# NP-completeness

Categorize problems → tractable / → non-tractable

↳ decision-problems

ANSWER = "YES" "NO"

max-flow :→ ∃ f of capacity F?

min-cut → ∃ cut S of size K?

P = { set of problems solvable in polynomial time }

eg:→ max-flow, min-cut, shortest paths etc ..

NP = { set of problems "verifiable" in polynomial time }

∃ polynomial size proof / certificate & a polynomial time-algorithm to verify the solutions to "YES" instances.

CNF-SAT.

# CNF-SAT.

**Given** :→ 1) Set of variables $x_1, x_2, \ldots, x_n$

2) Set of clauses $C_1, C_2, \ldots, C_m$.

$$C_i = (x_{i_1} \vee \neg x_{i_2} \vee x_{i_3} \vee \ldots)$$

$C_i$ is satisfied if

$x_{i_1} = T$, or $x_{i_2} = F$ or ...

**Question** :→ $\exists$ Assignment to the variables such that all clauses are satisfied.

**Non-deterministic Alg**

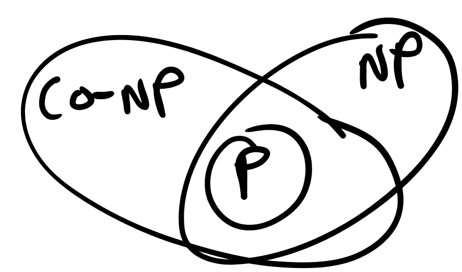
1) Guess

$$\left.\begin{aligned} x_1 &= T \text{ or } F \\ x_2 &= T \text{ or } F \\ &\vdots \\ x_n &= T \text{ or } F \end{aligned}\right\} \text{poly-size proof}$$

Verification Alg { 2) For each clause $C_i$, check whether $C_i$ is satisfied or not

if satisfied o/p YES

if not o/p NO




**P = NP ?**

**Reductions**: Reduction $R$ is a mapping from instances of decision problem $A$ to instances of decision problem $B$ s.t.

$$A(I) = \text{"YES"} \iff B(R(I)) = \text{"YES"}$$

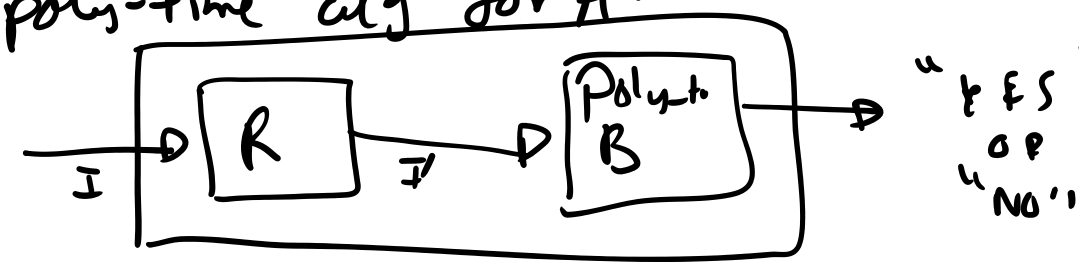
**Poly-time Reductions (Karp Reductions)**

A poly-time ALG that maps any instance $I$ of $A$ to $I'$ of $B$, s.t.

$$A(I) = \text{"YES"} \iff B(I') = \text{"YES"}$$

$$\underline{A \leq_p B}$$

**Claim**:- If $\exists$ poly-time alg for $B$, then $\exists$ poly-time alg for $A$.




$R$ = poly-time alg

$|I'| = \text{Poly}(|I|)$

Running time = Running time $(R)$ + $\text{Running-time}_{ALG_B}(I')$

$= Poly(|I|) + Poly(|I'|)$

$= Poly(|I|) + poly(Poly(A))$

$\in Poly(|I|)$

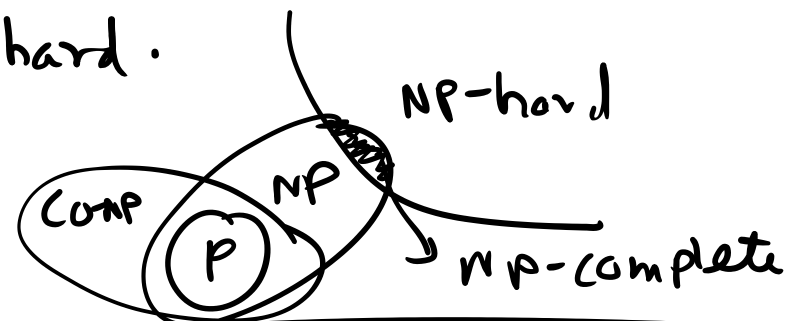
$A \le_p B$

---

**NP hard** :-> problem $\Pi$ is NP-hard if $\forall$ problem $A \in NP$, $A \le_p \Pi$

**NP-complete** :-> problem A is NP-complete

1) $A \in NP$
2) A is NP-hard.




---

**Thm** :-> Cook-Levin Thm

CNF-SAT is NP-complete.

---

$\text{CNF-SAT} \le_p \text{K-CLIQUE} \le_p \text{IS} \le_p \boxed{\text{VC}}$

$\text{CNF-SAT} \le_p$ Hamiltoni. Cycles $\le_p HP \le_p \boxed{TSP}$

---

**K-clique** :->

Given:- 1) Undirected Graph $G=(V,E)$
2) integer $K \in V$.

Question:- $\exists$ clique of size $k$ in $G$.

CNF-SAT $\leq_p$ 3-SAT

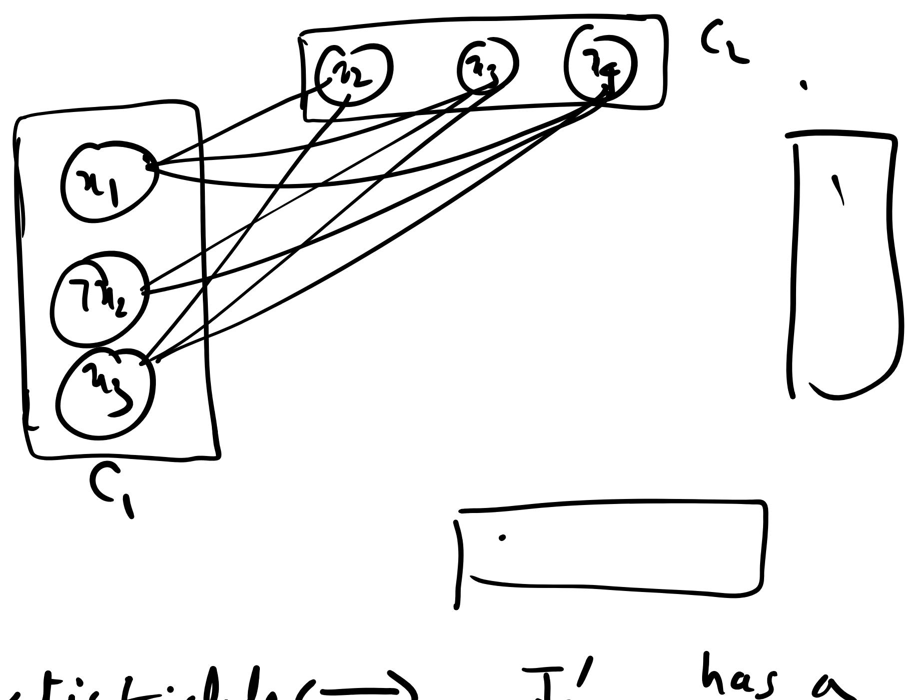
Thm:- k-clique is NP-complete

1) k-clique $\in$ NP (obvious)

2) 3-SAT $\leq_p$ k-clique

We need a poly-time alg that maps every instance I of 3-SAT to I' of k-clique

$C_1 = x_1 \vee \neg x_2 \vee x_3$
$C_2 = x_2 \vee x_3 \vee x_4$
$\vdots$




I is satisfiable $\iff$ I' has a clique of size m.

$\Rightarrow$ I is Satisfiable.
For each clause $C_r$, pick a literal $l_r \in C_r$ that Satisfies $C_r$

$$G[l_1 \cup l_2 \cup \dots \cup l_m, E] = \text{clique}$$

$\Leftarrow$ $I'$ has a clique $C$ of size $m$
$\Rightarrow$ $C$ has one vertex from each Clause set.

$\Rightarrow$ say $l_r$ is chosen from $C_r$ in $C$
Set $l_r$ to appropriate value to satisfy $C_r$

---

**IS**: Given:→ Und. Graph $G=(V,E)$
integer $k \in [|V|]$

Question:→ $\exists$ IS of size $k$
$S \subseteq V$ s.t. $\forall\, u,v \in S$
$(u,v) \notin E$

Thm: IS is NP-complete.
1) IS $\in$ NP (obvious)
2) $k$-clique $\leq_p$ IS

$I$ of $k$-clique $\Longleftrightarrow$ $I'$ of $IS$

$(G, E)$ $\underline{k}$ $\qquad\qquad$ $\bar{G}$, $k$

---

## Hamiltonian-Cycles :-

Given :- 1) Directed Graph $G = (V, E)$

2) $\exists$ cycle $c$ in $G$ that visits each part exactly once

Thm :- HC is NP-complete

1) $HC \in NP$ ✓

2) $3SAT \leq_p HC$

---

$(x_1, x_2, \ldots, x_n)$ = variables

$(c_1, c_2, \ldots, c_m)$ = clauses

Variable gadgets

$C = (x_1 \vee \neg x_2 \vee$

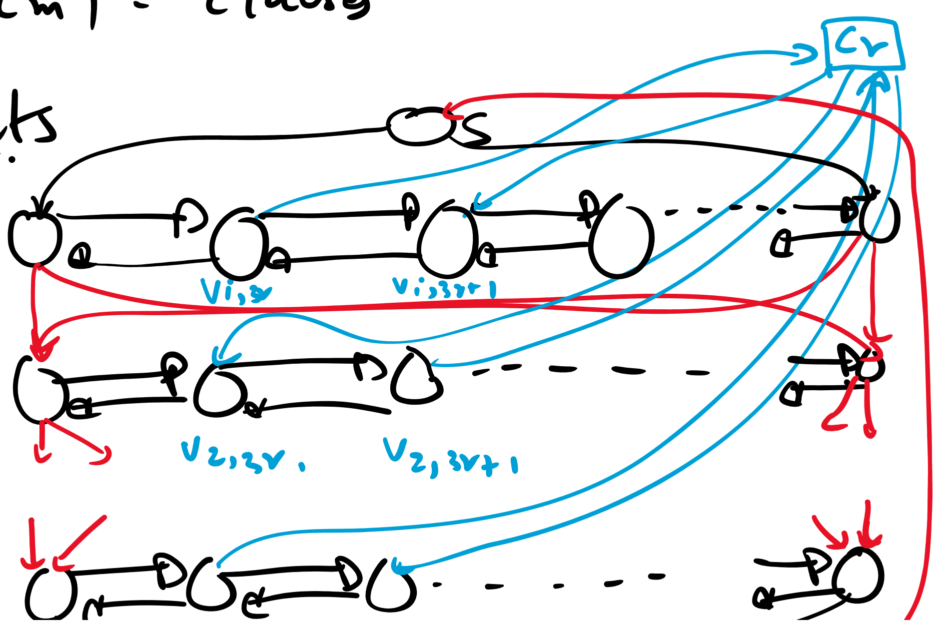
$x_1$

$v_u$

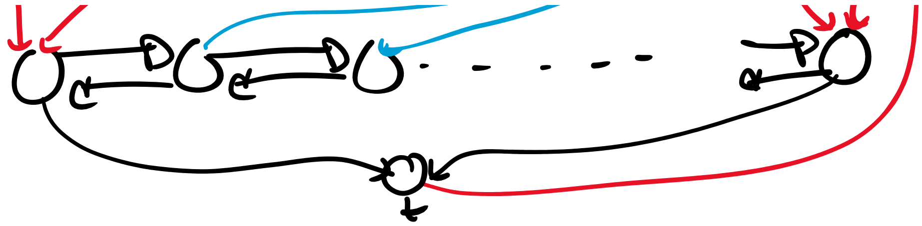
$x_2$

$\vdots$

$x_n$



$\neg x_2$ $\vdots$ $x_n$



## Clause-gadgets

$$\left.\begin{array}{l} v_{i,3r} \to C_r \\ C_r \to v_{i,3r+1} \end{array}\right\} \text{ if } x_i \text{ occurs in } C_r,$$

$$\left.\begin{array}{l} C_r \to v_{i,3r} \\ v_{i,3r+1} \to C_r \end{array}\right\} \text{ if } \neg x_i \text{ occurs in } C_r$$

---

$I$ has a satisfying assignment $\Leftrightarrow$ $I'$ has a HC

$\Rightarrow$ Straightforward.

$\Leftarrow$ Let $C$ be a HC in $I'$

if $C$ enters $C_r$ through $v_{i,3r}$ then it must leave $C_r$ to $v_{i,3r+1}$

4

<u>Symmetry</u>

---