6.895 Randomness and Computation May 12, 2008

Lecture 25

Lecturer: Ronitt Rubinfeld *Scribe: Yoong Keok Lee*

Today, we will prove that if one-way permutations exist then pseudorandom generators exist. After that, we will look at some relationships between hardness and pseudorandomness. Before we begin, let us review some materials from previous lectures.

1 Review

Definition 1 (Pseudorandom generator (PRG)) *A function* $G : \{0,1\}^{l(n)} \to \{0,1\}^k$ *is a* (t, ϵ)-PRG *if*

1. $l(n) < n$
2. $G(U_{l(n)})$ *is* ϵ*-computation indistinguishable to* U_n *according to* $t(n)$ *non-uniform statistical test*

Definition 2 (One-way function (OWF)) *A function* f *is* one-way *if*

1. *for all input* x, $f(x)$ *is computable in polynomial time*
2. *for all probabilistic polynomial time (ppt) algorithm* A, $\Pr_{x,coins\ of\ A}[A(f(x) \in f^{-1}(f(x))]$ *is negligible*

Definition 3 (Hardcore bit (hcb)) *A function* $b : \{0,1\}^* \to \{0,1\}$ *is a* hcb *for OWF* f*, if for all ppt algorithm* A*, there exists negligible* ϵ *such that*

$$\Pr_{x \in \{0,1\}^l}[A(f(x)) = b(x)] \leq \frac{1}{2} + \epsilon(l)$$

Theorem 4 *Efficient PRG exists* $\iff$ *OWF exists*

Definition 5 (One-way permutation (OWP)) OWPs *are OWFs that are one-to-one and onto.*

Theorem 6 *Efficient PRG exists* $\iff$ *OWP exists*

Claim 7 *If function* $f : \{0,1\}^n \to \{0,1\}^{2n}$ *is a PRG, then* f *is a OWF*

Proof See last lecture ■

2 If OWP exists, then PRG exists

Theorem 8 *If OWP exists, then PRG exists*

Let x be an input. If function b is a hcb for OWP f, then the concatenation of their output $G(x) = f(x) \circ b(x)$ is a PRG (proven in last lecture). In the remaining of this section, we will show that:

1. Not only can we obtain one bit of stretch, we can also get polynomially long bits of stretch with a hcb.
2. If a OWP exists, we can construct a new OWP and its hcb.

Thus, if OWP exists, PRG exists. The result actually also holds for OWF although we will not see it today.

Theorem 9 *If function $f : \{0,1\}^l \to \{0,1\}^l$ is a OWP with efficiently computable hcb b, then $G(x) = b(f^{n-1}(x)) \circ b(f^{n-2}(x)) \circ \ldots b(f(x)) \circ b(x)$ is a PRG for all $n = \text{poly}(l)$.*

Proof We will prove the theorem by contraction. Suppose $G(x)$ is not a PRG, then $G(X)$ is not next-bit-unpredictable, i.e. $\exists$ ppt P such that

$$\Pr_{x,i}[P(b(f^{(n-1)}(x), b(f^{(n-2)}(x), \ldots, b(f^{(n-i+1)}(x)) = b(f^{(n-1)}(x))] - \frac{1}{2} \geq \frac{1}{n^k}$$

Let $y = f^{(n-i)}(x)$, i.e. $f^{(n-i+1)}(x) = f(y)$. Because $x \in_r U_l \Rightarrow y \in_r U_l$ (since f is a permutation),

$$\begin{aligned} & \Pr_{y,i}[P(b(f^{(i)}(x), \ldots, b(f(x)) = b(y)] - \frac{1}{2} \geq \frac{1}{n^k} \\ \Rightarrow \quad & \Pr_{y}[A(f(y)) = b(y)] - \frac{1}{2} \geq \frac{1}{n^k} \end{aligned}$$

where algorithm $A(z)$ does the following:

1. pick $i \in_r \{1, \ldots, n\}$
2. output $P(b(f^{i-2}(z)), b(f^{i-2}(z)), \ldots, b(z))$

Since b and f can be efficiently computed, A is computable in polynomial time, and so we arrive at a contradiction that b is not hardcore. ■

Theorem 10 (Goldreich Levin) *If function f is a OWF, then function $b(x,r) = \langle x, r \rangle$ is a hcb for OWF $f'(x,r) = (f(x), r)$, where $\langle x, r \rangle$ denotes the inner product.*

Sketch of Proof We will sketch the proof of a weaker statement — for OWP instead of OWF and also over the boolean space, i.e. function $f : \{0,1\}^n \to \{0,1\}^n$. We will prove by contradiction. Assuming $b(x,r)$ is not a hcb, then $\exists A$ such that

$$\Pr_{x,r}[A(f(x), r) = \langle x, r \rangle] > \frac{1}{2} + \epsilon$$

Let $h_x(r) \equiv A(f(x), r)$. Call an input x "good" if $\Pr_r[h_x(r) = \langle x, r \rangle] > \frac{1}{2} + \epsilon$. We know that at least $\frac{\epsilon}{2}$ fraction of all inputs are "good"; otherwise it leads to a contradiction (by using conditional probabilities): $\Pr_{x,r}[A(f(x), r) = \langle x, r \rangle] < \frac{\epsilon}{2} \cdot 1 + 1 \cdot (\frac{1}{2} + \frac{\epsilon}{2}) = \frac{1}{2} + \epsilon$. In other words, we know that for a non-negligible fraction of inputs, function $h_x(r)$ is close to a linear boolean function $\langle x, r \rangle$. The remaining plan is to obtain an inverse for the image of each of the "good" inputs.

From boolean analysis, we obtain the following Fourier decomposition: $h_x(r) = \sum_{S \subseteq [n]} \hat{h}(S) \chi_S(r)$. Recall that $\hat{h}(S) = \epsilon \Rightarrow \Pr_r[h_x(r) = \chi_S(r)] = \frac{1}{2} + \frac{\epsilon}{2}$. So for each image $f(x)$, applying the Goldreich-Levin-Kushilevitz-Mansour algorithm for learning heavy Fourier coefficients (we can query h_x without the need to know its inverse), we can get all bases $\chi_S(r)$ such that $\hat{h}(S) \geq \epsilon$. If x happens to be "good", we will be able to find its inverse by testing each candidate basis.

For a non-negligible fraction of the inputs, we can invert their image efficiently, therefore we arrive at contradiction that f is not one-way. ■

3 Hardness and Pseudorandomness

Definition 11 ((l,a)-design) *$S_1, \ldots, S_m \subseteq [d]$ is a (l,a)-*design *if*

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

Remark We want m to be big, d small, and a small.

Theorem 12 *$\exists (l,a)$-design with $a = \gamma \log m, b = O(l^2/a)$ constructible in time* $\text{poly}(m,d)$.

Proof Idea Greedy approach. ■

Proof of [: Weaker statement] Wlog let $d = l^2$ for prime l. Let $S_i = \{(j, q_i(j) | 1 \leq jl\}$ where $q_i(j)$ is a univariate polynomial (mod l) of deg $\leq a$. So, $\forall i, |S_i| = l$. And, $\forall i, j, |S_i \cap S_j| \leq a$. Hence, $m = l^{a+1}$ ■

Definition 13 *Function $f : \{0,1\}^l \to \{0,1\}$ is (t, α)-average case hard if $\forall$ non-uniform algorithm A time $t(l)$*

$$\Pr_x[A(x) = f(x)] < 1 - \alpha(l)$$

for large enough l.

Theorem 14 *If function f is $(t, \frac{1}{2} - \epsilon)$-average case hard then function $G(y) = y \circ f(y)$ is a (t, ϵ)-PRG.*

Theorem 15 (Nisan Wigderson) *If*

1. *$\exists f : \{0,1\}^l \to \{0,1\} \in \mathbf{E} = \mathbf{DTIME}(2^{O(l)})$ such that f is $(\frac{1}{2} - \frac{1}{t(l)})$-hard for non-uniform time t*
2. *$\exists (l,a)$-design with $S_1, \ldots, S_m \subseteq [d]$ where $m = t(l)^{1/3} a = \frac{1}{3} \log t(l)$*

then

$$G : \{0,1\}^d \to \{0,1\}^m$$
$$G(x) = f(x|_{s_1}) f(x|_{s_2}) \ldots f(x|_{s_m})$$

is $\frac{1}{m}$-PRG against non-uniform time m (where $x|_{s_i}$ denotes the indices of x given by set S_i)

Proof Next lecture ■

Corollary 16 *If $\mathbf{E}$ has $(t(l), \frac{1}{2} - \frac{1}{t(l)})$-average case hard function $f : \{0,1\}^l \to \{0,1\}$ for*

$$\begin{aligned} t(l) &= 2^{\Omega(l)} && \text{then } \mathbf{P} = \mathbf{BPP} \\ &= 2^{l^{\Omega(1)}} && \text{then } \mathbf{BPP} \subseteq \tilde{\mathbf{P}} \\ &= l^{\omega(1)} && \text{then } \mathbf{BPP} = \mathbf{SUBEXP} \end{aligned}$$