Chapter 35
Approximation Algorithm

Algorithm Analysis

School of CSEE

How to solve problems that are NP-hard

1. Try to develop the most efficient algorithm possible
2. For sufficiently small input → applying exponential algorithm might be O.K.
3. Restrict input → isolate special cases that are solvable in polynomial time
4. Accept potentially incorrect solution in return for polynomial running time → **approximation algorithm**

Formalism

Given optimization problem on input of size n, let

C^*= cost of optimal solution

C = cost of approximation algorithm solution.

Ratio bound: $\rho(n)$

$$\max\{\frac{C}{C^*},\frac{C^*}{C}) \le \rho(n)$$

Relative error bound: $\varepsilon(n)$

$$\frac{|C - C^*|}{C^*} \le \varepsilon(n)$$

Traveling salesperson problem

- Input: undirected complete graph with edge length $c(u,v)$
- Output: a cyclic tour of minimum length that visits each vertex exactly once
- This problem is NP-hard.

Most efficient one?

- Brute force
- Greedy : does not work
- Dynamic Programming
- Branch and Bound

An approximation algorithm for the TSP

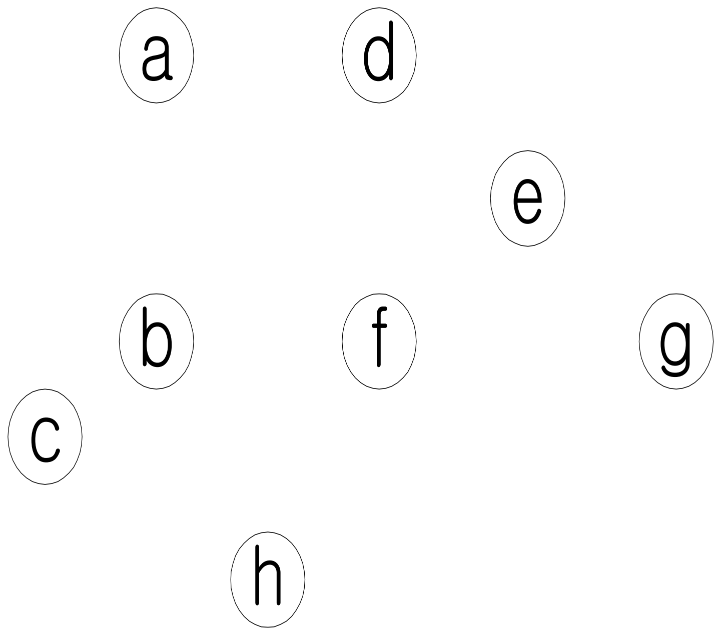
Approx-TSP-Tour(G,c)

1. select a vertex $r \in V(G)$ to be a root vertex
2. compute a minimum spanning tree T for G from root r
3. let L be the list of vertices visited in a preorder tree walk of T
4. return the hamiltonian cycle H that visits the vertices in the order L

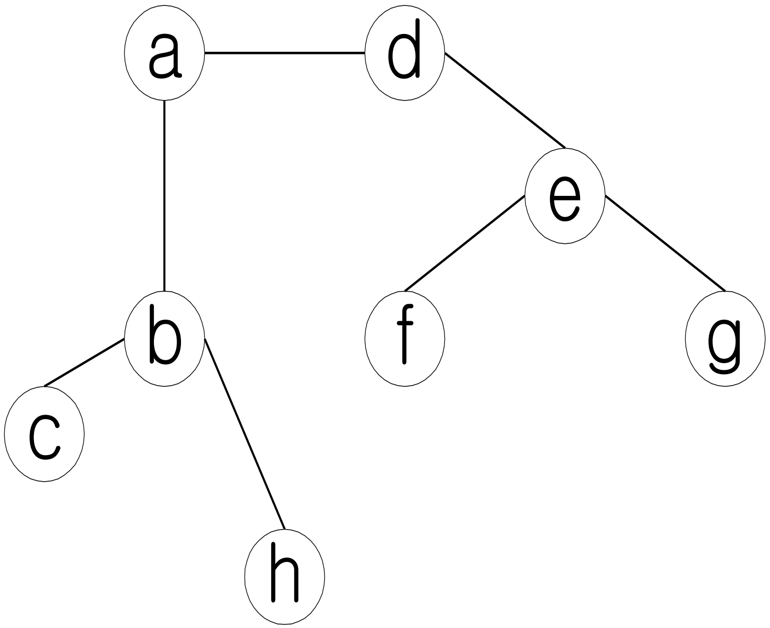
The running time is $\Theta(V^2)$.

Example: Approx-TSP-Tour

The input graph

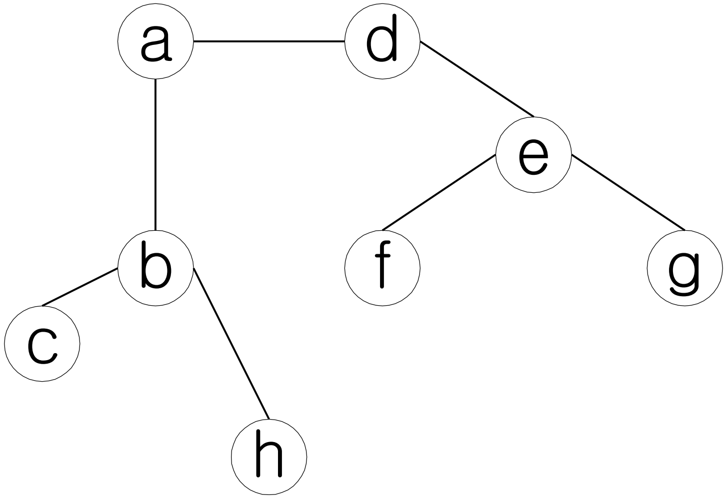


Step 1: a minimum spanning tree

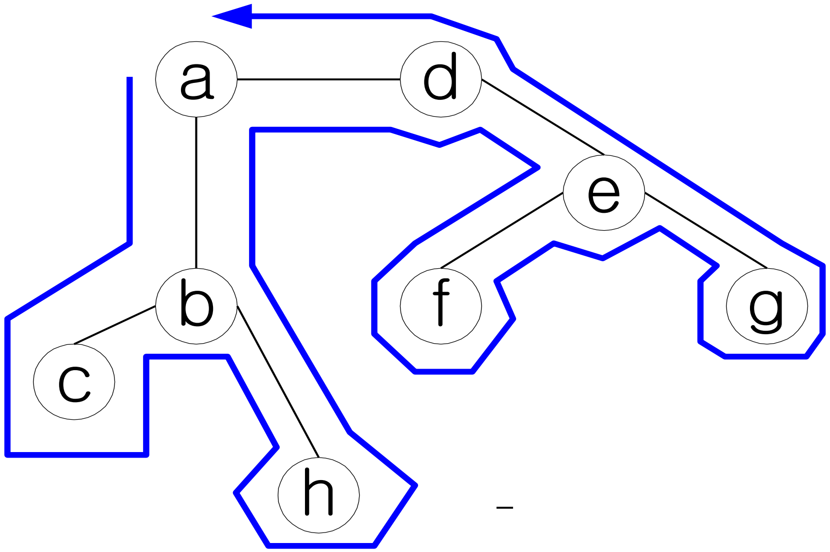


Example: Approx-TSP-Tour

Step 1: a minimum spanning tree



Step 2: a walk of T

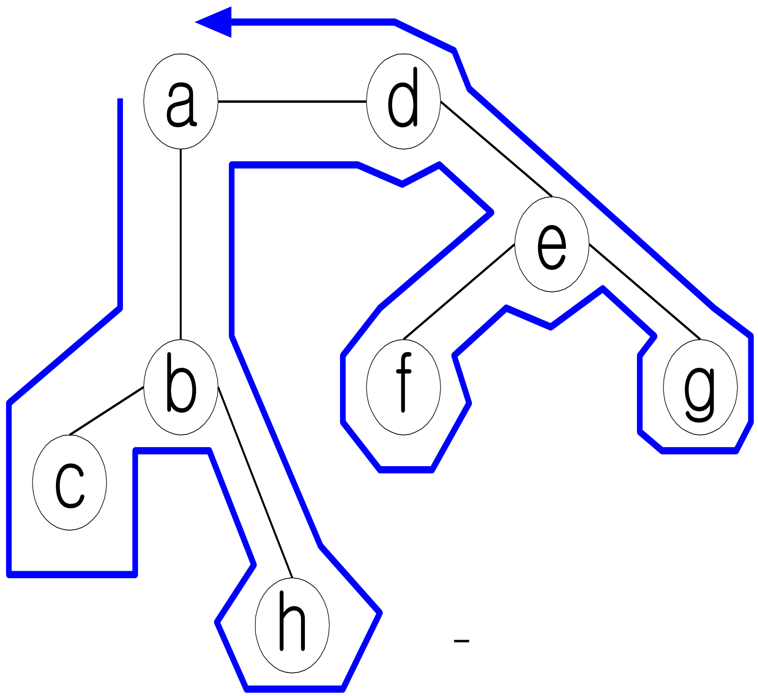


A full walk of T:

W={a,b,c,b,h,b,a,d,e,f,e,g,e,d,a}

Example: Approx-TSP-Tour

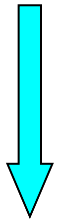
Step 2: a walk of T



W={a,b,c,b,h,b,a,d,e,f,e,g,e,d,a}

Step 3: remove from W all but the first visit to each vertex.

W={a,b,c,b,h,b,a,d,e,f,e,g,e,d,a}



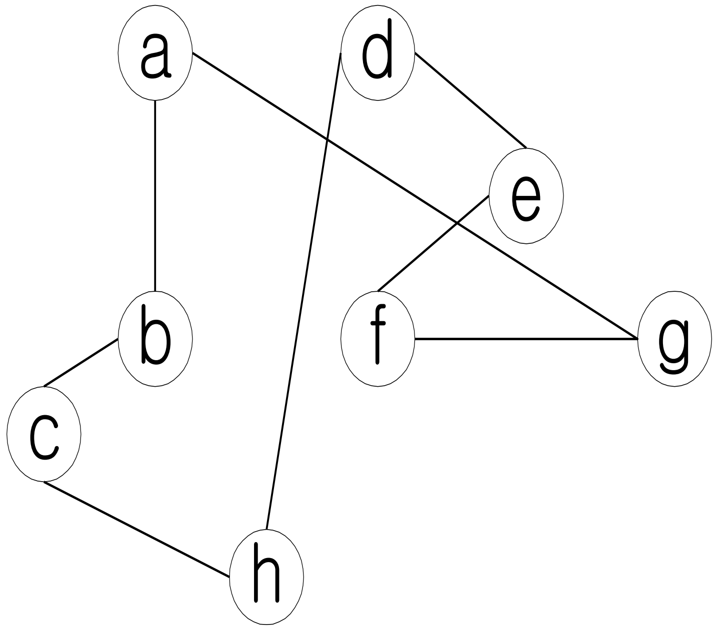
H={a,b,c,h,d,e,f,g}

: preorder walk

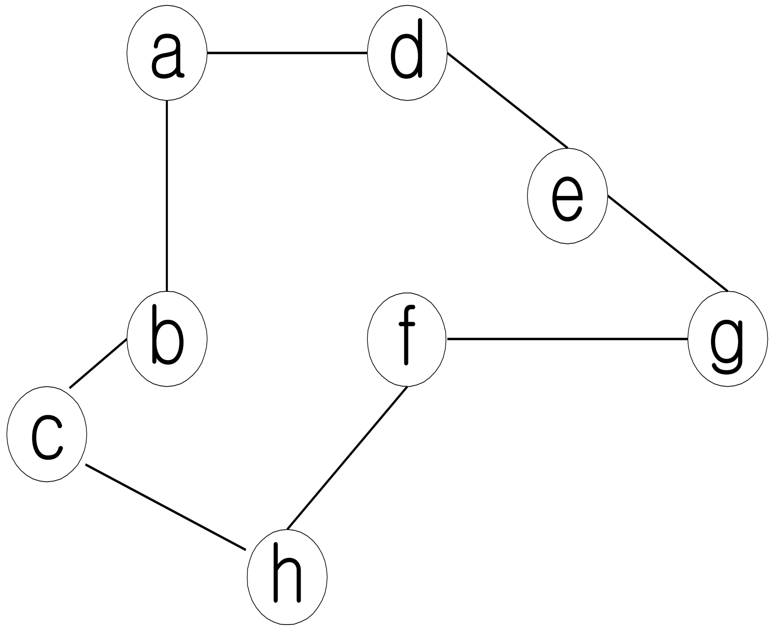
Example: Approx-TSP-Tour

A tour by Approx-TSP-Tour

H={a,b,c,h,d,e,f,g}



An optimal tour H*



$c(H^*) \le c(H) \le 2c(H^*)$

Approx-TSP-Tour is a polynomial-time 2-approximation algorithm for the traveling salesman problem