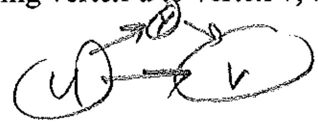
Data Structures (CS 1520) Lecture 29 Name:__________________

Approximation Algorithm for TSP with Triangular Inequality

Restrictions on the weighted, undirected graph G=(V, E):

1. There is an edge connecting every two distinct vertices.
2. Triangular Inequality: If W(u, v) denotes the weight on the edge connecting vertex u to vertex v, then for every other vertex y,

$$W(u, v) \le W(u, y) + W(y, v).$$



NOTES:

- These conditions satisfy automatically by a lot of natural graph problems, e.g., cities on a planar map with weights being as-the-crow-flys (Euclidean distances).
- Even with these restrictions, the problem is still NP-hard.

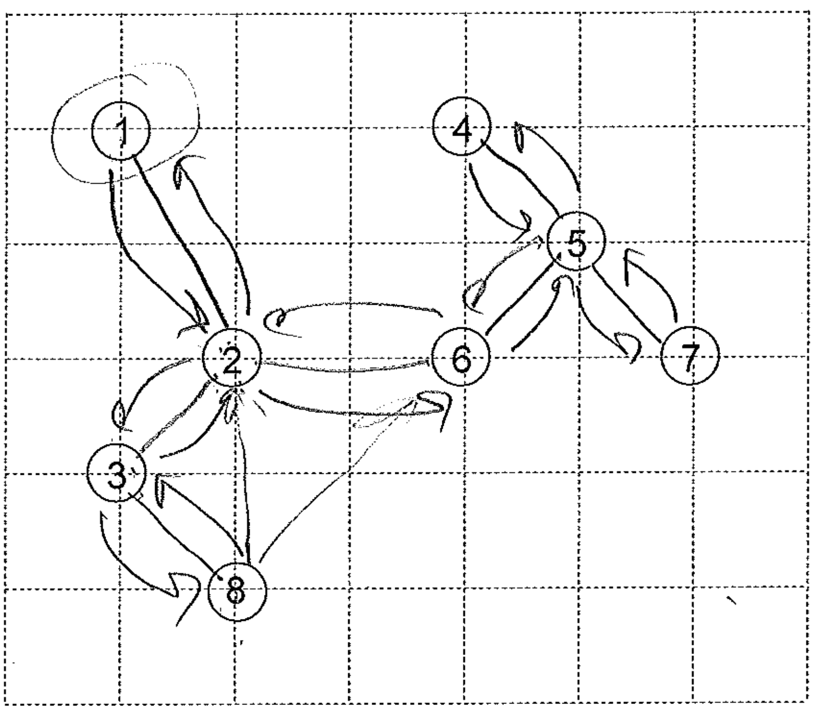
<u>A simple TSP approximation algorithm:</u>

Step 1. Determine a Minimum Spanning Tree (MST) for G (e.g., Prim's Algorithm section 4.1)

Step 2. Construct a path that visits every node by performing a preorder walk of the MST. (A *preorder walk* lists a tree node every time the node is encounter including when it is first visited and "backtracked" through.)

Step 3. Create a tour by removing vertices from the path in step 2 by taking shortcuts.

1) (Step 1) Determine a Minimum Spanning Tree (MST) for G (e.g., Prim's Algorithm) if we start with vertex 1 in the MST. (Assume edges connecting all vertices with their Euclidean distances)



Prim's algorithm is a greedy algorithm that performs the following:

a) Select a vertex at random to be in the MST.

b) Until all the vertices are in the MST:
   - Find the closest vertex not in the MST, i.e., vertex closest to any vertex in the MST log n

   Add this vertex using this edge to the MST

$O(n \log n)$

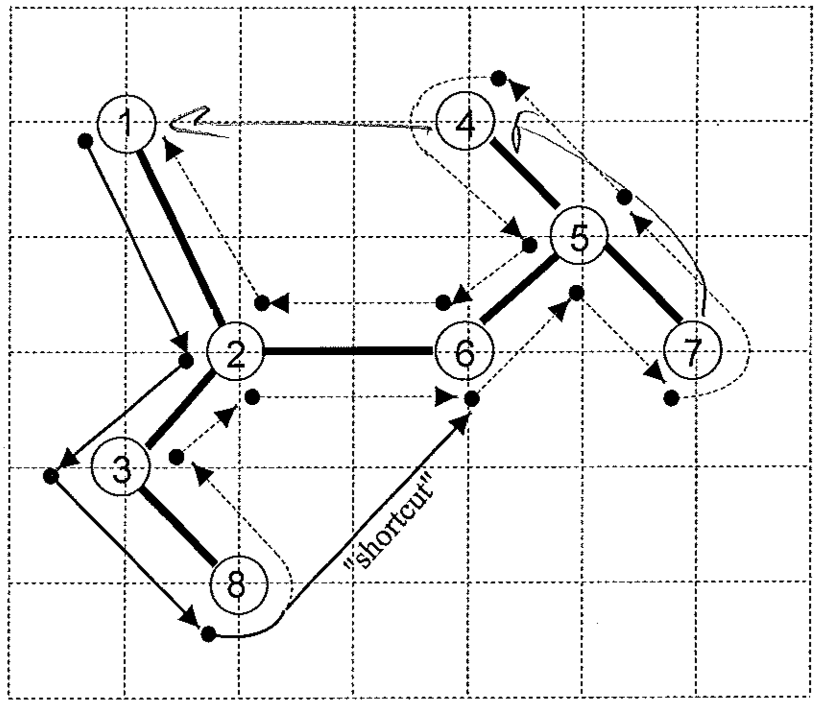
$n = |V|$

2) (Step 2) Determine the preorder walk of the MST.

1, 2, 3, 8, 8, 2, 6, 5, 7, 5, 4, 5, 6, 2, 1

$O(n)$

3) (Step 3) Complete a tour by removing vertices from the path in step 2 by taking shortcuts.



a) Finish removing vertices from the preorder-walk path to create a tour by taking shortcuts:
[1 2 3 8 ~~3~~ ~~2~~ 6 5 7 ~~5~~ 4 ~~5~~ ~~6~~ ~~2~~ 1]

b) When scanning the above path, how did you know which vertices to eliminate to take a shortcut?

Seen them before so can be eliminated except the last 1

4) Let's determine how close our approximation algorithm gets to the actual TSP tour.

opt TSP tour

a) If we take the optimal TSP tour and remove an edge, what do we have?

no cycle
complete connected
spanning tree

b) What is the relationship between the distance of the MST and the optimal TSP tour?

MST $\le$ spanning from opt TSP tour

c) What is the relationship between the distance of the MST and the distance of the preorder-walk of the MST?

(pre-order walk of MST) = 2×MST or $\frac{1}{2}$(preorder walk of MST) = MST

d) What is the relationship between the distance of the preorder-walk of the MST and the tour obtained from the preorder-walk of the MST?

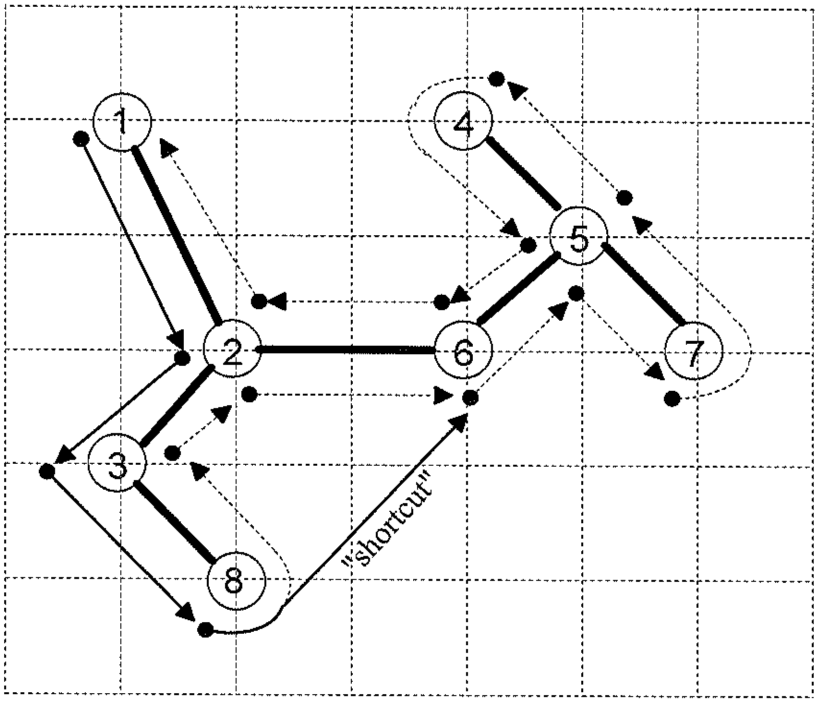
tour from preorder-walk of MST $\le$ preorder walk of MST

e) What is the relationship between the tour obtained from the preorder-walk of the MST and the optimal TSP tour?

$\frac{1}{2}$(tour from preorder walk of MST) $<$ opt TSP tour

Data Structures (CS 1520) Lecture 29 Name:_______________

1. Complete a tour by removing vertices from the path in step 2 by taking shortcuts.



a) Finish removing vertices from the preorder-walk path to create a tour by taking shortcuts:
[1 2 3 8 ~~3~~ ~~2~~ 6 5 7 ~~5~~ 4 ~~5~~ ~~6~~ ~~2~~ 1]

b) When scanning the above path, how did you know which vertices to eliminate to take a shortcut?

If a value has been seen before in the path, except for start vertex of 1.

c) If we take the optimal TSP tour and remove an edge, what do we have?

A spanning tree since it includes all the vertices and contains no cycles.

d) What is the relationship between the distance of the MST and the optimal TSP tour?

distance of the MST <= spanning tree by removing edge from optimal tour < the optimal TSP tour

distance of the MST < the optimal TSP tour

e) What is the relationship between the distance of the MST and the distance of the preorder-walk of the MST? Because every edge in the MST is traversed twice in the preorder-walk, then: distance of the MST * 2 = distance of the preorder-walk of the MST, or

0.5 * distance of the preorder-walk of the MST = distance of the MST

f) What is the relationship between the distance of the preorder-walk of the MST and the tour obtained from the preorder-walk of the MST?

tour obtained from the preorder-walk of the MST <= distance of the preorder-walk of the MST

0.5 * tour obtained from the preorder-walk of the MST <= 0.5 * distance of the preorder-walk of the MST

g) What is the relationship between the tour obtained from the preorder-walk of the MST and the optimal TSP tour?

0.5 * tour obtained from the preorder-walk of the MST <= 0.5 * distance of the preorder-walk of the MST = distance of the MST < the optimal TSP tour

0.5 * tour obtained from the preorder-walk of the MST < the optimal TSP tour

Thus, tour obtained from the preorder-walk of the MST < 2 * the optimal TSP tour