Lecture 23

Today

Coding In Complexity / Crypto

① Hardcore Predicates For One-Way Permutations (List-Decoding)

② Worst-Case To Avg. Case Reductions (Local (List) Decoding)

Defn:

① $f: \{0,1\}^k \to \{0,1\}^k$ permutation if $\exists\ f^{-1}: \{0,1\}^k \to \{0,1\}^k$ s.t. $\forall x\ f^{-1}(f(x)) = x$.

② f one-way permutation (o.w.p.) if

- (a) f easy (in poly time) on worst-case
- (b) f^{-1} "very hard" on average.

VERY HARD:

- $g: \{0,1\}^k \to \{0,1\}^k$ is $\epsilon(k)$-approximable if $\exists$ polytime A s.t. $\Pr_x[A(x) = g(x)] \geq \epsilon(k)$
- Very hard if not $\frac{1}{p(k)}$-approximable $\forall$ poly $p(k)$

Exercise: if $f: \{0,1\}^k \to \{0,1\}^k$ is OWP

then so is $F: \{0,1\}^{2k} \to \{0,1\}^{2k}$

where $F(x,y) = (f(x), y)$

given $F(x,y)$ can you compute y?

HARD CORE PREDICATE for $f: \{0,1\}^k \to \{0,1\}^k$

- Predicate $b: \{0,1\}^k \to \{0,1\}$
- Hard-core $\Rightarrow$ (1) Easy to compute given x
  (2) hard to compute given $f(x)$
  (2') very hard to guess $b(x)$ given $f(x)$

  ↑ Diff. notion for Boolean functions.

Specifically: $b \circ f^{-1}$ is not $\left(\frac{1}{2} + \frac{1}{p(k)}\right)$-approx. for every polynomial $p()$.

Exercise: if b is hardcore for f, and f is easy, then f is OWP.

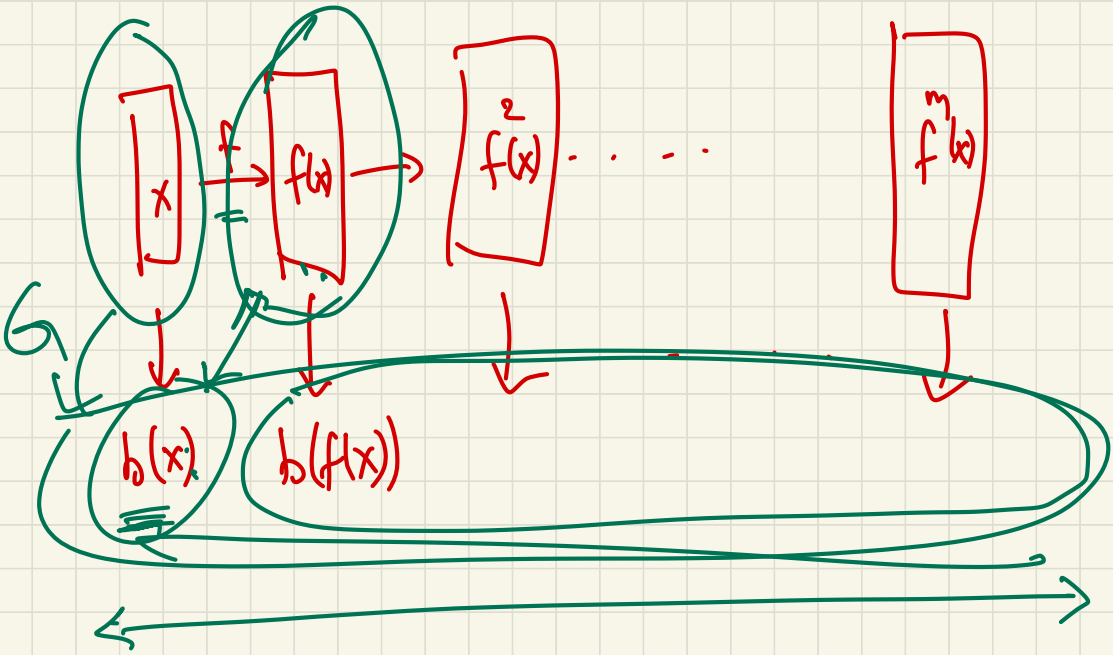
Motivation: Pseudo-random Generators (PRGs)

Defn: $G: \{0,1\}^n \to \{0,1\}^m$ is a PRG if (1) $m > n$

(2) $\forall$ poly A, $\forall$ poly p

$$\left\{A(G(x))\right\} \overset{\frac{1}{p(n)}}{\approx_\epsilon} \left\{A(\text{Unif}\{0,1\}^m)\right\}$$

$\epsilon \Rightarrow \epsilon$-PRG



$$A\left(\left(b(x),\ f(x)\right)\right) \simeq_s A(\mathrm{Unif})$$

PRGs from OWP + Hardcore Predicate

Theorem:

- Suppose $f: \{0,1\}^k \to \{0,1\}^k$ OWP
  & $b: \{0,1\}^k \to \{0,1\}$ hardcore for f
- Then $G(x) = (b(x), b(f(x)), b(f^2(x)) \dots b(f^m(x)))$
  is a PRG, for every $m = m(k)$ poly.

Proof:

(i) Let $G_i(x) = (b(x), G_{i-1}(f(x)))$; $G_0(x) = x$

(ii) $G_1(x)$ ϵ-PRG $\Rightarrow$ G_m $m\cdot\epsilon$-PRG (Δ-ineq.)

(iii) G_1 ϵ-PRG $\Longleftarrow$ b hardcore for f (Exercise)

- Constructing hardcore predicate for OWP
  - Specific pairs known:
    eg f = RSA ; b = msb
  - Specific counterexamples known:
    eg f = RSA ; b = lsb
  - General? - $\exists b$ s.t. $\forall f$ owp, b is hardcore?
    (not likely).

Theorem: $\exists\, b: \{0,1\}^m \to \{0,1\}$ s.t. $\forall$ owf $f: \{0,1\}^k \to \{0,1\}^k$

$\exists$ padding $F: \{0,1\}^m \to \{0,1\}^m$ s.t. F is owp & b is hardcore for F.

$F(x,y) = f(x), y$

Ingredient: list-decodable code $C: \{0,1\}^k \to \{0,1\}^n$

- Suppose C is efficiently encodable and list-decodable from $(\frac{1}{2} - \epsilon)$-fraction error
- Then $m = k + \log n$
  - $F(x, i) = (f(x), i)$
  - $b(x, i) = C(x)_i$

$i \in [n]$ } Construction

- **Analysis**: $\gamma + \delta$
  - Suppose $A(f(x), i) = C(x)_i$ w.p. $\frac{1}{2} + \alpha$ (over x, i)

$$\Rightarrow \Pr_x\left[\underbrace{\Pr_i\left[A(f(x), i) = C(x)_i\right] \geq \frac{1}{2} + \frac{\alpha}{2}}_{\gamma}\right] \geq \frac{\alpha}{2} \quad \delta$$

  - Call such x **good**
  - if x good then can compute x from $f(x)$

f-inverter (y)

- for $i = 1$ to n let

$$w_i = A(y, i)$$

- List-Decode $(w) = \{x_1 .. x_L\}$
- if $f(x_i) = y$ output x_i

Claim: • if x = good then

$$\delta(w, C(x)) \leq \frac{1}{2} - \frac{\alpha}{2}$$

• if $\frac{\alpha}{2} \geq \epsilon$ $[\alpha \geq 2\epsilon]$ then $x \in \{x_1 .. x_L\}$

Δ inverter (y) outputs x.

• $\Rightarrow$ Invert f w.p. $\frac{\alpha}{2} = \epsilon$ if $b(x)$ 2ϵ-approximable (in time poly(n) = poly(k/ϵ))

$\Rightarrow$ f ϵ-inapproximable in time poly$(\frac{k}{\epsilon})$

$\Rightarrow$ b $(\frac{1}{2} + 2\epsilon)$-inapproximable in time poly(n)

Part 2: Worst-Case $\leq$ Average-Case

Example: Permanent:

- $M = \begin{bmatrix} x_{11} & \cdots & x_{1n} \\ \vdots & & \\ x_{n1} & \cdots & x_{nn} \end{bmatrix}$:

$$\text{Perm}(M) = \text{Perm}(x_{11} \cdots x_{nn}) = \sum_{\substack{\pi : [n] \to [n] \\ \text{perm.}}} \prod_{i=1}^{n} x_{i\,\pi(i)}$$

- **Properties:** Perm(M) is #P-complete

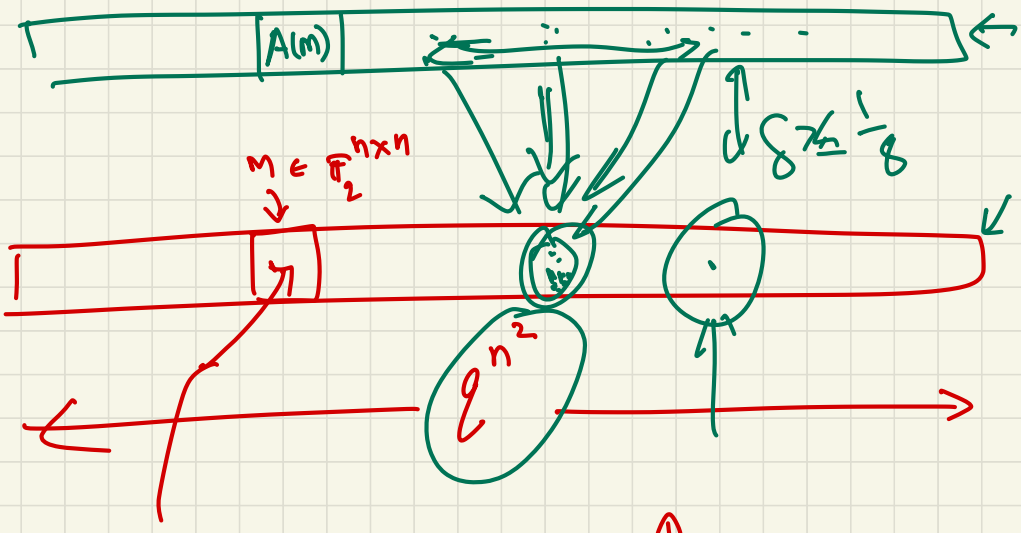
  ($SAT \leq_p \text{Perm}(M)$) ...

- $\text{Perm}(x_{11} \cdots x_{nn}) =$ deg n poly in $x_{11}, \cdots x_{nn}$.

- **Theorem** [Lipton, GLRSW, Gemmell+S]:

  Polytime alg that is correct w.p. $\frac{7}{8}$ for "random" matrix R

  $\Rightarrow$ Polytime alg that is correct on all matrices! whp.



Π

$RM(q, r, m)$

" " "

$q \quad n \quad n^2$

$Perm(m)$

Lemma: if $q > n^2$ & $\Pr_{R \sim \mathbb{F}_q^{n \times n}}[A(R) = \mathrm{Perm}(R)] \geq \frac{7}{8}$

then $\exists B$ s.t. $\forall M \in \mathbb{F}_q^{n \times n}$ $B(M) = \mathrm{Perm}(M)$ w.p. $1 - o(1)$.

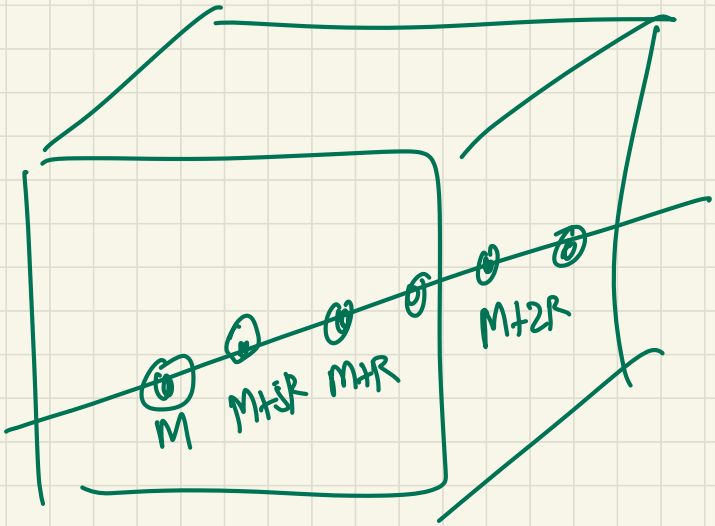
Proof: Use local decoder for RM codes!

- $B(M)$:
  - Pick $R \in \mathbb{F}_q^{n \times n}$ randomly
  - $\{\alpha_1, \ldots \alpha_{10n}\}$ distinct from $\mathbb{F}_q^*$
  - $\beta_i = A(M + \alpha_i \cdot R) \quad \forall i \in [10n]$
  - Reed-Solomon-Decode $(\{(\alpha_i, \beta_i)\}_i) \to P$
  - Output $P(0)$.

- Claim:
  - $g(t) \triangleq \mathrm{Perm}(M + t \cdot R)$

    $\deg g \leq n$; $g(0)$ = our goal.

  - $\forall i$ $\Pr_M[\beta_i = g(\alpha_i)] \geq 7/8$
  - $\Rightarrow \Pr_M\left[\#\underbrace{\{i \mid \beta_i \neq g(\alpha_i)\}}_{10n} > \frac{1}{4}\right] \leq \frac{1}{4}$



$$F_g \propto \frac{1}{n^2}$$

- **Claim:**
  - $g(t) \triangleq \text{Perm}(M + t \cdot R)$

    $\deg g \le n$; $g(0) =$ our goal.

  - $\forall i$
    - $\Pr_M [\beta_i = g(\alpha_i)] \ge 7/8$
    - $\Rightarrow \Pr_M [\#\{i \mid \beta_i \ne g(\alpha_i)\} > \frac{1}{4}] \le \frac{1}{4}$

      (ion)

$\Rightarrow$ Decoder outputs $g \Rightarrow$ B outputs perm(M).

Improvements: $\frac{7}{8} \to \underbrace{\frac{1}{2} + \epsilon}$ (Better decoding + reduction)

$\Rightarrow [M + \alpha R_1 + \alpha^2 R_2]$

- $\frac{1}{2} + \epsilon \to \frac{1}{\text{poly}(n)} \to$ "Local list decoding".

General Thm: [S.-Trevisan-Vadhan]

$f \in \text{TIME}(2^{O(n)})$ but not in $P/poly$ — size $(2^{\varepsilon n})$

$\Rightarrow \exists\ F \in \text{TIME}(2^{O(n)})$ but "very hard" for $P/poly$ — size $(2^{\varepsilon' n})$

Proof: Let $C: \{0,1\}^K \to \{0,1\}^N$ be a
- locally list decodable code with locality $(\log K)^{O(1)}$
- from $\frac{1}{2} - \frac{1}{poly}$ errors.

- View $f: \{0,1\}^k \to \{0,1\}$ as vector in $\{0,1\}^K$, $K = 2^k$
- $F = C(f) \in \{0,1\}^N$ viewed as $F: \{0,1\}^n \to \{0,1\}$ for $n = \log N$
- Thm holds for F. Proof (Exercise)

Summary

- Hardcore Predicate for owp $\Rightarrow$ List-decoding (ordinary)
- Average Case $\geq$ Worst-case $\Rightarrow$ Local (list) decoding