Computer Science & Engineering 423/823
Design and Analysis of Algorithms
Lecture 12 — Approximation Algorithms (Chapter 34)

Stephen Scott and and Vinodchandran N. Variyam

Meanwhile, back at Evil Corp

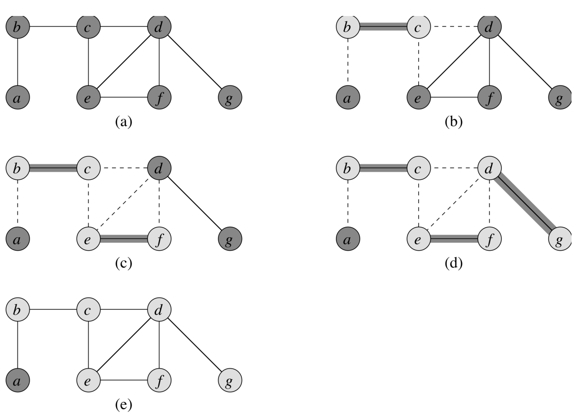
- Your boss wants you to do develop and implement an algorithm that
  1. Takes as input a building's floor plan, with hallways and junctions indicated
  2. Determines, in polynomial time, if one can place k omnidirectional cameras at junctions on a floor, such that each hallway is "covered" by at least one camera
     - (And if placement exists, output it)
- What should be your response? Why?
- Should you start updating your résumé?

Perhaps not all is lost

- This is, of course, our old friend (?) VERTEX-COVER where E = set of hallways and V = junctions
- What if you tried this:
  1. Let $E' = E$ and $C = \emptyset$
  2. Choose an arbitrary edge $(u, v) \in E'$ and add u and v to the cover C
  3. Delete from E' all edges covered by u or v
  4. Repeat until $E' = \emptyset$

Notes and Questions

Example

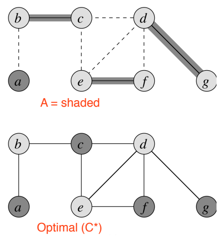


Is C a vertex cover?

Notes and Questions

So what?

Yes, C is a vertex cover, but can we say more?



- Let C^* be an optimal (smallest) vertex cover of G, and $A \subseteq E'$ be edges chosen in line 2
- No two edges from A can be covered by the same vertex, so $|C^*| \geq |A|$
- Since we add two vertices per chosen edge, $|C| = 2|A|$

$\Rightarrow$ $|C| \leq 2|C^*|$, i.e., the algorithm's output will be at most twice optimal

Theorem: This algorithm is a polynomial-time **2-approximation algorithm**

Notes and Questions

Approximation algorithms

- An algorithm is a polynomial time $\rho(n)$**-approximation algorithm** if it has a guaranteed **approximation ratio** of $\rho(n)$, where

$$\rho(n) \geq \max\left(\frac{C}{C^*}, \frac{C^*}{C}\right) \quad ,$$

  where C is the cost of the algorithm's solution and C^* is the cost of an optimal solution
- Note that the ratio can depend on n, the size of the input (VERTEX-COVER algorithm had a constant ratio)
- Definition applies both to minimization and maximization problems

Notes and Questions

Another Approximation Algorithm: TSP with Triangle Inequality

- Optimization version of the NP-complete problem TSP: Given a complete, undirected, weighted graph G, find a Hamiltonian cycle of minimum weight (cost)
- Approximation algorithm exists if the cost function c satisfies the **triangle inequality**: for all $u, v, w \in V$,

$$c(u, w) \leq c(u, v) + c(v, w)$$

  (I.e., a direct edge from u to w is never worse than going through some intermediate vertex v)
- Holds if, e.g., c is Euclidean distance

Notes and Questions

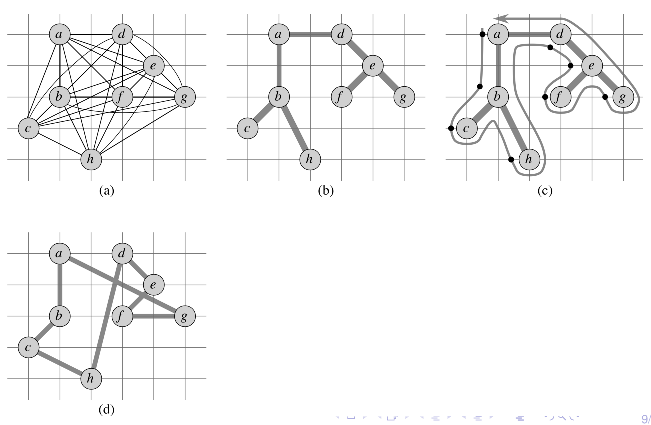
The algorithm

1. Select arbitrary vertex $r \in V$ to be root
2. Compute MST T of G from r via Prim's algorithm
3. Let H be a list of vertices in the order of a preorder walk of T
4. Return the Hamiltonian cycle H

- G is complete, so H is guaranteed to be a Hamiltonian cycle

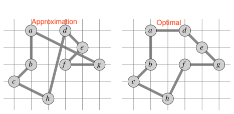
Notes and Questions

Example



Notes and Questions

Approximation ratio



- Let H^* be an optimal (smallest) tour of G
- Deleting any edge from H^* yields a spanning tree, so $c(T) \leq c(H^*)$, since T is an MST
- A **full walk** W of tree T is a listing of each vertex every time it's visited in preorder traversal, e.g., $W = \langle a, b, c, b, h, b, a, d, e, f, e, g, e, d, a \rangle$
- W traverses every edge in T twice: $c(W) = 2c(T)$, so $c(W) \leq 2c(H^*)$
- Transform walk W into tour H by listing each vertex only when it first appears: $H = \langle a, b, c, h, d, e, f, g \rangle$
- Because of triangle inequality, can go directly from u to w, skipping v, without increasing cost, e.g., $c(f, g) \leq c(f, e) + c(e, g)$, so $c(H) \leq c(W) \leq 2c(H^*)$

Theorem: This algorithm is poly-time **2-approximation algorithm** for TSP when triangle inequality holds

Notes and Questions

Why do we need the triangle inequality?

Theorem: If P $\neq$ NP, then for any constant $\rho \geq 1$, there is no polynomial-time algorithm with approximation ratio ρ for general TSP

- **Proof:** Reduce HAM-CYCLE to this problem
- Transform instance $\langle G \rangle$ of HAM-CYCLE to instance $\langle G', c \rangle$ of TSP (optimization) where G' is a complete graph and

$$c(u,v) = \begin{cases} 1 & \text{if } (u,v) \in E \\ \rho|V| + 1 & \text{otherwise} \end{cases}$$

- If G has a Hamiltonian cycle, there is a TSP tour of cost $|V|$, so a ρ-approximation tour would have cost $\leq \rho|V|$
- If G has no Hamiltonian cycle, the cheapest tour's cost is at least

$$(\rho|V| + 1) + (|V| - 1) = \rho|V| + |V| > \rho|V|$$

$\Rightarrow$ If in polynomial time we can get a ρ-approximation of an optimal TSP tour, then we can compare its cost to $\rho|V|$ to solve HAM-CYCLE in polynomial time :-)

Notes and Questions