Notes for Lecture 5

Today we give the definition of the polynomial hierarchy and prove two results about boolean circuits and randomized algorithms.

1 Polynomial hierarchy

Remark 1 (Definition of NP and *co*NP) *A problem is in* **NP** *if and only if there is a polynomial time computable* $F(\cdot,\cdot)$ *and a polynomial time bound* $p()$ *such that*

$$x \textit{ is a YES-instance} \Leftrightarrow \exists y.\ y \in \{0,1\}^{p(|x|)} \wedge F(x,y)$$

*co***NP** *is the class of problems whose complement (switch YES-instance to NO-instance) is in* **NP**. *Formally, a problem is in co***NP** *if and only if there is a polynomial time computable* $F(\cdot,\cdot)$ *and a polynomial time bound* $p()$ *such that*

$$x \textit{ is a YES-instance} \Leftrightarrow \forall y : y \in \{0,1\}^{p(|x|)}, F(x,y)$$

The polynomial hierarchy starts with familiar classes on level one: $\Sigma_1 = \mathbf{NP}$ and $\Pi_1 = co\mathbf{NP}$. For all $i \geq 1$, it includes two classes, Σ_i and Π_i, which are defined as follows:

Definition 2 Σ_k *is the class of all problems such that there is a polynomial time computable* $F(\cdot,...,\cdot)$ *and* k *polynomials* $p_1(),...,p_k()$ *such that*

$$x \textit{ is a YES-instance} \Leftrightarrow$$

$$\exists y_1 \in \{0,1\}^{p_1(|x|)}.\forall y_2 \in \{0,1\}^{p_2(|x|)}.\ \ldots$$
$$\ldots \underset{k \textit{ is odd/even}}{\forall/\exists} y_k \in \{0,1\}^{p_k(|x|)}.\ F(x,y_1,\ldots,y_k)$$

Definition 3 Π_k *is the class of all problems such that there is a polynomial time computable* $F(\cdot,...,\cdot)$ *and* k *polynomials* $p_1(),...,p_k()$ *such that*

$$x \textit{ is a YES-instance} \Leftrightarrow$$

$$\forall y_1 \in \{0,1\}^{p_1(|x|)}.\exists y_2 \in \{0,1\}^{p_2(|x|)}.\ \ldots$$
$$\ldots \underset{k \textit{ is odd/even}}{\forall/\exists} y_k \in \{0,1\}^{p_k(|x|)}.\ F(x,y_1,\ldots,y_k)$$

One thing that is easy to see is that $\Pi_k = \mathrm{co}\Sigma_k$. Also, note that, for all $i \leq k-1$, $\Pi_i \subseteq \Sigma_k$ and $\Sigma_i \subseteq \Sigma_k$. These subset relations hold for Π_k as well. This can be seen by noticing that the predicates F do not need to "pay attention to" all of their arguments, and so can represent classes lower on the hierarchy which have a smaller number of them.

Exercise 1 $\forall k.\Sigma_k$ *has a complete problem.*

One thing that is easy to see is that $\Pi_k = \text{co}\Sigma_k$. Also, note that, for all $i \le k-1$, $\Pi_i \subseteq \Sigma_k$, $\Sigma_i \subseteq \Sigma_k$, $\Pi_i \subseteq \Pi_k$, $\Sigma_i \subseteq \Pi_k$. This can be seen by noticing that the predicates F do not need to "pay attention to" all of their arguments, and so a statement involving k quantifiers can "simulate" a statement using less than k quantifiers.

Theorem 4 *Suppose* $\Pi_k = \Sigma_k$. *Then* $\Pi_{k+1} = \Sigma_{k+1} = \Sigma_k$.

PROOF: For any language $L \in \Sigma_{k+1}$, we have that there exist polynomials $p_1, \dots, p_{k+1}$ and a polynomial time computable function F such that

$$x \in L \Leftrightarrow \exists y_1.\forall y_2. \dots Q_{k+1} y_{k+1}.F(x, y_1, \dots, y_{k+1}) = 1$$

where we did not explicitly stated the conditions $y_i \in \{0,1\}^{p_i(|x|)}$. Let us look at the right hand side of the equation. What is following $\exists y_1$ is a Π_k statement. Thus, there is a $L' \in \Pi_k$ such that

$$x \in L \Leftrightarrow \exists y_1 \in \{0,1\}^{p_1(|x|)}.(x, y_1) \in L'$$

Under the assumption that $\Pi_k = \Sigma_k$, we have $L' \in \Sigma_k$, which means that there are polynomials $p'_1, \dots, p'_k$ and a polynomial time computable F' such that

$$(x, y_1) \in L' \Leftrightarrow \exists z_1.\forall z_2. \dots Q_k z_k.F'((x, y_1), z_1, \dots, z_k) = 1$$

where we omitted the conditions $z_i \in \{0,1\}^{p'_i(|x|)}$. So now we can show that

$$\begin{aligned} x \in L &\Leftrightarrow \exists y_1.(x, y_1) \in L' \\ &\Leftrightarrow \exists y_1.(\exists z_1.\forall z_2. \dots Q_k z_k.F'((x, y_1), z_1, \dots, z_k) = 1) \\ &\Leftrightarrow \exists (y_1, z_1).\forall z_2. \dots .Q_k z_k.F''(x, (y_1, z_1), z_2, \dots, z_k) = 1) \end{aligned}$$

And so $L \in \Sigma_k$.

Now notice that if $\mathcal{C}_1$ and $\mathcal{C}_2$ are two complexity classes, then $\mathcal{C}_1 = \mathcal{C}_2$ implies $\text{co}\mathcal{C}_1 = \text{co}\mathcal{C}_2$. Thus, we have $\Pi_{k+1} = \text{co}\Sigma_{k+1} = \text{co}\Sigma_k = \Pi_k = \Sigma_k$. So we have $\Pi_{k+1} = \Sigma_{k+1} = \Sigma_k$. □

2 BPP $\subseteq \Sigma_2$

This result was first shown by Sipser and Gács. Lautemann gave a much simpler proof which we give below.

Lemma 5 *If L is in* **BPP** *then there is an algorithm A such that for every x,*

$$\mathop{\mathbb{P}}_r(A(x, r) = \textit{right answer}) \ge 1 - \frac{1}{3m},$$

where the number of random bits $|r| = m = |x|^{O(1)}$ and A runs in time $|x|^{O(1)}$.

Proof: Let $\hat{A}$ be a **BPP** algorithm for L. Then for every x,

$$\mathbb{P}_r(\hat{A}(x,r) = \text{wrong answer}) \leq \tfrac{1}{3},$$

and $\hat{A}$ uses $\hat{m}(n) = n^{o(1)}$ random bits where $n = |x|$.

Do $k(n)$ repetitions of $\hat{A}$ and accept if and only if at least $\dfrac{k(n)}{2}$ executions of $\hat{A}$ accept. Call the new algorithm A. Then A uses $k(n)\hat{m}(n)$ random bits and

$$\mathbb{P}_r(A(x,r) = \text{wrong answer}) \leq 2^{-ck(n)}.$$

We can then find $k(n)$ with $k(n) = \Theta(\log \hat{m}(n))$ such that $\frac{1}{2^{ck(n)}} \leq \frac{1}{3k(n)\hat{m(n)}}$. □

Theorem 6 **BPP** $\subseteq \Sigma_2$.

Proof: Let L be in **BPP** and A as in the claim. Then we want to show that

$$x \in L \iff \exists y_1, \ldots, y_m \in \{0,1\}^m \forall z \in \{0,1\}^m \bigvee_{i=1}^{m} A(x, y_i \oplus z) = 1$$

where m is the number of random bits used by A on input x.
Suppose $x \in L$. Then

$$\begin{aligned}
&\mathbb{P}_{y_1,\ldots,y_m}(\exists z A(x, y_1 \oplus z) = \cdots = A(x, y_m \oplus z) = 0) \\
&\leq \sum_{z \in \{0,1\}^m} \mathbb{P}_{y_1,\ldots,y_m}(A(x, y_1 \oplus z) = \cdots = A(x, y_m \oplus z) = 0) \\
&\leq 2^m \frac{1}{(3m)^m} \\
&< 1.
\end{aligned}$$

So

$$\begin{aligned}
\mathbb{P}_{y_1,\ldots,y_m}\left(\forall z \bigvee_i A(x, y_i \oplus z)\right) &= 1 - \mathbb{P}_{y_1,\ldots,y_m}(\exists z A(x, y_1 \oplus z) = \cdots = A(x, y_m \oplus z) = 0) \\
&> 0.
\end{aligned}$$

So a sequence $(y_1, \ldots, y_m)$ exists, such that $\forall z. \bigvee_i A(x, y_i \oplus z) = 1$.
Conversely suppose $x \notin L$. Then fix a sequence $(y_1, \ldots, y_m)$. We have

$$\begin{aligned}
\mathbb{P}_z\left(\bigvee_i A(x, y_i \oplus z)\right) &\leq \sum_i \mathbb{P}_z(A(x, y_i \oplus z) = 1) \\
&\leq m \cdot \frac{1}{3m} \\
&= \frac{1}{3}.
\end{aligned}$$

So

$$\mathbb{P}_z(A(x, y_1 \oplus z) = \cdots = A(x, y_m \oplus z) = 0) = \mathbb{P}_z\left(\bigvee_i A(x, y_i \oplus z) = 0\right)$$
$$\geq \frac{2}{3}$$
$$> 0.$$

So for all $y_1, \ldots, y_m \in \{0,1\}^m$ there is a z such that $\bigvee_i A(x, y_i \oplus z) = 0$. □

3 The Karp-Lipton Theorem

Theorem 7 (Karp-Lipton) *If* **NP** $\subseteq$ **SIZE**$(n^{O(1)})$ *then* $\Sigma_2 = \Pi_2$ *and therefore the polynomial hierarchy would collapse to its second level.*

Before proving the above theorem, we first show a result that contains some of the ideas in the proof of the Karp-Lipton theorem.

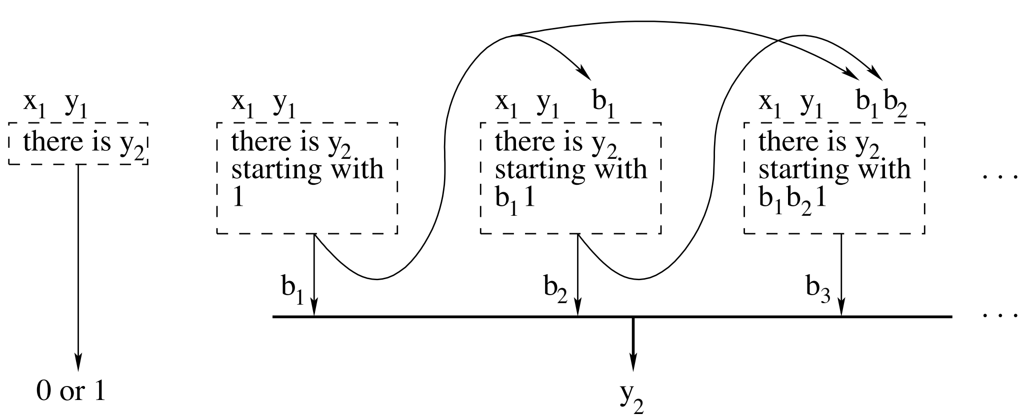
Lemma 8 *If* **NP** $\subseteq$ **SIZE**$(n^{O(1)})$ *then for every polynomial time computable* $F(\cdot,\cdot)$ *and every polynomial* $p(\cdot)$*, there is a family of polynomial size circuits such that*

$$C_{|x|}(x) = \begin{cases} y : F(x,y) = 1 & \textit{if such a y exists} \\ \textit{a sequence of zeroes} & \textit{if otherwise} \end{cases}$$

PROOF: We define the circuits $C_n^1, \ldots, C_n^{p(n)}$ as follows:

C_n^i, on input x and bits $b_1, \ldots, b_{i-1}$, outputs 1 if and only if there is a satisfying assignment for $F(x,y) = 1$ where $y_1 = b_1, \ldots, y_{i-1} = b_{i-1}, y_i = 1$.

Also, each circuit realizes an **NP** computation, and so it can be built of polynomial size. Consider now the sequence $b_1 = C_n^1(x)$, $b_2 = C_n^2(b_1, x)$, …, $b_{p(n)} = C_n^{p(n)}(b_1, \ldots, b_{p(n)-1}, x)$, as shown in the following picture:



The reader should be able to convince himself that this is a satisfying assignment for $F(x,y) = 1$ if it is satisfiable, and a sequence of zeroes otherwise. □

We now prove the Karp-Lipton theorem.

Proof: [Of Theorem 7] We will show that if **NP** $\subseteq$ **SIZE**$(n^{O(1)})$ then $\Pi_2 \subseteq \Sigma_2$. By a result in a previous lecture, this implies that $\forall k \geq 2\ \ \Sigma_k = \Sigma_2$.

Let $L \in \Pi_2$, then there is a polynomial $p(\cdot)$ and a polynomial-time computable $F(\cdot)$ such that

$$x \in L \leftrightarrow \forall y_1.|y_1| \leq p(|x|)\exists y_2.|y_2| \leq p(|x|).F(x, y_1, y_2) = 1$$

By using Lemma 8, we can show that, for every n, there is a circuit C_n of size polynomial in n such that for every x of length n and every y_1, $|y_1| \leq p(|x|)$,

$$\exists y_2.|y_2| \leq p(|x|) \land F(x, y_1, y_2) = 1 \text{ if and only if } F(x, y_1, C_n(x, y_1)) = 1$$

Let $q(n)$ be a polynomial upper bound to the size of C_n.

So now we have that for inputs x of length n,

$$x \in L \leftrightarrow \exists C.|C| \leq q(n).\forall y_1.|y_1| \leq p(n).F(x, y_1, C(x, y_1)) = 1$$

which shows that L is in Σ_2. □