CS 6810: Theory of Computing **Fall 2023**

Lecture 20: Oct 31, 2023

Lecturer: Eshan Chattopadhyay *Scribe: Yanyi Liu*

In this lecture, we will see connections between hard functions (with respect to non-uniform machines) and pseudorandom generators (PRG) (with respect to non-uniform machines). Finally, we will show that the existence of a "dream" PRG implies that $\mathsf{BPP} = \mathsf{P}$.

For any $n \in \mathbb{N}$, let $\mathcal{U}_n$ denote the uniform distribution over $\{0,1\}^n$.

1 Definitions

We start by introducing what it means for a function to be hard. Roughly speaking, if a function f is (S, ε)-hard, then no S-size circuit can compute f with probability $\geq 1/2 + \varepsilon$. We also consider worst-case hardness where we only require each circuit fails to compute f on some input.

Definition 1.1. *Let $f : \{0,1\}^n \to \{0,1\}$ be a function. We say that f is (S, ε)-hard if for every circuit C of size $\leq S$, it holds that*

$$\Pr[x \leftarrow \{0,1\}^n : C(x) = f(x)] < \frac{1}{2} + \varepsilon$$

We simply say that f is S-hard if the above probability is < 1.

We proceed to defining pseudorandom generators (PRG). Roughly speaking, a function g is a (S, ε)-PRG if no S-size circuit can distinguish between the output of PRG and the uniform distribution with advantage $\geq \varepsilon$.

Definition 1.2. *Let $g : \{0,1\}^{s(n,\varepsilon)} \to \{0,1\}^n$ be a function. We say that g is a (S, ε)-pseudorandom generator ((S, ε)-PRG) if for every circuit C of size $\leq S$, it holds that*

$$|\Pr[x \leftarrow \{0,1\}^{s(n,\varepsilon)} : C(g(x)) = 1] - \Pr[r \leftarrow \{0,1\}^n : C(r) = 1]| < \varepsilon$$

Remark 1.3. *In the above definitions, we only consider functions defined over a specific input length. We can also consider functions $f = \{f_n\}_{n \in \mathbb{N}}$ defined over all input lengths, and we say that f is a $(S(\cdot), \varepsilon(\cdot))$-hard function (resp $(S(\cdot), \varepsilon(\cdot))$-PRG) if it is $(S(n), \varepsilon(n))$-hard (resp $(S(n), \varepsilon(n))$-pseudorandom) for all sufficiently large $n \in \mathbb{N}$.*

2 Hardness from Pseudorandomness

We will show that we can get a hard function from any PRG $g : \{0,1\}^{n-1} \to \{0,1\}^n$. We consider the function $f : \{0,1\}^n \to \{0,1\}$ defined as $f(x) = 1$ iff $\exists y \in \{0,1\}^{n-1}$, $x = g(y)$.

Lemma 2.1. *Assume that $g : \{0,1\}^{n-1} \to \{0,1\}^n$ be an $(S, 1/2 - \delta)$-PRG for some $\delta > 0$. It holds that f is S-hard.*

Proof. Assume for the sake of contradiction that f is not S-hard; i.e., there exists a circuit C of size S that computes the function f. We will show that the circuit C will distinguish between the output of g and the uniform distribution with advantage $\geq 1/2$, which contradicts to the $(S, 1/2 - \delta)$-pseudorandomness of g. Observe that $\Pr[C(g(\mathcal{U}_{n-1})) = 1] = 1$ since C computes f and

f will output 1 if the input is in the range of g. On the other hand, $\Pr[C(\mathcal{U}_n) = 1] \leq 1/2$ since the PRG g can output at most 2^{n-1} strings which can occupy at most a $1/2$ fraction of n-bit strings. Taken together, it follows that

$$|\Pr[C(g(\mathcal{U}_{n-1})) = 1] - \Pr[C(\mathcal{U}_n) = 1]| \geq 1/2$$

which concludes our proof. □

3 Pseudorandomness from Average-Case Hardness

We move on to show that we can obtain a PRG from average-case hard functions. For any function $f : \{0,1\}^n \to \{0,1\}$, define $g : \{0,1\}^n \to \{0,1\}^{n+1}$ as

$$g(x) = (x, f(x))$$

where g outputs x concatenated with $f(x)$.

We turn to proving that g is indeed a PRG. The proof uses essentially the same idea as in Yao's indistinguishibility vs. unpredictability Theorem.

Lemma 3.1. *Assume that f is (S, ε)-hard. It holds that g is a $(S-1, \varepsilon)$-PRG.*

Proof. Suppose for contradiction that there exists circuit C' of size $\leq S-1$ such that

$$|\Pr[C'(g(\mathcal{U}_n)) = 1] - \Pr[C'(\mathcal{U}_{n+1}) = 1]| \geq \varepsilon$$

It follows that there exists a circuit $C \in \{C', C' \oplus 1\}$ such that

$$\Pr[C(g(\mathcal{U}_n)) = 1] - \Pr[C(\mathcal{U}_{n+1}) = 1] \geq \varepsilon$$

and we consider the circuit C.

We will show that the circuit C will output 1 with higher probability when the input is sampled from $(x, f(x))$, $x \leftarrow \mathcal{U}_n$ than $(x, f(x) \oplus 1)$, $x \leftarrow \mathcal{U}_n$. Observe that

$$\begin{aligned}
&\Pr[x \leftarrow \mathcal{U}_n : C(x, f(x)) = 1] - \Pr[x \leftarrow \mathcal{U}_n : C(x, f(x) \oplus 1) = 1] \\
&= \Pr[C(\mathcal{U}_n, f(\mathcal{U}_n)) = 1] - \Pr[C(\mathcal{U}_n, f(\mathcal{U}_n) \oplus 1) = 1] \\
&= 2\Pr[C(\mathcal{U}_n, f(\mathcal{U}_n)) = 1] - (\Pr[C(\mathcal{U}_n, f(\mathcal{U}_n)) = 1] + \Pr[C(\mathcal{U}_n, f(\mathcal{U}_n) \oplus 1) = 1]) \\
&= 2\Pr[C(g(\mathcal{U}_n)) = 1] - 2\Pr[C(\mathcal{U}_{n+1}) = 1] \\
&\geq 2\varepsilon
\end{aligned}$$

Therefore, we can use the circuit C to compute the function f. Consider the following randomized algorithm A: On input x, toss a random coin $b \leftarrow \{0,1\}$, and output b if $C(x, b) = 1$ (since b is more "likely" to be $f(x)$); otherwise output $b \oplus 1$. In other words, $A_b(x) = C(x, b) \oplus b \oplus 1$ where $b \leftarrow \{0,1\}$.

We proceed to showing that A computes f with probability $\frac{1}{2} + \varepsilon$. Note that

$$\begin{aligned}
&\Pr[x \leftarrow \mathcal{U}_n, b \leftarrow \{0,1\} : A_b(x) = f(x)] \\
&= \Pr[x \leftarrow \mathcal{U}_n, b \leftarrow \{0,1\} : b = f(x)] \Pr[x \leftarrow \mathcal{U}_n, b \leftarrow \{0,1\} : A_b(x) = f(x) \mid b = f(x)] \\
&\quad + \Pr[x \leftarrow \mathcal{U}_n, b \leftarrow \{0,1\} : b = f(x) \oplus 1] \Pr[x \leftarrow \mathcal{U}_n, b \leftarrow \{0,1\} : A_b(x) = f(x) \mid b = f(x) \oplus 1] \\
&= \frac{1}{2}\Pr[x \leftarrow \mathcal{U}_n : C(x, f(x)) = 1] + \frac{1}{2}\Pr[x \leftarrow \mathcal{U}_n : C(x, f(x) \oplus 1) = 0] \\
&= \frac{1}{2}\Pr[x \leftarrow \mathcal{U}_n : C(x, f(x)) = 1] + \frac{1}{2}(1 - \Pr[x \leftarrow \mathcal{U}_n : C(x, f(x) \oplus 1) = 1]) \\
&\geq \frac{1}{2} + \varepsilon
\end{aligned}$$

Finally, it remains to show that A can be implemented by a circuit of size S. Since A_b computes f with probability at least $\frac{1}{2} + \varepsilon$ over a random choice of $b \in \{0,1\}$, it follows that there exists $b_0 \in \{0,1\}$ such that A_{b_0} computes f with probability $\geq \frac{1}{2}+\varepsilon$. Recall that $A_{b_0}(x) = C(x, b_0) \oplus b_0 \oplus 1$, and notice that the operator $\oplus 1$ can be implemented by adding a NOT gate in the end of the circuit. It follows that A_{b_0} is just C' with the last input fixed to b_0, and with (or without) a NOT gate in the end (depending on the value of b_0 and which of $\{C', C' \oplus 1\}$ C is), where the circuit size is increased by at most 1. $\square$

4 Derandomization from PRGs

Finally, we show that $\mathsf{BPP} = \mathsf{P}$ if there exists a $(O(n), 1/6)$-PRG $g : \{0,1\}^{O(\log n)} \to \{0,1\}^n$ computable in time $\mathsf{poly}(n)$.

Lemma 4.1. *Assume that there exists a $(O(n), 1/6)$-PRG $g : \{0,1\}^{O(\log n)} \to \{0,1\}^n$ where g (on input of length $O(\log n)$) is computable in time $d(n) \in \mathsf{poly}(n)$. Then,* $\mathsf{BPP} = \mathsf{P}$.

Proof. For any $L \in \mathsf{BPP}$, let M be the poly-time probabilistic machine such that $M(x, r)$ decides L on instance x using random tape r. Let $t(n)$ denote the running time of M on instance $x \in \{0,1\}^n$, and we can without loss of generality assume that $|r| = t(|x|)$.

We will give a deterministic poly-time machine A such that A decides L. Roughly speaking, A will replace the random tape r of M by the output of g, and the average can now be computed using brute-force since the seed length to g is only logarithmic in its output length. Our machine A, on input $x \in \{0,1\}^n$, enumerates all possible $s \in \{0,1\}^{O(\log t(n))}$, and outputs the majority of $M(x, g(s))$ for all s (where $|g(s)| = t(|x|)$). Notice that A runs in time $2^{O(\log t(n))}(d(n) + t(n)) \in \mathsf{poly}(n)$ time.

We turn to arguing that for every $n \in \mathbb{N}$, $x \in \{0,1\}^n$, $A(x) = L(x)$. Consider the circuit $C(r)$ defined as $C(r) = M(x, r)$. Since g is a PRG, it follows that

$$|\Pr[s \leftarrow \{0,1\}^{O(\log t(n))} : C(g(s)) = 1] - \Pr[r \leftarrow \{0,1\}^{t(n)} : C(r) = 1]| < \frac{1}{6}$$

Therefore, it follows that $A(x)$ will output 1 if $\Pr_r[M(x,r) = 1] \geq \frac{2}{3}$, or output 0 if $\Pr_r[M(x,r) = 0] \geq \frac{2}{3}$ which concludes our proof. $\square$