Previously.

- NP
  - Simplified NTM
  - Verifier Definition

Verifier Prover

$V(x, c) \leftarrow c$ Good / Malicious

$L \in NP$ if $\exists$ a verifier $V(x, c)$ s.t

- $\forall x \in L, \exists c$ s.t $V(x, c) = 1$ ✓ (Comp)
- $\forall x \notin L, \forall c$ $V(x, c) = 0$ ✓ (Soundness)

- CNF, IND SET,

- Reduction

CNF $\longrightarrow$ 3CNF

$\varphi \longrightarrow \varphi'$

Tarjan's alg
2CNF

s.t φ is satisfiable iff φ' is satisfiable

3CNF $\longrightarrow$ INDSET

$\varphi \longrightarrow (G, m)$

s.t φ is satisfiable iff G has an indepset of size $\geq m$

3CNF $\leq$ INDSET

$L_1 \leq L_2$

CNF $\leq$ 3CNF
$\wedge$
CNF

L_2 is atleast as hard as L_1

$L_1 \leq L_2$ if there is a poly time deterministic TM M s.t

$\forall x \in L_1, \quad M(x) \in L_2$

$\forall x \notin L_1, \quad M(x) \notin L_2$

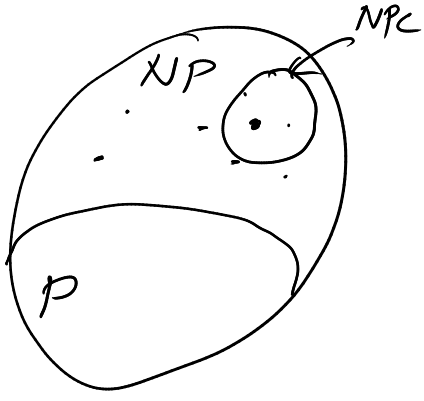
- Reductions are transitive

$$L_1 \overset{M_1}{\leq} L_2 \overset{M_2}{\leq} L_3 \qquad M_2(M_1(x))$$

$$\Rightarrow L_1 \leq L_3$$

NP-Complete

$= \{ L \in NP : \forall L' \in NP, \; L' \leq L \}$



L_A and L_B are NPC

then $L_A \leq L_B$

and $L_B \leq L_A$ (by definition)

- NPC is the set of hardest problems in NP.
- Suppose there is a polytime TM for deciding an NPC language then $P = NP$

- Is there an NP-complete problem?

- **Cook-Levin Theorem**

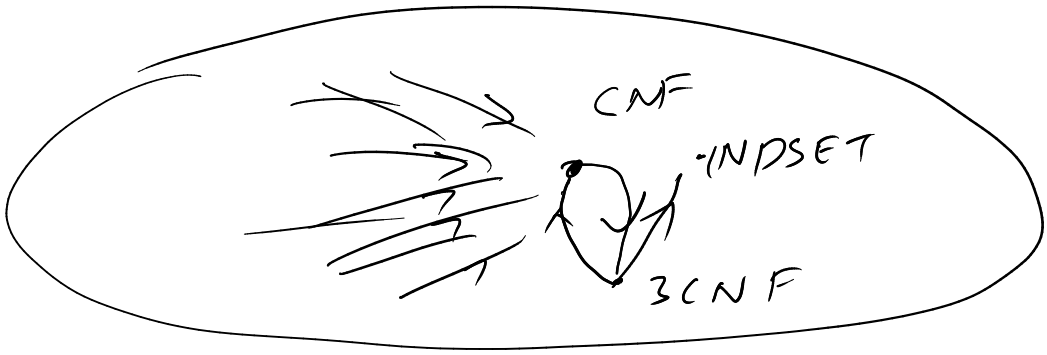
  CNF is NP-complete.

60-70

Karp

$CNF \leq 3CNF$

$\Rightarrow$ 3CNF is NP Complete.



$3CNF \leq INDSET$

$\Rightarrow$ INDSET is NP-Complete.

If we want to show, $L \in$ NP-Complete then we just need to give a reduction that converts an instance of an existing NP-Complete language to L

Karp showed over 30 combinatorial problem are NP-Complete.

$P \stackrel{?}{=} NP$ will mean coming up with a proof is as easy as verifying a proof: → mechanical.

FACTORING (n)

N, m, n

$m \cdot n = N$

Cook-Levin Theorem.

$\forall L \in NP, \quad L \leq CNF$

Proof:

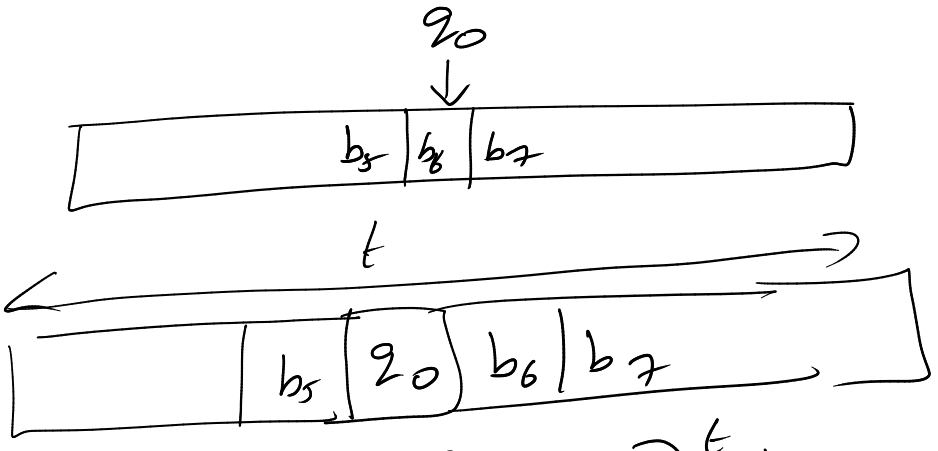
$L \in NP$, => There is a verifier T_M s.t (clock, poly time.)

$\forall x \in L, \exists c, \quad V(x,c)$ accepts

$\forall x \notin L, \forall c, \quad V(x,c)$ rejects.

Configuration of the verifier.

- current state
- position of tape head
- contents of work tape



a string $\in (\Gamma \cup Q)^t$

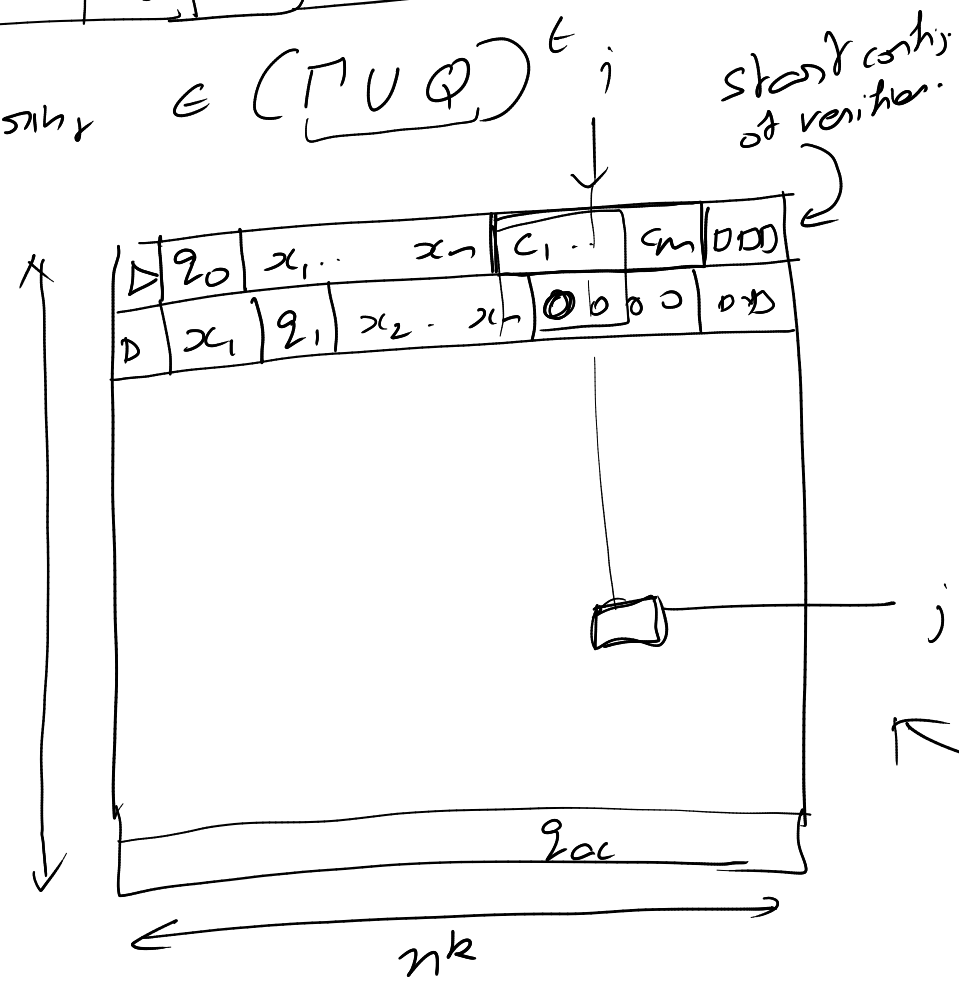


tableau.

$x \in L$ iff $M(x, c) = 1$

$\Rightarrow$ ∃ a tableau of $n^k \times n^k$ s.t

① first row is the start configuration

② last row is the accept configuration

③ Every consecutive row is obtainable by application of a δ (transition rule)

Proof Idea: Check all of above using a CNF formula

$x \longrightarrow \varphi$

Cell $[i, j]$ = symbol at i, j^{th} posi in the tableau

$\Gamma \cup Q$

$\forall i, j \in [1, \ldots n^k]$, $s \in \Gamma \cup Q$ → constant.

$y_{i,j,s} = 1$ iff Cell $(i,j) = s$

How many such variables?

$n^k \cdot n^k \cdot (|\Gamma| + |Q|) = O(n^{2k})$

- New constraint:

(4) – Every cell can have exactly 1 symbol. ✓

$y_{11q_1} = 1$ $\quad$ $y_{11q_2} = 1$

↳ not possible in a valid tableau.

$v_1 \ldots v_t$

Find CNF s.t exactly one of $v_1 \ldots v_t$ is 1?

- atleast one of $v_1, \ldots v_k$ is 1

$v_1 \vee v_2 \cdots \vee v_k$

- atmost one of $v_1 \ldots v_k$ is 1

$$\bigwedge_{i,j} (\bar{v}_i \vee \bar{v}_j)$$

$k=3$

$$(\bar{v}_1 \vee \bar{v}_2) \wedge (\bar{v}_2 \vee \bar{v}_3) \wedge (\bar{v}_3 \vee \bar{v}_1)$$

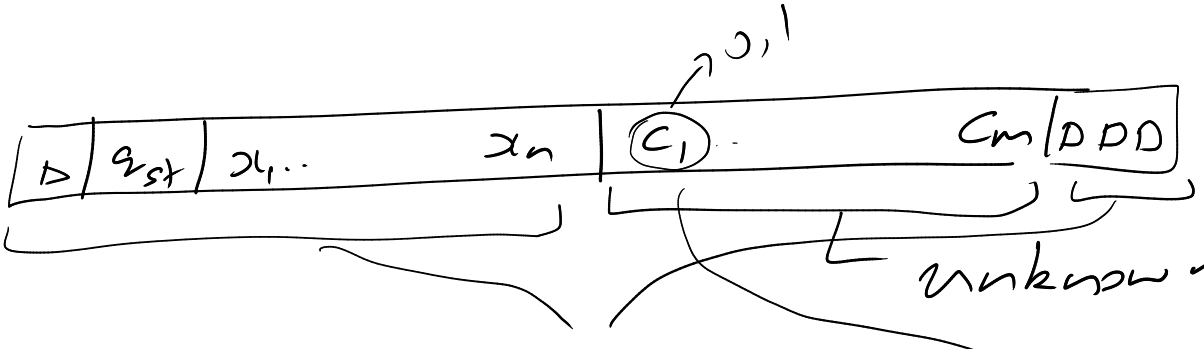
1 0 0 0 0 1

$\wedge$

For every call,

( ) $\wedge$ ( )

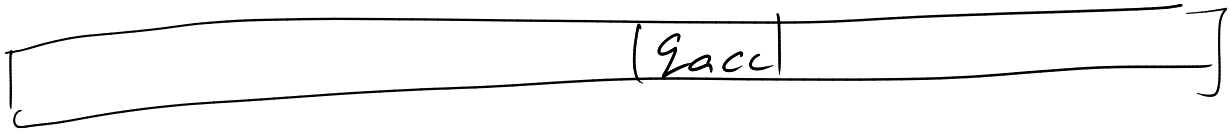
① First row is the start configuration



$$\left(y_{11\triangleright} \wedge y_{12 q_{st}} \wedge y_{13 x_1} \wedge y_{14 x_2} \cdots \right.$$

$$\left. \wedge \left(\left(y_{1 n+2 0} \vee y_{1, n+2, 1} \right) \wedge \right) \right)$$

② last row is the acc configuration.



$$\left(y_{n^t 1 q_{acc}} \vee y_{n^t 2 q_{acc}} \cdots \vee y_{n^t n^t q_{acc}} \right)$$

3. Consecutive configuration are obtained by δ transitions.

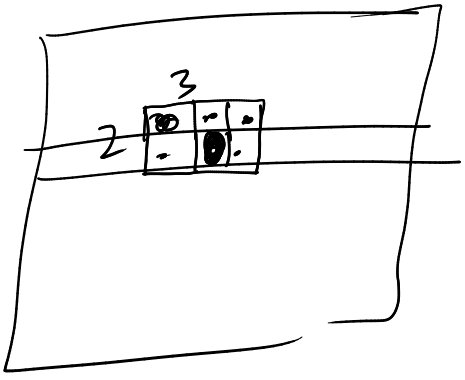
C_1 ▷ b a a q_1 b c b #

C_2 ▷ b a q_2 a b c b #

C_3 ▷ b q_2 a a c c b #

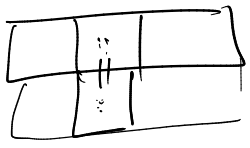
$$\delta(q_1, a) = (q_2, a, L)$$

2×3 window in tableau.



Every 2×3 window can have some limited possibilities.

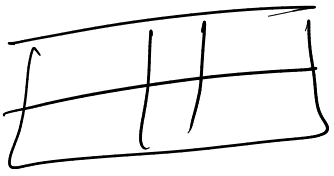
↳ constant.



$\exists$ no 's.

$S \subseteq (Q \cup \Gamma)^6$

$\in (Q \cup \Gamma)^6$



↳ $\forall i, j \in n^k$,

check if the 2×3 window at $i,j \in S_{i,j}$

Can be encoded by a constant size formulae.

∴ Total n^{2k} size formulae.

$x \longrightarrow \varphi$

$O(n^{2k})$

Size.

$x \in L \iff \varphi$ is satisfiable.

$M(x)$

$x \in L$ if $\exists$ a path at accepts.

$x \notin L$ if $\forall$ paths reject.