CS 7800: Advanced Algorithms

Class 15: More Intractability

- NP-Completeness
- More hardness: knapsack, hamiltonian path

Jonathan Ullman

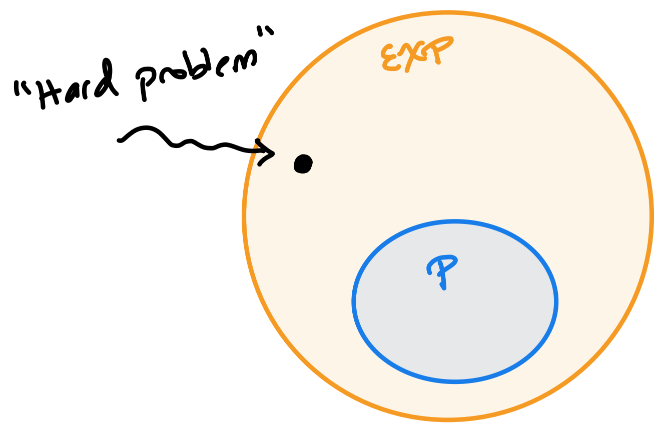
October 24, 2025

Tractable and Intractable Problems

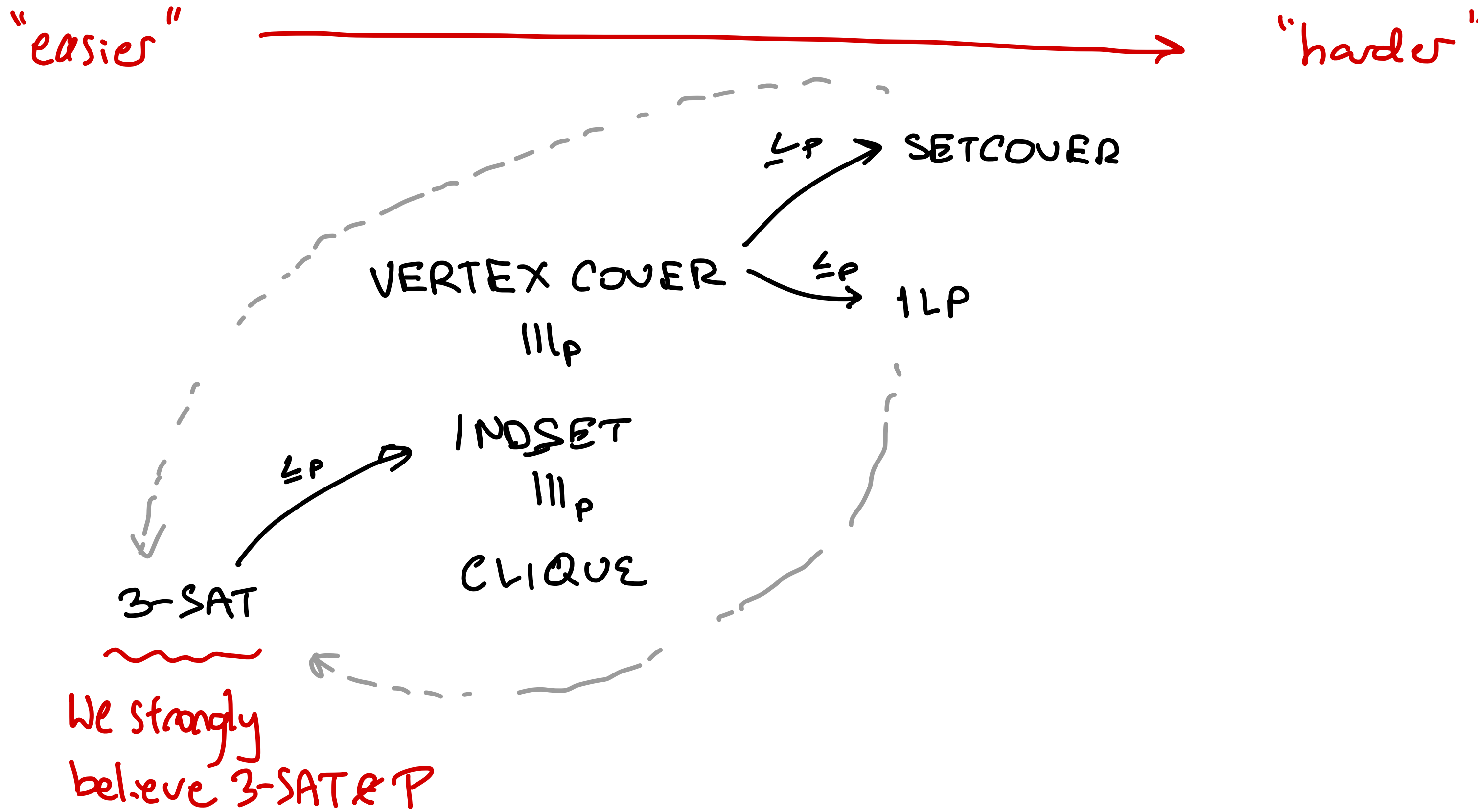
- **Definition:** $\mathcal{P}$ is the set of decision problems that can be solved in polynomial time

problems with a yes/no answer

- **Definition**: $\mathcal{EXP}$ is the set of decision problems that can be solved in exponential time
- **Theorem:** $\mathcal{P} \neq \mathcal{EXP}$

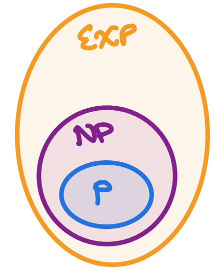


Allegedly Intractable Problems



Note: Reductions are transitive

The Class NP



- **Definition:** $\mathcal{NP}$ is the class of problems for which there is an efficient verifier for solutions
  - An algorithm V is an efficient verifier for problem A if
    - (1) V takes as input I and a solution S
    - (2) V is a polynomial-time algorithm
    - (3) $I \in A$ if and only if there exists a polynomial-size solution S such that $V(I, S) = \text{YES}$

If answer on input I is YES

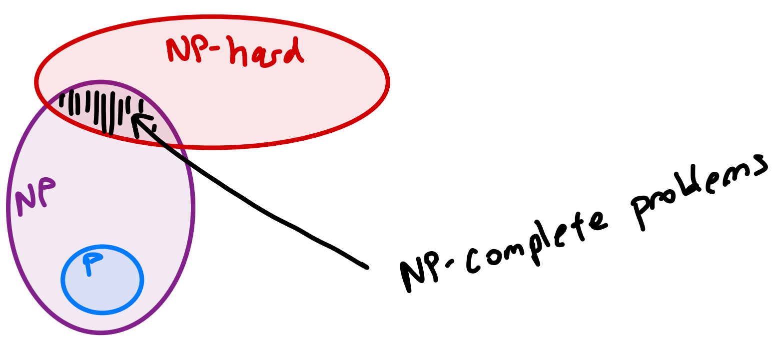
- $\mathcal{P}$ = easy to solve, $\mathcal{NP}$ = easy to check solution

- Natural hard optimization problems are in $\mathcal{N}P$
  - 3-SAT, Vertex-Cover, Independent-Set…

Does $\mathcal{P} = \mathcal{NP}$?

- We do not know, but we believe ~~it~~ $\mathcal{P} \neq \mathcal{NP}$ very strongly!
  - One of the Millenium Problems

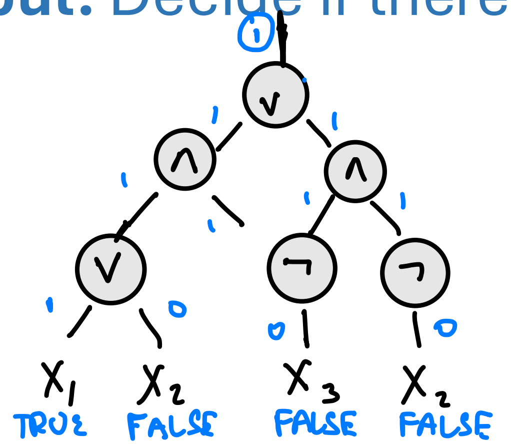
- If we believe $\mathcal{P} \neq \mathcal{NP}$ what does that tell us about problems we care about?
  - **Def:** B is $\mathcal{NP}$-hard if for $A \in \mathcal{NP}$, $A \leq_P B$
  - **Def:** B is $\mathcal{NP}$-complete if $B \in \mathcal{NP}$ and B is $\mathcal{NP}$-hard
  - If B is $\mathcal{NP}$-hard and $B \in \mathcal{P}$ then $\mathcal{P} = \mathcal{NP}$

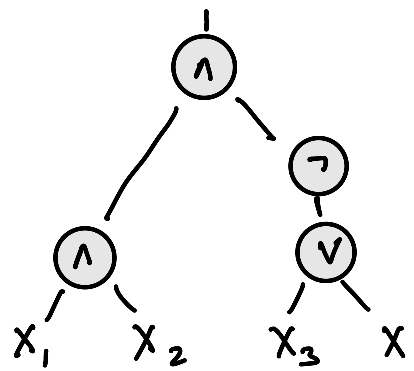


What problems are $\mathcal{NP}$-complete?

- The Circuit Satisfiability Problem (CKT-SAT)
  - **Input:** Circuit C with n wires and AND/OR/NOT gates
  - **Output:** Decide if there exists x such that $C(x) = 1$

10 wires, 6 gates
satisfiable





8 wires, 4 gates
not satisfiable

- **Thm:** CIRCUIT-SAT is $\mathcal{NP}$-complete

Cook '71, Levin '73 Part 1

$A \in NP$ with verifier V $\qquad V(I, \cdot)$

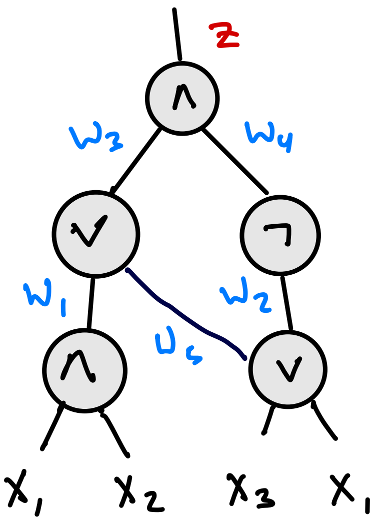
I be an input

What problems are $\mathcal{NP}$-complete?

($\Rightarrow$ 3SAT is NPC)

- **Thm (Cook '71, Levin '73):** CKT-SAT $\leq_P$ 3-SAT

CKT-SAT $\leq_P$ 3-SAT



Given a circuit with m wires and n variables, decide if there exists x such that $C(x) = 1$

Variables: $\underbrace{x_1, \dots, x_m}_{\text{inputs}}, \underbrace{w_1, \dots, w_n}_{\text{wires}}, \underbrace{z}_{\text{output}}$

Idea: Does there exist $x_1 \dots x_m, w_1, \dots, w_n, z$ s.t.

① $z = 1$ ② $w_1 \dots w_n, z$ are the correct values for the wires on input $x_1 \dots x_m$

Gadget for each of the three gates

AND $w = a \wedge b \longrightarrow (w \vee \bar{a} \vee \bar{b}) \wedge (\bar{w} \vee a) \wedge (\bar{w} \vee b)$

OR $w = a \vee b \longrightarrow (\bar{w} \vee a \vee b) \wedge (w \vee \bar{a}) \wedge (w \vee \bar{b})$

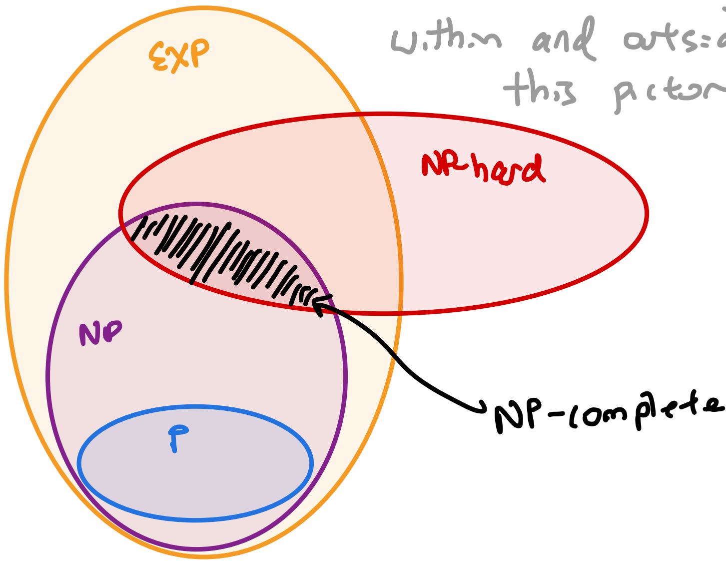
NOT $w = \neg a \longrightarrow (\bar{w} \vee \bar{a}) \wedge (w \vee a)$

What problems are $\mathcal{NP}$-complete?

($\Rightarrow$ 3-SAT is NPC)

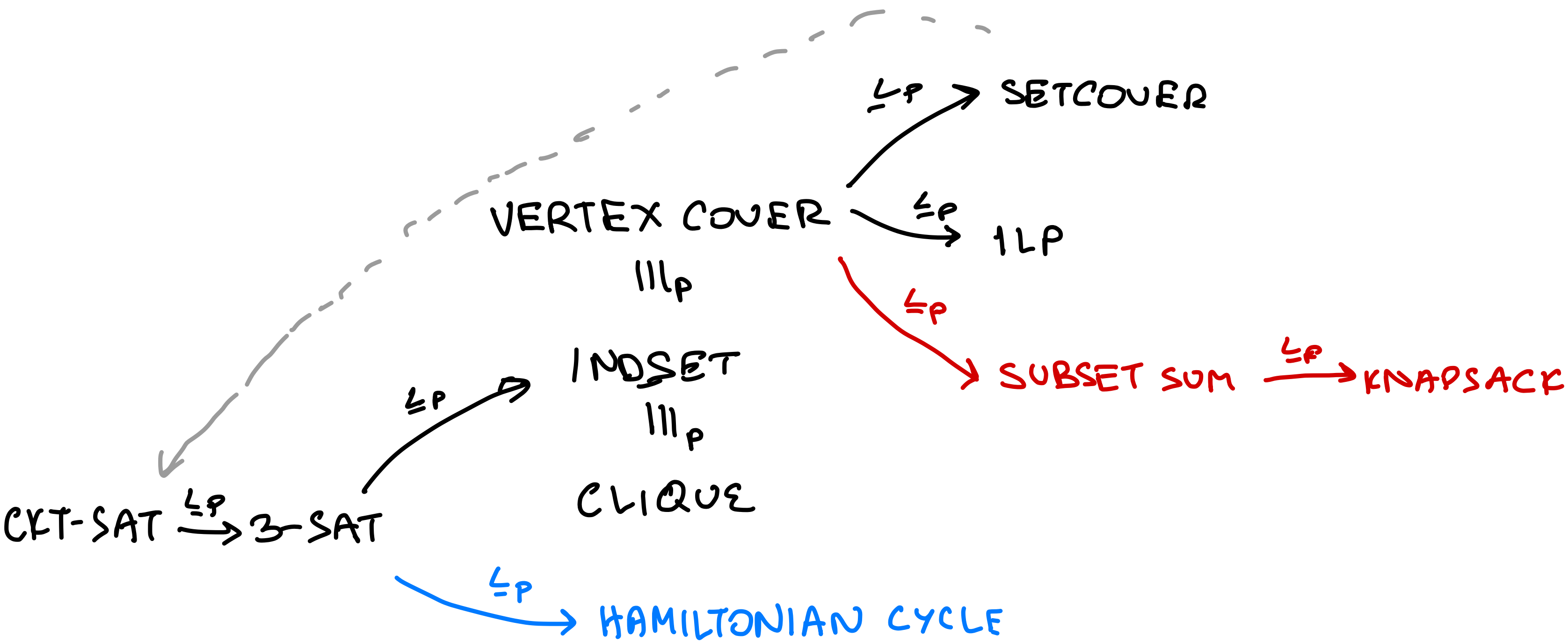
- **Thm (Cook '71, Levin '73):** CKT-SAT $\leq_P$ 3-SAT
  - Now we know IND-SET, CLIQUE, VERTEX-COVER, SET-COVER, IP, and 3-SAT are all $\mathcal{NP}$-complete
  - There are thousands more known $\mathcal{NP}$-complete problems in essentially every area within CS

Lots of interesting classes within and outside of this picture.



NP- Complete Problems

~~Allegedly Intractable Problems~~



SUBSET-SUM / KNAPSACK

SUBSET-SUM:

Input: integers $z_1 \dots z_n \geq 0$

target $T \geq 0$

Output: decide if there exists $S \subseteq \{1, \dots, n\}$ such that $T = \sum_{i \in S} z_i$

- Special case of KNAPSACK

$\Longrightarrow$ Can solve in time $\underbrace{O(n2^n)}_{\text{brute force}}$ or time $\underbrace{O(nT)}_{\text{dynamic programming}}$

Not a P-time algorithm

- Is SUBSET-SUM $\in P$?

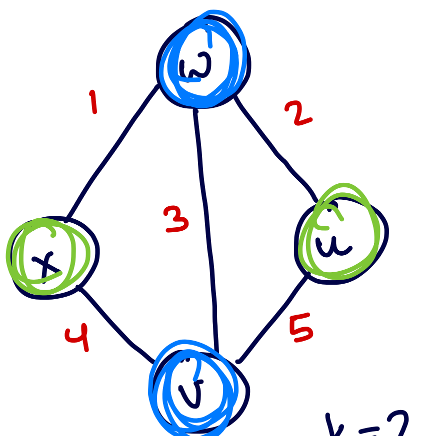
# of bits is $(n+1)\log T$

VERTEX COVER $\leq_p$ SUBSET SUM

VERTEX COVER $\leq_p$ SUBSET SUM

Graph $G=(V,E)$

Number k



$k=2$

Does G have a vertex cover of size exactly k

Want: A set of numbers $z_1, \ldots, z_\ell$ and T such that there is a subset summing to T iff there is a vertex cover

MSD ⟵ Digits ⟶ MSD

	size	xw	wu	uv	xv	wv
a_u	1	0	1	0	0	1
→ a_v	1	0	0	1	1	1
→ a_w	1	1	1	1	0	0
a_x	1	1	0	0	1	0
→ b_{xw}	0	1	0	0	0	0
→ b_{wu}	0	0	1	0	0	0
b_{uv}	0	0	0	1	0	0
→ b_{xv}	0	0	0	0	1	0
→ b_{wv}	0	0	0	0	0	1

211011

211211

$\underbrace{3}\underbrace{12212}$

blue → #s

$= \underbrace{2}_{\text{size}}22222$

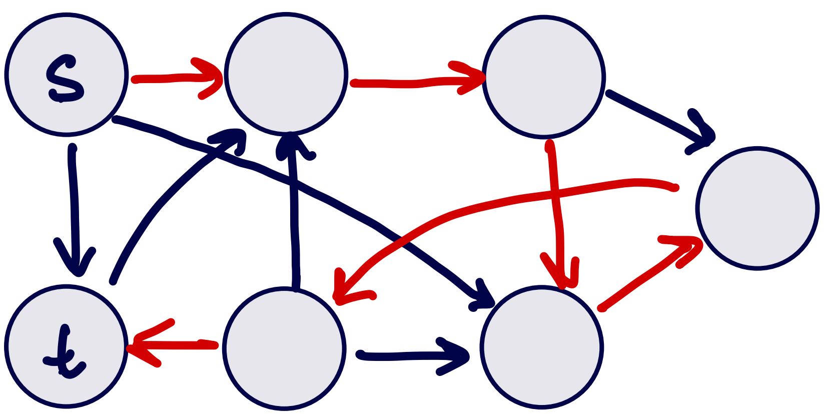
set $T = 222222$

HAMILTONIAN PATH

HAMP:

Input: A directed graph $G=(V,E)$ and nodes $s,t \in V$

Output: Decide if there is an s-t path that visits every node exactly once

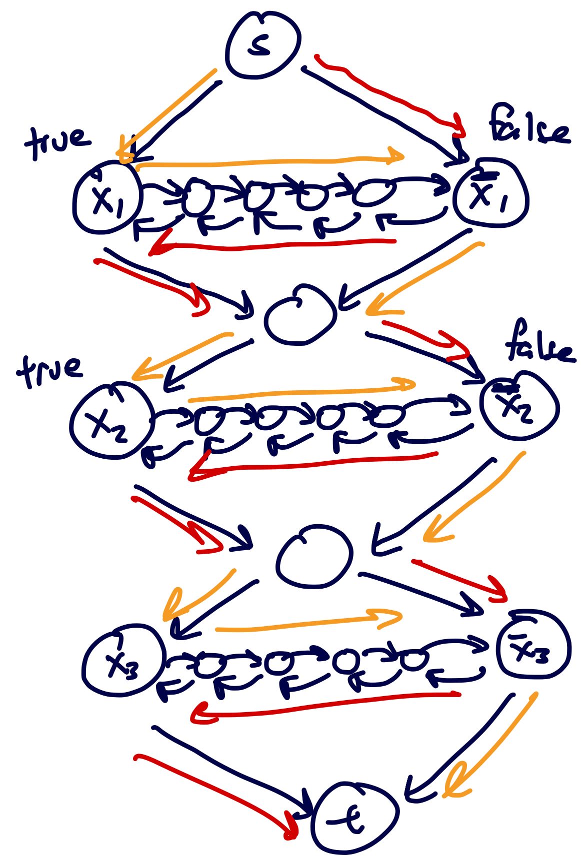


3-SAT $\leq_p$ HAMP

3-SAT $\leq_p$ HAMP

$$\varphi(x) = (x_1 \vee x_2 \vee x_3) \wedge (\overline{x}_1 \vee \overline{x}_2 \vee x_3) \wedge (x_1 \vee \overline{x}_2 \vee \overline{x}_3)$$

Variable gadgets



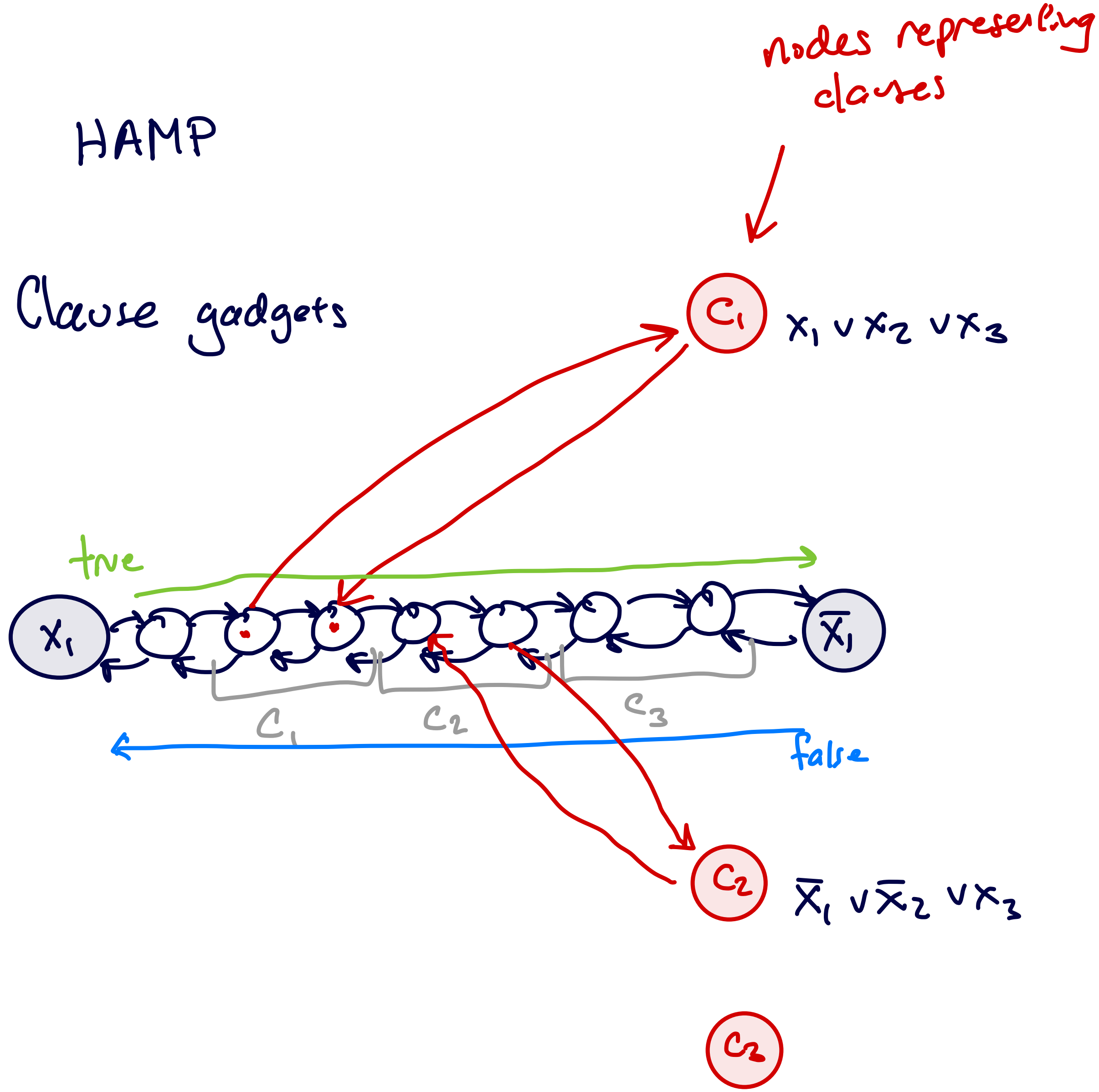
Any path traverses each variable left-to-right (TRUE) or right-to-left (FALSE)

3-SAT $\leq_p$ HAMP

3-SAT $\leq_p$ HAMP

$$\varphi(x) = (x_1 \vee x_2 \vee x_3)^{C_1} \wedge (\overline{x}_1 \vee \overline{x}_2 \vee x_3)^{C_2} \wedge (x_1 \vee \overline{x}_2 \vee \overline{x}_3)^{C_3}$$

Clause gadgets



3-SAT $\leq_p$ HAMP

3-SAT $\leq_p$ HAMP

$$\varphi(x) = (x_1 \vee x_2 \vee x_3)^{C_1}$$
$$\wedge\ (\overline{x}_1 \vee \overline{x}_2 \vee x_3)^{C_2}$$
$$\wedge\ (x_1 \vee \overline{x}_2 \vee \overline{x}_3)^{C_3}$$

Clause gadgets

NP- Complete Problems

~~Allegedly Intractable Problems~~

VERTEX COVER $\xrightarrow{\le_p}$ SETCOVER

VERTEX COVER $\xrightarrow{\le_p}$ ILP

VERTEX COVER $\xrightarrow{\le_p}$ SUBSET SUM $\xrightarrow{\le_p}$ KNAPSACK

VERTEX COVER $\equiv_p$ INDSET $\equiv_p$ CLIQUE

CKT-SAT $\xrightarrow{\le_p}$ 3-SAT $\xrightarrow{\le_p}$ INDSET

3-SAT $\xrightarrow{\le_p}$ HAMILTONIAN CYCLE

What problems are $\mathcal{NP}$-complete?

($\Rightarrow$ 3-SAT is NPC)

- **Thm (Cook '71, Levin '73):** CKT-SAT $\leq_P$ 3-SAT
  - Now we know IND-SET, CLIQUE, VERTEX-COVER, SET-COVER, IP, and 3-SAT are all $\mathcal{NP}$-complete
  - There are thousands more known $\mathcal{NP}$-complete problems in essentially every area within CS

Lots of interesting classes within and outside of this picture.