The Design and Analysis of Algorithms

Lecture 17 Intractability II

Zhenbo Wang

Department of Mathematical Sciences, Tsinghua University

Content

Polynomial-Time Reduction

- Suppose we could solve Y in polynomial-time. What else could we solve in polynomial time?
- *Reduction*. Problem X polynomial reduces to problem Y if arbitrary instances of problem X can be solved using:

  Polynomial number of standard computational steps, plus

  Polynomial number of calls to oracle that solves problem Y.
- *Notation*. $X \leq_P Y$.

Note. We pay for time to write down instances sent to oracle $\Rightarrow$ instances of Y must be of polynomial size.

Polynomial-Time Reduction

- *Design algorithms*. If $X \leq_P Y$ and Y can be solved in polynomial time, then X can be solved in polynomial time.
- *Establish intractability*. If $X \leq_P Y$ and X cannot be solved in polynomial time, then Y cannot be solved in polynomial time.
- *Establish equivalence*. If both $X \leq_P Y$ and $Y \leq_P X$, we use notation $X \equiv_P Y$.

  In this case, X can be solved in polynomial time iff Y can be.

NP-Complete

- *NP-complete*. A problem $Y \in NP$ with the property that for every problem $X \in NP$, $X \leq_p Y$.

Theorem 1

Suppose $Y \in NP - complete$. Then $Y \in P$ iff $P = NP$.

Pf. $\Leftarrow$ If $P = NP$, then $Y \in P$ because $Y \in NP$.

$\Rightarrow$ Suppose $Y \in P$.

Consider any problem $X \in NP$. Since $X \leq_P Y$, we have $X \in P$.

This implies $NP \subseteq P$.

We already know $P \subseteq NP$. Thus $P = NP$. $\square$

NP-Complete

Theorem 2

If $X \in NP - complete$, $Y \in NP$, and $X \leq_P Y$, then $Y \in NP - complete$.

Pf. Consider any problem $W \in NP$. Then, both $W \leq_p X$ and $X \leq_p Y$.

By transitivity, $W \leq_p Y$.

Hence $Y \in NP - complete$. □

- *Fundamental question.* Do there exist "natural" *NP-complete* problems?

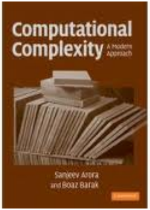
The “First” NP-complete Problem: 3-SAT

Theorem 3 (Cook 1971, Levin 1973)

3-SAT is NP-Complete.

Pf. The proof is deferred to *Computational Complexity* (Spring, 2016), or referred to (Arora and Barak 2009).

S. Arora and B. Barak. Computational Complexity: A modern Approach, Cambridge University Press, 2009.

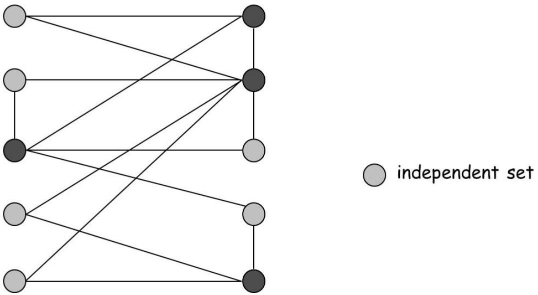


Independent Set

- *INDEPENDENT SET*: Given a graph $G = (V, E)$ and an integer k, is there a subset of vertices $S \subseteq V$ such that $|S| \geq k$, and for each edge at most one of its endpoints is in S?

Ex. Is there an independent set of size ≥ 6? Yes.

Ex. Is there an independent set of size ≥ 7? No.



3-Satisfiability Reduces to Independent Set

Theorem 4

3-SAT $\leq_P$ INDEPENDENT-SET.

Pf. Given an instance Φ of *3-SAT* with n variables and m clauses, we construct an instance (G, k) of *INDEPENDENT-SET* that has an independent set of size m iff Φ is satisfiable.

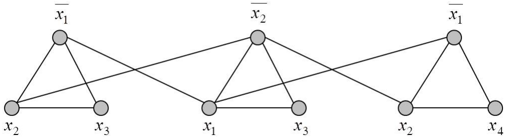


Figure 1: $\Phi = (\bar{x}_1 \vee x_2 \vee x_3) \wedge (x_1 \vee \bar{x}_2 \vee x_3) \wedge (\bar{x}_1 \vee x_2 \vee x_4)$

Construction: G contains 3 nodes for each clause, one for each literal.

Connect 3 literals in a clause in a triangle.

Connect literal to each of its negations.

3-Satisfiability Reduces to Independent Set

Claim. G contains independent set of size m iff Φ is satisfiable.

$\Rightarrow$ Let S be independent set of size m.

S must contain exactly one node in each triangle.

Set these literals to *true*.

Truth assignment is consistent and all clauses are satisfied.

$\Leftarrow$ Given satisfying assignment, select one true literal from each triangle.

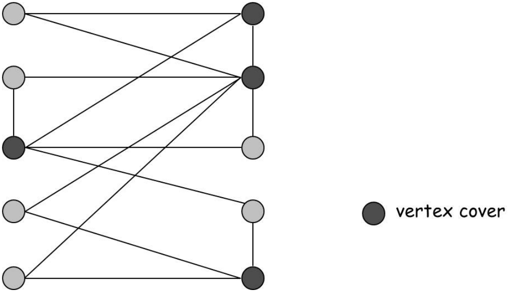
This is an independent set of size m. $\square$

Vertex Cover

- *VERTEX-COVER*: Given a graph $G = (V, E)$ and an integer k, is there a subset of vertices $S \subseteq V$ such that $|S| \leq k$, and for each edge at least one of its endpoints is in S?

Ex. Is there a vertex cover of size ≤ 4? Yes.

Ex. Is there a vertex cover of size ≤ 3? No.

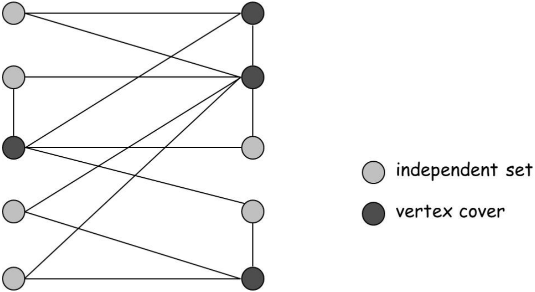


Vertex Cover and Independent Set

Theorem 5

VERTEX-COVER $\equiv_P$ *INDEPENDENT-SET*.

Pf. We show S is an independent set of size k iff $V - S$ is a vertex cover of size $n - k$.



$\Rightarrow$ Let S be any independent set of size k.

$V - S$ is of size $n - k$.

Consider an arbitrary edge (u, v).

Vertex Cover and Independent Set–Con't

$\Leftarrow$ Let $V - S$ be any vertex cover of size $n - k$.

S is of size k.

Consider two nodes $u \in S$ and $v \in S$.

Observe that $(u, v) \notin E$ since $V - S$ is a vertex cover.

Thus, no two nodes in S are joined by an edge $\Rightarrow$ S independent set. $\square$

Set Cover

- *SET COVER*: Given a set U of elements, a collection $S_1, S_2, \cdots, S_m$ of subsets of U, and an integer k, does there exist a collection of $\leq k$ of these sets whose union is equal to U?

Ex. $U = \{1, 2, 3, 4, 5, 6, 7\}, k = 2$

$S_1 = \{3, 7\}$, $S_2 = \{3, 4, 5, 6\}$, $S_3 = \{1\}$, $S_4 = \{2, 4\}$, $S_5 = \{5\}$ and $S_6 = \{1, 2, 6, 7\}$

S_2 and S_6 is a set cover.

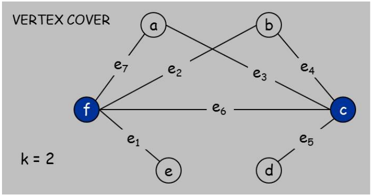
Vertex Cover Reduces to Set Cover

Claim. *VERTEX-COVER* $\leq_P$ *SET-COVER*.

Pf. Given a *VERTEX-COVER* instance $G = (V, E)$, we construct a SET-COVER instance (U, S) that has a set cover of size k iff G has a vertex cover of size k.

Construction: Universe $U = E$; $S_v = \{e \in E$: e incident to v$\}$.

Set-cover of size $\leq k$ iff vertex cover of size $\leq k$. □



SET COVER

$U = \{1, 2, 3, 4, 5, 6, 7\}$
$k = 2$

$S_a = \{3, 7\}$	$S_b = \{2, 4\}$
$S_c = \{3, 4, 5, 6\}$	$S_d = \{5\}$
$S_e = \{1\}$	$S_f = \{1, 2, 6, 7\}$

Review

- Basic reduction strategies.

  Simple equivalence: *INDEPENDENT-SET* $\equiv_P$ *VERTEX-COVER*.

  Special case to general case: *VERTEX-COVER* $\leq_P$ *SET-COVER*.

  Encoding with gadgets: *3-SAT* $\leq_P$ *INDEPENDENT-SET*.

- *Transitivity*. If $X \leq_P Y$ and $Y \leq_P Z$, then $X \leq_P Z$.

Ex. *3-SAT* $\leq_P$ *INDEPENDENT-SET* $\leq_P$ *VERTEX-COVER* $\leq_P$ *SET-COVER*.

Three Versions of Problems

- *Decision problem*. Does there exist a vertex cover of size $\leq k$?
- *Search problem*. Find a vertex cover of size $\leq k$.
- *Optimization problem*. Find a vertex cover of minimum size.

Claim. *VERTEX-COVER* $\equiv_P$ *FIND-VERTEX-COVER* $\equiv_P$ *OPTIMAL-VERTEX-COVER*.

FIND-VERTEX-COVER reduces to *VERTEX-COVER* via iteratively deleting a node.

OPTIMAL-VERTEX-COVER reduces to *FIND-VERTEX-COVER* by binary searing the optimal size k^*.

Hamiltonian Cycle

- *HAM-CYCLE*: given an undirected graph $G = (V, E)$, does there exist a simple cycle that contains every node in V?

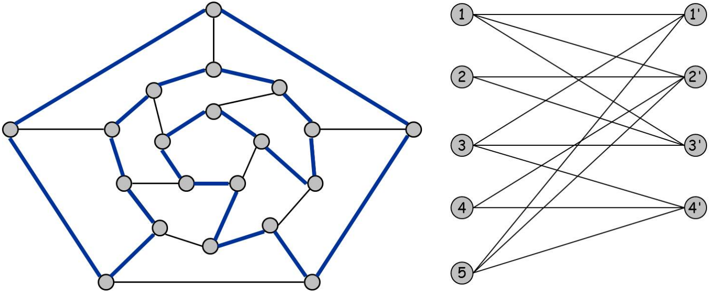


Figure 2: Left: yes; right: no

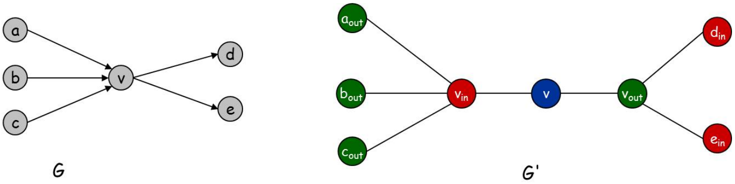
Directed Hamiltonian Cycles

- *DIR-HAM-CYCLE*: given a digraph $G = (V, E)$, does there exists a simple directed cycle that contains every node in V?

Theorem 6

DIR-HAM-CYCLE $\leq_P$ *HAM-CYCLE.*

Pf. Given a digraph $G = (V, E)$, construct a graph G' with $3n$ nodes.



Directed Hamiltonian Cycles

Claim. G has a Hamiltonian cycle iff G' does.

$\Rightarrow$ Suppose G has a directed Hamilton cycle Γ.

Then G' has an undirected Hamilton cycle (same order).

$\Leftarrow$ Suppose G' has an undirected Hamilton cycle Γ'.

Γ' must visit nodes in G' using one of following two orders:

$$\cdots, B, G, R, B, G, R, B, G, R, B, \cdots$$
$$\cdots, B, R, G, B, R, G, B, R, G, B, \cdots$$

Blue nodes in Γ' make up directed Hamilton cycle Γ in G, or reverse of one. $\square$

Homework

- Read Chapter 8 of the textbook.
- Exercises 3 & 22 in Chapter 8.