Michael Forbes
mifabes@illinois.edu

2018-01-22.4 → 2018-01-24 ← 2018-01-24.1

cs579

cs 579 : Computational Complexity : Lecture 3

admin : bugfix on ps1 #4

~~next week~~: I'm away

Robert: lecturing

office hours T2, R1, F2-4

today : NP

NP completeness

~~NP intermediate~~

Q : what is computing ?

decision - primality $\in P$ [AKS04]

search - factoring $\stackrel{?}{\in} P$

verification - multiplication $\in P$ ]

ex : integer $n$, has proper factorization $n = a \cdot b$? [ $a, b$ may not be prime ]

Q (P vs NP) : easy verification $\stackrel{?}{\Rightarrow}$ easy search ?

defn : NP is nondeterministic polynomial time [explain name here]

is those $L = \{ x : \exists y \in \{0,1\}^{f(n)} \text{ s.t. } (x,y) \in L' \}$

- $y$: witness
- $f(n)$: time constructible, $f(n) \leq n^{O(1)}$
- $L' \in P$ [ verification ]

defn : a nondeterministic TM is


$Q$ set of states

$\Gamma$ tape alphabet

$\sigma_0, \sigma_1$ transition functions

it computes : initial setup [ head, state, tape ]

applies $\sigma_0$ or $\sigma_1$ at each step [ guess ]

$L(N)$ = some choices of $\sigma_0/\sigma_1$ at each step reach $q_{acc} \Rightarrow x \in L$

$q_{acc} \Rightarrow x \notin L$ [ includes not halting ]

$NTIME(f(n)) = \{ L = L(N), N \text{ is NTM running in time } \leq O(f(n)) \}$

all branches reach $q_{acc}$, $q_{rej}$

$NSPACE$

Prop : $NP = NTIME(\text{poly}(n))$ [ original defn ] [ two years ]

pf : $\subseteq$ : $L = \{ x : \exists y \in \{0,1\}^{\text{poly}(n)} \ (x,y) \in L(M) \}$

$N =$ " on input $x$ :

1) guess $y$ [ nondeterministic guess ]

2) $N_{acc}$ iff $M_{acc}(x,y)$ " [ polynomial time verification ]

Michael Forbes
mmforbes@illinois.edu
2018-01-24.2 ← 2018-01-24.1
2018-01-24.2 → 2018-01-24.3
cs579

---

2: NTM $N$ ← length $\le$ time($N$) $\le$ poly($|x|$)

$L := \{(x,y)\}$: $y$ is a sequence of $\sigma_0, \sigma_1$ that makes $N$ accept $x$ }

↑ $\mathbb{P}$ // universal simulation ?

Prop: $TIME(f(n)) \subseteq NTIME(f(n)) \subseteq TIME(2^{O(f(n))})$

Pf: ignore $\sigma_1$

assume $f(n)$ time constructible // can also without, proof only slightly harder ?

$L \in NTIME(f(n)) \Rightarrow L = \{x : \exists y \in \{0,1\}^{f(n)} \ (x,y) \in L(M)\}$

$2^{f(n)}$ choices; $L(M)$ in $TIME(poly(|x|))$

$M' =$ { on input $x$:

1) for all $y$

run $M$ on $(x,y)$

2) if any ↗ accept, else reject

---

defn: $NEXP = NTIME(2^{poly(n)})$

$\Rightarrow$ $P \subseteq NP \subseteq EXP \subseteq NEXP$

$=?$ vs $\neq$

---

Thm [NTIME hierarchy ?] $f(n) << g(n)$ time constructible $\Rightarrow NTIME(f(n)) \subsetneq NTIME(g(n))$ ] $\Rightarrow NP \neq NEXP$

"Proof" $D$: "on input $\langle N, 1^k \rangle$:

1) simulate $N$ on $\langle N, 1^k \rangle$

2) accept $\langle N, 1^k \rangle$ if ↑ rejects"

problem: how to negate?

eg: $n$ not prime $\Rightarrow$ $n = a \cdot b$

prime $\Rightarrow$ ???

---

def: coNP is $L$ s.t. $L = \{0,1\}^* \setminus L'$, $L' \in NP$

$= \{x : \forall y \in \{0,1\}^{poly(|x|)} \ (x,y) \notin L(M)\}$ $\in P$

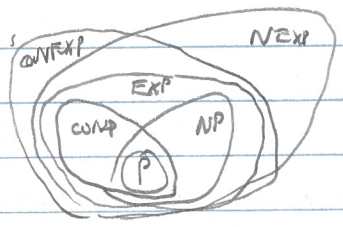
equiv: $(x,y) \in \overline{L(M)}$ ↑ $P$

equiv:

---

Q: NP vs coNP

= Question: the world




∃ resource counting view ?

∀ problem counting view ?

Michael Forbes
miforbes@illinois.edu

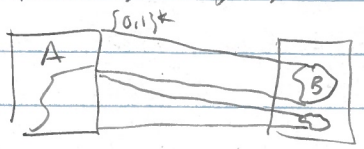
2018-01-242 ←
2018-01-244 →

2018-01-243
CS574

**def:** $A, B \subseteq \{0,1\}^*$. "$A$ is polynomial time reducible to $B$", $A \leq_p B$ if exists $f: \{0,1\}^* \to \{0,1\}^*$ computable in poly time s.t. ∥ use $B$ as way to solve $A$ ∥ easier than ∥

$x \in A \Rightarrow f(x) \in B$

$x \notin A \Rightarrow f(x) \notin B$




**lem:** $A \leq_p B$, $B \in P \Rightarrow A \in P$

**pf:** $|f(x)| \leq |x|^{O(1)}$ ⇒ "$f(x) \in B$" takes $|f(x)|^{O(1)} \leq |x|^{O(1)}$ time

**lem** $\in NP$ $\in NP$

$\in coNP$ $\in coNP$

**lem:** $A \leq_p B$ iff $\bar{A} \leq_p \bar{B}$ ∥ complement ∥

**lem:** $A \leq_p B$, $B \leq_p C \Rightarrow A \leq_p C$

**def:** $A$ is NP-hard if $\forall L \in NP$ $L \leq_p A$

-complete if " , $A \in NP$

**Prop:** $L$ NP-complete, $P = NP$ iff $L \in P$

**pf:** $\Rightarrow$: easy

$\Leftarrow$: $A \in NP \Rightarrow A \leq_p L \Rightarrow A \in P$.

**Prop:** coNP-NP iff $L \in coNP$ iff $\bar{L} \in NP$

**Q:** do NP-complete problems exist?

**def.** $BA = \{ \langle M, x, 1^w, 1^t \rangle : \exists y \in \{0,1\}^w$ $M$ accepts $(x,y)$ in $\leq t$ steps$\}$ $\in P$

**Prop:** BA is NP-complete

**pf:** $\in NP$:

NP-hard: $L = \{ x : \exists y \in \{0,1\}^{p(n)}$ $(x,y) \in L' \}$ ($L' \in P$)

$L \leq_p BA$

$M'$ acc $(x,y)$ in $\leq p(n) = n^{O(1)}$ steps

$x \mapsto \langle M', x, 1^{t(n)}, 1^{p(n)} \rangle$

time constructible $\leq \text{poly}(n)$

**Q:** natural NP-complete problems? ∥ artificial vs more vs natural ∥

**def:** a boolean formula is an expression comprised of

- literals: $x_i$ or $\neg x_i$ ∥ vars ∥ eg. $\neg((x \vee y) \wedge z) \vee \neg w$
- AND
- OR
- NOT

Michael Forbes
mforbes@illinois.edu

2018-01-24.4 ← 2018-01-24.3
CS 579 → 2018.

def: SAT = { $\langle \varphi \rangle$ : $\varphi$ is a boolean formula that is satisfiable }
exists $x \in \{0,1\}^n$ $\varphi(x) = 1$

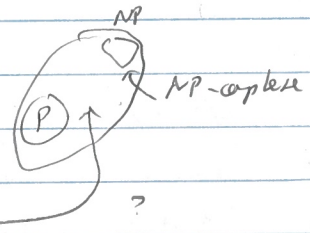
eg: $(x \vee y)(0,1) = 1$ satisfiable
$x \wedge \bar{x}$ unsat

def: a CNF (conjunctive normal form) formula is AND-OR-levels
kCNF AND-OR (≤ k levels)

eg: 3CNF: $(x \vee \neg y \vee z) \wedge (x \vee w \vee \neg x) \wedge (\cdots)$ ...

Thm [Cook Levin]: 3CNF-SAT is NP-complete [∈ NP, NP-hard later]

Thm [Karp ...]: 1000's of natural NP problems are NP-complete [do a hw reduce from 3SAT]




Q:

next time: NP-intermediate problems
space complexity

admin: Robert next week