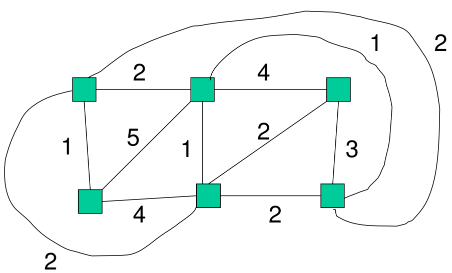
CSEP 521
Applied Algorithms
Autumn 2009

Traveling Salesman Problem

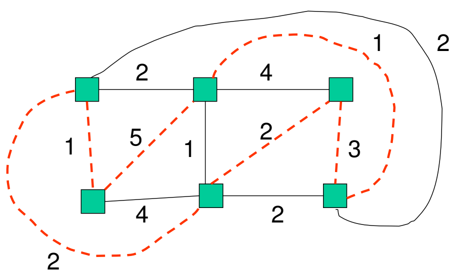
Traveling Salesman Problem

- Input: Undirected Graph G = (V,E) and a cost function C from E to the reals. C(e) is the cost of edge e.
- Output: A cycle that visits each vertex exactly once and is minimum total cost.

Example



Example



Cost = 1 + 5 + 1 + 3 + 2 + 2 = 14

Variations

- Hamiltonian Cycle
  - Is there a cycle that visits each vertex exactly once
  - Ignores costs
- Triangle inequality constraint
  - $C(u,v) \leq C(u,x) + C(x,v)$
- Euclidean Traveling Salesman
  - Vertices are points on the plane and the cost is the Euclidian distance between them
  - Implies triangle inequality

Applications

- Telescope planning
- Route planning
  - coin pickup
  - mail delivery
  - book order pickup in the Amazon warehouse
- Circuit board drilling

Why Traveling Salesman?

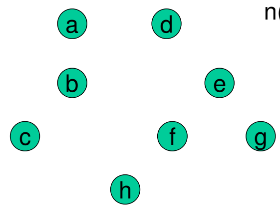
- Old well-studied problem
- Example of an NP-hard problem
  - These problems are very hard to solve exactly
  - No polynomial time algorithms known to exist
- Interesting and effective approximation algorithms
  - Good practical algorithms
  - Simple algorithms with provable approximation bounds

Approximation Alg. vs. Heuristic

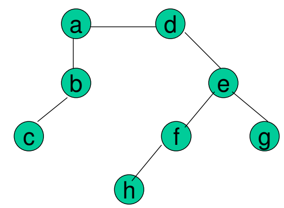
- Approximation Algorithm
  - There is a provable guarantee of how close the algorithm's result is to the optimal solution.
- Heuristic
  - The algorithm finds a solutions but there is no guarantee how good the solution is.
  - Heuristics often outperform provable approximation algorithms.

A Simple Approximation Algorithm

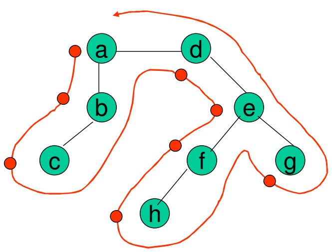
Euclidean distance
n(n-1)/2 edges



1. Find a Minimum Spanning Tree

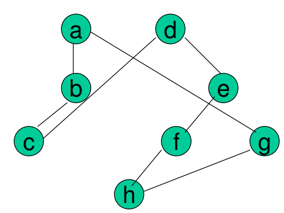


2. Depth-First Search of Tree



Marking Order = a, b, c, d, e, f, h, g

3. Connect Vertices in Marking Order



Marking Order = a, b, c, d, e, f, h, g

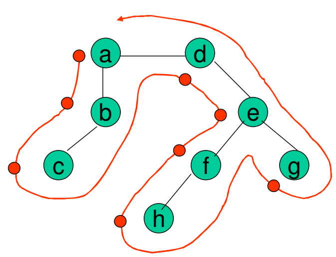
Evaluation

- Time and Storage
  - Time $O(n^2 \log n)$ with Kruskal's Algorithm
  - Storage $O(n^2)$
- Quality of Solution H
  - $C(H) \leq 2\,C(H^*)$ where H* is an optimal tour
  - This is a "2-approximation algorithm"
- Same approximation bound applies to case of triangle inequality

Proof of Approximation Bound

- Setup
  - T minimum spanning tree
  - W the depth-first walk of T
  - H the tour computed by the algorithms
  - H* an optimal tour

Depth-First Walk



$C(W) = 2\,C(T)$
$C(H) \leq C(W)$
triangle inequality

```
Depth-first walk = a,b,c,b,a,d,e,f,h,f,e,g,e,d,a
   Marking order = a,b,c,    d,e,f,h,     g
```


Proof of Approximation Bound

1. $C(W) = 2\,C(T)$
2. $C(H) \leq C(W)$, triangle inquality
3. $C(H) \leq 2\,C(T)$, last two lines
4. $C(T) \leq C(H^*)$, minus an edge H* is a spanning tree
5. $C(H) \leq 2\,C(H^*)$, last two lines

Solving TSP Exactly

- Branch-and-Bound
  - $n < 25$?
- Linear Programming
  - $n < 100$
- Cutting Plane Methods for Euclidian case
  - $n < 15{,}000$ with "concord"
  - see http://www.math.princeton.edu/tsp/

Solving TSP Approximately

- 3/2 – approximation algorithm of Christofedes
- Polynomial approximation scheme for Euclidian TSP by Aurora (1998), Mitchell (1999)
  - To get within $(1+\varepsilon)$ of optimal can be done in time polynomial in $1/\varepsilon$ and n.
  - These are not practical

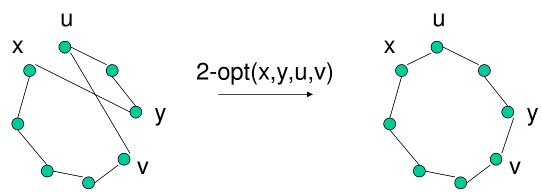
Solving TSP Approximately, Practically

- Local Search
  - Lin-Kernighan method
- Simulated Annealing
- Genetic Algorithms
- Neural Networks

Local Search Algorithms

- Start with an initial solution that is usually easy to find, but is not necessarily good.
- Repeatedly modify the current solution to a better nearby one. Until no nearby one is better.

2-Opt Neighborhood



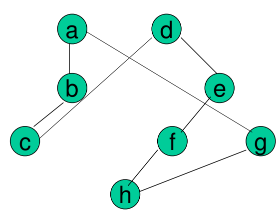
2-opt Algorithm

Lin-Kernighan (1973)

Find an initial tour T
1. For every pair of distinct edges (x,y), (u,v) in T
   if C(x,u) + C(y,v) < C(x,y) + C(u,v) then
      T := T – {(x,y),(u,v)} union {(x,u),(y,v)}
      exit for loop and go to 1
Return T

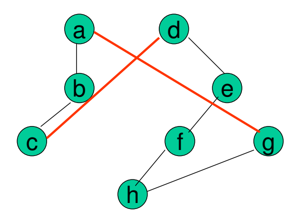
Example of LK

Euclidian case



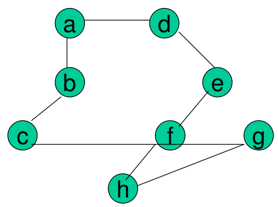
Example of LK

Euclidian case



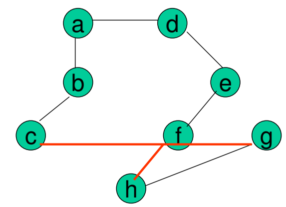
Example of LK

Euclidian case



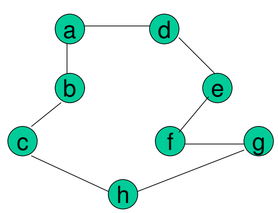
Example of LK

Euclidian case



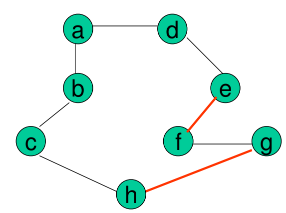
Example of LK

Euclidian case



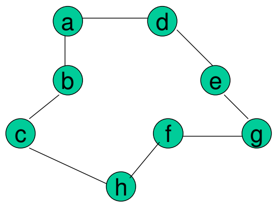
Example of LK

Euclidian case



Example of LK

Euclidian case



Lin-Kernighan

- Empirical $O(n^{2.2})$ time
- Finds optimal in most examples < 100 points
- Excellent Implementations
  - Can easily handle hundreds of thousands of points