Independent Set (I.S.) is NP-Complete

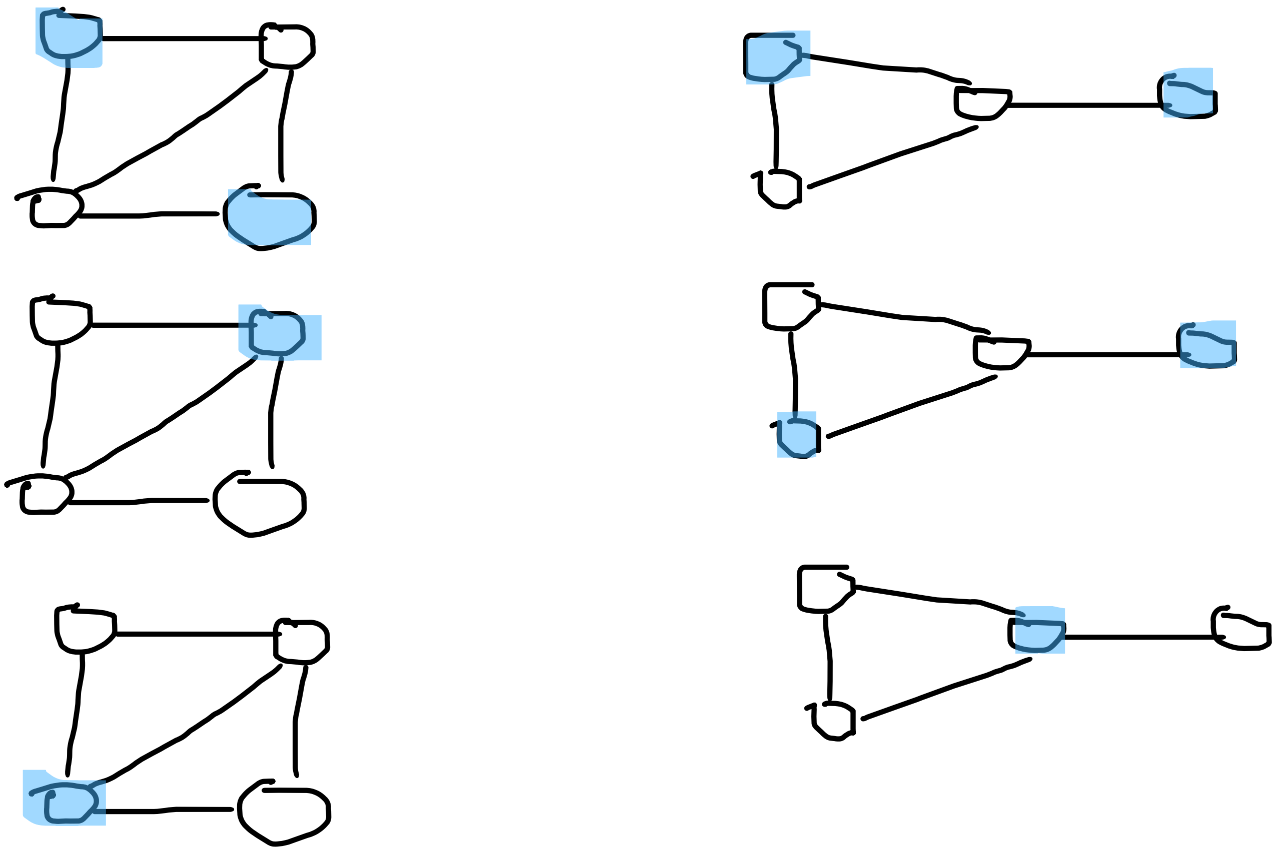
Dianne Foreback

Topics

- Independent Set
  - Definition
  - Decision Problem
  - Verification Problem
- 3-SAT and 3-CNF
- Proving Independent Set Problem is NP-Complete
  - Will use 3-SAT to Independent Set Reduction in proof

Independent Set

- independent set of a graph G is a subset S of vertices such that there is no edge between vertices in S



Each drawing shows a graph G with an independent set highlighted

Independent Set Problems

Independent Set Decision Problem: Given a graph G, does G contain an independent set with k vertices?

Independent Set Verification Problem: Given a graph G, an integer k, and a set of vertices I, is I an independent set of G with exactly k vertices?

Goal: prove an unknown problem is in NP-Complete

- To prove an unknown (decision) problem U is in NP-Complete must:
  1. Prove U = NP
     - state U as a verification problem and provide an algorithm that verifies it in polynomial time)
  2. Prove it is in NP-Hard
     - given a problem H that is assumed to be in NP-complete, provide a reduction from H to U in polynomial time. Written $H \leq_P U$
       - ❑ the reduction is an "if and only if" (must prove in both directions)
       - ❑ A problem H is polynomial-time reducible to problem U, if any algorithm that can solve U can be used to solve H such that the increase in time is no more than polynomial

Part 1: Independent Set Verification Solvable in Polynomial Time

Independent Set Verification Problem: Given a graph G, an integer k, and a set of vertices I, is I an independent set of G with exactly k vertices?

Proof Sketch: Independent set means vertices in the set do not share an incident edge.

- For each vertex $v \in I$, verify that it is not adjacent to any other vertex $u \in I$. This can be done in polynomial time by examining vertex $v's$ adjacency list…explain this.
- Count the number of vertices in I and make sure it is equal to k.
- If both of the above hold, then I is an independent set of G otherwise it is not.

Part 2: Independent Set is in NP-Hard

- We'll do this by showing a problem in NP-Complete. "Known" problem, the 3-SAT problem is reducible to the "unknown" problem Independent Set

$$H \leq_P U$$
$$3SAT \leq_P I.S.$$

Boolean Formula

- Boolean Formula Example: $x_1 \wedge (x_2 \vee (\overline{x_2} \wedge x_3)) \wedge x_4$
- Parts of Boolean Formula
  - Disjunction symbol (or): $\vee$
  - Conjunction symbol (and): $\wedge$
  - Boolean variables: x_1, x_2, x_3
  - Literals: a Boolean variable or its negation. $\overline{x_2}, x_2$ where $\overline{x_2}$ is the negation of x_2
  - clause: a disjunction of Boolean literals, or the Boolean literal itself.
    3 examples

    $x_1 \vee x_2 \vee \overline{x_3}$ $\quad$ $\overline{x_1}$ $\quad$ $x_1 \vee \overline{x_3}$

Formula in 3-CNF

- Boolean formula in 3-Conjunctive Normal Form (3-CNF) : a conjunction of clauses with each clause having exactly 3 literals. 2 examples follow

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

- Boolean formula is satisfiable if there is a truth assignment to the Boolean variables that make the formula true

example 1: $x_1 \vee \overline{x_2} \vee \overline{x_3}$
assigning $x_1 = true$ is a satisfying assignment
assigning $x_1 = false, x_2 = x_3 = true$ is not satisfying

example 2: $(x_1 \vee x_2 \vee \overline{x_3}) \wedge (\overline{x_1} \vee \overline{x_2} \vee x_3)$
assigning $x_1 = x_2 = x_3 = true$ is a satisfying assignment
assigning $x_1 = x_2 = true, \ \ x_3 = false$ is not satisfying

3-SAT (decision) Problem

- 3-SAT Decision Problem: Given a Boolean formula in 3-CNF, does it have a satisfying assignment?
- Known to be in NP-Complete

Famous Theorem by Samuel Cook: 3-SAT is NP-Complete

- Why we can use reductions – Cook (and Levin) proved independently as the first problem to be in NP-Complete
- Reductions are transitive (so can reduce one problem in NP-Complete to unknown problem and state transitively all other problems reducible to it

Transitivity of Reducibility

- If A $\rightarrow$ B and B $\rightarrow$ C, then A $\rightarrow$ C.
  - An input x for A can be converted to x' for B, such that x is in A if and only if x' is in B. Likewise, for B to C.
  - Convert x' into x'' for C such that x' is in B iff x'' is in C.
  - Hence, if x is in A, x' is in B, and x'' is in C.
  - Likewise, if x'' is in C, x' is in B, and x is in A.

Theorem: Independent Set is in NP-Complete

Theorem: Independent Set (decision) Problem is in NP-Complete

1. Independent Set is in NP – verification problem is solvable in polynomial time ✓
2. Independent Set is in NP-Hard. Show/prove

   $3SAT \leq_P I.S.$

Independent Set is in NP-Hard

Claim:

If 3-SAT is not solvable in polynomial time, then Independent Set is not solvable in polynomial time.

Proof by Contraction: Assume there is a polynomial time algorithm A to decide Independent Set, which, given a graph G, an integer k, returns whether G has an independent set of size k.

…Explain the algorithm that is to be used for the transformation (next slides)

… then prove the reduction (in both directions)

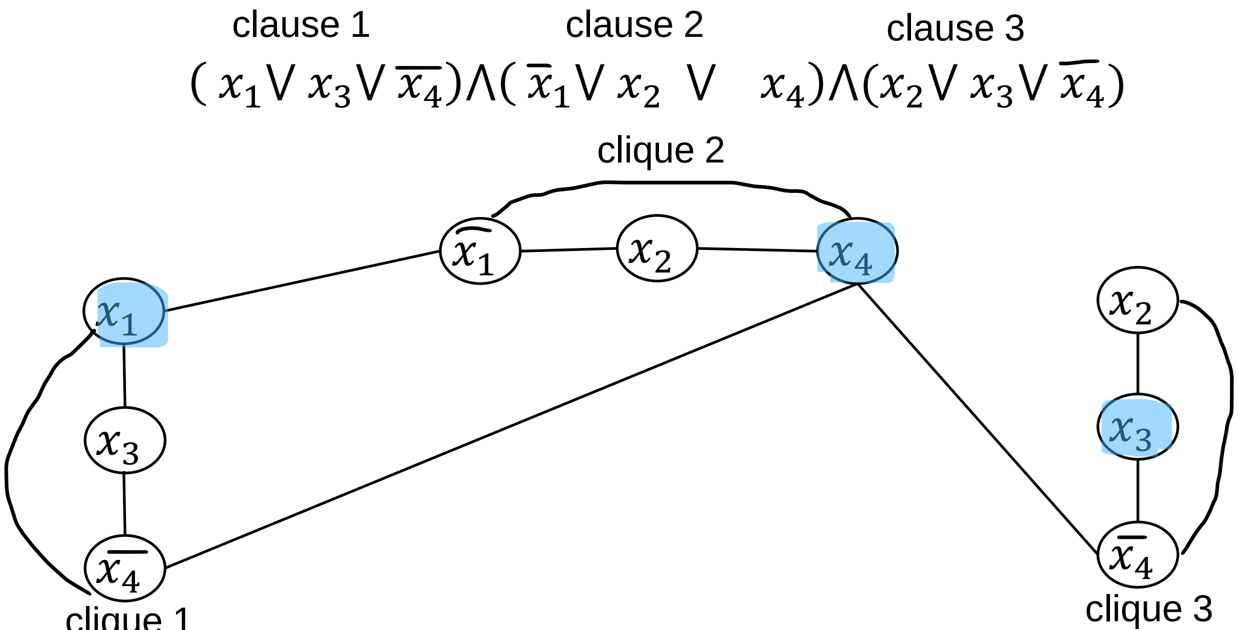
Compare Independent Set and NP-Hard

Algo for 3-SAT
Input: a Boolean formula in 3-CNF
Output: true if the formula is satisfiable, false otherwise

…Explain the algorithm that is to be used for the transformation (next slides)
… then prove the reduction (in both directions)

Explaining the Transformation Algorithm: Transforming 3-CNF formula from 3-SAT problem to an I.S.

- for each clause, form a clique with the literals
- For each clique, if a vertex that represents a literal has its negation represented in another clique, place an edge between them (since assigning true to one negatives the other in 3-CNF, thus, means there is a conflict)



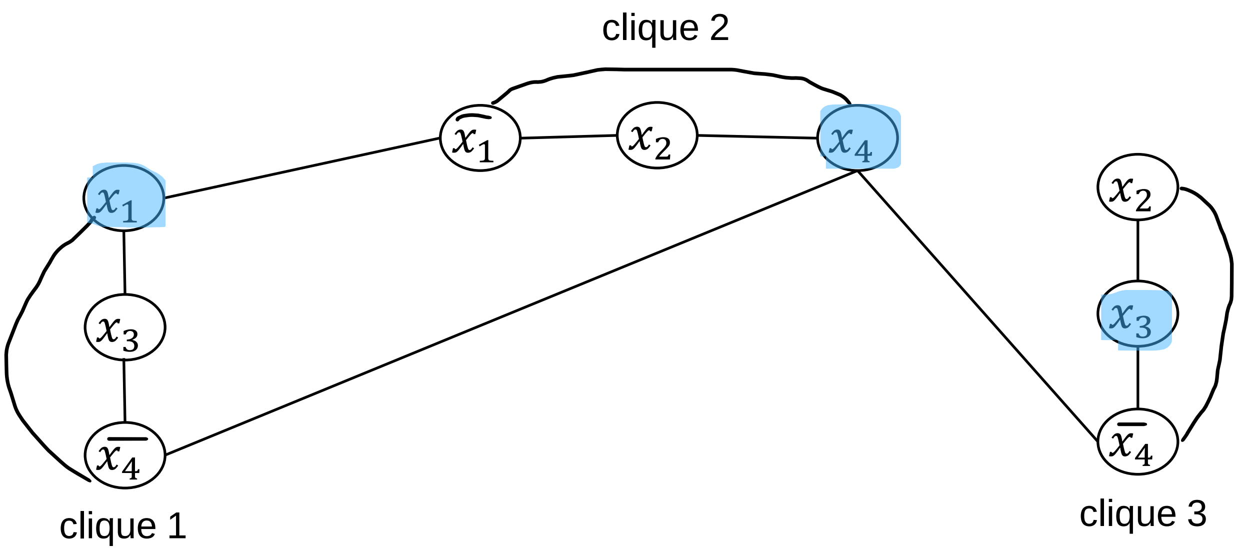
Questions	3-SAT	Independent Set
What is selected?	Boolean variables that are true	vertices that are in the IS
What are the requirements?	each clause is true	vertices in IS I cannot be adjacent
What are the constraints?	if x_1 = true then $\overline{x_1}$ = false	$\mid I \mid$ = k (size of IS is k)

Idea: Use 3-SAT Graph to form I.S.

- Independent Set – cannot choose adjacent vertices to be in it

clause 1 clause 2 clause 3

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



Independent Set can be formulated as follows:

- choose x_1 from clique 1, then cannot pick $x_3 \, or \, \overline{x_4}$ in clique 1, or x_1 in clique 2
- choose x_4 from clique 2, then cannot pick $\overline{x_1} \, or \, x_2$ in clique 2, or $\overline{x_4}$ in cliques 1 or 3
- choose x_3 from clique 3, then cannot pick $x_2 \, or \, \overline{x_4}$ in clique 3

Aside: notice, assigning $x_1 = x_4 = x_3 = true$ satisfies the 3-CNF formula

Theorem: Independent Set is in NP-Complete

Theorem: Independent Set (decision) Problem is in NP-Complete

1. Independent Set is in NP – verification problem is solvable in polynomial time ✓
2. Independent Set is in NP-Hard. Show/prove
   $3SAT \leq_P I.S.$

Recall: Independent Set is in NP-Hard

Claim:

If 3-SAT is not solvable in polynomial time, then Independent Set is not solvable in polynomial time.

Proof by Contraction: Assume there is a polynomial time algorithm A to decide Independent Set, which, given a graph G, an integer k, returns whether G has an independent set of size k.

…Explain the algorithm that is to be used for the transformation (next slides)✓

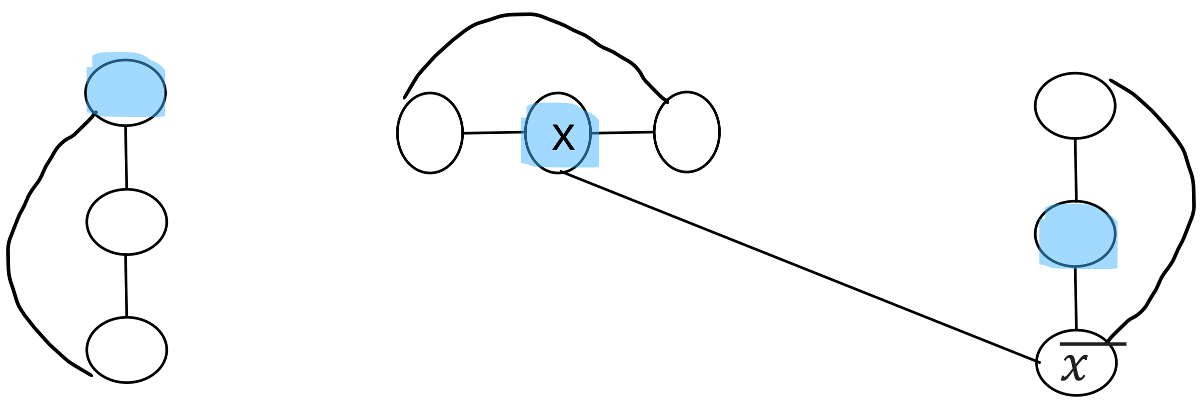
… then prove the reduction (in both directions)

Prove $3SAT \le_P I.S.$ (→)

(→) If the 3-CNF Boolean formula is satisfiable, then the graph has an independent set I of size | I | = k

Proof:

- If the formula is satisfiable, there is at least one variable per clause that is true. Pick one vertex per clique corresponding to one of the $true$ literals in the clause to put in the independent set. Since only one vertex is chosen from the clique, there is no edge within the same clique.
- Also, there is no edge across cliques because in a valid truth assignment, if Boolean variable x is chosen and set to true, its complement will not be chosen in another clique. (aside: you can only assign one value to a variable)



Prove $3SAT \leq_P I.S.$ (→)

(→) If the 3-CNF Boolean formula is satisfiable, then the graph has an independent set I of size | I | = k

Proof:

- If the formula is satisfiable, pick one vertex per clique corresponding to one of the $true$ literals in the clause. Since only one vertex is chosen from the clique, there is no edge within the same clique.
- Also, there is no edge across cliques because in a valid truth assignment, if Boolean variable x is chosen and set to true, it will not be also be set false or vice versa. (aside: you can only assign one value to a variable)

Aside:

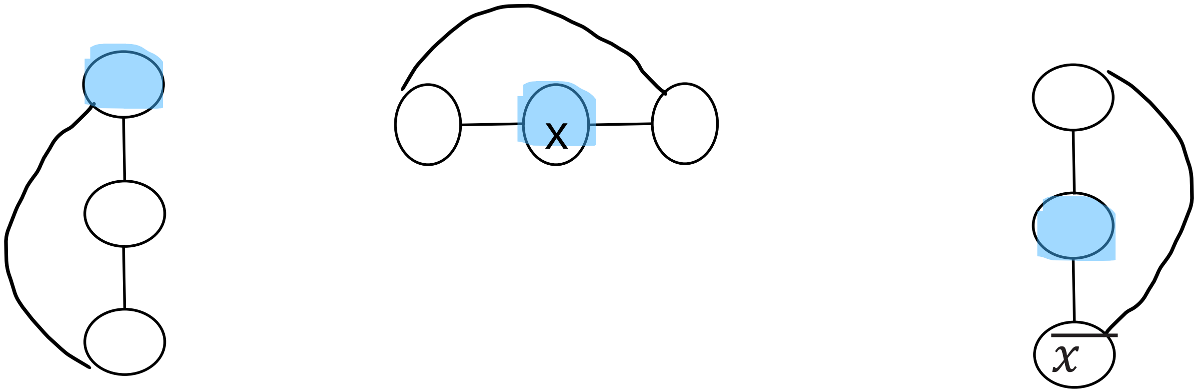
- This tells us that if the output of algorithm A returns true, the formula is satisfiable (→).
- BUT, if the formula is NOT satisfiable, algorithm A could return true or false! We have not shown that algorithm A correctly returns false if the formula is not satisfiable. So, we must prove the converse direction (←)

Prove $3SAT \le_P I.S.$ ($\leftarrow$)

($\leftarrow$) If a Graph has an independent set of size k, then the formula is satisfiable.

Proof:

- Independent Set I has at most 1 vertex per clique since the clique has incident edges for all pairs of vertices in the clique.
- Since $|I| = k, I$ has exactly one vertex per clique.
- Take the corresponding literal that is represented by this vertex to be true.
- The Boolean formula is satisfied because there is one true literal per clause.
- The truth assignment is consistent because if a literal is chosen, its compliment is not chosen.



Recall: Independent Set is in NP-Hard

Claim:

If 3-SAT is not solvable in polynomial time, then Independent Set is not solvable in polynomial time.

Proof by Contraction: Assume there is a polynomial time algorithm A to decide Independent Set, which, given a graph G, an integer k, returns whether G has an independent set of size k.

…Explain the algorithm that is to be used for the transformation (next slides) ✓

… then prove the reduction (in both directions) ✓

Theorem: Independent Set is in NP-Complete

Theorem: Independent Set (decision) Problem is in NP-Complete

1. Independent Set is in NP – verification problem is solvable in polynomial time ✓
2. Independent Set is in NP-Hard. Show/prove
   $3SAT \leq_P I.S.$ ✓

Thank You !

Questions ?