, HEUR$$

$$\frac{HEUR}{OPT} \leq \frac{\lceil \lg n \rceil + 1}{2}$$

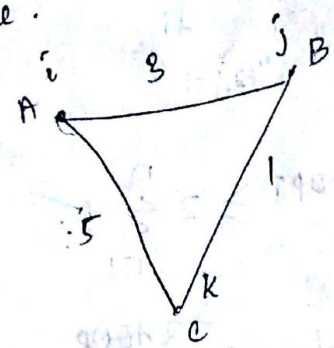
3) a) Prove that given an arbitrary symmetric cost matrix c of inter city distance, it is possible to force triangle inequality to hold by adding sufficiently large value to every element of c.

$$d(i,j) + d(j,k) \geq d(i,k)$$ for all i, j, k in N.

This condition is referrede to as the triangle inequality.

If the given matrix is symmetric cost matrix then $x_{ij} = x_{ji}$

example :



$$d(i,j) + d(j,k) = 3+1 = 4$$

$$d(i,k) = 5$$

$$d(i,j) + d(j,k) < d(i,k)$$

The triangle inequality does not hold.

Now let us add ~~every element~~ a large value to every element of c.

$$d(i,j) + n = 3+n$$

$$d(j,k) + n = 1+n$$

$$d(i,k) + n = 5+n$$

$$3+n + 1+n \geq 5+n$$

Two times the large value is added to other side while only one value is added to $d(i, k)$. In order for triangle inequality to hold:

$$3+1+n+n \geq 5n$$

$$4+2n \geq 5n$$

Let us substitute $n = 1$ then,

$$4+2 > 5$$

$$6 > 5$$

The triangle inequality ~~is~~ holds when a large value is added to every element of c.

Hence Proved.

3b) Does this mean that the bound on the quality of the closest insertion heuristic holds for symmetric cost matrices c for which the triangle equality does not hold?

a) Edge (x,y) in T minimizes

$$d(x,k) + d(k,y) - d(x,y)$$

b) delete edge (x,y) and add edges (x,k) and (k,y) to obtain TOUR(T,k).

$$\text{COST}(T,k) \leq 2 \cdot d(k,j)$$

When T has one node result is obvious. more than one node, j is an endpoint of some edge (i,j) in T. Because k is inserted to minimise

$$\text{Cost}(T,k) \leq d(i,k) + d(k,j) - d(i,j)$$

$$\frac{\text{INSERT}}{\text{OPTIMAL}} \leq \lceil lg(n) \rceil + 1$$

Scanned by CamScanner

4) a) Prove that if c is a symmetric cost matrix that satisfies the triangle inequality.

$$\frac{\text{cost of cheapest insertion tour}}{\text{cost of optimal tour}} < 2$$

We denote the length of tour obtained by cheapest insertion as INSERT and the optimal tour as OPTIMAL.

We can prove

$$\frac{\text{INSERT}}{\text{OPTIMAL}} < 2 \quad — \quad ①$$

By proving the following lemma:

Lemma 1: Suppose that, for a travelling salesman graph (N, d) with n nodes, a tour of length INSERT is constructed by the insertion method.

Suppose further that for i satisfying $1 < i < n$, the tour T_i and node a_i selected by the insertion method satisfy.

$$COST(T_i, a_i) \leq 2d(p, q) \quad — ②$$

for all node p and q such that p is in T_i and q is not in T_i. Then

$$INSERT \leq 2 \cdot TREE \quad — ③$$

where TREE is the length of a minimal spanning tree for (N, d).

Proof: Let M be a minimal spanning tree. The idea of the proof is to establish a correspondence between steps in the insertion procedure and edges of M. For the steps of inserting node a_i into T, the corresponding edge of M will have one endpoint in T_i and the other endpoint in $N-T_i$. Thus (2) can be used to show that the cost of each step is no more than twice the corresponding edge. Since M is a tree, there is a unique path in M connecting each pair of nodes. For each node a_i with $i>0$, we say that node a_i is compatible with node a_j if $j<i$ and all the intermediate nodes in the unique path in M connecting a_i and a_j have indices greater than i. Thus a_j compatible with a_i implies that a_j is the first node in T_i encountered in the path from a_i to a_j. For each a_i with $i>0$, the critical node is the node with the largest index that is compatible with a_i. The unique path in M between a_i and critical noble is the critical path for a_i critical edge for a_i is the edge in the critical path and one of its end is connected to critical node. The critical edge of a_i one endpoint is T_i and other is $N-T_i$.

Scanned by CamScanner

~~Q~~ Now let us see two nodes cannot have same critical edge.

Let us assume the contrary that a_i and a_j have same critical edge ($j > i$).

The endpoints of the critical edge be a_k and a_l with $l > k$. For any critical edge, the node with the lower index is the critical node and the node with the higher index is on the critical path, so node a_k is the critical node for both a_i and a_j

The critical paths for a_i and a_j both pass through a_l before reaching a_{k_i}. So there is a path P in M connecting a_j and a_i, where every edge in P belongs to either the critical path for a_j or a_i or both. Every intermediate node on P has an index greater than i. Since the path P from a_j reaches a node of lower index (a_i), some node a_m, along path P is compatible with a_j. Now $m \geq k$ because a_m is on path P and $i > k$ because a_k is a compatible node for a_i. This implies $m > k$ and so a_m is a compatible node for a_j with a higher index than a_k. This contradicts the assumption that a_k is critical for a_j. Therefore no two nodes can have the same critical edge. Thus given a minimal spanning tree we can associate a unique edge in that tree with each node inserted by the insertion method.

Let e_i be the critical edge for node a_i. Since one endpoint of e_i is in T and other endpoint is not, from (2)

$$\text{COST}(T, a_i) \leq 2d(e_i) \quad \text{— (4)}$$

Summing (4) gives.

$$\sum_{i=1}^{n-1} \text{COST}(T_i, a_i) \leq 2\sum_{i=1}^{n-1} d(e_i) \quad \text{— (5)}$$

The left hand side of (5) is INSERT. Since M consists of $n-1$ edges, and each e_i is distinct, the right hand side of (5) is 2.TREE. Thus (5) implies (3).

Now we will show ② holds for the cheapest insertion method. For the cheapest insertion, there is for each i a node y_i in T_i such that

$$d(y_i, a_i) \le d(p, q) \quad — ⑥$$

for all p in T_i and q in $N - T_i$

$$COST(T_i, a_i) \le 2d(y_i, a_i) \quad — ⑦$$

⑥ and ⑦ imply ②.

Hence Proved.

$$\frac{\text{Cost of cheapest insertion tour}}{\text{Cost of optimal tour}} < 2$$

b) How many operations as a function of number of cities, are needed to implement this algorithm?

Each time a new node k is added, we need to find nodes $i, j \in T$ and $k \notin T$ such that $C_{ik} + C_{kj} - C_{ij}$ is minimized. Each node can maintain a min-heap

Scanned by CamScanner

storing the costs of inserting it to every edge of the tour and update the heap each time new node is inserted. Since n nodes will be inserted to the tour and updating the heap takes $O(\log n)$ time, the total running time is $O(n^2 \log n)$.

c) How far from optimal can the cheapest insertion tour actually be?

The optimal tour can be made into a tree by deleting its longest edge and this longest edge has a length at least OPTIMAL/n where n is the number of nodes in the problem. Since the minimal spanning tree is no longer than this tree.

$$\text{TREE} \leq \left(1 - \frac{1}{n}\right) \cdot \text{OPTIMAL}$$

which implies

$$\frac{\text{INSERT}}{\text{OPTIMAL}} \leq 2\left(1 - \frac{1}{n}\right)$$

References:

An analysis of several heuristics for the traveling salesman problems.

https://www.m9.matum.de

lcbb.epfl.ch/algs10/notes09/oct15.pdf.

cdn.intechopen.com/pdfs.