Lecture 21

Lecturer: Dr. Meera Sitharam Scribe: Hongyu Guo

Our main goal is to prove Nisan-Wigderson theorem(1988), which relates hardness to pseudorandomness.

Theorem 1 *(Nisan and Wigderson,1988)*
For every s, where $l \leq s(l) \leq 2^l$, the following are equivalent:
(1) For some $c > 0$, $\exists$ some function f_n in $EXPTIME$ that cannot be approximated by circuits of size $s(l^c)$.

(2) For some $c > 0$, $\exists$ a f_n in $EXPTIME$ with hardness $s(l^c)$.

(3) For some $c > 0$, $\exists$ a $DTIME(2^l)$ pseudorandom generator $G : l \longrightarrow s(l^c)$, such that $\forall$ circuit C of size $s(l^c) = n$,

$$|P[C(y) = 1] - P[C(G(x)) = 1]| \leq \frac{1}{s(l^c)}, \tag{21.1}$$

where y is uniformly distributed on $\{0,1\}^n$ and x is uniformly distributed on $\{0,1\}^l$.

Corollary
$RAC^0 \subseteq DSPACE(poly/log(n)) \subseteq DTIME(2^{poly/log(n)}) = DTIME(n^{poly/log(n)}) \subseteq$ constant depth polynomiao size circuits,
where RAC^0 is Randomized AC^0, or randomized constant depth polynomial size circuits. A circuit C in RAC^0 takes I as regular input and x as random inputs. If $I \in S$, then $C(I,x) = 1$ with probability $\geq \frac{1}{2} + \epsilon$ and if $I \notin S$, then $C(I,x) = 0$ with probability $\geq \frac{1}{2} + \epsilon$.

Note to find 1 bit that looks random to circuit of size $s(m^c)$ we can simply take the output of f. It is a problem to come up with lots of bits. To get a pseudo random generator from f to satisfy condition c1 we need the XOR Lemma. First we give some definitions.

Definition 1 *Given a Boolean function $f_n : \{0,1\}^n \longrightarrow \{0,1\}$, we say f_n is (γ, s) hard if for any circuit of size s*

$$|P[C(x) = f(x)] - \frac{1}{2}| < \frac{\gamma}{2}$$

$$|P[C(x) \neq f(x)]| \geq \frac{1}{2} - \frac{\gamma}{2}$$

where $0 < \gamma < 1$.

Definition 2 *We say that f cannot be approximated by circuit of size $s(n)$ if for some constant k, all large enough n and circuits C_n of size $s(n)$:*

$$Prob[C_n(x) \neq f(x)] > \frac{1}{n^k}$$

where x is chosen uniformly in $\{0,1\}$.

Definition 3 *Let $f : \{0,1\}^n \longrightarrow \{0,1\}$ be a Boolean function uniformly defined and let f_m be restrictions of f to strings of length m. The hardness of f at m $H_f(m)$ is the maximum integer h_m such that f is $(1/h_m, h_m)$ - hard.*

Notice this is pretty much the same as the hardness definition given in Erwin's talk.

Lemma 2 *(Hardness amplification, Yao's XOR Lemma)*
Let $f_1, ..., f_k$ be all (γ, s) hard. Then for any $\mu > 0$ the function $f(x_1, ..., x_k) = \sum_{i=1}^{k} f_i(x_i) mod 2$ is $(\gamma + \mu, \mu^2(1-\gamma)^2 s)$-hard, where x_i's are all strings.

Idea The output of G is a sequence of bits, where each bit is $f(x_i)$, where x_i is a small seed. We want to choose x_i's such that the x_i's are not highly correlated. Intuitively, choose x_i, x_j in the set such that $|x_i \cap x_j|$ is small.

Definition 4 *A collection $\{S_1, ..., S_n\}$ of sets where $S_i \subseteq \{1, ..., l\}$ is called a (k,m) design, if $\forall i$,*
(1) $|S_i| = m$, (each x_i does not have too many 1's)
(2) $\forall i \neq j |S_i \cap S_j| \leq k$.

Our pseudo random generator G takes a seed x of length l. The x_i's are chosen as a (k,m) design on the set of 1-bits of x. $G(x) = f(x_1)f(x_2)...f(x_n)$.
Recall $n = s(l^c)$ in the statement of the theorem.

Lemma 3 *Let m, n, l be integers. $f : \{0,1\}^m \longrightarrow \{0,1\}$. $H_f(m) \geq n^2$ and let A be a Boolean $n \times l$ matrix which is a $(logn, m)$ design. Then $G : \{0,1\}^l \longrightarrow \{0,1\}^n$ given as above is a pseudo random generator satisfying (21.1) in the statement of the main theorem.*