CS 6815: Pseudorandomness and Combinatorial Constructions **Fall 2022**

Lecture 13: October 4, 2022

Lecturer: Eshan Chattopadhyay *Scribe: Tomas Alvarez*

1 Introduction

Recall the following claim from the previous lecture:

Claim 1.1. *Suppose $f : \{0,1\}^n \to \{0,1\}$ is (S,ϵ)-hard. Then $G : \{0,1\}^n \to \{0,1\}^{n+1}$ defined as $G(x) = x \circ f(x)$ is an (S',ϵ') pseudorandom generator (PRG) with $S' = S - 1$ and $\epsilon' = \epsilon$.*

This shows that the assumption of a hard function f allows us to extend n bits to $n+1$ bits. The following idea from Nisan and Wigderson will show the construction of a much better PRG based on the same hardness assumption.

2 Nisan-Wigderson

Claim 2.1. *Suppose $f : \{0,1\}^n \to \{0,1\}$ is (S,ϵ)-hard. Then there is an (S',ϵ') PRG, $G : \{0,1\}^r \to \{0,1\}^m$ where $S' = S - O(m2^k)$ and $\epsilon' = m\epsilon$.*

Before we prove this claim we will need to define (n,k) designs which are an essential component of these PRG's. It will be the case that if we can pick better designs then we will get better PRG's.

Definition 2.2. *An (n,k) design is a set system $S_1, S_2, \cdots S_m \subseteq [r]$ such that $|S_i| = n$ and $\forall i, j$ where $i \neq j$ we have $|S_i \cap S_j| \leq k$ (small intersection) and the size of such a design is m.*

The amount of sets m, and intersection size k will depend on the choice n. Also, since the size of the sets is at least n we must have $r \geq n$. With this definition we can now provide a construction of the PRG's claimed in 2.1.

Claim 2.3. *Suppose $f : \{0,1\}^n \to \{0,1\}$ is (S,ϵ)-hard. Let $x \in \{0,1\}^r$ (a seed). For any $W \subseteq [r]$ let $x|_W$ denote the projection of x onto coordinates W. Let $z \in \{0,1\}^m$ be $G(x)$ where $z_i = f(x|_{S_i})$ and $S_1, \cdots, S_m$ form an (n,k) design. That is to say $G(x) = z = f(x|_{S_1}) \circ f(x|_{S_2}) \circ \cdots \circ f(x|_{S_m})$, the concatenation of f evaluated at x projected onto each of the sets S_i. Then G is an (S',ϵ') PRG with parameters described in Claim 2.1.*

Essentially we are using many evaluations of the hard function f to construct a PRG. The runtime of G will then be $m \cdot T(n)$ where $T(n)$ is the runtime of f ($f \in DTIME(T(n))$) plus the construction of an (n,k) design. For a dream PRG we would want m to be exponential in r and r polynomial in n. Is it reasonable to assume such sets exist? (We shall see a construction in the next class)

We now prove by way of contradiction that G is an (S',ϵ') PRG.

Proof. Suppose G is not an (S',ϵ') PRG. We will use the hybrid argument to show this is a contradiction. Consider the following sequence of hybrid distributions:

$$H_0 = f(x|_{S_1}) \circ f(x|_{S_2}) \circ \cdots \circ f(x|_{S_m})$$

$$H_1 = b_1 \circ f(x|_{S_2}) \circ \cdots \circ f(x|_{S_m})$$
$$H_2 = b_1 \circ b_2 \circ \cdots \circ f(x|_{S_m})$$
$$H_i = b_1 \circ b_2 \circ \cdots \circ b_i \circ f(x|_{S_{i+1}}) \circ \cdots \circ f(x|_{S_m})$$
$$\vdots$$
$$H_m = b_1 \circ b_2 \circ \cdots \circ b_m$$

where $x \sim U_r$ and $b_1, b_2, \cdots b_m \sim U_1$. For any i the distribution replaces the first i function values with uniform random bits. The hybrids go from from H_0 which is just the distribution of $G(x)$ until H_m which is the uniform distribution on m random bits. Due to our assumption that G is not an (S', ϵ') PRG, there must exist some distinguisher D that distinguishes between the output of G and the uniform distribution. Rephrasing this in terms of the hybrids, we have $|Pr[D(H_m) = 1] - Pr[D(H_0) = 1]| \geq \epsilon'$. We can manipulate this probability into the following telescoping sum

$$|Pr[D(H_m) = 1] - Pr[D(H_0) = 1]| = \left| \sum_{i=0}^{m-1} Pr[D(H_{i+1}) = 1] - Pr[D(H_i) = 1] \right|$$

By the triangle inequality we have

$$\left| \sum_{i=0}^{m-1} Pr[D(H_{i+1}) = 1] - Pr[D(H_i) = 1] \right| \leq \sum_{i=0}^{m-1} |Pr[D(H_{i+1}) = 1] - Pr[D(H_i) = 1]|$$

and thus $\epsilon' \leq \sum_{i=0}^{m-1} |Pr[D(H_{i+1}) = 1] - Pr[D(H_i) = 1]|$. Notice that it cannot be the case that all m terms in the summation are less than $\frac{\epsilon'}{m}$ or they would not sum to at least ϵ'. Therefore, there must be some i for which $|Pr[D(H_{i+1}) = 1] - Pr[D(H_i) = 1]| \geq \frac{\epsilon'}{m}$. Therefore, by assuming that there was a distinguisher for H_0 and H_m with ϵ', we necessarily have a distinguisher between at least one H_i and H_{i+1} with $\frac{\epsilon'}{m}$.

We will now consider a randomized algorithm $\mathcal{A}$ (which can eventually into a circuit). We will show that under conditions on $\frac{\epsilon'}{m}$, we can construct an ϵ distinguisher from $\mathcal{A}$ which will contradict the fact that f is (S, ϵ)-hard.

Algorithm 1

Require: x, b'
  $z \leftarrow 0^r$ (an r-length string of 0's)
  $j \leftarrow 1$
  for $j \leq r$ **do**
    if $j \in S_{i+1}$ **then** $z_j = x_j$
    else $z_j \sim U_1$
    end if
  end for
  Sample $b_1, b_2, \cdots, b_i \sim U_1$
  Output $D(b_1 \circ b_2 \circ \cdots \circ b_i \circ b' \circ f(z|_{S_{i+2}}) \circ f(z|_{S_{i+3}}) \circ \cdots, f(z|_{S_m}))$

For this algorithm $\mathcal{A}$, we are given a seed x and a bit b' and $\mathcal{A}$ has knowledge of both i (the value for which $|Pr[D(H_{i+1}) = 1] - Pr[D(H_i) = 1]| \geq \frac{\epsilon'}{m}$) and an (n, k) design. Importantly this bit b' is either chosen uniformly at random or $b' = f(x)$. If b' is a random bit then over all choices of random bits and values of x we see that $b_1 \circ b_2 \circ \cdots \circ b_i \circ b' \circ f(z|_{S_{i+2}}) \circ f(z|_{S_{i+3}}) \circ \cdots , f(z|_{S_m})$ is the distribution H_{i+1}. Similarly, over all choices of random bits and values of x, we see that if $b' = f(x)$ then the distribution is H_i. Since D is an $\frac{\epsilon'}{m}$ distinguisher this means that

$$\underset{b_1,\cdots,b_i,z|_{\overline{S_{i+1}}}}{Pr} \underset{x}{Pr}[\mathcal{A}(x, f(x)) = 1] - \underset{b_1,\cdots,b_i,z|_{\overline{S_{i+1}}}}{Pr} \underset{x}{Pr}[\mathcal{A}(x, b') = 1] \geq \frac{\epsilon'}{m}$$

Where $z|_{\overline{S_{i+1}}}$ is the set of uniformly sampled bits (the indices which are not in S_{i+1}). Using similar reasoning to the previous result with sums, we know that if the total probability of this is greater than $\frac{\epsilon'}{m}$, then there must exist some values $b_1, \cdots, b_i, Z_{\bar{S}_i}$, for which $\underset{x}{Pr}[\mathcal{A}(x, f(x)) = 1] - \underset{x}{Pr}[\mathcal{A}(x, b') = 1] \geq \frac{\epsilon'}{m}$. Therefore, this algorithm constructs an $\frac{\epsilon'}{m}$ distinguisher for f and if we set $\epsilon \geq \frac{\epsilon'}{m}$ then this an ϵ distinguisher. It remains to find S which is the size of a circuit made by derandomizing algorithm $\mathcal{A}$. Notice that the we must calculate D during the algorithm which means we need a circuit of size S'. We also need $r - n$ random bits as inputs to fully construct z and depending on i we might need at most another m bits for $b_1, b_2, \cdots$. Knowledge of i can also be counted to take $\log(m)$ bits. Finally, we make at most m function calls to f so this adds size $m \cdot c(f)$ size to the circuit where $c(f)$ is the circuit size needed to compute f. However, by the property of the (n, k) design, each set S_j has an intersection of size at most k with S_{i+1}. Therefore, computing $f(z|_{S_j})$ depends only on at most k bits. Therefore, we can include a lookup table in our circuit for these calculations which has size 2^k. Thus the total size of the circuit is $S' + O(m2^k)$. If we let $S' + O(m2^k) < S$ then we have constructed a size less than S circuit which ϵ approximates f. This contradicts the hardness assumption on f. Hence, G must be an $(S - O(m2^k), m\epsilon)$ PRG. $\square$

We also include the following lemma which relates PRG's to hard languages.

Lemma 2.4. *Suppose $L \subset \{0,1\}^*$ is in $DTIME(T(n))$ and is (S, ϵ)-hard. Then there exists a PRG $\{G_n\}_{n\geq 1}$ where $G_n : \{0,1\}^{r(n)} \to \{0,1\}^{m(n)}$ that is $(S - O(m(n) \cdot 2^{k(n)}), m(n) \cdot \epsilon)$ hard.*