CSE 421
Algorithms
Richard Anderson
Lecture 27
NP-Completeness Proofs
A problem X is NP-complete if
- X is in NP
- For every Y in NP, \( Y \leq_p X \)
Cook’s Theorem

• The Circuit Satisfiability Problem is NP-Complete

• Proof ideas
  – Let A be an arbitrary problem in NP
  – Show that an instance x of A can be transformed in polynomial time into an instance y of Circuit SAT, such that x is a true instance of A iff y is a satisfiable circuit
  – $A \prec_P \text{Circuit SAT}$
Populating the NP-Completeness Universe

- Circuit SAT $\leq_P$ 3-SAT
- 3-SAT $\leq_P$ Independent Set
- 3-SAT $\leq_P$ Vertex Cover
- Independent Set $\leq_P$ Clique
- 3-SAT $\leq_P$ Hamiltonian Circuit
- Hamiltonian Circuit $\leq_P$ Traveling Salesman
- 3-SAT $\leq_P$ Integer Linear Programming
- 3-SAT $\leq_P$ Graph Coloring
- 3-SAT $\leq_P$ Subset Sum
- Subset Sum $\leq_P$ Scheduling with Release times and deadlines
Satisfiability

Literal: A Boolean variable or its negation. \( x_i \) or \( \overline{x}_i \)

Clause: A disjunction of literals. \( C_j = x_1 \lor \overline{x}_2 \lor x_3 \)

Conjunctive normal form: A propositional formula \( \Phi \) that is the conjunction of clauses.
\[ \Phi = C_1 \land C_2 \land C_3 \land C_4 \]

SAT: Given CNF formula \( \Phi \), does it have a satisfying truth assignment?

3-SAT: SAT where each clause contains exactly 3 literals.

Ex: \( (\overline{x}_1 \lor x_2 \lor x_3) \land (x_1 \lor \overline{x}_2 \lor x_3) \land (x_2 \lor x_3) \land (\overline{x}_1 \lor \overline{x}_2 \lor \overline{x}_3) \)

Yes: \( x_1 = \text{true}, x_2 = \text{true}, x_3 = \text{false} \).
3-SAT is NP-Complete

Theorem. 3-SAT is NP-complete.

Pf. Suffices to show that CIRCUIT-SAT ≤_P 3-SAT since 3-SAT is in NP.

– Let K be any circuit.
– Create a 3-SAT variable \( x_i \) for each circuit element \( i \).
– Make circuit compute correct values at each node:
  • \( x_2 = \neg x_3 \) \( \Rightarrow \) add 2 clauses: \( x_2 \lor \overline{x}_3, \overline{x}_2 \lor \overline{x}_3 \)
  • \( x_1 = x_4 \lor x_5 \) \( \Rightarrow \) add 3 clauses: \( x_1 \lor x_4, x_1 \lor x_5, x_1 \lor x_4 \lor x_5 \)
  • \( x_0 = x_1 \land x_2 \) \( \Rightarrow \) add 3 clauses: \( \overline{x}_0 \lor x_1, \overline{x}_0 \lor x_2, x_0 \lor \overline{x}_1 \lor \overline{x}_2 \)

– Hard-coded input values and output value.
  • \( x_5 = 0 \) \( \Rightarrow \) add 1 clause: \( \overline{x}_5 \)
  • \( x_0 = 1 \) \( \Rightarrow \) add 1 clause: \( x_0 \)

– Final step: turn clauses of length < 3 into clauses of length exactly 3.
Proving a problem A is NP Complete

• Show A is in NP (usually easy)
• Choose an NP complete problem B
  – Convert an instance of B into an equivalent instance of A
3 Satisfiability Reduces to Independent Set

Claim. $3$-SAT $\leq_p$ INDEPENDENT-SET.

Pf. Given an instance $\Phi$ of $3$-SAT, we construct an instance $(G, k)$ of INDEPENDENT-SET that has an independent set of size $k$ iff $\Phi$ is satisfiable.

Construction.

- $G$ contains 3 vertices for each clause, one for each literal.
- Connect 3 literals in a clause in a triangle.
- Connect literal to each of its negations.

$$\Phi = \overline{x_1} \lor x_2 \lor x_3 \land (x_1 \lor \overline{x_2} \lor x_3) \land (\overline{x_1} \lor x_2 \lor x_4)$$
3 Satisfiability Reduces to Independent Set

Claim. G contains independent set of size \( k = |\Phi| \) iff \( \Phi \) is satisfiable.

Pf. \( \Rightarrow \) Let \( S \) be independent set of size \( k \).
   - \( S \) must contain exactly one vertex in each triangle.
   - Set these literals to true. \( \leftarrow \) and any other variables in a consistent way
   - Truth assignment is consistent and all clauses are satisfied.

Pf \( \Leftarrow \) Given satisfying assignment, select one true literal from each triangle. This is an independent set of size \( k \). □

\[
\Phi = \left( \bar{x}_1 \lor x_2 \lor x_3 \right) \land \left( x_1 \lor \bar{x}_2 \lor x_3 \right) \land \left( \bar{x}_1 \lor x_2 \lor x_4 \right)
\]

\( k = 3 \)
Analysis of 3-SAT to IS reduction

• Clause satisfaction
  – Only one literal per clause can be selected, so to get $k$ literals, a literal from every clause must be selected
Analysis of 3-SAT to IS reduction

• Truth setting
  – $X$ is true if at least one $X$ is in the independent set
  – $X$ is false if at least one $\overline{X}$ is in the independent set

• Truth consistency
  – Edges between all copies of $X$ and $\overline{X}$ ensure variables are true or false
IS $\leq_P$ VC

- Lemma: A set $S$ is independent iff $V - S$ is a vertex cover

- To reduce IS to VC, we show that we can determine if a graph has an independent set of size $K$ by testing for a Vertex cover of size $n - K$
IS $\leq^p$ VC

Find a maximum independent set $S$

Show that $V - S$ is a vertex cover
Clique

- Clique
  - Graph $G = (V, E)$, a subset $S$ of the vertices is a clique if there is an edge between every pair of vertices in $S$
Complement of a Graph

- Defn: $G'=(V,E')$ is the complement of $G=(V,E)$ if $(u,v)$ is in $E'$ iff $(u,v)$ is not in $E$. 

![Diagram of a graph and its complement]
IS $\leq_p$ Clique

- Lemma: S is Independent in G iff S is a Clique in the complement of G

- To reduce IS to Clique, we compute the complement of the graph. The complement has a clique of size K iff the original graph has an independent set of size K.
Hamiltonian Circuit Problem

- Hamiltonian Circuit – a simple cycle including all the vertices of the graph
Thm: Hamiltonian Circuit is NP Complete

• Reduction from 3-SAT
Hamiltonian Path

• Is there a simple path that visits all the vertices?
• Is there a simple path from s to t that visits all the vertices?
Reduce HC to HP

$G_2$ has a Hamiltonian Path iff $G_1$ has a Hamiltonian Circuit
Traveling Salesman Problem

• Given a complete graph with edge weights, determine the shortest tour that includes all of the vertices (visit each vertex exactly once, and get back to the starting point)

Find the minimum cost tour
Thm: \( HC \preceq_p TSP \)
Graph Coloring

• NP-Complete
  – Graph K-coloring
  – Graph 3-coloring

• Polynomial
  – Graph 2-Coloring