Lecture 24:

Hardness vs. Randomness

NW 1
Sp2014

A lemma for next time:

<u>def</u> $\mathcal{I} = \{I_1 \cdots I_m\} \subseteq [\ell]$ is (ℓ, n, d)-design $(\ell > n > d)$

if 1) $|I_j| = n \quad \forall j$

2) $|I_j \cap I_k| \le d \quad \forall j \ne k$

<u>Thm</u>. $\exists$ algorithm running in $2^{O(\ell)}$ time s.t. for $n \ge d$, $\ell > 20n^2/d$ which outputs (ℓ, n, d)-design s.t. $m = 2^{d/10}$

<u>Pf</u>. ..

Greedy - best parameters, use prob method to show can progress:

GreedyAlg: after have $I_1 \cdots I_\ell$ for $\ell < 2^{d/10}$

search all subsets to find I^* s.t. $|I^* \cap I_j| \le d \quad \forall j \in [\ell]$

runtime: $\text{poly}(m) \cdot 2^\ell$

Why doesn't it get stuck?

if pick I^* randomly: prob $x \in [\ell]$ gets chosen $= \frac{2n}{\ell}$ (and truncate later)

$E[|I^*|] = 2n \longrightarrow \Pr[|I^*| \ge n] \ge 0.9$

$E[|I^* \cap I_j|] = \frac{4n^2}{\ell} < \frac{d}{5} \longrightarrow \Pr[|I^* \cap I_j| \ge d] \le \frac{1}{2} \cdot 2^{-d/10}$ } chernoff

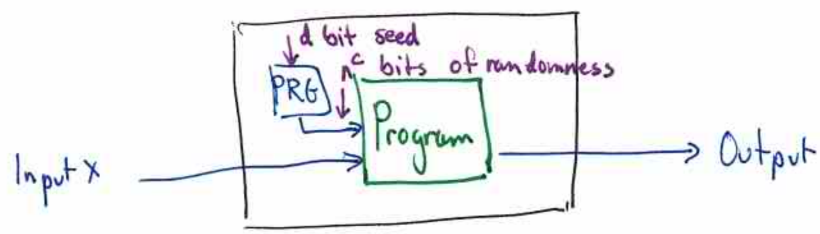
since $m < 2^{d/10}$, via union bnd, with prob ≥ 0.4

I^* will be good. ∎

Other constructions:

based on polynomials –
computable in parallel, small space, low sequential time

Recall our goal: Derandomizing algorithms



total time: $\underbrace{\#\text{seeds}}_{2^d} \times$ (run time of PRG + runtime of program)

- to derandomize all ptime algs, need PRG which takes $O(\log n)$ bits, outputs n^c bits which $\overset{c}{\equiv} U_{n^c}$ + runs in poly(n) time

- today: derandomize parallel algorithms. i.e., need PRG which outputs bits that look uniform to parallel algs

- more generally: hard on average fctns (on $\le S(n)$ size, get advantage $\le \frac{1}{S(n)}$)
  $\Rightarrow$ PRGs (stretch $S(\ell)^{\delta}$ s.t. ckts of size $\le S(\delta\ell)^{\Theta(\delta)}$ get adv $\le \frac{1}{10}$)

NW ③
Sp2014

<u>Def</u>. $f:\{0,1\}^\ell \to \{0,1\}$ is $(\overset{\downarrow \text{time}}{t}, \overset{\swarrow \text{error}}{\varepsilon})$ - average case hard

if $\forall$ nonuniform A in time $t(\ell)$

$\Pr_{x, \text{\$ of } A}[A(x) = f(x)]$ for large enough ℓ

$\le \frac{1}{2} + \varepsilon(\ell)$ pick $\varepsilon(\ell) < \frac{1}{t(\ell)}$

so

$\le \frac{1}{2} + \frac{1}{t(\ell)}$

f is t- ave case hard if for nonunif A in time $t(\ell)$, adv A is $\le \frac{1}{t(\ell)}$

<u>Thm</u> If $f:\{0,1\}^\ell \to \{0,1\}$ is (t, ε)- ave case hard
then $G(y) = y \circ f(y)$ is (t, ε)-PRG

note:
f <u>not</u> a permutation!
how do we extend >1 bit?

passing through

<u>Pf</u>. as for HCB ∎

How to stretch?

Define N-W generator...

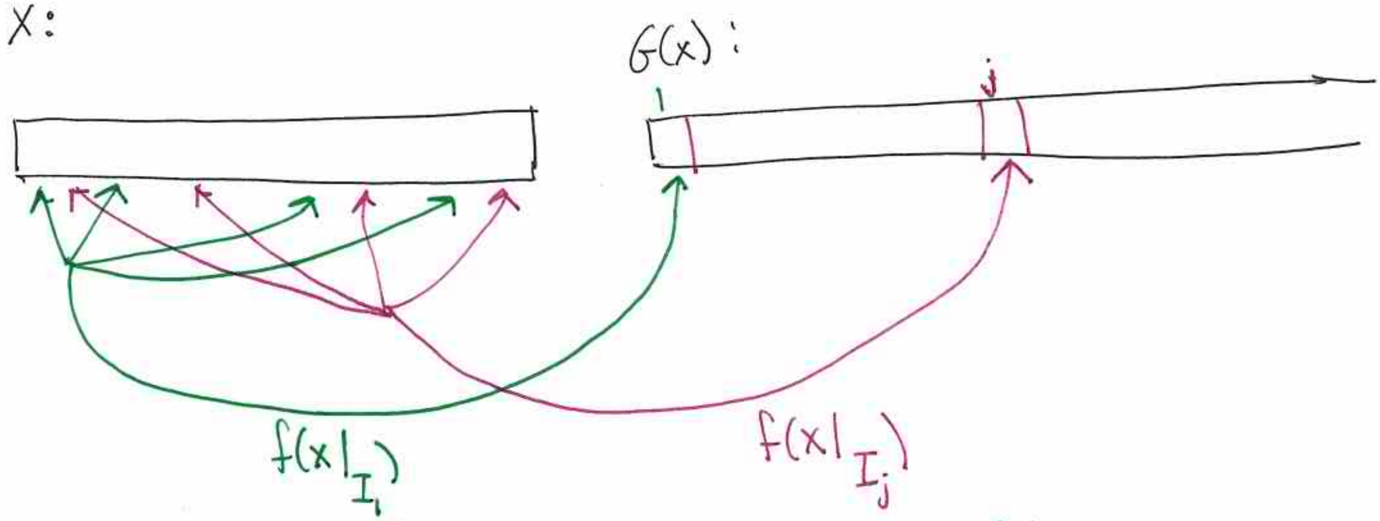
Def. Nisan–Wigderson generator

Given (ℓ, n, d)-design $\mathcal{I} = \{I_1 \dots I_m\} \subseteq [\ell]$

$G : \{0,1\}^\ell \to \{0,1\}^m$

is $G(x) = f(\underbrace{x|_{I_1}}_{\substack{\text{string of}\\ \text{length } n\\ \text{selecting bits}\\ \text{indexed by } I_i}}) \circ f(x|_{I_2}) \circ \dots \circ f(x|_{I_m}) \overset{\text{new notation}}{=} f_1(x) \circ \dots \circ f_m(x)$

where $f_i(x) = f(x|_{I_i})$



Thm [NW] If (1) $\exists f : \{0,1\}^n \to \{0,1\}$ s.t. $f \in E = DTIME(2^n)$

s.t. f is t-ave case hard

$\leq t(\ell)/2$

(2) $\exists$ (ℓ, n, d) design with m sets, + constructable in time $2^{O(\ell)}$

s.t. $m = 2^{d/10} = t(\ell)^{1/c}$, $\ell > 10n^2/d$, $n > d$

← e.g. $c=2$ works

then G is $\frac{\varepsilon}{m}$-PRG against nonuniform time m.

↑ can think of $\varepsilon = \frac{1}{10}$

Pf.

if G not $\frac{1}{m}$-PRG against time m,

$\exists$ n.b. predictor P st.

$$\Pr_{i,x}\left[P(f_1(x)\, f_2(x) \ldots f_{i-1}(x)) = f_i(x)\right] \geq \frac{1}{2} + \frac{\varepsilon}{m} \quad \& \ \text{time}(P) = \text{time}(T) + O(m)$$

time of PRG distinguisher from n.b. test which is $O(m)$

Plan: use this to approx f with $\frac{\varepsilon}{m}$ adv in $O(t(\ell))$ time to contradict f's hardness where $m \approx t^{1/3}$ i.e. $t \approx m^3$

As usual, averaging $\Rightarrow \exists\, i^*$ st. attain expectation

$\Rightarrow \exists$ $\underbrace{\text{choice of}}_{\text{call it } z}$ bits of X $\underbrace{\text{not in } I_{i^*}}_{\text{in } \overline{I}_{i^*}}$ attaining expectation

notation $Y \leftarrow X$ with bits in $\overline{I}_{i^*}$ set to this choice z & others picked randomly

so $$\Pr_Y\left[P(f_1(Y)\, f_2(Y) \ldots f_{i^*-1}(Y)) = f_{i^*}(Y)\right] \geq \frac{1}{2} + \frac{\varepsilon}{m}$$

properties of (ℓ, n, d)-design give this

each depends on $\leq d$ bits of Y since $|I_{i^*} \cap I_j| \leq d$

Since depend on $\leq d$ bits, and $f \in E$, can compute each f_j in time 2^d or with ckt that has encoded lookup table

not when trying to prove P=BPP

$A(y) = P\left(f_1(Y)\, f_2(Y) \ldots f_{i^*-1}(Y)\right)$

- predicts $f_{i^*}(Y)$ with adv $\geq \frac{\varepsilon}{m} = \frac{1}{10\, 2^{d/10}}$

time(P) time to construct design

- runtime $\underbrace{\tilde{O}(d 2^d)}_{\text{compute fctn } f_i \text{ on } d \text{ bits}} \cdot \underbrace{O(i^*)}_{O(m) \text{ such fctns}} + O(m) + \frac{t(\ell)}{2}$ to find "design" bits + compute P

set d to be $\frac{\log t}{10}$

So it is $\tilde{O}(t^{\frac{1}{10}})$

$= 2^{d/10}$

but $\tilde{O}(d 2^d) \cdot O(m) + O(m) = \tilde{O}(t^{\frac{1}{10}}) \cdot O(t^{1/c}) + \cdot \frac{t}{2}$

$< t$

which contradicts hardness of f