Nonuniform complexity

\[ P \subseteq P/poly \]

poly-time TM
  * poly-size Boolean circuits.
  * poly-time TM with poly-size advice.

1. Boolean circuits
   1.1 Definition
   1.2 \( P \subseteq P/poly \)
   1.3 \( P \subseteq P/poly \) (\( P/poly \) contains TM-undecidable problems)

2. Advice
   2.1 Definition, equivalence
   2.2 \( \text{NP} \subseteq P/poly \Rightarrow \text{PH} = \Sigma_2 \)
   2.3 Existence of hard functions.
1. Boolean circuits

1.1. **def** Boolean circuit $C$ directed acyclic graph

vertices:
- input $x_1, \ldots, x_n \in \{0, 1\}$
- gates:
  - $\land, \lor$ fan in 2
  - $\neg$ fan in 1.
- output $C(x_1, \ldots, x_n)$

- one number of vertices, $|C|$
- depth length of longest path

**def** $T : \mathbb{N} \rightarrow \mathbb{N}$
- $T(n)$-size circuit family: sequence $(C_n)_{n \in \mathbb{N}}, |C_n| \leq T(n)$

$L \in \text{SIZE}(T(n))$ if $\exists T(n)$-size circuit family $(C_n)_n$
$\forall x \in \{0, 1\}^n \quad x \in L \iff C_n(x) = 1.$

$\text{P/poly} := \bigcup_{c \in \mathbb{N}} \text{SIZE}(n^c)$.
1.2 \text{ Th } \text{ P} \subseteq \text{ P}/\text{poly}.

\textbf{Sketch} Similar to proof of Cook-Levin result: SAT is \text{ NP-complete}. Let \mathcal{L} \in \text{ P}. Let \text{ M } \text{ a polytime TM deciding } \mathcal{L}. \text{ Wlog M is oblivious (head movements depend only on input length). Suppose k tapes. For } x \in \{0, 1\}^n \text{ snapshot of } \text{ M}(x) \text{ at time } i : z_i = \langle \text{state, symbols under head } \rangle

\text{encode as constant-size binary string.}

\text{transcript of } \text{ M}(x) : z_1, z_2, \ldots, z_{T(n)}

There is a constant-size circuit

\[ C(x, z_{i-1}, z_i, \ldots, z_{i+\varepsilon}) = z_i \]

\[ i_{\downarrow} = \text{previous time that head 1 was at the same position as it is at time } i \]

\[ \vdots \]

\text{Concatenate these circuits to get } z_{T(n)}. \text{ From this snapshot compute output } \text{ Out}. \quad \square
\[ P \subseteq P/poly \]

**Prop** Any \( L \subseteq \{ 1^n \mid n \in \mathbb{N} \} \) (unary language) is in \( P/poly \).

**Proof**

\[
\begin{align*}
\text{if } 1^n &\in L & C_n(x_1, \ldots, x_n) &= x_1 \land \cdots \land x_n. \\
\text{if } 1^n &\notin L & C_n(x_1, \ldots, x_n) &= 0 .
\end{align*}
\]

**Th** \( P \not\subseteq P/poly \)

**Proof** \( \text{UHALT} = \{ 1^n \mid \text{len } = \langle M, x \rangle \text{ s.t. } M(x) \text{ halts} \} \in P/poly \) but undecidable. \( \square \)
2. Advice.

2.1. Alternative definition: advice.

\[ T : \mathbb{N} \rightarrow \mathbb{N} \text{ time bound} \]
\[ a : \mathbb{N} \rightarrow \mathbb{N} \text{ advice size bound} \]

\( \mathcal{L} \in \text{DTIME}(T(n)/a(n)) \) if \( \exists \text{ } T(n) \text{-time TM } M \) and \( a(n) \in \text{poly} \) "advice" \( \forall x \in \text{poly} \)

\[ x \in \mathcal{L} \iff M(x, a(n)) = 1. \]

**Theorem (Equivalence)** \( \text{P/poly} = \bigcup_{c \in \mathbb{N}} \text{DTIME}(n^c)/a(n) \)

- **Boolean circuit advice**

**Proof** \( \subseteq \) \( \mathcal{L} \in \text{P/poly} \), so decidable by polytime \( C_n \).

Advice \( a(n) \) is description of \( C_n \). Define

\[ \text{TM } M(x, C_{1|x|}) : \text{simulate } C_{1|x|}(x). \]

2. Let \( \mathcal{L} \) decidable by polytime TM \( M \) with \( a(n) \) -size advice. There is a polytime circuit \( D_n \) such that

\[ \forall x \in \text{poly} \exists a \in \text{poly} \mathcal{L} \iff D_n(x, a) = M(x, a_n). \]

Hardwire \( a \) to get \( C_n(x) = D_n(x, a_n) \). \( \Box \)
If $NP \not\subseteq P/poly$, then $NP \neq P$. It is also possible that $NP \subseteq P/poly$ and $NP \neq P$. That would however collapse the PH.

**Th** $NP \subseteq P/poly \Rightarrow PH \subseteq \Sigma_2$ (Karp–Lipton)

**Proof** Suppose $NP \subseteq P/poly$. It suffices to show $\Pi_2 \subseteq \Sigma_2$. It suffices to show $\Pi_2 \text{SAT} \in \Sigma_2$ (since $\Pi_2$-complete)

\[
\Pi_2 \text{SAT} = \{ \phi(x_1, \ldots, x_n) : \forall \psi_0 \forall_1 \exists \psi_1 \forall \psi_1 \forall_1 \exists \psi_1 \phi(u, v) = 1 \}
\]

\[
\{ \langle \phi(x_1, \ldots, x_n), u \rangle : \exists \psi_0 \forall_1 \exists \psi_1 \forall \psi_1 \forall_1 \exists \psi_1 \phi(u, v) = 1 \}
\]

\[
\in \text{NP} \subseteq P/poly
\]

exists poly p, poly(n)-sized $C_n$ $\forall \phi(x_1, \ldots, x_n) \forall \psi_0 \forall_1 \exists \psi_1 \forall \psi_1 \forall_1 \exists \psi_1$ $C_n(\phi, u) = 1 \iff \exists \psi_0 \forall_1 \exists \psi_1 \forall \psi_1 \forall_1 \exists \psi_1 \phi(u, v) = 1$

Binary search to find a $v$ if it exists.

\[
C_n(\phi|v_1=0, u) = 1 \quad C_n(\phi|v_1=0, v_2=0, u) = 0 \quad \ldots
\]
\[
C_n(\phi|v_1=1, u) = 0 \quad C_n(\phi|v_1=0, v_2=1, u) = 1
\]

$v_1 = 0 \quad v_1 = 0, v_2 = 1$
Can be done with poly-size circuit $C'$. Encode $C'$ as we $\phi \in \text{poly}(n)$.

$$\phi(x_1, \ldots, x_n) \in \Pi_2 \text{SAT}$$

$$\iff \exists \phi \in \text{poly}(n) \forall u \in \{0,1\}^n \text{ describes circuit } C' \text{ and } \phi(u, C(\phi(u))) = 1$$

so $\Pi_2 \text{SAT} \in \Sigma_2$. \(\square\)

2.3 Existence of hard functions.

One can show that any $f: \{0,1\}^n \to \{0,1\}$ can be computed by a circuit of size $n(2^n+1)$.

Th (Shannon) \forall n \exists f: \{0,1\}^n \to \{0,1\} that cannot be computed by any circuit of size $\leq 2^n/5n$.

Proof

$$A_n := \{ f: \{0,1\}^n \to \{0,1\} \text{ has size } 2^{(2^n)} \}$$

$$B_n, s := \{ \text{circuits } C_n \text{ of size } s \} \text{ has size } \leq 2^{4s \log s}.$$
Set \( S = 2^n/n \). Then

\[
|B_{n,s}| = 2^{4 \frac{n \log S}{5n}} = 2^{\frac{2^n}{5n} \log \frac{2^n}{5n}} \leq 2^{\frac{2^n}{5n} n} = 2^{\frac{n}{5} 2^n} < 2^{(2^n)} = |A_n|.
\]

Proof of \( * \): \( |B_{n,s}| \leq 2^n 8 \log S \).

Let \( C_n \) circuit of size \( S \). Encode as adjacency list.

\[
\text{Vertex } 1 \quad \text{Vertex } 2
\]

\[
(\text{type, parent 1, parent 2}), (\text{type, parent 1, parent 2}), \ldots)
\]

\[\uparrow\]

\[A, V, T, \text{ input}\]

There are \( S \) vertices. Type takes 2 bits. Parent takes \( \log S \) bits. Adjacency list takes \( S (2 + 2 \log S) \) \leq 4 S \log S \) bits. There are \( \leq 2^{n \log S} \) possibilities. \( \square \).