Foundations of Computing

Lab 10 – $\mathcal{NP}$-Completeness

April 9, 2025

Outline

Verifiability

A verifier for a language L is an algorithm V where

$$L = \{x \mid V \text{ accepts } (x, w) \text{ for some string } w\}$$

- Runtime of V is measured as a function of $|x|$
- V is a polynomial time verifier if it runs in time $poly(|x|)$
- L is polynomial time verifiable if it has a polynomial time verifier
- Strng w is called a witness that $x \in L$

The Class $\mathcal{NP}$

Definition

$\mathcal{NP}$ is the class of languages that have polynomial-time verifiers.

How to show A is $\mathcal{NP}$-complete

1. Show that $A \in \mathcal{NP}$ – provide a poly-time verification algorithm for A
2. Reduce some other $\mathcal{NP}$-complete problem to A (e.g., $SAT \leq_P A$)

Important

Make sure you understand why this is the correct direction for the reduction.

Outline

The SAT Problem

Satisfiability Problem

$$SAT = \{\langle \phi \rangle \mid \phi \text{ is a satisfiable Boolean formula}\}$$

Example: $\phi = (\overline{x} \wedge y) \vee (x \wedge \overline{z})$

Can you think of a formula that is not satisfiable?

We proved in class that SAT is $\mathcal{NP}$-complete

The 3-SAT Problem

- Recall that SAT asks if a Boolean formula has a satisfying assignment
- 3-SAT asks the same question for 3-CNF formulas

3-CNF formulas

- A literal is a (possibly negated) Boolean variable – x or $\overline{x}$
- A clause is several literals connected with $\vee$'s – $x_1 \vee \overline{x_2} \vee x_3$
- A Boolean formula is in conjunctive normal form (CNF) if it consists of clauses connected by $\wedge$'s

$$(x_1 \vee \overline{x_2} \vee x_3 \vee x_4) \wedge (\overline{x_3} \vee x_5)$$

- A Boolean formula is a 3-CNF if all the clauses have exactly 3 literals

$$(x_1 \vee \overline{x_2} \vee x_3) \wedge (\overline{x_3} \vee x_4 \vee x_5) \wedge (\overline{x_1} \vee x_4 \vee x_2)$$

3-SAT

$$\text{3-SAT} = \{\langle \phi \rangle \mid \phi \text{ is a satisfiable 3-CNF formula}\}$$

Can show that 3-SAT is $\mathcal{NP}$-complete using similar proof to SAT

Outline

Recall The Clique Problem

Clique

A clique in and undirected graph is a subset of nodes s.t. every two nodes are connected by an edge. A k-clique is a clique containing k nodes

$$CLIQUE = \{\langle G, k\rangle \mid G \text{ is an undirected graph with a } k\text{-clique}\}$$

Goal:

Prove that CLIQUE is $\mathcal{NP}$-complete

1. CLIQUE $\in \mathcal{NP}$
2. 3SAT $\leq_P$ CLIQUE

3SAT $\leq_P$ CLIQUE – We saw this in class

Need to show reduction f from 3SAT formula ϕ to $\langle G, k \rangle$ where

3SAT $\leq_P$ CLIQUE – We saw this in class

Need to show reduction f from 3SAT formula ϕ to $\langle G, k \rangle$ where

- If ϕ is satisfiable, G has a clique of size k

3SAT $\leq_P$ CLIQUE – We saw this in class

Need to show reduction f from 3SAT formula ϕ to $\langle G, k\rangle$ where

- If ϕ is satisfiable, G has a clique of size k
- If ϕ is not satisfiable, G has no clique of size k

3SAT $\leq_P$ CLIQUE – We saw this in class

Need to show reduction f from 3SAT formula ϕ to $\langle G, k \rangle$ where

- If ϕ is satisfiable, G has a clique of size k
- If ϕ is not satisfiable, G has no clique of size k

Reduction Procedure f

1. Create a vertex for each term (i.e., x_1 or $\overline{x_1}$) in each clause – label the vertex with the variable
2. Draw an edge between two vertices if
   - They are in different clauses
   - They can be set to 1 at the same time (i.e., they are not contradictory)

- Exercise 1: Show the graph resulting from the formula $\phi = (x_1 \vee x_1 \vee x_2) \wedge (\overline{x_1} \vee \overline{x_2} \vee \overline{x_2}) \wedge (\overline{x_1} \vee x_2 \vee x_2)$. Is this equation satisfiable?
- Exercise 2: Show the graph resulting from the formula $\phi = (x_1 \vee \overline{x_1} \vee x_2) \wedge (\overline{x_3} \vee x_1 \vee x_2) \wedge (x_3 \vee x_1 \vee \overline{x_2})$. Is this satisfiable?

Proving This Works

- Case 1: ϕ is satisfiable:
  - This means that there is an assignment x that satisfies all clauses at once (assume ϕ has k clauses)
  - Since ϕ is a CNF, this means that at least one term in each clause is set to 1 by x
  - This means that these terms are non-contradictory and so, would all have edges between them in $f(\phi)$
  - Then, these nodes form a k-CLIQUE so $(G, k) \in CLIQUE$
- Case 2: ϕ is not satisfiable
  - There is no assignment x that satisfies all clauses at once
  - For every assignment, at least one clause must have all its terms set to 0
  - Since there are no edges between vertices in same clause, this means that largest clique has at most $k - 1$ nodes
  - $(G, k) \notin CLIQUE$

Independent Set

Given an undirected graph $G = (V, E)$, an independent set of G is a set of vertices $S \subseteq V$ such that no two members of S are connected by an edge. That is, if $u, v \in S$, then $(u, v) \notin E$.

Independent Set Problem

Given a graph G and an integer k, does G have an independent set of size $\geq k$?

$$L_{IS} = \{(G, k) \mid G \text{ has an independent set of size } \geq k\}$$

Exercise 3: Prove that L_{IS} is $\mathcal{NP}$-Complete

Independent Set

Given an undirected graph $G = (V, E)$, an independent set of G is a set of vertices $S \subseteq V$ such that no two members of S are connected by an edge. That is, if $u, v \in S$, then $(u, v) \notin E$.

Independent Set Problem

Given a graph G and an integer k, does G have an independent set of size $\geq k$?

$$L_{IS} = \{(G, k) \mid G \text{ has an independent set of size } \geq k\}$$

Exercise 3: Prove that L_{IS} is $\mathcal{NP}$-Complete

Hints:

1. Remember that you have to prove two things

Independent Set

Given an undirected graph $G = (V, E)$, an independent set of G is a set of vertices $S \subseteq V$ such that no two members of S are connected by an edge. That is, if $u, v \in S$, then $(u, v) \notin E$.

Independent Set Problem

Given a graph G and an integer k, does G have an independent set of size $\geq k$?

$$L_{IS} = \{(G, k) \mid G \text{ has an independent set of size } \geq k\}$$

Exercise 3: Prove that L_{IS} is $\mathcal{NP}$-Complete

Hints:

1. Remember that you have to prove two things
2. What is the relationship between CLIQUE and IS?