Lecture #26:

Complexity theory

NP-completeness

Cook-Levin theorem

Reductions

Next class: recap/review

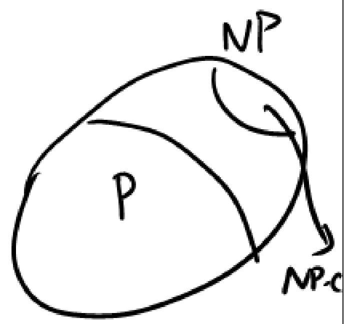
$$\mathrm{PSPACE} : \bigcup_{k \geq 1} \mathrm{SPACE}(n^k)$$

$$\mathrm{EXPTIME} - \bigcup_{k \geq 1} \mathrm{TIME}(2^{n^k})$$

$$\begin{tikzcd}
P \arrow[r, phantom, "\subseteq"] \arrow[rrr, bend right, "\subsetneq"'] & NP \arrow[r, phantom, "\subseteq"] & \mathrm{PSPACE} \arrow[r, phantom, "\subseteq"] & \mathrm{EXPTIME}
\end{tikzcd}$$

L is NP-complete:

- L is in NP and
- Every problem $A \in NP$, $A \leq_p L$



$A \leq_p L$ if $\exists f : \Sigma^* \to \Sigma^*$

that is poly time computable

$\forall x \in \Sigma^* \quad x \in A \iff f(x) \in L$

L is NP-complete

Then if $L \in P$ then $NP \subseteq P$

$\Rightarrow P = NP$

If $L \notin P$, since $L \in NP$, $P \neq NP$.

So $L \in P \iff P = NP$.

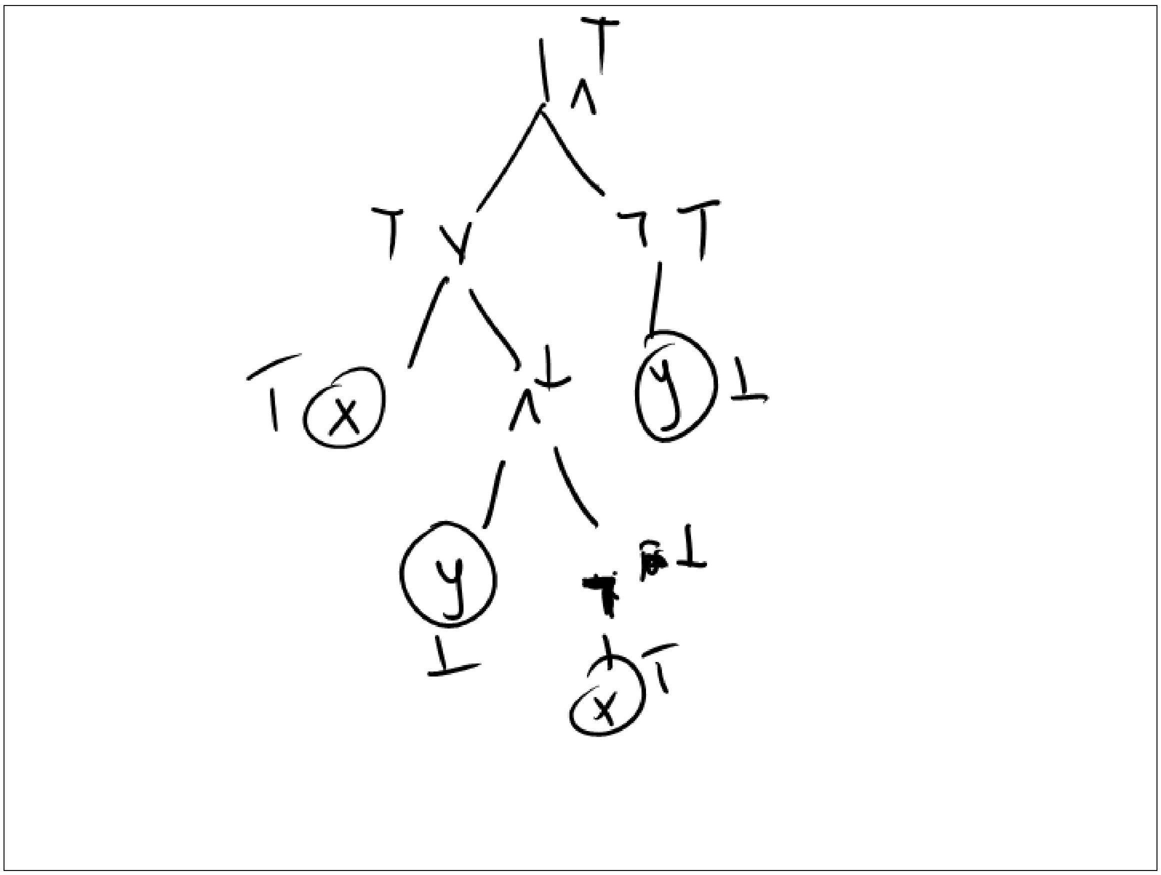
SAT is NP-complete

SAT = $\{ \varphi \mid \varphi$ is a Boolean formula that is satisfiable $\}$

$$BF \quad \varphi ::= \underset{x \in Var}{x} \mid \neg\varphi \mid \varphi \vee \varphi' \mid \varphi \wedge \varphi' \mid \cancel{T \mid F}$$

<u>SAT $\in$ NP</u>.

- Generate a valuation $Val : V \to \{T, \bot\}$
- Check if under Val, φ is satisfied.
- If it is, accept else reject.

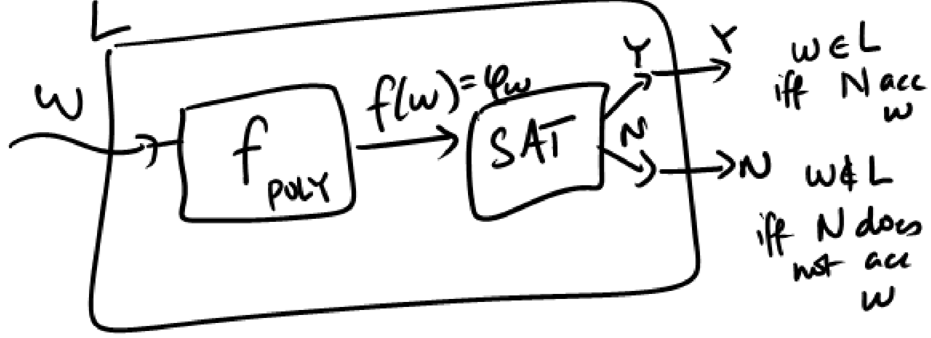


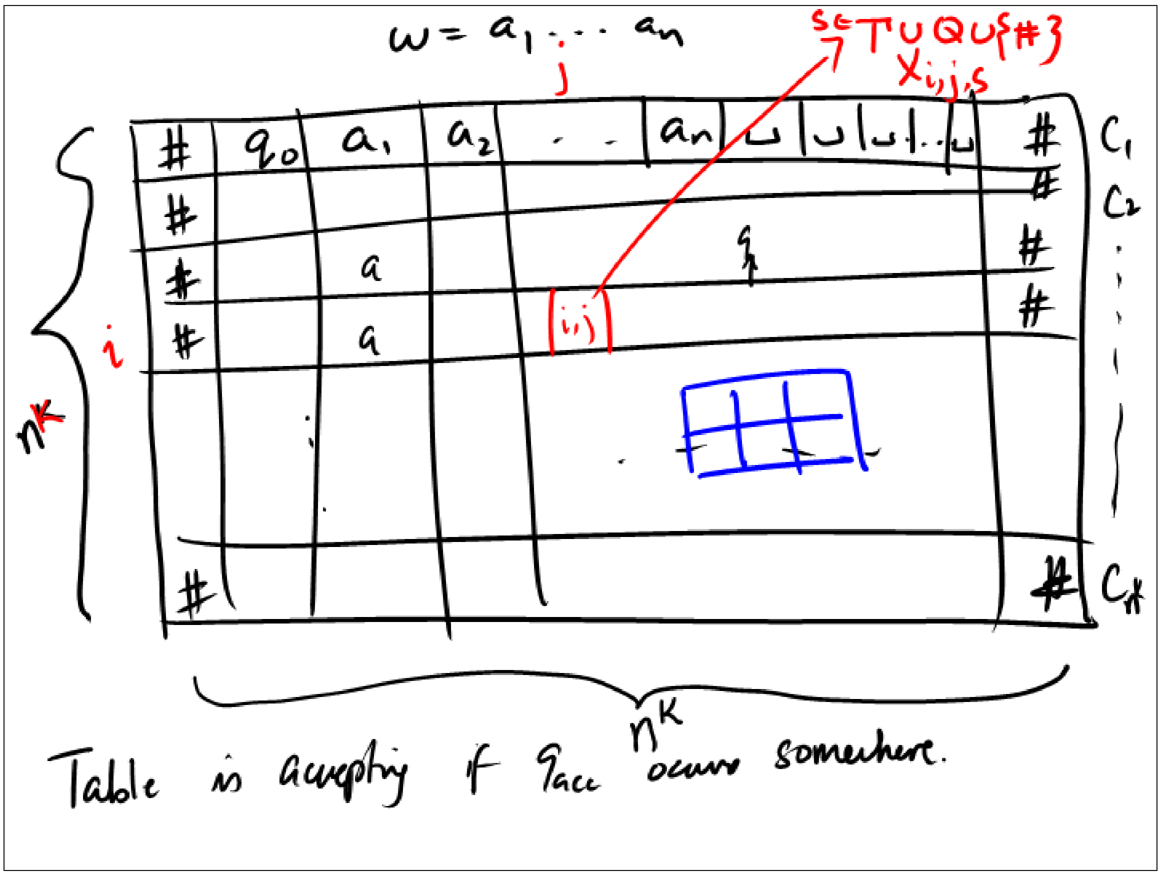
⊤
∧
⊤
∨
¬
⊤
⊤
x
⊥
∧
y
⊥
y
⊥
⊥
¬
x
⊤

SAT is NP-complete.

$\forall L \in NP$, $L \leq_p SAT$

Since L is in NP, $\exists$ NTM N running in time $O(n^k)$ and accepting L. for a fixed k,





w = a1 ... an
j
s ∈ T ∪ Q ∪ {#}
x i,j,s
#
q0
a1
a2
an
#
C1
C2
a
a
i
(i,j)
#
#
#
#
#
#
Cn^k
n^k
n^k
Table is accepting if q_acc occurs somewhere.

Given w, construct φ_w

s.t. φ_w is satisfiable

iff there is an accepting table of N on w

iff $w \in L$.

$x_{i,j,s}$ variable

$x_{i,j,s} = T$ would mean cell (i,j) has symbol s

$x_{i,j,s} = \perp$ means (i,j) does not have symbol s.

$$\varphi_w = \varphi_{cell} \wedge \underline{\underline{\varphi_{start}}} \wedge \underline{\underline{\varphi_{move}}} \wedge \underline{\underline{\varphi_{acc}}}$$

φ_{cell} : Every cell has a symbol
& has only one symbol.

$$\bigwedge_{1 \le i,j \le n^k} \left(\bigvee_{s \in C} x_{i,j,s} \right) \qquad C = Q \cup \Gamma \cup \{\#\}$$

$$\wedge \bigwedge_{1 \le i,j \le n^k} \left(\bigwedge_{\substack{s,t \in C \\ s \ne t}} \left(\overline{x_{i,j,s}} \vee \overline{x_{i,j,t}} \right) \right)$$

$$n^k \cdot n^k \cdot |C|^2 = n^{2k} \cdot |C|^2.$$

$$\varphi_{start} = x_{1,1,\#} \wedge x_{1,2,q_0} \wedge x_{1,3,a_1} \wedge \ldots x_{1,n+2,a_n}$$
$$\wedge\ x_{1,n+3,\sqcup} \wedge \ldots\ x_{1,n^k-1,\sqcup}$$
$$\wedge\ x_{1,n^k,\#}$$

$O(n^k)$

$$\varphi_{acc} : \bigvee_{1 \le i,j \le n^k} x_{i,j,q_{acc}}$$

Legal 2×3 windows

$$\delta(q_1, b) \ni (q_2, c, L)$$

$$\delta(q_1, b) \ni (q_3, d, R)$$

a	q_1	b
q_2	a	c

a	q_1	b
a	d	q_3

a	b	c
a	b	c

a	b	c
a	d	c

× illegal

a	b	c
d	b	c

legal

$$\varphi_{move} = \bigwedge_{\substack{1 < i < n^k \\ 1 \leq j < n^k}} \left(\begin{array}{l} \text{the } (i,j) \text{ window is legal} \\ (i,j-1)\,(i,j)\,(i,j+1) \\ (i+1,j-1)\,(i+1,j)\,(i+1,j+1) \end{array} \right)$$

$\underbrace{\hspace{6em}}_{\text{constant}}$

$O((n^k)^2) = O(n^{2k})$

So given w, φ_w is sat iff $w \in L$

and $|\varphi_w|$ is $poly(|w|)$ and computable in PTIME

SAT is NP-complete.

If SAT $\leq_p$ L and L is in NP, then L is also NP-complete.

↳ L is in NP

↳ $A \in NP$, to show $A \leq_p L$

$A \leq_p SAT \leq_p L$

$A \leq_p L$

3SAT Literal : x or $\bar{x}$

Clause : OR of literals

$(x \vee y \vee \bar{x})$

$(x \vee \bar{y} \vee z \vee \bar{z})$

CNF : $C_1 \wedge C_2 \wedge \ldots C_k$

where each C_i is a clause.

$\varphi \longrightarrow$ CNF φ' $\quad |\varphi'| \leq \text{poly}(|\varphi|)$

and φ is sat iff φ' is sat.

3CNF – CNF $C_1 \wedge C_2 \dots \wedge C_k$ where every clause has exactly 3 literals.

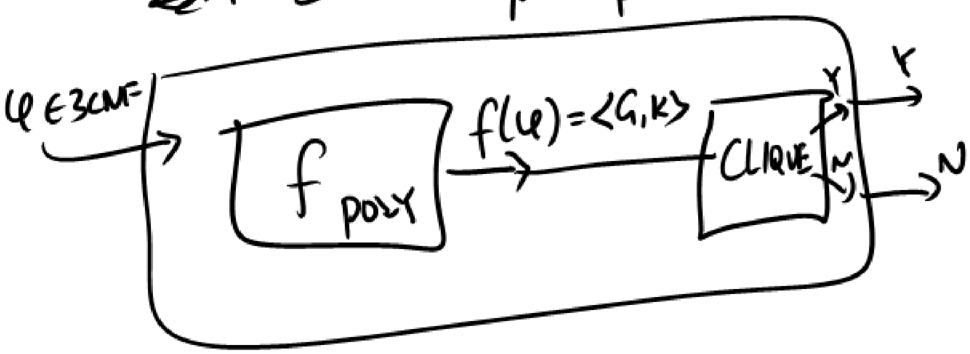
$$(x_1 \vee x_2 \vee x_3 \vee x_4)$$

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

3CNF is NP-complete.

$$CLIQUE = \{ \langle G, k \rangle \mid G \text{ has a clique of size } k, \ G \text{ is undirected} \}$$

A $3CNF \le_p CLIQUE$



CLIQUE is in NP.

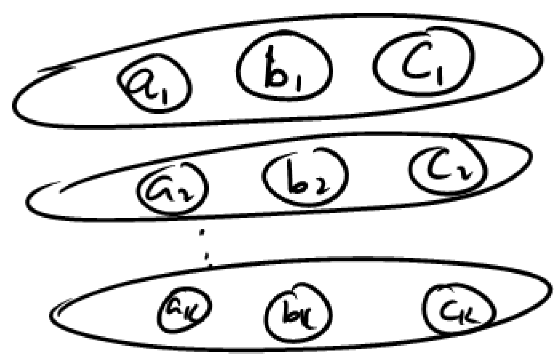
- Given k vertices
- Check if it is clique.

~~\$~~ $3CNF \leq_P$ ~~SAT~~ CLIQUE

$$\varphi = (a_1 \vee b_1 \vee c_1) \wedge (a_2 \vee b_2 \vee c_2) \wedge \dots (a_k \vee b_k \vee c_k)$$

$\rightsquigarrow \langle G, k \rangle$

φ is sat iff G has k-clique.

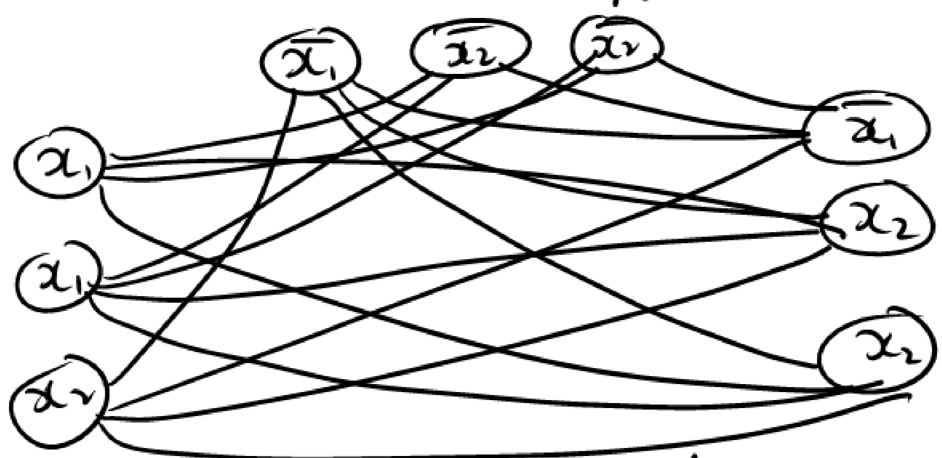


u — v

unless u, v are in same triple

or u, v are contradicting

$(x_1 \vee x_1 \vee x_2) \wedge (\bar{x}_1 \vee \bar{x}_2 \vee \bar{x}_2)$
$\wedge (\bar{x}_1 \vee x_2 \vee x_2)$



φ sat $\Rightarrow$ P ∃ k-clique exists
Take one vertex that gets satisfied in the satisfying assgn. in each triple.

k-clique $\Rightarrow \varphi$ sat.