CS761 Derandomization and Pseudorandomness **2022-23 Sem I**

Lecture 15: 03-10-2022

Scribe: Amit Rajaraman *Lecturer: Rohit Gurjar*

In the previous lecture, we had said the following (after defining what a PRG is).

Theorem 15.1. *If a $2^{\Omega(\ell)}$-PRG exists,* $\mathsf{BPP} = \mathsf{PP}$.

Note that a 2^ℓ-PRG does not exist. To see this, suppose instead that it did, and consider a circuit that is 1 precisely at each point in $\{0,1\}^{2^\ell}$ that is mapped to by the PRG, and is 0 everywhere else. Then,

$$\Pr_{r\sim D}[C(r) = 1] = 1 \text{ and } \Pr_{r\sim U_m}[C(r) = 1] = \frac{m}{2^m},$$

which are clearly not within 1/10 of each other.
While no PRGs are known that fool *all* circuits of size bounded by $m(\ell)^3$, there are PRGs known under more specific conditions on the circuit. For example, we can get a PRG that fools any randomized algorithm that is log-space.[1] It is also known that there exist (non-trivial) PRGs which fool constant-depth circuits.

Now, what are *circuit lower bounds*? We had remarked in the previous lecture that they imply the existence of PRGs.

Definition 15.2 (Worst-case hardness)**.** *For $f : \{0,1\}^n \to \{0,1\}$, its* worst-case hardness $H_{worst}(f)$ *is the largest number S such that for any circuit of size at most S, there exists some $x \in \{0,1\}^n$ such that $C(x) \neq f(x)$.*

We cannot compute the function using a circuit of size any smaller than its worst-case hardness. The implementation of the truth table yields that the worst-case hardness of any function is at most about $O(2^n)$.

Does there exist any function which is actually this hard? There are 2^{2^n} functions from $\{0,1\}^n \to \{0,1\}$, and there are (about) $S2^S$ circuits of size at most S. Consequently, some functions do require an S of at least about $2^n/n$.
However, no such function is explicitly known – this is another huge open question! In fact, the hardest explicit function we know has worst-case hardness just $3n - o(n)$.

Definition 15.3 (Average-case hardness)**.** *For $f : \{0,1\}^n \to \{0,1\}$, its* average-case hardness $H_{avg}(f)$ *is the largest number S such that for any circuit of size S,*

$$\Pr_{x\sim U_n}[C(x) \neq f(x)] > \frac{1}{2} + \frac{1}{S}.$$

Note that we can trivially get a circuit that is equal to f with probability $\geq 1/2$, setting it as either the constant 0 or the constant 1 (depending on which value f takes more often).
Clearly, the average-case hardness of any function is at least the worst-case hardness.

Theorem 15.4 (Nisan-Wigderson)**.** *If there exists a function computable in time $2^{O(n)}$ with $H_{avg}(f) \geq 2^{2n/3}$, then there exists a $(2^{\ell/45})$-PRG and in particular,* $\mathsf{BPP} = \mathsf{PP}$.

This links the worlds of algorithms (in the time complexity of f), circuits, and derandomization.

[1] This does not make sense in our current framework, but it is possible to modify the definition of PRGs appropriately. In this setting, we do not have exponential stretch, but we can go from $\Omega(\log^2 m)$ to m. The question of whether $\mathsf{RL} = \mathsf{L}$ is a huge open question.

To go from just ℓ to $\ell+1$, a logical idea is to use the hardness of the function to generate a new bit that is difficult to predict given all the previous bits.
That is, letting $f : \{0,1\}^\ell \to \{0,1\}$ be such that $H_{\text{avg}}(f) \geq \ell^4$, G defined by

$$G(r) = (r_1, \ldots, r_\ell, f(r)) = (r, f(r))$$

is a PRG.
The above merely says that unpredictability implies indistinguishability.

Theorem 15.5 (Yao)**.** *Let D be a distribution on $\{0,1\}^m$. Suppose that for any i and any circuit of size $2S$,*

$$\Pr_{y \sim D}[C(y_1, \ldots, y_i) = y_{i+1}] < \frac{1}{2} + \epsilon.$$

Then, for any circuit B of size S,

$$\left| \Pr_{y \sim D}[B(y) = 1] - \Pr_{y \sim U_m}[B(y) = 1] \right| < m\epsilon.$$