Lecture Outline 5
Topics in Graph Algorithms (CSCI5320-22S)

CAI Leizhen
Department of Computer Science and Engineering
The Chinese University of Hong Kong
lcai@cse.cuhk.edu.hk

February 15, 2022

Keyword: NP-completeness.

Three steps in proving NP-completeness of a problem Π:

1. Prove that $\Pi \in NP$ – usually very easy.
2. Choose an NP-complete problem Π' – choosing the right problem Π' is not that easy but experience helps a lot.
3. Construct a polynomial reduction from Π' to Π – key step whose difficulty varies greatly.

1 Basic NP-complete problems

Problems frequently used to prove NP-completeness of many graph problems.

CLIQUE
INSTANCE: Graph G and positive integer k.
QUESTION: Does G contain a clique of size at least k?

VERTEX COVER
INSTANCE: Graph G and positive integer k.
QUESTION: Does G contain a vertex cover of size at most k?

DOMINATING SET
INSTANCE: Graph G and positive integer k.
QUESTION: Does G contain a dominating set of size at most k?

HAMILTONIAN CYCLE
INSTANCE: Graph G.
QUESTION: Does G contain a Hamiltonian cycle, i.e., a cycle with n vertices?

3-SATISFIABILITY (3SAT)
INSTANCE: Set U of Boolean variables, collection C of clauses over U with $|c| = 3$ for each $c \in C$.
QUESTION: Is there a truth assignment for U that satisfies all clauses in C?

Two related NP-complete problems.

INDEPENDENT SET: Does the input graph G contain an independent set of size at least k?

HAMILTONIAN PATH: Does the input graph G contain a Hamiltonian path, i.e., a path on n vertices?

2 Easy proofs

An easy way to prove the NP-hardness of problem Π is by *restriction*: show that Π contains a known NP-complete problem Π' as a special case. This is done by showing that all instances of Π' can be obtained from instances of Π by setting input parameters of Π properly.

1. DENSE INDUCED SUBGRAPH

   INSTANCE: Graph G, positive integers k and l.

   QUESTION: Does G contain an induced subgraph on k vertices that has at least l edges?

   (Proof: Set $l = \binom{k}{2}$ and we get CLIQUE.)

2. EDGE PACKING

   INSTANCE: Graph G and positive integer l and k.

   QUESTION: Does G contain l edges that are incident with at most k vertices?

   (Proof: Set $l = \binom{k}{2}$ and we get CLIQUE.)

3. HITTING RECTANGLES

   INSTANCE: Set of rectangles R and set P of points on the plane, and positive integer k.

   QUESTION: Are there k points from P that hit every rectangles in R?

   (Proof: VERTEX COVER is a special case. Draw graph G on the plane such that an edge uv of G corresponding to a straight line l_{uv} connecting points u and v (make sure that l_{uv} intersects no other vertices). Take points corresponding to vertices as P, and for each edge uv, regard l_{uv} as a rectangle.)

4. EULERIAN SUBGRAPH

   INSTANCE: Graph G and positive integer k.

   QUESTION: Does G contain an Eulerian subgraph with exactly k edges?

   (Proof: For cubic graphs, the problem for $k = n$ is equivalent to HAMILTONIAN CYCLE, which is NP-complete on cubic graphs.)

3 Not-hard proofs

To choose a proper NP-complete problem Π' for a reduction to our target problem Π, we can consider an NP-complete problem that is similar to Π. Often we can use *local replacement*: identify some basic units of Π', and replace each basic unit uniformly with a "gadget" to get an instance of Π.

1. FEEDBACK VERTEX SET

   INSTANCE: Graph G and positive integer k.

   QUESTION: Can we make G acyclic by removing at most k vertices?

   Connection with VERTEX COVER: A vertex cover uses vertices to cover edges, and a feedback vertex set uses vertices to cover cycles.

   Reduction from VERTEX COVER by replacing each edge of G by a triangle, i.e., for each edge e of G, add a new vertex v_e and two edges connecting v_e with the two ends of e.

2. ONE-SIDED DOMINATING SET

   INSTANCE: Bipartite graph $G = (X, Y; E)$ and positive integer k.
   QUESTION: Are there $\leq k$ vertices $X' \subseteq X$ that dominate all vertices in Y, i.e., every vertex in Y is adjacent to some vertex in X'?

   Proof. Reduction from VERTEX COVER. For graph $G = (V, E)$, construct bipartite graph $G = (X, Y; E')$ with $X = V$ and $Y = E$ such that $xy \in E'$ where $x \in X$ and $y \in Y$ iff vertex x is incident with edge y.

3. MULTICUT

   INSTANCE: Graph G, collection C of pairs of vertices, and positive integer k.
   QUESTION: Does G contain at more k edges E' such that no component of $G - E$ contains a pair of C?

   Connection between VERTEX COVER and MULTICUT on stars $K_{1.n}$: To disconnect a pair u, v in a star T with center vertex x, we need to remove either edge ux or edge $vx \iff$ To cover an edge ab, we use either vertex a or vertex b.

   Reduction from VERTEX COVER $(G, k) \to (T, C, k)$: Construct a star T with center x and vertices of G as leaves, and for each edge uv of G add pair $\{u, v\}$ to C.

4 Not-easy proofs

1. DUAL SEPARATOR FOR TWO TERMINALS (Cai and Ye 2014)

   INSTANCE: Edge-bicolored graph G, two vertices s and t of G, and positive integer k.

   QUESTION: Can we remove $\leq k$ vertices from G to disconnect s and t in both blue and red graphs?

   VERTEX COVER remains NP-complete on cubic graphs, and we give a reduction from this special case of VERTEX COVER. Given a cubic graph $G = (V, E)$, we construct an edge-bicolored graph G' from G as follows:

(a) Partition edges of G into two bipartite graphs $G_r = (X_r, Y_r; E_r)$ and $G_b = (X_b, Y_b; E_b)$, colour all edges of G_r red and all edges of G_b blue.

(b) Introduce two new vertices s and t as terminals.

(c) For G_r, connect s with every vertex in X_r by a red edge and connect t with every vertex in Y_r by a red edge.

(d) Similarly for G_b, connect s with every vertex in X_b by a blue edge and connect t with every vertex in Y_b by a blue edge.

(e) Turn the above multigraph into a simple graph by subdividing each blue edge incident with s or t.

2. Vertex 3-Colorability

   Instance: Graph $G = (V, E)$.

   Question: Is there a coloring $f : V \to \{1, 2, 3\}$ so that for every edge uv, $f(u) \neq f(v)$?

   Reduction from 3SAT.

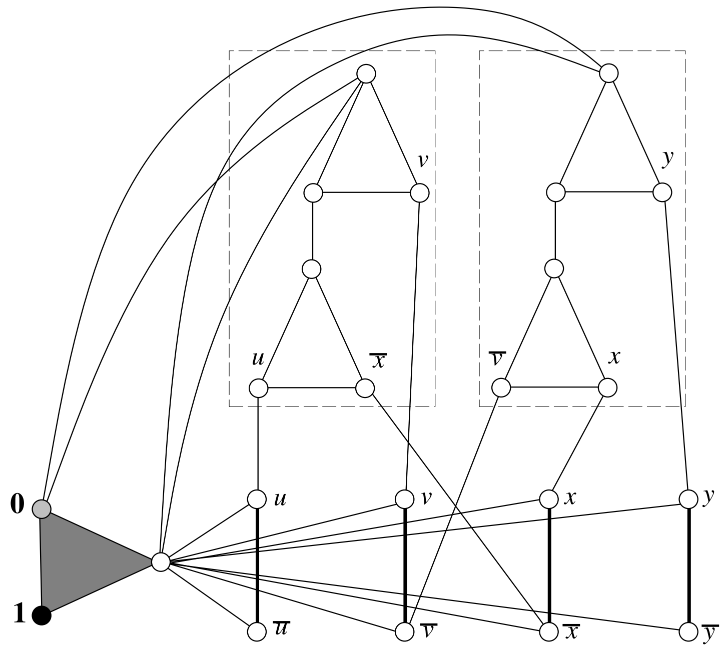

Figure 1: An example of reduction from 3SAT to Vertex 3-Colorability for clauses $\{\{u, \overline{x}, v\}, \{\overline{v}, x, y\}\}$, where thick edges indicate truth-setting components, dashed rectangles are satisfaction testing components, and the shaded triangle sets the truth value of a vertex to **1** if its color is identical to the black vertex in the triangle.

3. Path Avoiding Forbidden Pairs

   Instance: Graph G, collection F of pairs of vertices, and two vertices s and t.

   Question: Is there an (s,t)-path in G that contains at most one vertex from each vertex pair in F?

   Reduction from 3SAT, which is illustrated in Figure 2.

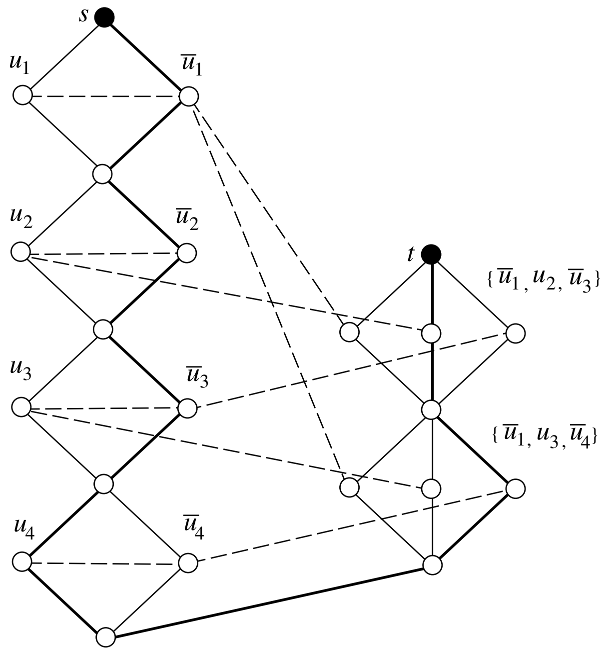

Figure 2: An example of reduction from 3SAT to Path Avoiding Forbidden Pairs for clauses $\{\{\overline{u}_1, u_2, \overline{u}_3\}, \{\overline{u}_1, u_3, \overline{u}_4\}\}$, where dashed lines indicate forbidden pairs and thick edges indicate a legal (s,t)-path.