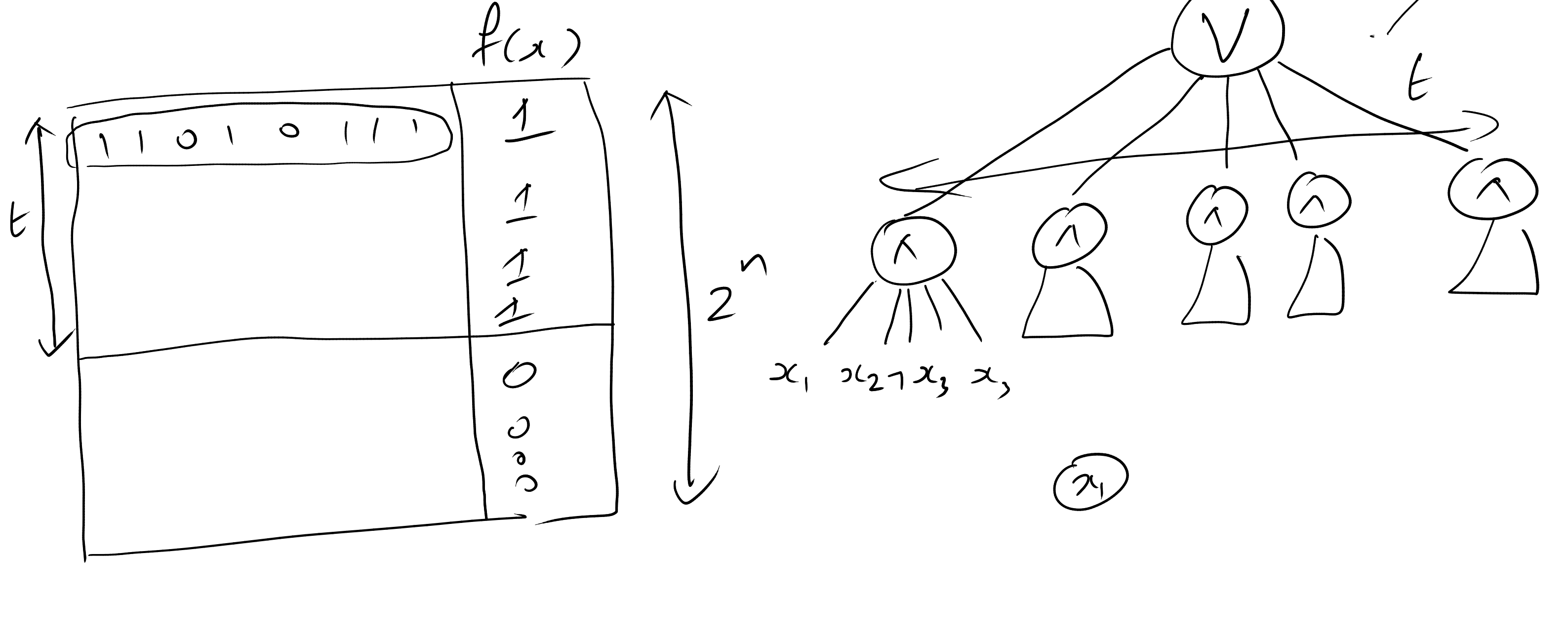
$$P = \bigcup_{k \in \mathbb{N}} DTIME(n^k)$$

$$NP = \bigcup_{k \in \mathbb{N}} NTIME(n^k)$$

DTIME
NTIME

Circuits.

- DAG, $\wedge$, $\vee$, $\neg$, leaf x_1 x_2 ... x_n
  root node.
- $f : \{0,1\}^n \longrightarrow \{0,1\}$ has a a Boolean Formula representation



Size (C) = # internal nodes

Depth (C) = len of longest path from root to leaf.

Any $f : \{0,1\}^n \to \{0,1\}$ has a Boolean formula of

$$\text{size} \le (t+1) + t(n) \le n 2^n$$

$$\text{Depth} \le 3$$

__Shannon__: The no. of fns computable by circuits of size $s \le 2^{s \log(s)}$

$$2^{2^n} \ge 2^{s \log(s)}$$

Open Problem:

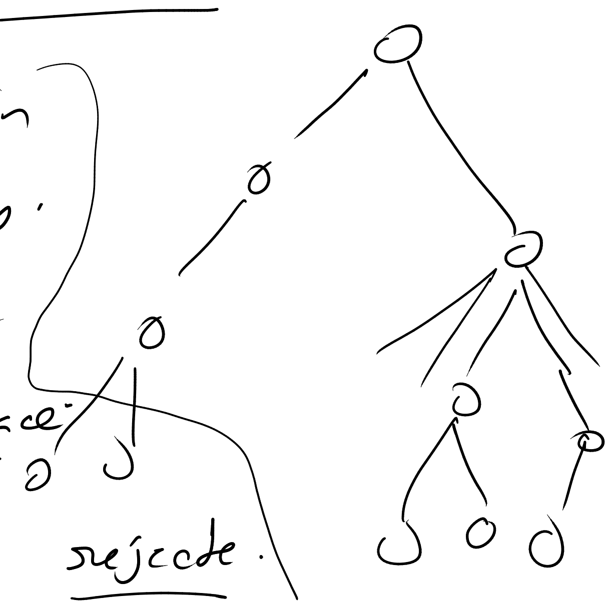
Find a $f : \{0,1\}^n \to \{0,1\}$, s.t

f cannot be computed by

size(n^k) circuits.

$L \in NP$ iff $\exists$ a Nondet polynomial time TM M that decides L

- all nondet paths in config graph halts.
- $x \in L \Rightarrow \exists$ a path that result in acceptance.
- $x \notin L \Rightarrow \forall$ path result in rejecte.



- length of the path $\leq n^k$

$$\delta \subseteq Q \times \Gamma^3 \times Q \times \Gamma^3 \times \{L, R\}^3$$

Simplified NTM:

$$\delta_0 : Q \times \Gamma^3 \longrightarrow Q \times \Gamma^3 \times \{L, R\}^3$$

$$\delta_1 : Q \times \Gamma^3 \longrightarrow \quad ''$$

From any config, NTM can transition using δ_0 or δ_1.

NP (Alternate Definition)

NP is the class of languages for which there is a polynomial time verifiable certificates def poly time TM

$L \in NP$ if $\exists$ TM M s.t

$x \in L \iff \exists u \in \{0,1\}^{n^k}, M(x,u)=1$

- u is a certificate
- $x \in L \Rightarrow \exists u, M(x,u)=1$
- $x \notin L \Rightarrow \forall u \in \{0,1\}^{n^k}, M(x,u)=0$

HAM-CYCLE $= \{ \langle G \rangle \mid G$ has a hamilton cycle $\}$

$M(\langle G \rangle, \underline{\langle p \rangle})$ { $\leftarrow$ path in G. $[u_1, u_2, u_3 \ldots x_n]$

- verifies p is a cycle in G.
- verify if p has every vertex exactly once

}

Verifier (poly time)

$V(x, c)$

$\xleftarrow{c}$

Prover:

Good

Malicious

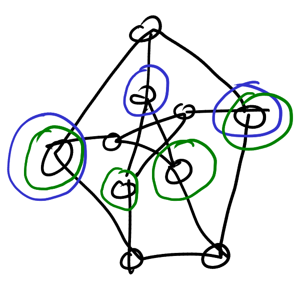
Interactive Proofs

INDSET = { <G, k>: G has an independent set of size $\geq k$ }

independent set is a set vertices in G, s.t there is no edges between any pair.

maximum independent set = 4

maximal independent set.



Verifier$(\langle G, k \rangle, \langle S \rangle)$ {

- $|S| \geq k$
- verify that there is no edge in S

}

$3CNF = \{ \langle \varphi \rangle \mid \varphi \text{ is satisfiable} \}$

clauses $\quad\quad\quad\quad\quad\quad\quad\quad$ variables.

$C_1 \quad \dots \quad C_m \quad\quad\quad x_1 \quad \dots \quad x_n$

$$C_i = \underbrace{x_3 \vee \neg x_1 \vee x_5}_{\text{3 literals.}}$$

2CNF

$x_5 \Rightarrow x_3$

$\neg x_3 \rightarrow \neg x_5$

$$C_i = \underbrace{x_3 \vee \neg x_5}_{\text{2 literals.}}$$

$$x_1 \Rightarrow x_2 \equiv \neg x_1 \vee x_2$$

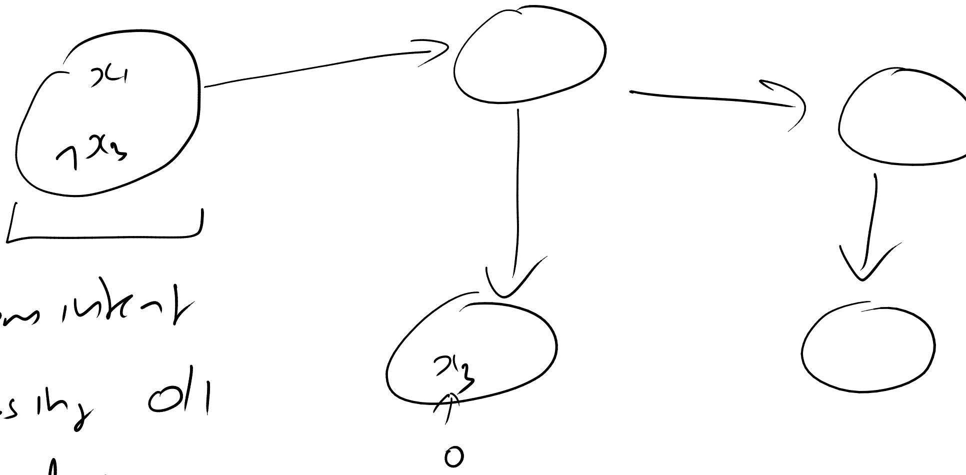
Poly time algorithm for 2CNF (Tarjan)

- Construct Implication graph
- For every clause put 2 directed edges.

- Construct Strongly Connected Components.
- $x_i, \neg x_i \in$ same SCC then φ does not have a satisfying assignment.
- else φ is satisfying

Graph of SCC

DAG



consistent assign 0/1 values

3CNF

- Can convert a general CNF formulae to 3CNF

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

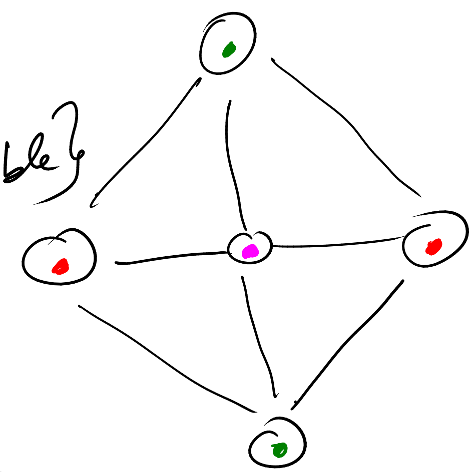
$(x_5 \vee \neg x_3 \vee x_2 \vee \neg x_1)$

$(x_1 \vee x_2 \vee y_1)$ $(\neg y_1 \vee x_3 \vee x_4)$

$(x_5 \vee \neg x_3 \vee y_2)$ $(\neg y_2 \vee x_2 \vee \neg x_1)$

Graph Coloring:

Coloring $= \{ \langle G, k \rangle :$ check if G can be colored using $\leq k$ colors. $\}$



2-Coloring $= \{ \langle G \rangle :$ G is 2-colorable $\}$

3-Coloring $= \{ \langle G \rangle :$ G is 3-colorable $\}$

Claim: If we solve IND-SET we can solve 3-CNF.

3-Col1

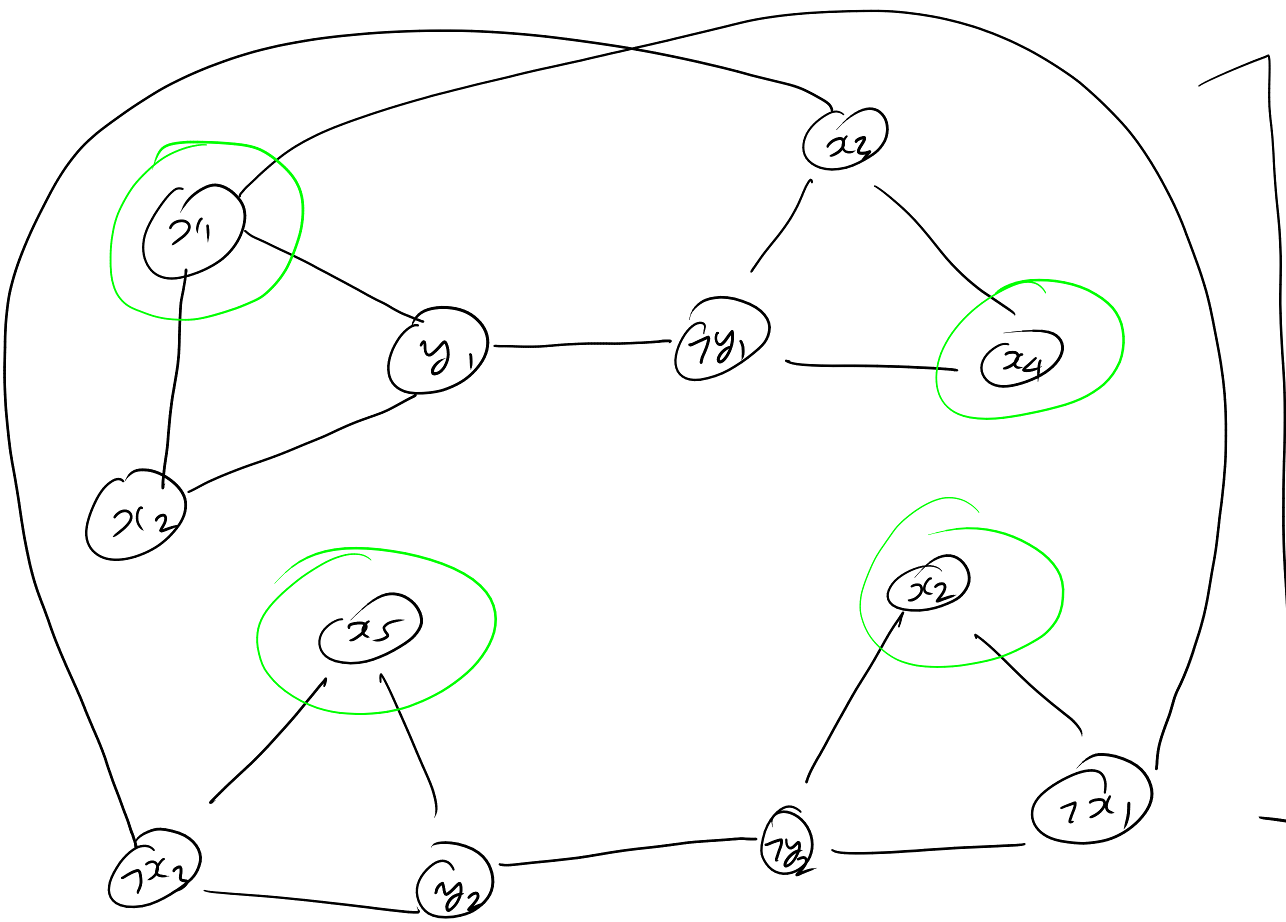
$\varphi \quad c_1 \cdot\cdot c_2 \cdot\cdot c_m \quad x_1 \cdot\cdot x_n$

$\downarrow$

G, m

$(x_1 \vee x_2 \vee y_1) \quad (\neg y_1 \vee x_3 \vee x_4)$

$(x_5 \vee \neg x_3 \vee y_2) \quad (\neg y_2 \vee x_2 \vee \neg x_1)$



3m vertices.

φ has a satisfying assignment $\iff$ G has an independent set of size m