# Sipser 7.4 Polynomial reductions and NP-completeness.

**Definition** Let $A, B$ be languages over $\Sigma$

A polynomial reduction from $A$ to $B$ is a function $f: \Sigma^* \to \Sigma^*$ such that

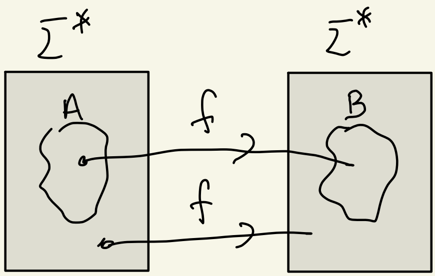
1. $x \in A \iff f(x) \in B$
2. There exist a positive integer $k = k(A,B)$ such that

   $f(x)$ can be calculated in time $O(|x|^k)$

If such a function exists, then we write

$$A \leq_p B$$

very similar to mapping reduction




Important difference: we only have polynomial time to calculate $f$

**Lemma** If $A \le_p B$ and $B \in \mathbf{P}$ then $A \in \mathbf{P}$

P: Suppose that $A_B$ decides $B$ in time $O(n^c)$ and $C_{A\to B}$ computes $f$ s.t. $x \in A \Leftrightarrow f(x) \in B$ in time $O(n^k)$

$n = |x|$

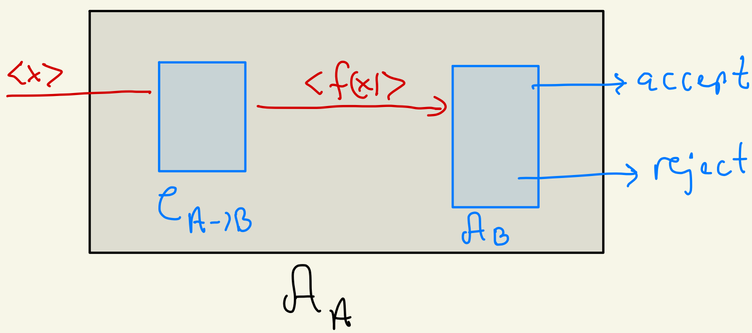
then

$$x \xrightarrow{C_{A\to B}} f(x) \xrightarrow{A_B} \{accept, reject\}$$

$n = |x|$ $\qquad |f(x)| \in O(n^k)$

$A_B$ produces answer in time

$$O(|f(x)|^c) = O((n^k)^c) = O(n^{ck})$$




$A_A$ accepts $x \Leftrightarrow A_B$ accepts $f(x)$

$\Leftrightarrow x \in A$

So $A_A$ decides $A$ in polynomial time

Recall that, via the universal alphabet, we can that all languages in NP are coded over the same alphabet $\Sigma$.

**Definition 7.34** A language $L$ is called NP-complete (written $L \in \mathbf{NPC}$) if

1. $L \in \mathbf{NP}$
2. $\forall L' \in \mathbf{NP}: \; L' \leq_p L$

> **NB!** not clear at all that there are such problems. We prove it in a separate lecture.

**Theorem** If $L \in \mathbf{NPC}$ and $L \leq_p \hat{L}$ then $\hat{L} \in \mathbf{NPC}$

P:

$$
\begin{tikzcd}
\underset{L'}{\bullet} \arrow[r, bend left, "f"', "O(n^r)"] & \underset{L}{\bullet} \arrow[r, bend left, "g"', "O(n^s)"] & \underset{\hat{L}}{\bullet}
\end{tikzcd}
\qquad
\begin{tikzcd}
\underset{x}{\bullet} \arrow[r, bend left, "g \circ f", "O(n^{rs})"'] & g(f(x))
\end{tikzcd}
$$

$x \qquad f(x) \qquad g(f(x)) \qquad x \in L' \iff g(f(x)) \in \hat{L}$

Hence $L' \leq_p \hat{L} \quad \forall L' \in \mathbf{NP}$

A boolean variable $X$ takes two values true and false $(T, F)$ sometimes written as $1$ and $0$

The negation $\bar{X}$ of a boolean variable $X$ is

$$\bar{X} = \begin{cases} \text{true} & \text{if } x = \text{false} \\ \text{false} & \text{if } x = \text{true} \end{cases}$$

A truth assignment to a boolean variable $X$ is an assignment of a value true or false to $X$

## SATISFIABILITY (SAT)

Given boolean variables $x_1, x_2, \dots, x_n$ and clauses $C_1, C_2, \dots, C_m$ over the literals $x_1, \bar{x}_1, x_2, \bar{x}_2, \dots, x_n, \bar{x}_n$ e.g $C_i = (X_{i_1} \vee \bar{X}_{i_2} \vee X_{i_3} \vee \bar{X}_{i_4})$

Question: does there exist a truth assignment $\varphi : \{X_1, X_2, \dots, x_n\} \to \{T, F\}^n$ such that $f = C_1 \wedge C_2 \wedge \dots \wedge C_m$ is true?

## SAT $\in$ **NP**:

certificate is just a truth assignment $\varphi$ s.t each $C_i$ evaluates to true. Given $\varphi$ we can check in time $O(|f|)$ whether $f$ is true under $\varphi$.

## Theorem (Cook-Levin) SAT $\in$ **NPC**

We prove this in a separate lecture.

**3-SAT:** SAT restricted to each clause having exactly 3 literals.

e.g. $f = (x_1 \vee x_2 \vee x_3) \wedge (\bar{x}_1 \vee x_2 \vee x_3) \wedge (x_1 \vee \bar{x}_2 \vee x_3) \wedge (x_1 \vee \bar{x}_2 \vee \bar{x}_3)$

($\varphi : \{x_1, x_2, x_3\} \to \{T, T, T\}$ satisfies $f$)

## Theorem 3-SAT $\in$ **NPC**

Proof
1. 3-SAT $\in$ **NP** is clear certificate = good truth assignment
2. We prove that SAT $\leq_p$ 3-SAT

We show how to transform an instance $f = C_1 \wedge C_2 \wedge \dots \wedge C_m$ of SAT over the variables $x_1, x_2, \dots x_n$ into an instance $f' = C_1' \wedge \dots \wedge C_{m'}'$ of 3-SAT such that $f$ is satisfiable $\iff$ $f'$ is satisfiable

We call $f$ ($f'$) a 'Yes' instance of SAT (3-SAT) if $f$ ($f'$) is satisfiable (has a satisfying truth assignment)

Method replace each clause $C_j$ whose length (no of literals) is $\neq 3$ by several equivalent clauses.

$|C_i| \geq 4$: $\quad C_i = (\lambda_1 \vee \lambda_2 \vee \dots \vee \lambda_k)$ $k \geq 4$ $\lambda_i$ literal over $\{x_1, \dots, x_n, \bar{x}_1, \dots, \bar{x}_n\}$

Introduce new variables $y_1, y_2, \dots, y_{k-3}$ private to this clause $C_i$ and replace $C_i$ in $\varphi$ by

$$X_i = (\lambda_1 \vee \lambda_2 \vee y_1) \wedge (\bar{y}_1 \vee \lambda_3 \vee y_2) \wedge (\bar{y}_2 \vee \lambda_4 \vee y_3) \wedge \dots \wedge (\bar{y}_{k-4} \vee \lambda_{k-2} \vee y_{k-3}) \wedge (\bar{y}_{k-3} \vee \lambda_{k-1} \vee \lambda_k)$$

claim $X_i$ is true $\Leftrightarrow$ at least one of the $\lambda_j$'s is true $j \in [k]$

$|C_i| = 2$: $\quad C_i = (\lambda_1 \vee \lambda_2) \to X_i = (\lambda_1 \vee \lambda_2 \vee z) \wedge (\lambda_1 \vee \lambda_2 \vee \bar{z})$

claim $X_i$ is true $\Leftrightarrow$ at least one of $\lambda_1, \lambda_2$ is true

$$C_i = (\lambda) \to X_i = (\lambda \vee x \vee y) \wedge (\lambda \vee x \vee \bar{y}) \wedge (\lambda \vee \bar{x} \vee y) \wedge (\lambda \vee \bar{x} \vee \bar{y})$$

claim $X_i$ is true $\Leftrightarrow$ $\lambda$ is true

So $\quad \varphi = C_1 \wedge C_2 \wedge \dots \wedge C_m \xrightarrow{f} \varphi' = X_1 \wedge X_2 \wedge \dots \wedge X_m$

satisfies $\quad \phi : \{x_1, \dots, x_n\} \to \{T, F\}^n$ satisfies $\varphi$

$\Updownarrow$ Every extension of $\phi$ to the new variables in $\varphi'$ satisfies $\varphi'$

and we can calculate $\varphi'$ from $\varphi$ in polynomial time measured in $|\varphi|$

**Clique:** Given $\langle G, k\rangle$ where $G$ is a graph and $k \in \mathbb{Z}_+$

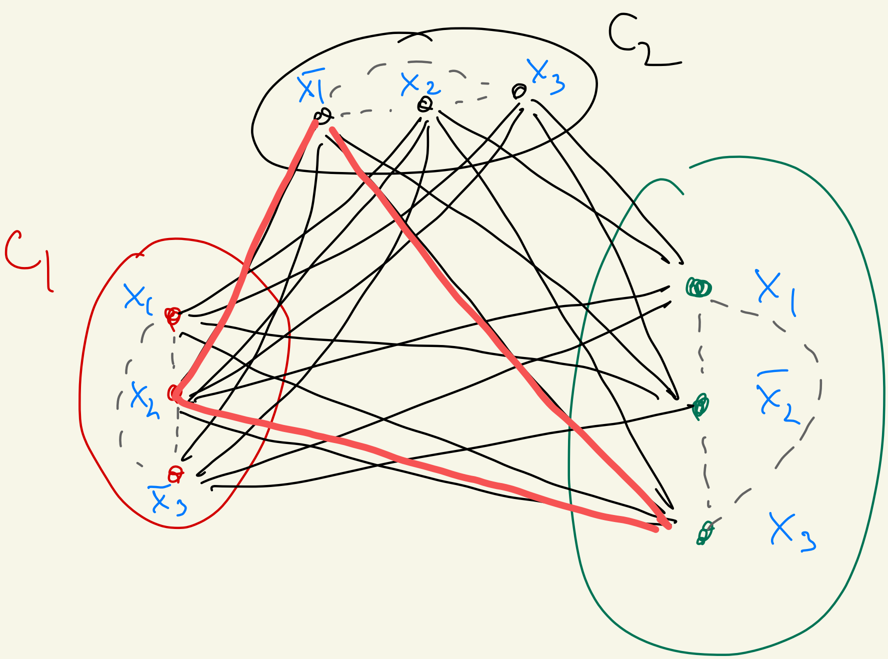
Does $G$ have a $k$-clique?

Clique $\in$ **NP** : certificate is a set of $k$ vertices $v_{i_1}, v_{i_2}, \ldots, v_{i_k}$ of $G$ s.t. $v_{i_q} v_{i_p}$ is an edge of $G$ for all $q, p \in \{1, 2, \ldots, k\}$, $q \neq p$.

**Theorem** 3-SAT $\leq_p$ CLIQUE

**Proof** first by an example to show idea:

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




$x_1 = F$
$x_2 = T$
$x_3 = T$
is a satisfying truth assignment

## General construction

Given an instance $\varphi = C_1 \wedge C_2 \wedge \cdots \wedge C_k$ of 3-SAT where $C_i = (\lambda_{i_1} \vee \lambda_{i_2} \vee \lambda_{i_3})$ with $\lambda_{i_j} \in \{x_1, x_2, \ldots, x_n, \overline{x}_1, \ldots, \overline{x}_n\}$

Construct an instance $\langle G, k \rangle$ of CLIQUE as follows:

$$V(G) = \bigcup_{i=1}^{k} \{v_{i,1}, v_{i,2}, v_{i,3}\}$$

when $v_{i,1}, v_{i,2}, v_{i,3}$ correspond to the literals $\lambda_{i_1}, \lambda_{i_2}, \lambda_{i_3}$ respectively

$$E(G) = \{ v_{i,j} v_{i',j'} \mid i \neq i' \text{ and } \lambda_{i_j} \neq \overline{\lambda_{i'_{j'}}} \}$$

Think of $v_{i,j}$ as being labelled by the literal $\lambda_{i_j}$

(as in the example)

Claim  G has a k-clique $\iff f$ is satisfiable

$\Rightarrow$ Let H be a k-clique in G. Then

- $|H \cap \{v_{i,1}, v_{i,2}, v_{i,3}\}| = 1 \quad \forall i \in \{1, 2, \dots, k\}$
- If a vertex labelled $x_j$ is in H, then no vertex of H is labelled $\bar{x}_j$

set $\varphi(x_i) = \begin{cases} T & \text{if some vertex of } H \text{ is labelled } x_i \\ F & \text{if some vertex of } H \text{ is labelled } \bar{x}_i \\ & \underline{\text{or}} \text{ no vertex of } H \text{ is labelled by } x_i \text{ or } \bar{x}_i \end{cases}$

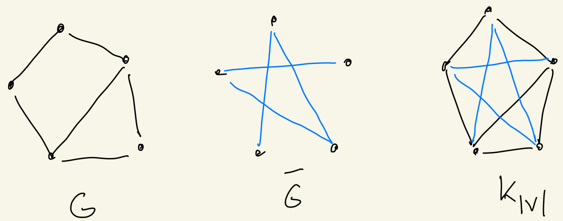
$\varphi$ is a satisfying truth assignment:
we set at least one literal true in each clause $C_i$

$\Leftarrow$ : Suppose $\varphi : \{x_1, x_2, \dots, x_n\} \to \{T, F\}^n$ satisfies $f$

- pick one true literal $\lambda_{i_j}$ in $C_i$ for $i = 1, 2, \dots, k$
- For $i := 1$ to $k$
  put the vertex labelled by $\lambda_{i_j}$ in H
- H is a k-clique

Given $f$ (clauses and variables)
we can construct $\langle G, k \rangle$ in time $O(|f|^2)$

**Definition** The complement of a graph $G=(V,E)$ is the graph $\bar{G}=(V,\bar{E})$ where $uv \in \bar{E} \iff uv \notin E$



**Definition** Let $G=(V,E)$ be a graph. A subset $W \subseteq V$ is independent if no edge $uv \in E$ has $|\{u,v\} \cap W| = 2$ ($\iff u,v \in W$)

**INDEPENDENT SET (IS)**

Given a graph $G=(V,E)$ and $q \in \mathbb{Z}_+$. Does $G$ have an independent set of size $q$?

**Theorem** Independent set is NPC

P: 1. clearly $IS \in \mathbf{NP}$

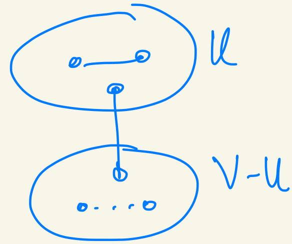
2. $\text{CLIQUE} \le_p IS$:

$X$ is a clique in $G \iff X$ is independent in $\bar{G}$

$\langle G,k\rangle \in \text{CLIQUE} \iff \langle \bar{G},k\rangle \in$ Independent set

polynomial reduction $\square$

**Definition** A vertex cover in a graph $G=(V,E)$ is a subset $U \subseteq V$ s.t. $|\{u,v\} \cap U| \geq 1$ $\forall uv \in E$




$$\boxed{\begin{array}{l}\text{VERTEX-COVER (VC)}\\ \text{Given } G=(V,E) \text{ and } p \in \mathbb{Z}_+\\ \text{Does } G \text{ have a vertex cover of size } p?\end{array}}$$

**Theorem** VERTEX-COVER $\in$ **NPC**

Proof:
- Vertex-cover $\in$ **NP**

  Certificate is a set $U \subseteq V$ s.t. removing $U$ kills all edges

- INDEPENDENT SET $\leq_p$ VERTEX-COVER

  $X$ is independent in $G$

  $\Updownarrow$

  $V \setminus X$ is a vertex cover in $G$

So

$\langle G, q \rangle \in$ INDEPENDENT-SET

$\Updownarrow$

$\langle G, |V(G)|-q \rangle \in$ VERTEX-COVER

$\square$.

Polynomial reductions seen so far:

$$\text{SAT} \le_p \text{3-SAT} \le_p \text{CLIQUE} \le_p \text{INDEPENDENT-SET} \le_p \text{VERTEX-COVER}$$