CS395T Pseudorandomness and Combinatorial Constructions
Lecture 19, March 27, 2001
Lecturer: Prof. David Zuckerman, Scribe: K. J. Balakrishnan

Last time we saw PRG's for space bounded computations. Today we will build PRG's from lower bounds. We will show the interesting direction of the following theorem.

Theorem 1 *(Nisan-Wigderson) Let $s(n)$ satisfy $m \leq s(m) \leq 2^m$ and for $c \geq 1; s(m)^c \leq s(m^c)$. Then the following are equivalent:*

1. $(\exists c > 0)$ **EXP** cannot be approximated by circuits of size $s(m^c)$
2. $(\exists c > 0)$ $\exists PRG\ G = \{G_m : \{0,1\}^m \rightarrow \{0,1\}^{s(m^c)}\}$ that runs in time $2^{O(m)}$ and fools circuits of size $s(m^c)$.

Proposition 1 *Suppose $\forall\ \epsilon,\ c > 0$, $\exists G = \{G_m : \{0,1\}^{m^\epsilon} \rightarrow \{0,1\}^{m^c}\}$ that fools poly-sized circuits and runs in exponential time. Then*

$$\mathbf{BPP} \subseteq \bigcap_{\epsilon>0} DTIME(2^{n^\epsilon}).$$

Proof: (Sketch) Run the **BPP** algorithm on the output of G. Cycle over all possibilites for the seed to make it deterministic. $\square$

Definition 1 *(Hard Functions) Let $f : \{0,1\}^n \rightarrow \{0,1\}$ be a function and C be a circuit. The bias of f with respect to C is defined as*

$$\textit{bias(C,f)} = |Pr_x[C(x) = f(x)] - Pr_x[C(x) \neq f(x)]|$$

(Think of f as having $\Pr[f(x) = 1] \approx 1/2$.)

Definition 2 *The hardness of a function f is defined as*

$$H_f(n) = \max\{S : \forall \textit{ circuits } C \textit{ of size} \leq S,\ \textit{bias(C,f)} \leq 1/S\}$$

1 PRG of Nisan-Wigderson

The Nisan-Wigderson **PRG** generates n bits from a seed of l bits in the following manner. The seed is first divided into n sets $S_1, S_2 \ldots S_n$ such that each $|S_i| = m < l$. Let $\mathbb{S}$ be such a division, $\mathbb{S} = (S_1, S_2 \ldots S_n)$. The generator is $f_{\mathbb{S}}(X) = f(X|_{S_1}) \circ f(X|_{S_2}) \circ \ldots f(X|_{S_n})$. We naturally require that the different S_i have very few elements in common. This leads to the following definition.

Definition 3 *$S_1, S_2, S_3..S_n$ is a (k,m)-design if $\forall\ i$ $|S_i| = m$ and $\forall\ i \neq j$ $|S_i \cap S_j| \leq k$*

We'll take $k = O(\log n)$. Given n, the number of sets and m, we want l, the size of domain of S_i, to be as small as possible. The naive limit is $l = mn$. Note that if $m = O(\log n)$, we can get $l = O(\log n)$ using a greedy algorithm.

The correctness of the **PRG** and the proof of the Theorem 1 are corollaries of the following lemma.

Lemma 1 *Let f satisfy $H_f(m) > n^2$, $\mathbb{S}$ is a $(\log n, m)$-design, $S_i \leq [l]$. Then $\forall$ circuits C of size n, $|Pr_Y[C(Y) = 1] - Pr_X[C(f_{\mathbb{S}}(X)) = 1]| < 1/n$.*

Proof:. Suppose $|Pr_Y[C(Y) = 1] - Pr_X[C(f_{\mathbb{S}}(X)) = 1]| \geq 1/n$. Let D_i be the distribution which takes the first i bits from $f_{\mathbb{S}}$ and the last $(n-i)$ bits from Y. Let $p_i = Pr[C(Z) = 1]$, $Z \in D_i$.
So $|p_0 - p_n| \geq 1/n$. Hence $\exists i$ such that $|p_i - p_{i-1}| \geq 1/n^2$.
Claim 1: Some probabilistic circuit of size n predicts the i^{th} bit of $f_{\mathbb{S}}$ from the first $i-1$ bits with $prob \geq 1/2 + 1/n^2$.
Proof: Say $p_i > p_{i-1}$. Let $w \in \{0,1\}^{i-1}$ and let $q_i = Pr[C(z) = 1]$, $z \oplus e_i \in D_i$. Therefore $(p_i + q_i)/2 = p_{i-1}$. Now run circuit C on the string wu, where $u \in_u \{0,1\}^{n-i+1}$. If the output is 1, predict the i^{th} bit as $(wu)_i = u_1$, otherwise predict the i^{th} bit as $(wu)_i = \bar{u_1}$. Now

$$\begin{aligned}
\Pr[\text{correct}] &= \Pr[u_1 \text{ correct}]\Pr[\text{predict correctly}|u_1 \text{ correct}] \\
&+ \Pr[u_1 \text{ incorrect}]\Pr[\text{predict correctly}|u_1 \text{ incorrect}] \\
&= 1/2p_i + 1/2(1-q_i) \\
&= 1/2 + (p_i - q_i)/2 \\
&= 1/2 + (p_i - p_{i-1}) \\
&\geq 1/2 + 1/n^2.
\end{aligned}$$

Claim 2: A deterministic circuit of the same size predicts as well.
Proof: Since the previous argument holds over all u, just take a u that does at least this well.

Now, let $r_j = f(X|_{S_j})$. Fix bits outside of S_i such that the circuit still predicts r_i from $r_{[i-1]}$. Since r_j, $j \neq i$, is a function on a maximum of $k = \log n$ bits of S_i, $\exists$ a circuit of size $\leq 2^{\log n} = n$ computing r_j from S_i. Insert these circuits for r_j, $j < i$, into the circuit predicting r_i from $r_{[i-1]}$. This gives a circuit predicting r_i from S_i of size $\leq (i-1)n + n \leq n^2$. This contradicts the hardness of f. $\square$