Announcements

Pset 4 due Tuesday

Pset 5 out

$$P = \bigcup_{k \geq 0} \mathrm{TIME}(n^k)$$ "polynomial time"

$$NP = \bigcup_{k \geq 0} \mathrm{NTIME}(n^k).$$ "nondet. poly time"

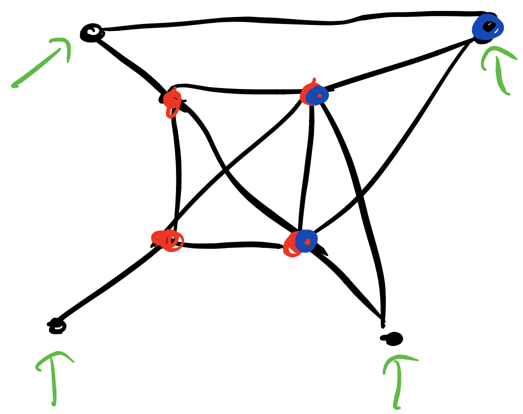
$P \subseteq NP$

Verifier for L is a det. algorithm V where

$L = \{ w \mid \exists$ a string c for which V accepts $\langle w, c \rangle \}$.

"certificate"

A k-clique is a complete subgraph of a graph $G=(V,E)$ with k vertices.



A poly-time verifier V is a verifier that runs in poly time in $\langle w, c \rangle$.

original graph (w) · list of k vertices (c)

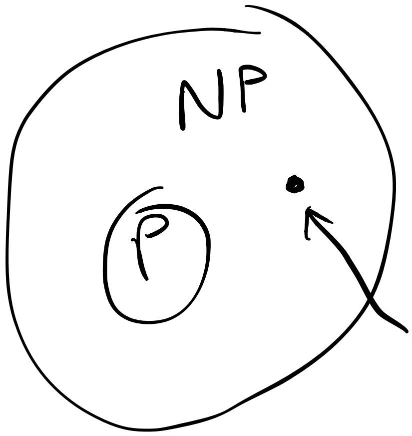
Thm: NP can also be defined in terms of poly time verifiers.

Proof:

If $L \in NP$, build a NTM that

1. Nondet. select a certificate c of poly length.
2. Det. simulate the verifier.

If $L \in NP$ and has a NTM for it, then let the certificate be the ACH of that machine on w.



(assume $P \neq NP$)

A polynomial time reduction from $A \to B$ ($A, B \in NP$) is a function $f: A \to B$ so that

$$w \in A \iff f(w) \in B.$$

We write this as $A \leq_p B$.

Say L is <u>NP-hard</u> if for <u>every</u> $B \in NP$, $B \leq_p L$.

($\Longrightarrow$ if $L \in P$, then $P = NP$).

L is NP-complete if L is NP-hard and $L \in NP$.

CNF Satisfiability

input: boolean variables $x_1, \dots x_n$
boolean operations AND, OR, NOT

vars can appear in

$\left.\begin{array}{l}\text{positive form: } x_i \\ \text{negative form: } \overline{x_i}\end{array}\right\}$ literals

Clause = or/"disjunction" of literals

$$\begin{array}{ccccc} (x_1 \overset{\uparrow}{\vee} & \overline{x_2} \vee & \overline{x_3} \vee & x_4 \vee & x_2) = T \\ T \ \text{"or"} & T & F & F & F \end{array}$$

formula: and/"conjunction" of clauses

$$\underbrace{(x \vee y \vee \overline{z})} \underset{\text{"and"}}{\overset{}{\wedge}} \underbrace{(\overline{x} \vee y)} \wedge \underbrace{(y \vee z \vee \overline{z})}$$

one assignment = $x = T$, $y = T$, $z = F$.

satisfiable if there exists an assignment to the vars to make the formula eval to true.

CNFSAT = $\{\langle \phi \rangle : \phi$ is a boolean CNF formula that is satisfiable$\}$.

CNFSAT $\in$ TIME(2^n).

CNFSAT is NP-complete [Cook, Levin 1971]

CNFSAT $\in$ NP:

the certificate is the assignment to the vars.

Then verify if the formula is satisfied by this assignment.

CNFSAT is NP-hard:

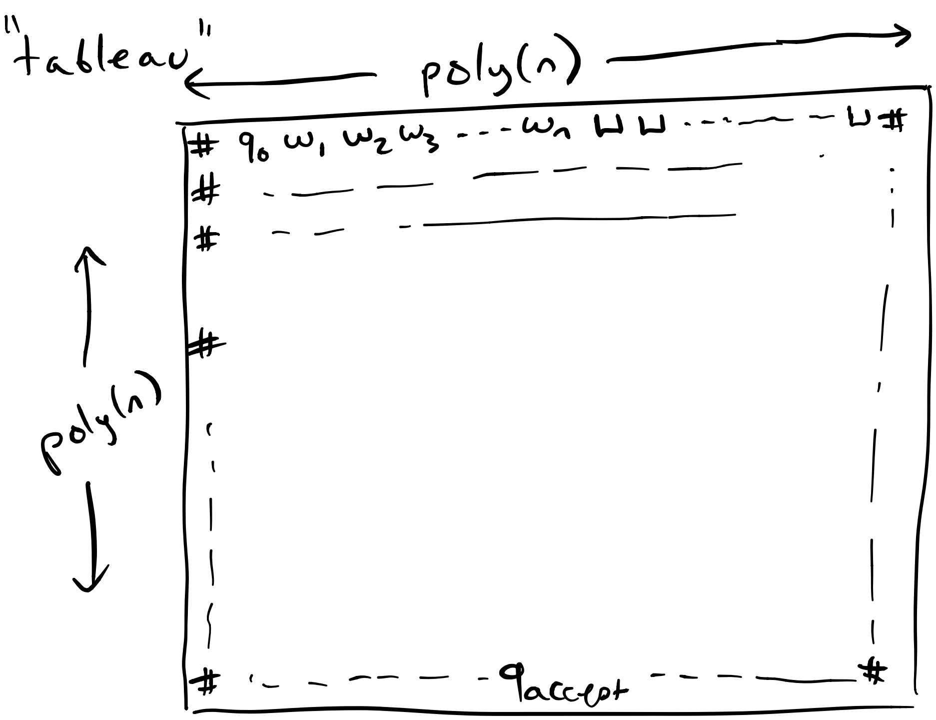
Let B be any language in NP.

$\Rightarrow \exists$ a NTM T for B that

runs in $O(\text{poly}(n))$ time.

Idea: build a formula that is satisfiable

iff $w \in B$.



Want: make a formula based off this that is satisfiable iff T accepts $w_1 - - - w_n$.

1. formula for start config (1st row)
2. formula for accepting config (last row)
3. formula to make sure each row yields the next.

There are s symbols (tape chars, states, & #).

Make a var. $x_{i,j,\ell}$ where i is any row ($1 \le i \le \text{poly}(n)$)

j is any column ($1 \le j \le \text{poly}(n)$)

and ℓ is any of the symbols

Q: could both $x_{i,j,\ell}$ and $x_{i,j,\ell'}$ ($\ell \neq \ell'$) be both true?

$$\phi_{cell} = \bigwedge_{i=1}^{\text{poly}(n)} \bigwedge_{j=1}^{\text{poly}(n)} \left[\bigvee_{\ell=1}^{s} x_{i,j,\ell} \wedge \underbrace{\bigwedge_{\substack{1 \le \sigma_1, \sigma_2 \le s \\ \sigma_1 \neq \sigma_2}} \left(\overline{x_{i,j,\sigma_1}} \vee \overline{x_{i,j,\sigma_2}} \right)} \right]$$

q_0 w_1 - - -

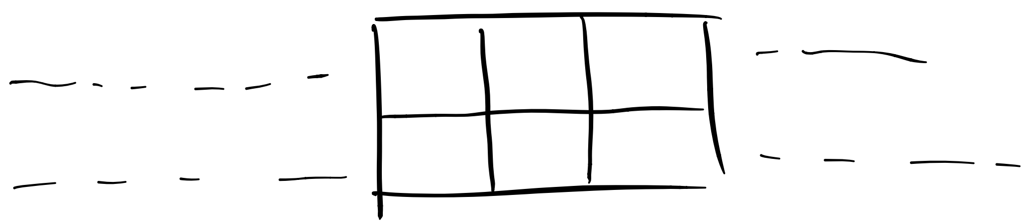
$\underline{(\overline{x_{1,1,\#}} \vee \overline{x_{1,1,q_0}})} \wedge \ldots$

formula for start config

$$\phi_{start} = X_{1,1,\#} \wedge X_{1,2,q_0} \wedge \bigwedge_{i=1}^{n} X_{1,i+2,w_i}$$
$$\wedge \bigwedge_{i=n+1}^{poly(n)-1} X_{1,i,\sqcup} \wedge X_{1,poly(n),\#}$$

formula for accept config

$$\phi_{accept} = \bigvee_{j=1}^{poly(n)} X_{poly(n),j,q_{accept}}$$



Define $\phi_{legal,i,j}$ to be true iff row i yields row j.

$$\phi_{move} = \bigwedge_{i=1}^{poly(n)-1} \bigwedge_{j=2}^{poly(n)-1} \phi_{legal,i,j}$$

Make

$$\phi = \phi_{cell} \wedge \phi_{start} \wedge \phi_{accept} \wedge \phi_{move}$$

ϕ is satisfiable iff T accepted w.

CNFSAT is NP-hard.

CLIQUE is NP-complete.

CLIQUE $\in$ NP — done.

CLIQUE is NP-hard:

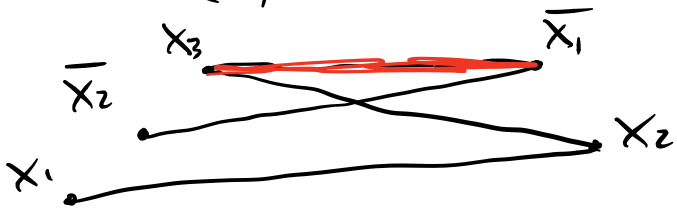
CNFSAT $\leq_p$ CLIQUE.

Let ϕ be a CNF formula with n vars and m clauses.

For each of the m clauses, make a "vertex gadget" $= (x_1 \vee x_2 \vee \cdots \vee x_n)$

$x_1 \quad x_2 \quad x_3 \quad x_4 \quad \cdots \quad x_n$

Ex: $\overbrace{(x_1 \vee \overline{x_2} \vee x_3)} \wedge \overbrace{(\overline{x_1} \vee x_2)}$



Connect all pairs of vertices iff they are not in the same clause nor are they complements of the same variable.

Claim: the graph has an m-clique iff the original formula is satisfiable.

Office Hours / Problem Solving

ab	a		
bd			

ac
