CS 6815: Lecture 21

Instructor: Eshan Chattopadhyay Scribes: Ziteng Sun, Cosmo Viola

November 8th

1 Recap

If $f : \{0,1\}^m \to \{0,1\}$ is (S,ϵ)-hard, then $(U_m, f(U_m))$ is (S,ϵ)-pseudorandom.

2 Pseudorandom Generators

2.1 Nisan-Widgerson (NW)-PRG

Definition 2.1 (Nisan-Widgerson (NW)-Designs). *$T_1, T_2, \dots, T_m \subseteq [r]$ is a (n,k)-design if*

1. *$|T_i| = n$*
2. *$|T_i \cap T_j| \leq k$ for all $i \neq j \in [m]$.*

The parameters of a NW-design are r, m, n, and k. The goal is to have as many sets as possible.

Remark 2.2. *Let $k = \frac{n}{1000}, r = 1000n$; then we can make m exponential in n. (Can be proved using probalistic method). We will look at an explicit NW construction with worse parameters in Section 2.2.*

Claim 2.3 (Nisan-Widgerson (NW)-Pseudorandom Generator). *Let $f : \{0,1\}^n \to \{0,1\}$ be a (S,ϵ)-hard function, with the seed z of length r. $T_1, T_2, \dots, T_m \subseteq [r]$ is a (n,k)- NW design. Define $z^i = z_{|T_i}$. Output: $W = f(z^1), f(z^2), \dots, f(z^m)$, with m the output length in bits. Then, W is (S', ϵ')-pseudorandom with $S' = S - 11m^2, \epsilon' = m\epsilon$.*

Proof. [Proof by contradiction.] Suppose W is not (S', ϵ')-pseudorandom. Then there is a distinguisher D of size at most S' such that

$$|\Pr[D(w) = 1] - \Pr[D(U_m)]| > \epsilon'$$

Then we define hybrids as following: suppose the r_i are fresh independent uniform bits. We define W_i as follows:

$$\begin{aligned}
W_0 :& f(z^1), f(z^2), \dots, f(z^m) \\
W_1 :& r_1, f(z^2), \dots, f(z^m) \\
W_i :& r_1, \dots, r_i, f(z^{i+1}), \dots, f(z^m) \\
W_{i+1} :& r_1, \dots, r_i, r_{i+1}, f(z^{i+2}), \dots, f(z^m) \\
W_n :& r_1, r_2, \dots, r_n
\end{aligned}$$

Then there must exist i such that

$$|\Pr[D(W_i) = 1] - \Pr[D(W_{i+1} = 1)]| > \frac{\epsilon'}{m}$$

Next we will try to come up with a circuit using D to break the hardness of f. Each W_i is a distribution on $\{0,1\}^m$, and two W_i, W_{i+1} differ only in the $(i+1)-$th bit. If we can distinguish W_0 from W_m, we should be able to distinguish bit $f(z^{i+1})$ from a uniform random bit using the triangle inquality.

Define algorithm $\mathcal{A}$ as following: Input $x \in \{0,1\}^n \sim U_n$, and b either from U_1, or $f(x)$.

1. Sample $r_1, r_2, \ldots, r_i$.
2. Sample length r seed z as follows: let the bits at locations of T_{i+1} be x, and sample uniformly at all other locations.
3. Compute: $f(z^j), j \geq i+2$.
4. Output: $D(r_1, r_2, \ldots, r_i, b, f(z^{i+2}), f(z^{i+3}), \ldots, f(z^m))$.

Then by previous assumption, over the randomness of $\mathcal{A}$, we have:

$$\left| \Pr_{x \sim U_n}[\mathcal{A}(x, f(x)) = 1] - \Pr_{x \sim U_n}[\mathcal{A}(x, U_1) = 1] \right| > \frac{\epsilon'}{m} = \epsilon$$

Since $\forall j > i+1$ (by a property of NW-designs), the intersection of T_j and T_{i+1} is at most k, so this is really a function of k variables. This means the total amount of space we used in this construction is at most $S' + m + r + m2^k$. By Remark 2.2, we know we can make $2^k = O(m)$. Hence we have:

$$S' + m + r + m2^k < S$$

A contradiction! Hence we know W is (S', ϵ')-pseudorandom. □

2.2 Explicit NW-designs

In this section, we will look at how to construct NW-designs explicitly using polynomials. Fix q. We will consider all univariate polynomials over $\mathbb{F}_q$ of degree at most k. Let $m \approx q^k$. For all polynomials p, let

$$T_p = \{(i, p(i)) : i \in \mathbb{F}_q\}.$$

Then $T_p \subset \mathbb{F}_q^2$ and we have:

$$|T_{p_1} \cap T_{p_2}| = \{i \in \mathbb{F}_q : p_1(i) - p_2(i) = 0\} \leq k.$$

The parameters of the design are:

1. $r = q^2 = n^2$
2. $m = q^k = n^k$
3. $n = q$

2.3 Trevisan's Extractor

Next we will see how this argument gives us an extractor for free, which is surprising, since extractors deal with statistical distance while these deal with circuits.

Define the extractor $Ext : \{0,1\}^n \times \{0,1\}^r \to \{0,1\}^m$ as follows: $X = f, U_r = z$. Given the set X, we can think of X as being the hard function $X : \{0,1\}^{\log n} \to \{0,1\}$ where $X(i)$ is the ith coordinate of X.

Let $T_1, T_2, \ldots, T_m \subseteq [r]$ $(r = d)$ be a $(\log n, k)$- NW design; note the similarity with the previous construction.

Let $z^i = z_{|T_i}$, $|T_i| = \log n$. Then the extractor is the coordinate of x indexed by z^i:

$$Ext(x,z) = x(z^1), \ldots, x(z^m)$$

If these are not statistically close, there is some statistical test D with:

$$|\mathbf{E}[D(Ext(x,z)] - \mathbf{E}[D(U_m)]| > \epsilon$$

Then, construct hybrids the same way as before:

$$\begin{aligned}
W_0 :& x(z^1), x(z^2), \ldots, x(z^m) \\
W_1 :& r_1, x(z^2), \ldots, x(z^m) \\
W_i :& r_1, \ldots, r_i, x(z^{i+1}), \ldots, x(z^m) \\
W_{i+1} :& r_1, \ldots, r_i, r_{i+1}, x(z^{i+2}), \ldots, x(z^m) \\
W_n :& r_1, r_2, \ldots, r_n
\end{aligned}$$

We know there is a distinguisher for W^0 and W^m with a certain ϵ and $|\mathbf{E}[D(W^0)] - \mathbf{E}[D(W^{i+1})] > \epsilon$, and this implies there exists an i such that

$$\left|\mathbf{E}[D(W^i)] - \mathbf{E}[D(W^{i+1})]\right| > \epsilon/m.$$

We will give the details of the algorithm construction in the next lecture. Here is a sketch of the idea behind the algorithm we will use:

1. Fix coordinates of z at locations of T_{i+1} to be z^{i+1}.

2. Randomly sample all other coordinates.

This algorithm can compute x on more than half of its coordinates. This is not enough, but we can come up with a contradiction if we start with, instead of x, an encoding of x using an error correcting code.