PH. recall def $\Sigma^i P$ $L \in \Sigma^i P$ if $\exists$ TM M runs in poly time s.t.

$$L = \{x : \exists y_1 \forall y_2 \cdots Q^i y_i \; M(x, y_1 \cdots y_i) = 1\}.$$

prop. if $\Sigma^i P = \Pi^i P$ for some i then $PH = \Sigma^i P$
if $P = NP$ then $PH = P$.

sketch: $\Sigma^{i+1} P = \Sigma^i P$

x: $\exists y_1 \forall y_2 \cdots Q^i y_i Q^{i+1} y_{i+1} \; M(\ldots) = 1$.

guess y_1, check w/ $\forall$. $\rightarrow$ C_{x,y_1}: is a lang. in $\Pi^i P$. $= \Sigma^i P$.

x: $\exists y_1 \forall y_2 \cdots Q^i y_i : Q^{i+1} y_{i+1}$ so $L \in \Sigma^i P$. ✓✓

prop: $\Sigma_1 = NP$; $\Sigma_2 = NP^{NP} = NP^{\Sigma_1}$ $\Sigma_3 = NP^{\Sigma_2}$ $\cdots$ $\Sigma_{i+1} = NP^{\Sigma_i}$

proof: $i=1$ by def. ($i=2$).

$\subseteq$ pick $L \in \Sigma_2 P$ by def $\exists M$ $L = \{x : \exists y \forall z \; M(x,y,z) = 1\}$.

$L' = \{(x,y) \forall z \; M(x,y,z) = 1\}$. $L' \in \Pi_1 P = coNP$.

build N nondd. poly time, oracle L': input x: guess y. query if $(x,y) \in L'$

N decide $L \Rightarrow L \in NP^{NP}$.

$\supseteq$ pick $L \in NP^{NP}$. by def $\exists M$ guesses (y) and makes queries $q_1 \cdots q_k$.
receives answers $a_1 \cdots a_k$.
oracle N

if q_i, a_i: 1 $\exists v_i$ st $N(q_i, v_i) = 1$
0 $\forall v_i$ $N(q_i, v_i) = 0$.

write in $\Sigma^2 P$ form, ie: $\exists \cdots \forall \cdots M'(\;)$

$\exists y, q_1, \ldots q_k, a_1 \cdots a_k, v_1, \ldots v_k \forall v_1, \ldots v_k$:

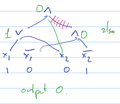
→ M accept (with $y, q_1 \cdots q_k, a_1 \cdots a_k$) (simulate M) ✓
→ $a_i = 1 \Rightarrow N(q_i, v_i) = 1$ ✓
→ $a_i = 0 \Rightarrow N(q_i, v_i) = 0$ ✓
$L \in \Sigma^2 P$ ✓✓

Circuits.

why? black-box TM $\Rightarrow$ cannot distinguish $P=NP$ from $P \neq NP$.
$\exists$ oracles A, B st $P^A = NP^A$, $P^B \neq NP^B$.

circuits can analyze structure: hope?

def circuit: (for input of size n): DAG (directed acyclic graph) vertices have labels $\wedge$ $\vee$ $\neg$.
$2n$ source vertices (without incoming edges)
1 sink (— outgoing edges).
internal vertices have exactly 2 incoming edges (indegree 2).



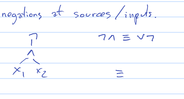
also OK.

input $x_1 = 1$
$x_2 = 0$
output 0.

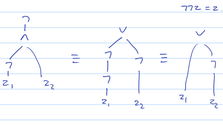
→ obs: indegree 2 does not matter

(unless limited circuits). $k/2 + k/4 + \cdots + 1$ gates $\leq k$.

→ obs: negations at sources/inputs.



worst case: each var. needs own negation.
size increases by $\leq 2n$.



→ obs: for each n can have a different circuit. "non-uniformity".

prop: circuits can compute every unary language.

$L = \{1^n$ for $n \in X\}$. $\{C_n\}_n$: if $n \in X$ then $C_n = \wedge x_1 \cdots x_n$.
$n \notin X$ then $C_n = 0$.

def $P/poly$: languages that can be decided by a family of circuits of polynomial size.

$P \neq P/poly$.

thm $P \subseteq P/poly$.

def family of circuits is uniform if $\exists$ TM M that on input 1^n outputs C_n ($\{C_n\}_n$)

prop languages decidable by uniform circuits $= P$.

sketch: input x: use M to build $C_{|x|}$, then eval. $C_{|x|}(x)$. //

are there TMs equivalent to circuits? yes! TMs with advice.

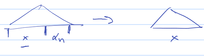
exists a DB of advice (sequence α_n, $|\alpha_n| = poly(n)$) M runs on input $(x, \alpha_{|x|})$

$C/\ell(n)$ = C with advice of size $\in \ell(n)$ $P/poly$

prop $P/poly$ is the same as languages decidable by circuits of poly size.

sketch: L $\{C_n\}_n$. advice $\alpha_n = C_n$. $M(x, C_n)$ simulate C_n on x.

$L \in P/poly$. build circuit that simulates $M(x, \alpha_n)$.



$P \subsetneq P/poly$. NP?

is $NP \subseteq P/poly$? unlikely!

thm (Karp–Lipton) If $NP \subseteq P/poly$ then $PH = \Sigma^2 P$.

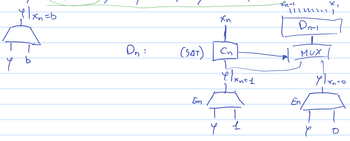
proof. enough to show $\Sigma^2 P = \Pi^2 P$. Π^2-SAT is $\Pi^2 P$-complete.
Π^2-SAT is in $\Sigma^2 P$

$\{\phi : \forall x \exists y \; \varphi(x,y)\}$.

assume $NP \subseteq P/poly$: $\exists \{C_n\}_n$ that decide SAT.

$\forall \varphi$, $|\varphi| = n$, $C_n(\varphi) = 1$ iff φ is SAT.

self-reducibility: $\exists \{D_n\}_n$ given φ output γ st $\gamma(\gamma) = 1$.



$\Sigma^2 P$ language: given $z : \forall x \exists y \; \varphi(x,y)$.

$\exists D_n \forall \varphi, x$: → $\varphi(D_n(\varphi)) \wedge$
→ $D_n(\varphi|_x) \neq \phi$.

Π^2-SAT $\in \Sigma^2 P$.
hence $\Sigma^2 P = \Pi^2 P$
$PH = \Sigma^2 P$ //.

thm [Shannon] $\exists f : \{0,1\}^n \rightarrow \{0,1\}$ req. circuits of size $2^n/10n$.

proof: counting. # functions 2^{2^n}.

circuits on s gates? $s \cdot (2\log s + 1) \leq 3 s \log s$.

circuits $\leq 2^{3 s \log s}$.

$3 s \log s \leq 2^n$ → not enough circuits!

$s > 2^n/10n$ //