DD2445 COMPLEXITY THEORY: LECTURE 9

Last time we started talking about

BOOLEAN CIRCUITS

- DAGs
- sources labelled by variables
- non-sources labelled $\wedge, \vee, \neg$
- value computed at unique sink: output of circuit
- size = # vertices

P/poly = languages decided by polynomial-size circuits

$$P \subsetneq P/poly$$

Don't necessarily know how to construct circuits, so there are small circuit families that decide undecidable problems (e.g., unary encoding of halting problem)

Equivalent definition of P/poly:
TMs that take advice — special, helpful input string α_n for inputs of length n

Does CNFSAT have polynomial-size circuits? I.e., can it be that $NP \subseteq P/poly$?

Not unless the polynomial hierarchy collapses

KL I

THEOREM [Karp & Lipton '80]
If $NP \subseteq P/poly$, then $PH = \Sigma_2^P$

Recall from previous lectures:

① THEOREM If $\Sigma_i^P = \Pi_i^P$, then $PH = \Sigma_i^P$

(We didn't prove this, but it could be a potential future pset problem)

② Complete problem for Σ_2^P (sets of variables)

$$\Sigma_2\text{-SAT} = \{\psi \text{ true} \mid \psi = \exists \vec{x} \forall \vec{y}\ \varphi(\vec{x}, \vec{y})\}$$

Complete problem for Π_2^P (just switching order of quantifiers)

$$\Pi_2\text{-SAT} = \{\psi \text{ true} \mid \psi = \forall \vec{y} \exists \vec{x}\ \varphi(\vec{x}, \vec{y})\}$$

(φ general propositional logic formula, say)

(Proving that these problems are indeed complete could also be a future pset problem)

Proof of Karp-Lipton theorem

Sufficient to show that $NP \subseteq P/poly$ implies $\Sigma_2^P = \Pi_2^P$ (by ①)

In fact, sufficient to show

$\Pi_2^P \subseteq \Sigma_2^P$. Why?

If $L \in \mathcal{C}$, then $\overline{L} \in co\mathcal{C}$ (by def).

So start with $L \in \Sigma_2^P$; take complements to get $\overline{L} \in \Pi_2^P$; use $\Pi_2^P \subseteq \Sigma_2^P$ to get $\overline{L} \in \Sigma_2^P$; and take complements again

How to show cplx class containment $\mathcal{C}_1 \subseteq \mathcal{C}_2$? Take complete problem L for $\mathcal{C}_1$ and show $L \in \mathcal{C}_2$

(if $\mathcal{C}_1$ has complete problems, which happily is the case here)

$L = \Pi_2\text{-SAT} = \{\psi \mid \psi = \forall \vec{y} \exists \vec{x} \varphi(\vec{x}, \vec{y}); \psi \text{ true}\}$

Show that assuming $NP \subseteq P/poly$ it holds that $\Pi_2\text{-SAT} \in \Sigma_2^P$

(can choose CNF to be)

i.e., exists poly-time TM M and poly-size witnesses $\vec{w}, \vec{z}$ such that

ψ true $\Leftrightarrow \exists \vec{w} \forall \vec{z}\ M(\psi, \vec{w}, \vec{z}) = 1$ (*)

$\psi = \forall y \exists x\ \varphi(x,y)$ is true iff

there exists a function $S: \{0,1\}^* \to \{0,1\}^*$, namely the existential player strategy, such that for all y

$\varphi(S(y), y)$ is true

Actually, let's switch back to standard order and write

$\psi = \forall \vec{x}\ \exists \vec{y}\ \varphi(\vec{x}, \vec{y})$ from now on...

Two observations

(i) Getting closer to Σ_2^P-style statement:

$\exists$ strategy S $\forall$ assignments x $\varphi(x, S(x))$ true

(ii) Consider problem

$$\{\langle \psi = \forall x \exists y\, \varphi, \alpha \rangle \mid \text{Does there exist } \beta \text{ s.t. } \varphi(\alpha, \beta) = \text{true}\}$$

This is clearly a problem in NP

Since $NP \subseteq P/poly$ $\exists$ poly-size circuit family $\{D_n\}_{n \in \mathbb{N}^+}$ deciding this problem

From this, can build multi-output circuits $\{C_n\}_{n \in \mathbb{N}^+}$ that compute assignment $\beta = \beta(\alpha)$, and these circuits also have polynomial size

[Figuring out the details here is a useful exercise]

If C_n has poly size $g(n)$, then can be described with $\leq (g(n))^2$ bits (for every gate, specify type and inputs)

But how can we *find* such circuits C_n? We don't need to — make a lucky NP-guess and then verify!

Here is how TM M in (*) will work

(1) Let $n = |\langle \psi, \alpha \rangle|$ [in some encoding we have agreed upon]

(2) Let $\vec{w}$ be guessed description of C_n

(3) Let $\vec{z}$ be any "challenge assignment" α to x

(4) Run $M(\psi, \vec{w}, \vec{z})$ to do the following

a) Check that $\vec{w}$ describes circuit C_n of correct type

b) Compute $C_n(\psi, \vec{z}) = \beta$

c) Accept iff $\varphi(\vec{z}, \beta)$ evaluates to true

Clearly, M can be made to run in polynomial time

Suppose $\psi = \forall x \exists y \varphi$ true. Then there is a circuit C_n that will compute valid response y for any x, so M accepts

Suppose ψ false. Then $\exists x$ for which it holds $\forall y$ that $\varphi(x, y)$ is false. For such inputs M will reject, regardless of what the guessed circuit computes.

Hence $\Pi_2\text{-SAT} \in \Sigma_2^P \Rightarrow \Sigma_2^P = \Pi_2^P \Rightarrow \Sigma_2^P = PH$, Q.E.D. ▨

KLV

If we believe in the polynomial hierarchy, then NP doesn't have poly-size circuits.

We can also prove that P/poly is unlikely to contain EXP

THEOREM [Meyer's theorem; from [KC80]]
If $EXP \subseteq P/poly$, then $EXP = \Sigma_2^P$

Proof (sketch) Let $L \in EXP$. Then $\exists$ TM M deciding L that

- runs in time $2^{p(n)}$ for some polynomial p
- is <u>oblivious</u> - head movements on input x only depends on length $|x|$

As in proof of Cook-Levin thm, can encode TM state at time i by snapshot Z_i:

- TM state (value of program counter)
- symbols read on all tapes

Given this snapshot information we know

- where TM jumps next
- which symbols written on tapes
- where tape heads move

Given M and x, can compute in exponential time (by simulating and observing M):

(a) ith snapshot Z_i
(b) positions $p_{i,1}, \ldots, p_{i,k}$ on k tapes
(c) last time steps $t_{i,1}, \ldots, t_{i,k}$ when tape heads were at the same positions

Given this information, can verify correctness at step i assuming steps $i' < i$ verified

- Compute time steps $t_{i,1}, \ldots, t_{i,k}$
- Compute snapshots $Z_i, Z_{t_{i,1}}, \ldots, Z_{t_{i,k}}$
- Check local consistency
- For $i = 1$, check that Z_1 correct starting state
- For $i = 2^{p(n)}$, check that $Z_{2^{p(n)}}$ is accept state

Now if $EXP \subseteq P/poly$, then $\exists$ poly-size circuits that on input i compute info in (a) - (c) [requires same kind of reduction from search to decision as in proof of Karp-Lipton]

We can guess such a circuit, but have to be a bit careful when verifying that guess is correct

Here is the set-up for the Σ_2^P-verifier | KL VII

(1) $\exists$-step Let C_n be guess for poly-size circuit that computes info (a) - (c)

(2) $\forall$-step Let $i', i'' < i$ be two arbitrary time steps

TM M uses C_n simulated on i to compute
- last time steps $t_{i,1}, \ldots, t_{i,k}$ for tapes
- snapshots $z_i, z_{t_{i,1}}, \ldots, z_{t_{i,k}}$

and checks local consistency
It also uses C_n on i and i' to compute $p_{i,1}, \ldots, p_{i,k}$ (current positions) and $p_{i',1}, \ldots, p_{i',k}$ (positions at time i')
and then checks that if $p_{i',\ell} = p_{i,\ell}$, then $t_{i',\ell} \geq p_{i,\ell}$ [later overwrites are not ignored]
Finally, for every $\ell = 1, \ldots, k$ M checks that $p_{i,\ell} = p_{t_{i,\ell},\ell}$ [the tape head was at the given position when claimed]

For the special cases $i = 1$ and $i = 2^{p(n)}$ we check that z_1 is a correct initial state and that $z_{2^{p(n)}}$ is an accepting state.

We just saw two highly nontrivial theorems.

But the actual proofs used completely elementary math! There was nothing fancy going on.

WHAT IS THE MORAL OF THIS STORY?

Complexity theory is deep and conceptually hard stuff (at least for me)

Suggested approach to survive the course:

(1) Skim chapter before lecture
Try to get a sense of what it is about, but don't get stuck — skip details you don't understand

(2) Attend the lecture (if it is helpful for you)

(3) Afterwards (and ASAP), read lecture notes and textbook chapter again, and more carefully

Can we prove any unconditional circuit lower bounds? Yes. Some functions must require very large circuits (worst-case hard)

THM 19 Shannon

For every $n \in \mathbb{N}^+$ there is a function $f: \{0,1\}^n \to \{0,1\}$ that cannot be computed by any circuit of size $2^n/(10n)$

Proof By counting.

functions from $\{0,1\}^n$ to $\{0,1\}$

2^n possible inputs; 2 choices for every input

2^{2^n} functions

How many circuits of size S?

Each gate can be

- $\wedge, \vee, \neg$, input — 4 choices
- for input specify which of $n < S$ variables
- for $\neg$ (not) specify input gate $\leq S$ choices
- for $\wedge, \vee$ specify two inputs $\leq S^2$

Total # circuits sth like (don't need to count very carefully):

$$(4S^2)^S = 2^{S \log(4S^2)} \overset{\text{base-change}}{\leq} 2^{9 S \log S}$$

Set $S = 2^n/(10n)$. Then # circuits $\leq$

$$2^{9 \cdot \frac{2^n}{10n} \log\left(\frac{2^n}{10n}\right)} \leq 2^{\frac{9}{10} 2^n \cdot \frac{1}{n} \log 2^n} = 2^{\frac{9}{10} 2^n}$$

$< 2^{2^n}$. Hence $\exists$ function $f: \{0,1\}^n \to \{0,1\}$ not computed by circuits of size $S = 2^n/(10n)$ ∎

A different take on Shannon's theorem

Pick function $f: \{0,1\}^n \to \{0,1\}$ at random by flipping a coin for every input x

For a fixed circuit C, probability that f and C agree on x is $= \frac{1}{2}$.

$\Pr[f \text{ and } C \text{ agree on every input}] = 2^{-2^n}$

$\Pr[\exists\, C \text{ which agrees with } f] = ?$

Union bound

For any events $A_1, A_2, \ldots, A_m$, probability that one of A_i happens is bounded by sum of probabilities

$$\Pr\left[\bigcup_{i=1}^{m} A_i\right] \leq \sum_{i=1}^{m} \Pr[A_i]$$

$\Pr[\exists \text{ Circuit } C \text{ agreeing with } f] \leq$

$$\sum_{\substack{\text{all circuits} \\ C_i}} \Pr[C_i \text{ agrees with } f] \leq$$

$$2^{\frac{9}{10}2^n} \cdot 2^{-2^n} = 2^{-\frac{1}{10}\cdot 2^n}$$

That is, vast majority of functions $f: \{0,1\}^n \to \{0,1\}$ are hard

Probabilistic method Prove that object $\mathcal{O}$ with property P exists by choosing $\mathcal{O}$ randomly and showing $\Pr[\mathcal{O} \text{ has property } P] > 0$

Since probability is strictly positive, such an element must exist! (Otherwise probability would be = 0)

In our case

$$\Pr[\text{randomly chosen } f \text{ hard}] \geq 1 - 2^{-\Omega(2^n)}$$

$\to 1$ as $n \to \infty$

Means that vast majority of random functions have this property

Counting argument can also be used to prove nonuniform time hierarchy theorem for circuits — see Thm 6.22 in textbook

SUMMING UP

Saw last time $P \subsetneq P/poly$

$P = NP$ iff NP has uniform poly-size circuits

Even dropping uniformity condition, believe $NP \not\subseteq P/poly$ (or PH collapses)

Most functions are hard (require exp size circuits)
What about lower bounds for explicit functions?
Ahem... SAD STORY... Will talk more about this later in the course.