Coping with NP-Completeness

Siddhartha Sen
Questions: sssix@cs.princeton.edu

Some figures obtained from Introduction to Algorithms, 2nd ed., by CLRS

Coping with intractability

Many NPC problems are important in industry and must be solved

Methods for coping:

- Special cases
- Average case
- Approximation algorithms
- Intelligent brute force
- Heuristics

Special cases

2-CNF-SAT can be solved in p-time, but 3-CNF-SAT and higher are NPC

- Uses strongly connected components

VERTEX-COVER for bipartite graphs can be solved in p-time, but NPC in general

- Uses maximum matching

Approximation algorithms

Returns near-optimal solution to a minimization/maximization problem

Algorithm has approximation ratio $\rho(n) \geq 1$ if cost C of solution is within factor $\rho(n)$ of cost C^* of optimal solution, for any input of size n:

Maximization: $0 < C \leq C^*$, ratio is C^*/C

Minimization: $0 < C^* \leq C$, ratio is C/C^*

Types of approximation algorithms

$\rho(n)$ may be:

- Constant, e.g. 2
- Growing function of input size n, e.g. $\log n$
- Approximation scheme: $(1 + \varepsilon)$ for fixed $\varepsilon > 0$

Approximation scheme that runs in p-time in n, e.g. $O(n^{2/\varepsilon})$, is a **polynomial-time approximation scheme** (PTAS)

If additionally p-time in $1/\varepsilon$, e.g. $O((1/\varepsilon)^2 n^3)$, it is a **fully polynomial-time approximation scheme**

Approximate VERTEX-COVER

VERTEX-COVER = $\{\langle G, k\rangle$: graph $G = (V, E)$ has vertex cover of size $k\}$

Optimization version: Find vertex cover of G of minimum size

Idea: Find maximal matching greedily

APPROX-VC (G)

$C \leftarrow 0$

$E' = E$

while $E' \neq \emptyset$

    Select from E' an arbitrary edge (u, v)

    $C \leftarrow C \cup \{u, v\}$

    Remove from E' every edge incident to u or v

return C

Theorem. APPROX-VC is a p-time 2-approximation algorithm for VERTEX-COVER

Proof. Running time is O(E). C is vertex cover since edge only removed if has endpoint in C. Show $|C| \leq 2|C^*|$.

Let A be set of edges selected from E'

A is a (maximal) matching since no two edges share common endpoint $\Rightarrow |C^*| \geq |A|$

Each selected edge has neither endpoint in C, so $|C| = 2|A|$
$\Rightarrow |A| \leq |C^*| \leq |C| = 2|A|$

Similar idea of lower-bounding optimal solution (when unknown) used in competitive analysis

Approximate TSP

TSP = $\{\langle G, d, k\rangle : G = (V, E)$ is a complete graph, d is cost function from $V \times V \rightarrow \mathbf{Z}$, $k \in \mathbf{Z}$, and G has tour of cost at most $k\}$

Tour is cycle that visits each vertex exactly once

Optimization version: Find a minimum tour of G

Use $d(L)$ to denote total cost of edges in L

Assume triangle inequality:

$d(x, y) + d(y, z) \geq d(x, z)$

Idea: Find a minimum spanning tree

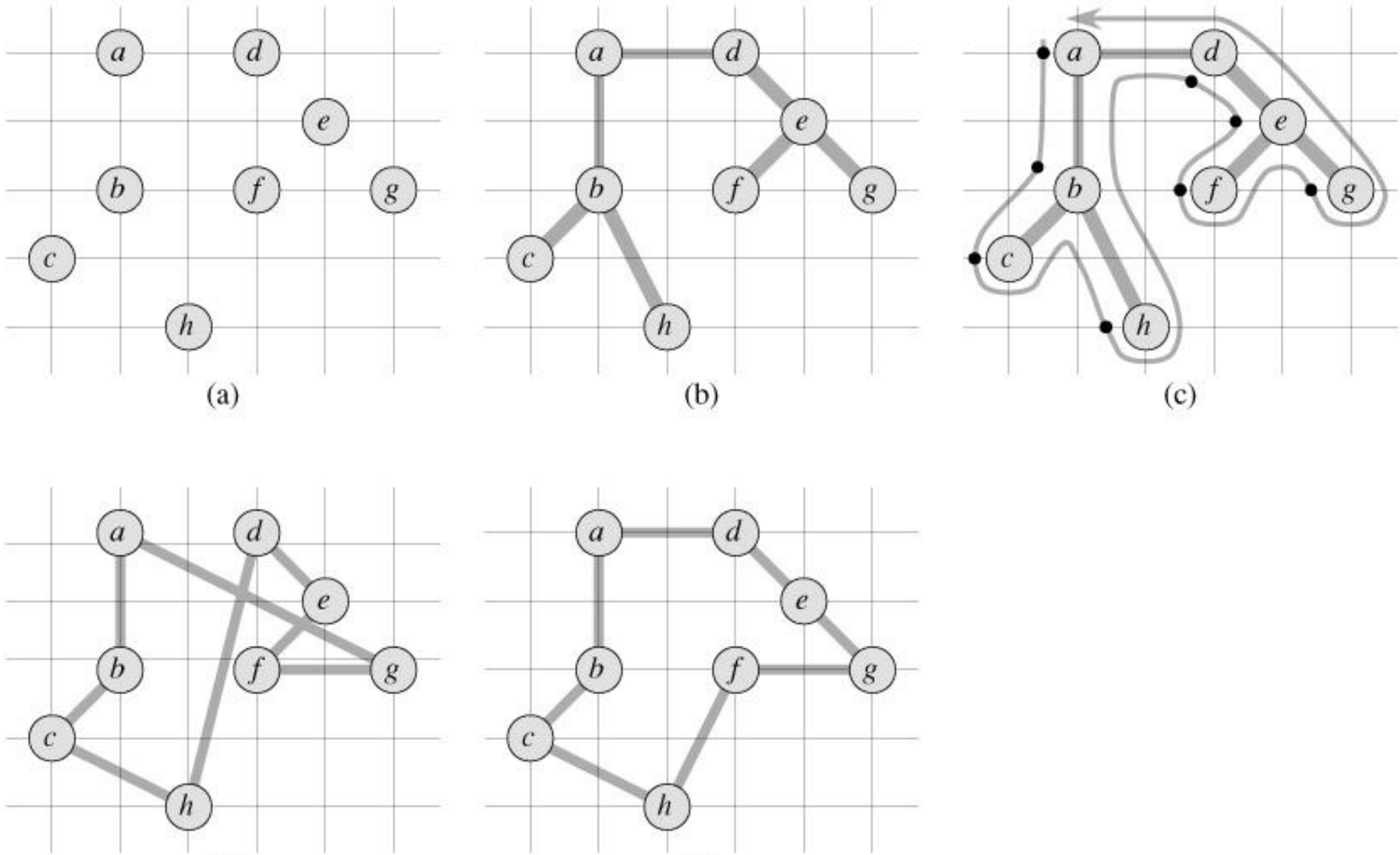
APPROX-TSP (G)

Select arbitrary vertex $r \in V$ to be root

Find a minimum spanning tree T from r in G

Let C be list of vertices from preorder walk of T

return C



(a) (b) (c) (d) (e)

APPROX-TSP's solution

Optimal solution

Theorem. APPROX-TSP is a p-time 2-approximation algorithm for TSP

Proof. Runtime is time to find MST + preorder traversal, i.e. p-time. C is tour because includes all vertices (definition of MST). Show $d(C) \leq 2d(C^*)$

Removing one edge from C^* gives spanning tree, so $d(T) \leq d(C^*)$

Full walk of T traverses each edge twice; since C is contracted version of full walk, $d(C) \leq 2d(T)$

- By triangle inequality, can delete any vertex from full walk without increasing cost

1.5-approximation for TSP

- Let O be set of odd-degree vertices in T; find minimum-cost perfect matching M over O
  - O is connected by complete graph and $|O|$ is even, so perfect matching exists
- Find Eulerian tour on $T \cup M$, delete repeated vertices
  - $T \cup M$ is connected, has even-degree vertices only, so Eulerian tour exists
  - Triangle inequality lets us delete vertices

Why is this a 1.5 approximation?

$d(T) \le d(C^*)$ as before

Show $d(M) \le d(C^*)/2$, see below

$\Rightarrow d(T \cup M) \le 1.5d(C^*)$

Let $C_O{}^*$ be optimal tour of O, let $e_1, e_2, \ldots, e_{2k}$ be its edge set (Eulerian path). Both $e_1, e_3, \ldots, e_{2k-1}$ and $e_2, e_4, \ldots, e_{2k}$ are perfect matchings, so one has cost $\le d(C_O{}^*)/2 \le d(C^*)/2$

PTAS for SUBSET-SUM

SUBSET-SUM = $\{\langle S, t\rangle : S \subset \mathbf{N}, t \in \mathbf{N}$ and $\exists$ a subset $S' \subseteq S$ s.t. $t = \sum_{s \in S'} s\}$

Optimization version: Find subset whose sum is closest to t without exceeding t

Idea: Maintain list of sums of all subsets so far, but trim list

APPROX-SUBSET-SUM(S, t, ε)

$n \leftarrow |S|$

$L_0 \leftarrow \langle 0 \rangle$

for $i \leftarrow 1$ **to** n

    do $L_i \leftarrow$ MERGE(L_{i-1}, $L_{i-1} + x_i$)

    ~~$L_i \leftarrow$ TRIM(L_i, $\varepsilon/2n$)~~

    remove from L_i every element > t

return largest value z* in L_n

APPROX-SUBSET-SUM(S, t, ε)

$n \leftarrow |S|$

$L_0 \leftarrow \langle 0 \rangle$

for $i \leftarrow 1$ **to** n

    do $L_i \leftarrow$ MERGE(L_{i-1}, $L_{i-1} + x_i$)

    $L_i \leftarrow$ TRIM(L_i, $\varepsilon/2n$)

    remove from L_i every element $> t$

return largest value z* in L_n

TRIM removes every y in L_i for which $\exists$ z still in L_i s.t.:

$$\frac{y}{1+\frac{\varepsilon}{2n}} \leq z \leq y$$

- $S = \{104, 102, 201, 101\}$, $t = 308$, $\varepsilon = 0.40$, $\varepsilon/2n = 0.05$

line 2:	L_0	=	$\langle 0 \rangle$
line 4:	L_1	=	$\langle 0, 104 \rangle$
line 5:	L_1	=	$\langle 0, 104 \rangle$
line 6:	L_1	=	$\langle 0, 104 \rangle$
line 4:	L_2	=	$\langle 0, 102, 104, 206 \rangle$
line 5:	L_2	=	$\langle 0, 102, 206 \rangle$
line 6:	L_2	=	$\langle 0, 102, 206 \rangle$
line 4:	L_3	=	$\langle 0, 102, 201, 206, 303, 407 \rangle$
line 5:	L_3	=	$\langle 0, 102, 201, 303, 407 \rangle$
line 6:	L_3	=	$\langle 0, 102, 201, 303 \rangle$
line 4:	L_4	=	$\langle 0, 101, 102, 201, 203, 302, 303, 404 \rangle$
line 5:	L_4	=	$\langle 0, 101, 201, 302, 404 \rangle$
line 6:	L_4	=	$\langle 0, 101, 201, 302 \rangle$

Theorem. APPROX-SUBSET-SUM is a fully PTAS for SUBSET-SUM

Proof. If y^* is optimal solution, need to show $y^*/z^* \leq 1 + \varepsilon$. By induction on *i* and recalling trimming equation, can show:

$$\frac{y}{\left(1+\frac{\varepsilon}{2n}\right)^i} \leq z \leq y$$

$$\Rightarrow \frac{y^*}{z^*} \leq \left(1+\frac{\varepsilon}{2n}\right)^n$$

$$\leq e^{\frac{\varepsilon}{2}}$$

$$\leq 1+\frac{\varepsilon}{2}+\left(\frac{\varepsilon}{2}\right)^2$$

$$\leq 1+\varepsilon$$

To show p-time, bound length of L_i. After trimming, successive elements z, z' must satisfy $z'/z > 1 + \varepsilon/2n \Rightarrow$ size of L_i is at most:

$$
\begin{aligned}
\log_{1+\varepsilon/2n} t + 2 &= \frac{\ln t}{\ln(1+\varepsilon/2n)} + 2 \\
&\le \frac{2n(1+\varepsilon/2n)\ln t}{\varepsilon} + 2 \\
&\le \frac{4n\ln t}{\varepsilon} + 2
\end{aligned}
$$