Approximation Algorithms I

2023/11/23

詹博华（中国科学院软件研究所）

After NP-complete

When a problem is shown to be NP-complete, there are various approaches to deal with it:

- If the actual inputs are small, then an exponential algorithm may be sufficient.
- Isolate special cases that can be solved in polynomial time.
- **Approximation algorithms.**
- Heuristic algorithms, including encoding into a boolean satisfiability problem, then use state-of-the-art SAT solvers.

Approximation Ratio

- For an optimization problem, we say an algorithm guarantees an **approximation ratio** $\rho(n)$, if for any input size n, the ratio between the cost of solution found and the optimal solution is bounded by $\rho(n)$.
- For minimization problems, let C be the cost of solution found and C^* be the cost of optimal solution, then $C^* \leq C$, and
$$\frac{C}{C^*} \leq \rho(n)$$
- For maximization problems, we have instead $C \leq C^*$, and
$$\frac{C^*}{C} \leq \rho(n)$$

Approximation Ratio

- When the guaranteed approximation ratio is independent of n, we say the algorithm has approximation ratio ρ.
- Approximation ratio of 1: the algorithm is actually optimal.
- Approximation ratio of 2: the algorithm always find a solution whose value is within a factor 2 of optimal.
- Approximation ratio of $1 + \epsilon$: a class of algorithms that can reach approximation ratio arbitrarily close to 1 (polynomial-time approximation scheme).

Vertex-cover problem

- Consider again the vertex-cover problem: given graph $G = (V, E)$, find a subset $V' \subseteq V$ of minimum size, such that each edge is incident on at least one vertex in V'.
- Optimal vertex cover: vertex cover of minimum size.
- We now present an approximation algorithm that returns a vertex cover that is at most twice the size of optimal vertex cover (**with approximation ratio 2**).

Approximate vertex-cover

Iteratively perform the following:

- Pick an arbitrary edge (u, v) remaining in G.
- Remove all edges incident on u or v from G.

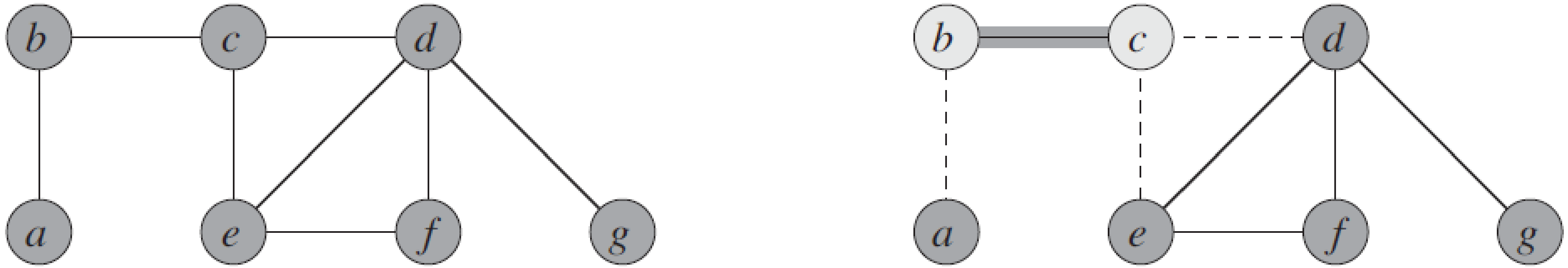
Until all edges are removed. Let V' be all vertices on the chosen edges.

APPROX-VERTEX-COVER(G)

$C = \emptyset$
$E' = G.E$
while $E' \neq \emptyset$
    let (u, v) be an arbitrary edge of E'
    $C = C \cup \{u, v\}$
    remove from E' every edge incident on either u or v
return C

Example: Step 1

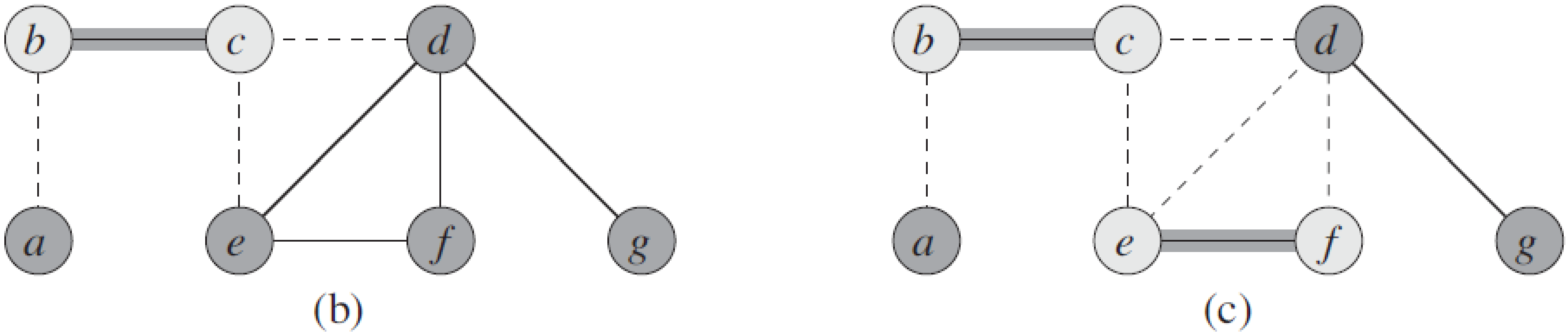
- Starting with graph on the left.
- Arbitrarily decide to choose edge (b, c). Then remove all edges incident on b and on c (right).



(a) (b)

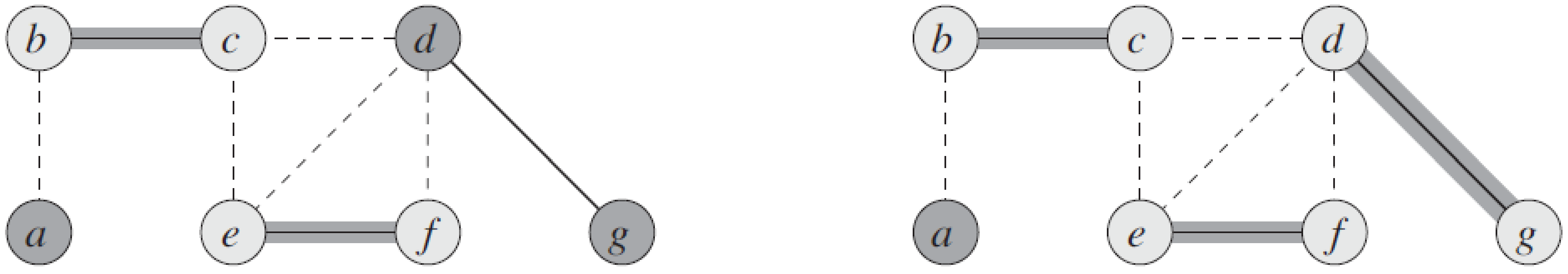
Example: Step 2

- Arbitrarily choose edge (e, f), then remove all edges incident on e and on f.



Example: Step 3

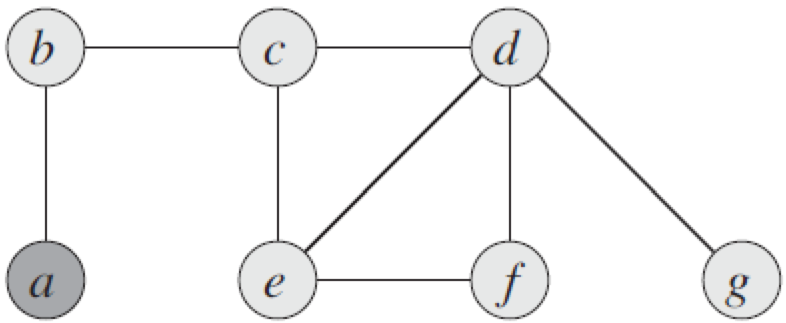
- Choose the final edge (d, g).
- The set of chosen vertices is $\{b, c, d, e, f, g\}$, forming a vertex cover.



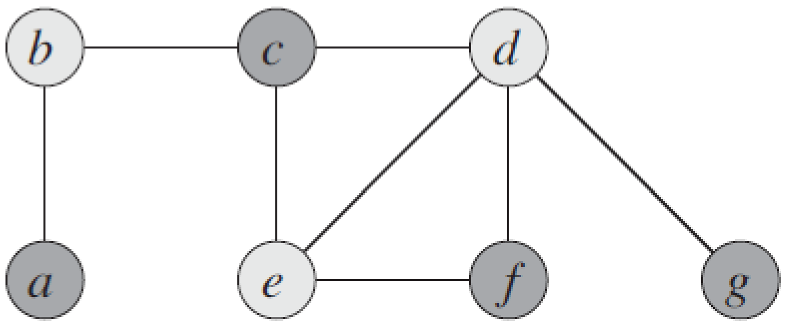
(c) (d)

Obtained vs. Best vertex cover

- Left: the obtained vertex cover $\{b, c, d, e, f, g\}$.
- Right: optimal vertex cover $\{b, d, e\}$.
- Although obtained vertex cover looks poor, it is at least within a factor of 2 within the optimal.



(e)



(f)

Proof of approximation ratio

Theorem: the given algorithm has approximation ratio 2.

Proof: the edges that are picked are all disjoint from each other (do not share any vertices). Hence, if m edges are picked, then any vertex cover must have at least m vertices (one to cover each of the m edges). Hence, the optimal vertex cover has at least m vertices, while the vertex cover returned has $2m$ vertices.

Main idea: we can obtain a lower bound on the optimal value, without knowing the optimal value itself!

Traveling-salesman problem

- **Recall the problem:** given a graph where each pair of vertices i, j is connected by an edge with cost $c(i, j)$, find the Hamiltonian cycle with smallest total cost.
- We now add an additional assumption: the **triangle inequality**

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

  holds between any three vertices u, v, w. This expresses the intuition that it is always no harder to go directly from u to w than adding an intermediate stop at v.
- This condition holds in many practical applications (e.g. Euclidean distance).

Approximation algorithm

- Traveling-salesman problem with triangle inequality is still NP-complete (our reduction from Hamiltonian cycle to TSP yesterday satisfies triangle inequality).
- We now show an approximation algorithm for traveling-salesman problem with triangle inequality, with approximation ratio 2.
- Without triangle inequality assumption, there is no polynomial algorithm with any fixed approximation ratio.

Approximation algorithm

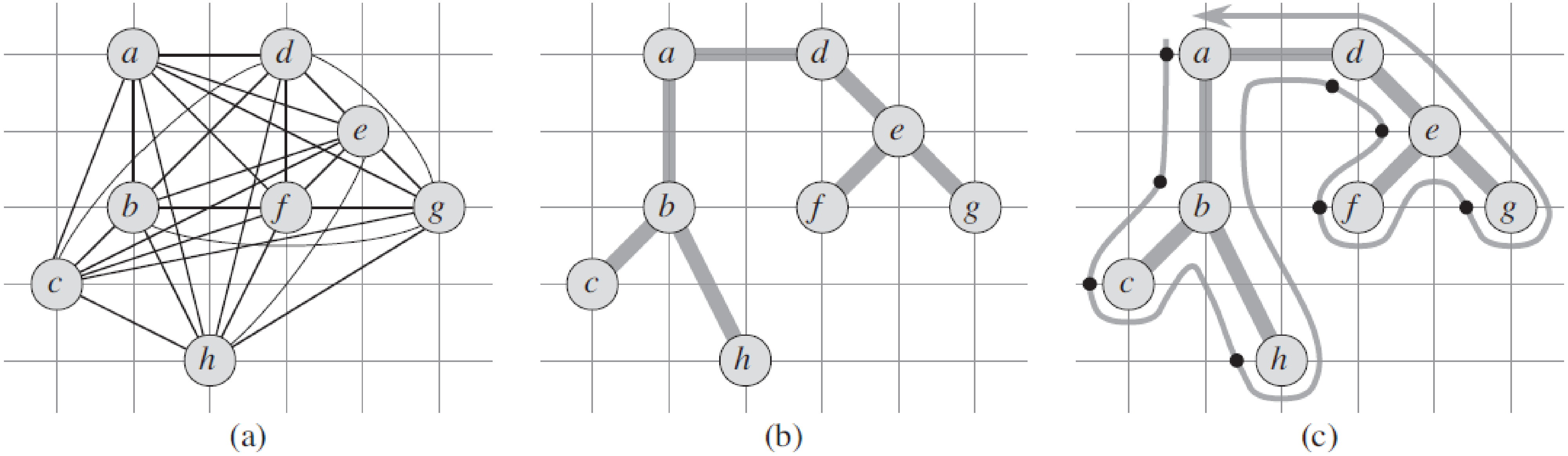
The algorithm has two steps:

1. Compute minimal spanning tree of G (starting from an arbitrary vertex).
2. Traverse the vertices of G according to the preorder traversal of the minimal spanning tree.

```
APPROX-TSP-TOUR(G, c)
select a vertex r ∈ G.V to be a "root" vertex
compute a minimum spanning tree T for G from root r
        using MST-PRIM(G, c, r)
let H be a list of vertices, ordered according to when they are first visited
        in a preorder tree walk of T
return the hamiltonian cycle H
```

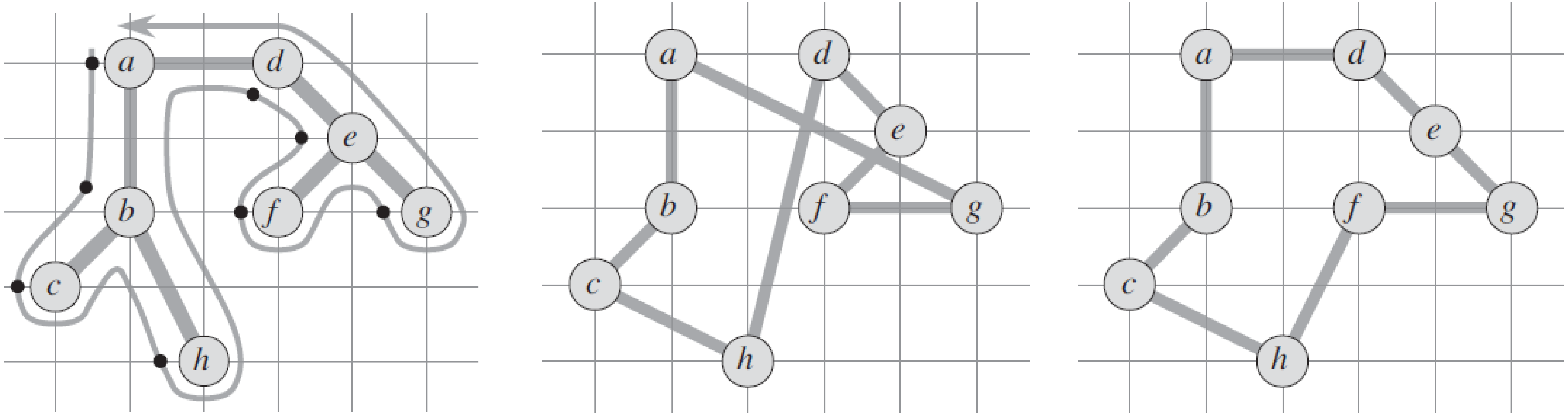
Approximation algorithm: Example

- (a) The original graph.
- (b) The minimum spanning tree from root a.
- (c) Preorder traversal on the minimum spanning tree.



Approximate algorithm: Example

- (d) Preorder traversal as a Hamiltonian cycle.
- (e) Optimal Hamiltonian cycle.



(c) (d) (e)

Approximation algorithm

Theorem: the given algorithm has approximation ratio 2.

Proof:

1. The cost of the minimal spanning tree gives a lower bound on the cost of the optimal Hamiltonian cycle:
$$c(T) \leq c(H^*).$$
2. The preorder traversal on the minimal spanning tree is a cycle that has twice the cost of T:
$$c(W) = 2c(T).$$
3. The cost of obtained Hamiltonian cycle is less than the preorder traversal (repeatedly apply triangle inequality).

Better Approximation Algorithms

- This result has been improved upon in subsequent work.
- Christofides improved on this algorithm and gave a 3/2-approximation algorithm.
- Arora and Mitchell have shown that for points on the Euclidean plane, there is a polynomial-time approximation scheme. (See textbook for references).

Proof of no approximation ratio

- The above algorithm crucially assumes the distances satisfy the triangle inequality.
- We next show that, if the assumption of triangle inequality is dropped, then there is no approximate ratio is possible unless $\mathrm{P} = \mathrm{NP}$.

Proof of no approximation ratio

- By contradiction, we assume there is a polynomial time algorithm A with approximation ratio ρ.
- We show how to use algorithm A to solve the (exact) Hamiltonian cycle problem.
- Given $G = (V, E)$ an instance of Hamiltonian cycle problem, convert into an instance G' of traveling salesman problem as follows:

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

Proof of no approximation ratio

- If the original graph G has a Hamiltonian cycle, then G' contains a tour of cost $|V|$.
- On the other hand, any tour of G' that use an edge not in G has cost at least
$$(\rho|V| + 1) + (|V| - 1) = (\rho + 1)|V| > \rho|V|$$
- Hence, the algorithm A with approximation ratio ρ must return a cycle corresponding to the Hamiltonian cycle in G if it exists, and so can solve the Hamiltonian cycle problem!