6.842 Randomness and Computation	May 16, 2012

Lecture 26

Lecturer: Ronitt Rubinfeld *Scribe: Andrei Frimu*

This lecture discusses the relationship between pseudorandomness and hard functions. The main result, accompanied by proof, is Theorem 3.

First, let's review the definition of a useful construction from the previous lecture:

Definition 1 *A collection of sets $S_1, \ldots, S_m \subseteq [d] = \{1, \ldots, d\}$ is an (l, a)-**design** if*

- $\forall i,\ |S_i| = l$
- $\forall i \neq j,\ |S_i \cap S_j| \leq a.$

Note that if $a = 0$, then the sets $S_1, \ldots, S_l$ are all disjoint as they each have l elements, d is forced to be at least $m \cdot l$. For the purposes of this lecture, it is useful to think of m as being big and a relatively small.

We will use the following theorem, which we don't prove here:

Theorem 1 *For any constant γ, there exists an (l, a)-design with $a = \gamma \log m$, constructible in time $2^{O(d)}$ and such that $d = O(l^2/a)$.*

We now introduce another definition

Definition 2 *$f : \{0,1\}^l \to \{0,1\}$ is (t, α)-**average case hard** if for any nonuniform (circuit with advice) algorithm A running in time $t(l)$ the following inequality holds for large l:*

$$\Pr_{x,A}[A(x) = f(x)] < 1 - \alpha(l)$$

Note that x is of size l. We will use $\alpha(l) = 1 - \epsilon(l)$ for $\epsilon(l) \leq \frac{1}{t(l)}$, hence $1 - \alpha(l) \leq \frac{1}{2} + \epsilon(l) \leq \frac{1}{2} + \frac{1}{t(l)}$.

The following theorem allows us to extend by 1-bit:

Theorem 2 *If f is $(t, 1-\epsilon)$-average case hard, then $G(y) := y \circ f(y)$ is a (t, ϵ)-PRG.*

We want to stretch this. Our approach is to use the *Nisan-Wigderson generator*, which we present here.

Definition 3 (Nisan-Wigderson generator) *Given (l, a)-design $S_1, \ldots, S_m \subseteq [d]$, define $G : \{0,1\}^d \to \{0,1\}^m$ to be*

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

where $x|_{S_i}$ is the string of length $l = |S_i|$ obtained by selecting the bits of x indexed by S_i. For convenience, use the notation $f_i(x) := f(x|_{S_i})$. Note that the domain of each f_i is $\{0,1\}^l$.

The intuition behind this construction is that if the sets S_i were completely disjoint, then the strings $x|_{S_i}$ would be completely independent, since they would have no common bits, making G hard to predict. However, in this case, as we saw, $d \geq ml$.

What we hope is that by trading independence of the strings $x|_{S_i}$, by allowing a bit of overlap (bounded above by $|S_i \cap S_j| \leq a$), we can still achieve satisfactory unpredictability. The following theorem quantifies these ideas:

Theorem 3 (NW) *Assume that the following two conditions hold (to be used in the Nisan-Wigderson generator):*

- *there exists $f : \{0,1\}^l \to \{0,1\}$ such that $f \in E := DTIME(2^{O(l)})$ and*

$$f \text{ is } \left(t, \frac{1}{2} - \frac{1}{\epsilon(l)}\right) - averagecasehard$$

- *there exists an (l,a)-design $S_1, \ldots, S_m \subseteq [d]$ such that*

$$m = t(l)^{1/3} \qquad \textit{and} \qquad a = \frac{1}{3}\log t(l)$$

Then the Nisan-Wigderson generator G is a $\frac{1}{m}$-PRG against non-uniform time m.

Before we move on to the proof of theorem 3, we mention two interesting corollaries.

Corollary 4 *If $f \in E = DTIME(2^{O(l)})$ such that f is $\left(t, \frac{1}{2} - t\right)$-average case hard for*

$$t = 2^{\Omega(l)} \implies P = BPP$$
$$t = 2^{l^{\Omega(1)}} \implies \tilde{P} = BPP$$
$$t = l^{\omega(1)} \implies BPP \subseteq SUBEXP$$

Corollary 5 *There exists $(m, 1/m)$ PRG for depth d circuits of size m such that the PRG is computable in polynomial time.*

Now we present the proof of theorem 3:

Proof

Suppose the result is not true. Then there exists a next-bit predictor P such that

$$\Pr_{i,x}\left[P\Big(f_1(x) \circ f_2(x) \circ \cdots \circ f_{i-1}(x)\Big) = f_i(x)\right] \geq \frac{1}{2} + \frac{\epsilon}{m}. \tag{1}$$

Note that the circuit size of P is the sum of the runtime of the PRG, which is m and the size of the advice we gave P in the proof, which is $O(m)$, hence size$(P) = O(m)$.

Using a standard argument (seen before in other lectures), there exists i^* that achieves the expectation, in other words

$$\Pr_{\text{bits of } x \text{ in } S_{i^*},\ \text{bits of } x \text{ not in } S_{i^*}}\left[P\Big(f_1(x) \circ f_2(x) \circ \cdots \circ f_{i^*-1}(x)\Big) = f_{i^*}(x)\right] \geq \frac{1}{2} + \frac{\epsilon}{m}. \tag{2}$$

Note that this is just inequality (1) as before, rewritten for i^* and with the probability split over two sets.

Now using an averaging process, we see that there must exist a setting Z of the bits of x *not in* S_i which achieves (2). We change notation and use the variable y to denote the x's that has its bits *not in* S_i set according to the setting Z. Then (2) becomes

$$\Pr_{y}\left[P\Big(f_1(y) \circ f_2(y) \circ \cdots \circ f_{i^*-1}(y)\Big) = f_{i^*}(y)\right] \geq \frac{1}{2} + \frac{\epsilon}{m} \tag{3}$$

Note that in (3), in $f_{i^*}(y)$, the unset variables are those indexed by S_{i^*} and f_{i^*} depends on all these. However, on the left hand side of the equality inside the probability in (3), each f_j, $1 \leq j \leq i^*-1$ depends only on the unset variables index by $S_j \cap S_i$, for the other variable of y have been fixed according to the setting Z chose above.

Hence, each f_j depends on $|S_i \cap S_j| \leq a$ variables. The 2^a values can be encoded as advice, giving a total advice size of $m \cdot 2^a$. This relatively small size of the advice (for special m and a) is crucial in what follows.

Define $A(y) = P(f_1(y) \circ \cdots \circ f_{i^*-1}(y))$.

- predicts $f(y)$ with advantage at least $\frac{\epsilon}{m} \approx \frac{1}{m^2}$

- has circuit size $m \cdot 2^a + \text{size of}(P)$. The latter we saw to be $O(m)$. Since we picked a, m to satisfy $a = \frac{1}{3} \log t(l)$ and $m = t(l)^{\frac{1}{3}}$, we have that

$$\text{size}(A(y)) = m \cdot 2^a + O(m) = t(l)^{\frac{1}{3}} \cdot t(l)^{\frac{1}{3}} + O(t(l)^{\frac{1}{3}}) \ll t(l),$$

contradicting the first assumption of theorem 3. (the average case hardness assumption)

■