Lecture 18: Ladner's Theorem

Date: October 26, 2023.

NP: $A \in \text{NP}$ iff there is a NTM M and k such that M runs in n^k and $\mathbf{L}(M) = A$.

Polynomial Verifiability: A is **verifiable in polynomial time** iff there is a (deterministic) TM V, k, ℓ such that (a) V runs in n^k time, and (b) $x \in A$ iff there is y, $|y| \leq |x|^\ell$ such that V accepts input $\langle x, y \rangle$.

$A = \{ x \mid \exists y.\ |y| \leq |x|^\ell \text{ and } \langle x,y \rangle \in L(V) \}$

Here V is said to be a **verifier** for A. For $x \in A$, a string y such that V accepts $\langle x, y \rangle$ is said to be the **proof** that $x \in A$ with respect to V.

Theorem 1. ~~Prove that~~ $A \in \text{NP}$ *iff* A *is verifiable in polynomial time.*

($\Rightarrow$) $A \in NP$. $\exists$ NTM M that runs in n^k time and $L(M) = A$.

"Proof" y: the sequence of nondet choices of M that lead to accepting x.

Verifier: Input $\langle x,y \rangle$ Run M on x on the choices y. Accept if M does.

($\Leftarrow$) V is a verifier for A.

Algo for A: Input x, Guess a string y and run V on $\langle x,y \rangle$. Accept if V

Logspace Computable Functions: A function f is **computable in logspace** if there is a Turing machine M such that on any input x, M halts with $f(x)$ written on its output tape, and M uses at most $O(\log |x|)$ cells on its work-tape.

Logspace Reducibility: A is reducible to B in **logspace** (denoted $A \leq_m^{\log} B$) if there is a logspace computable function f such that for any x, $x \in A$ iff $f(x) \in B$.

Proposition 2. *Let* $\mathcal{C} \in \{\text{NL}, \text{P}, \text{NP}, \ldots\}$. *If* $A \leq_m^{\log} B$ *and* $B \in \mathcal{C}$ *then* $A \in \mathcal{C}$.

Proposition 3. *If* $A \leq_m^{\log} B$ *and* $B \leq_m^{\log} C$ *then* $A \leq_m^{\log} C$.

Hardness and Completeness: Let $\mathcal{C} \in \{\text{NL}, \text{P}, \text{NP}, \ldots\}$. A problem B is **$\mathcal{C}$-hard** if for any $A \in \mathcal{C}$, $A \leq_m^{\log} B$. B is **$\mathcal{C}$-complete** if B is $\mathcal{C}$-hard and $B \in \mathcal{C}$.

Proposition 4. *If* A *is* NP*-hard and* $A \leq_m^{\log} B$ *then* B *is* NP*-hard.*

Need to show: $\forall L \in NP,\ L \leq_m^{\log} B$.

Know: $\forall L \in NP,\ L \leq_m^{\log} A,\ A \leq_m^{\log} B$.

Proposition 5. *If* A *is* NP*-complete and* $A \in \text{P}$ *then* $\text{NP} = \text{P}$.

$L \in NP$.

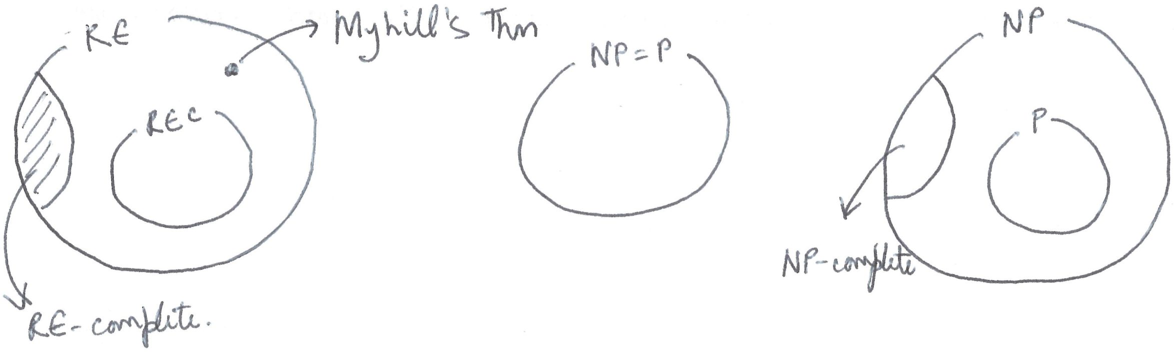
$L \leq_m^{\log} A,\ A \in P \Rightarrow L \in P$.

$L \in$ R.E. iff there is recursive relation R s.t.

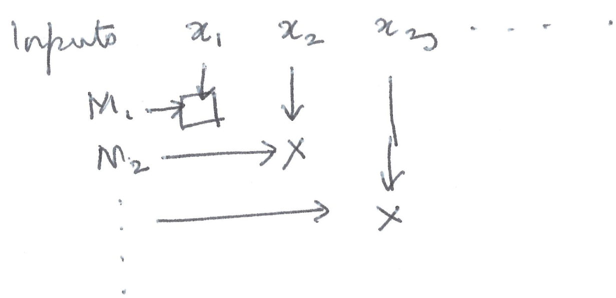
$$L = \{ x \mid \exists y.\ (x,y) \in R \}.$$

$L \in NP$ iff there is a polytime relation R s.t.

$$L = \{ x \mid \exists y.\ |y| \le |x|^l \text{ and } (x,y) \in R \}.$$



Diagonalization Technique



Satisfiability: SAT is the problem, where given φ a CNF formula, determine if there is an assignment a to the variables such that φ evaluates to 1 under a.

3SAT is the problem, where given φ a 3CNF formula, determine if there is an assignment a to the variables such that φ evaluates to 1 under a.

Theorem 6 (Cook-Levin). *SAT and 3SAT are NP-complete.*

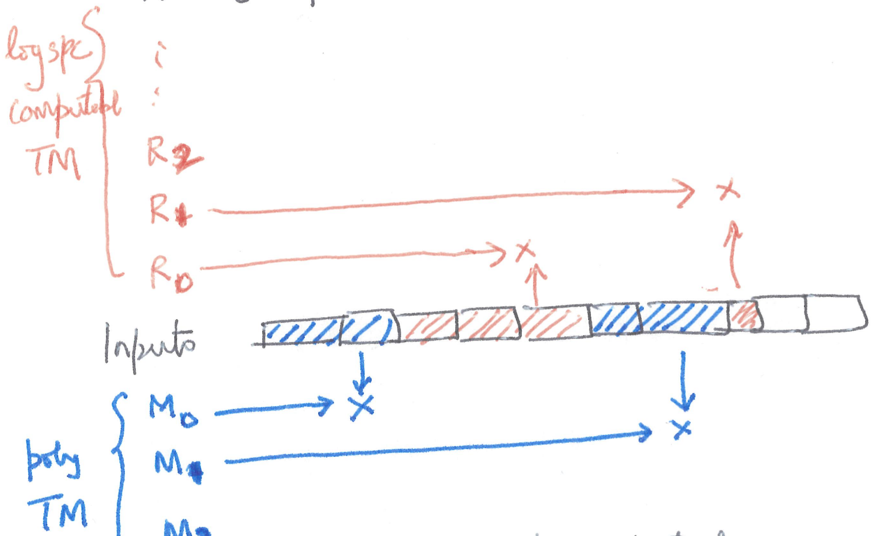
Theorem 7 (Ladner). *If* NP $\neq$ P *then there is a problem A such that $A \in$ NP $\setminus$ P and A is not NP-complete.*

$A \notin P$ and A is not NP-hard ($\Leftrightarrow$ SAT $\not\leq_m^{\log} A$)

A : merge SAT and ϕ

$f : \mathbb{N} \to \mathbb{N}$.

$A = \{ x \mid f(|x|) \text{ is even and } x \in \text{SAT} \}$



Defining f:

$f(0) = 0$

- if $f(n-1)$ is even ($2k$)
  - if $\exists x$ $|x| < n$ and $M_k(x) \neq$ ~~SAT~~ $A(x)$
    then $f(n) = 2k+1$
    else $f(n) = 2k$.
- if $f(n-1) = 2k-1$
  if $\exists x.$ $|R_k(x)| < n$.
  $A(R_k(x)) \neq \text{SAT}(x)$
  - then $f(n) = 2k$
  else $f(n) = 2k-1$

N_f — algo to compute f.

N_S — det algo for SAT.

N_A — algo for A (determined by f and N_S)

Algo N_f: Input 0^n (compute $f(n)$)

Phase I: $\left.\begin{array}{l}\text{Run } N_f \text{ on } 0 \\ \text{Run } N_f \text{ on } 1 \\ \vdots \end{array}\right\} \leq n$

Let i be the last value for which $f(i)$ was computed

And $f(i) = m$.

Phase II:

Case $m = 2k$:

For all x, $|x| < n$ $\left\{\begin{array}{l} \text{Run } N_A(x) \\ \text{Run } M_k(x) \end{array}\right\} \leq n$ steps

If witness found ($N_A(x) \neq M_k(x)$) then return $m+1$ else m.

Case $m = 2k+1$:

$\leq n$ steps $\left\{\begin{array}{l} \text{For } x, \\ \quad \text{Compute } R_k(x) \\ \quad N_S(x) \text{ and } N_A(R_k(x)) \end{array}\right.$

If $N_S(x) \neq N_A(R_k(x))$ then $m+1$

else m.

Need to show: $\lim_{n \to \infty} f(n) = \infty$