Hardness Amplification

- Our goal is to construct average-case hard functions using a function f that is only worst-case hard.

- Idea: View f as a 2^n-length string & apply a map φ that is a "very good" error-correcting code.

Definition: • For $x, y \in \{0,1\}^m$, the fractional Hamming distance $\Delta(x,y) := \frac{\#\{i \mid x_i \neq y_i\}}{m}$.

• For $\delta \in (0,1)$, a function $E: \{0,1\}^n \to \{0,1\}^m$ is an error-correcting code (ECC) with distance δ, if $\forall x \neq y \in \{0,1\}^n$,

$$\Delta(E(x), E(y)) \geq \delta.$$

• We call $\mathrm{Im}(E) := \{E(x) \mid x \in \{0,1\}^n\}$ the set of codewords.

- These have vast applications.

  In the real world they are used in physical communication channels & the storage media.

- For hardness amplification:

  Let f be a worst-case hard boolean function. Let f' be the $N = 2^n$-bit string expressing $tt(f)$.

  We encode f' by an ECC $E : \{0,1\}^N \longrightarrow \{0,1\}^{N^c}$. Thus, $E(f')$ is a 2^{cn}-bit string expressing $tt(g)$, for some $g : \{0,1\}^{cn} \to \{0,1\}$.

  We will show that if E with nice local decoding properties exists then g is an average-case hard function.

  (Also, $f \in \mathbb{E} \Rightarrow g \in \mathbb{E}$.)

Introduction to Error-correction

- Practical applications of ECC stem from the following situation:

  Alice wants to transmit Bob a string $x \in \{0,1\}^n$ on a channel that corrupts $\leq 10\%$ of bits.

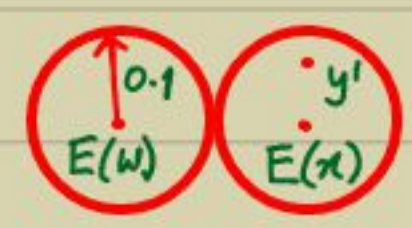
A $\qquad\qquad\qquad\qquad\qquad\qquad$ B

$x \in \{0,1\}^n$

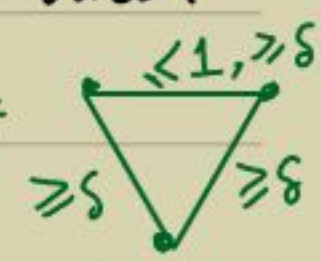
$E(x) \in \{0,1\}^m \xrightarrow{y = E(x)} y'$ (guarantee: $\Delta(y', y) \leq 0.1$)

- If E is of distance > 0.2 then $\exists$ unique w s.t. $\Delta(E(w), y') \leq 0.1$,

  $\Rightarrow w = x$.



- This motivates the design of codes with:
  - large distance δ. $\quad \delta \leq 1/2 \Leftarrow$ (unique decoding)
  - small length m.
  - efficient encoding & decoding.



- The first two conditions get satisfied for "most" E.

Lemma (Gilbert-Varshamov bound): $\forall \delta \in (0, \frac{1}{2})$ & large enough n, $\exists E: \{0,1\}^n \to \{0,1\}^m$ that is an ECC with distance δ & $m = 2n/(1-H(\delta))$, where $H(\delta) := -\delta \lg \delta - (1-\delta)\lg(1-\delta)$.

[Shannon's entropy. Eg. $0 = H(0) \le H(\delta) \le H(\frac{1}{2}) = 1$.]

Proof:

- In fact, we show that a "random" E works!
- Pick $y_1, y_2, \dots, y_{2^n} \in \{0,1\}^m$ at random.

  Define $E: x \mapsto y_x$.
- $\forall i \neq j \in [2^n]$, $\Pr_E[\Delta(y_i, y_j) < \delta]$

  $\leq \#(\leq \delta m)$-places in y_j / # possible y_j

$$\sum_{i=0}^{\delta m} \binom{m}{i} / 2^m \leq 0.01 \times 2^{-m(1-H(\delta))}$$

* Stirling's approximation

$$\Rightarrow \Pr_E[\exists i \neq j, \Delta(y_i, y_j) < \delta] \leq 0.01 \times 2^{2n - m(1-H(\delta))}$$

$$= 0.01$$

$$\Rightarrow \Pr_E[\forall i \neq j, \Delta(y_i, y_j) \geq \delta] > 0.99.$$

$\square$

- It can be seen in the analysis that:
  - For $\delta = 1/2$, $\exists$ code with $m = 2^{\Omega(n)}$.
  - For $\delta > 1/2$, $\nexists$ code for large n.

$\Rightarrow$ These codes might lead to <u>unique</u> decoding up to errors $< \delta/2 \leq 1/4$.

- Can we find encoding & decoding algorithms that run in poly(n)-time?

- We will study four explicit codes: linear!
  - <u>Walsh-Hadamard</u> ($\delta = 1/2$)
  - Reed-Solomon ($\delta < 1/2$ & efficient)
  - <u>Reed-Muller</u> (multivariate generalization)
  - <u>Concatenated codes</u>

- We will strengthen the notion of decoding from: <u>unique</u> $\rightarrow$ <u>local</u> $\rightarrow$ <u>list</u>.

Walsh-Hadamard code (1940s)

Defn: For $x, y \in \{0,1\}^n$ we define $x \odot y = \sum_{i=1}^{n} x_i y_i \pmod 2$. The <u>WH code</u> is

$$WH : \{0,1\}^n \to \{0,1\}^{m=2^n}, \quad x \mapsto z \text{ where}$$

$$z_y := x \odot y, \quad \forall y \in \{0,1\}^n.$$ (i.e. all projections of x modulo 2)

Lemma 1: WH is an ecc of distance $1/2$.

Pf:

- $\forall x \neq y$, $WH(x+y) = WH(x) + WH(y)$, where $x+y$ is the coordinate-wise sum mod 2.
- Thus, $wt(WH(x+y)) = \Delta(WH(x), WH(y)) \cdot m$.

- As $x+y \neq \bar{0}$, it is orthogonal to exactly $1/2$ of the vectors in $\{0,1\}^n$.

  $\Rightarrow \Delta(WH(x), WH(y)) = 1/2$. $\square$

- To get a shorter code we look at finite fields other than $\mathbb{F}_2$:

Reed-Solomon code (1960)

- We view the input string as a polynomial & consider all its evaluations.

Defn: Let $\mathbb{F}$ be a field & $n \le m \le |\mathbb{F}|$. RS code is

$$RS : \mathbb{F}^n \longrightarrow \mathbb{F}^m, \ (a_0, \dots, a_{n-1}) \mapsto (z_0, \dots, z_{m-1})$$

where $\forall j$, $z_j = \sum_{0 \le i < n} a_i \cdot f_j^i$ for the j-th element f_j of $\mathbb{F}$.

Lemma 2: RS is an ecc of distance $1 - \frac{n-1}{m}$.

Proof:

- $\forall a \ne b \in \mathbb{F}^n$, $RS(a-b) = RS(a) - RS(b)$, for coordinate-wise sums.
- Thus, $wt(RS(a-b)) = \Delta(RS(a), RS(b)) \cdot m$.
- As $a - b \ne \bar{0}$, $RS(a-b)$ is a set of m evaluations of a nonzero polynomial $\sum_{0 \le i < n} (a_i - b_i) x^i$.

$\Rightarrow$ $< n$ of these could be zero.

binary-distance is $\frac{m-n+1}{m \cdot \lg|\mathbb{F}|}$

$\Rightarrow \Delta(RS(a), RS(b)) \geq \frac{m-n+1}{m}$. $\square$

Reed-Muller code (1954)

- Here we view the input as a multivariate polynomial, and consider evaluations.

Defn: Let $\mathbb{F}$ be a finite field; $\ell, d \in \mathbb{N}$ & $d < |\mathbb{F}|$.

- RM code is $RM: \mathbb{F}^{\binom{\ell+d}{d}} \to \mathbb{F}^{|\mathbb{F}|^\ell}$ that maps every ℓ-variate d-deg polynomial P, over $\mathbb{F}$, to all evaluations.

Note: $d=1 \Rightarrow$ WH $\to$ & $\ell=1 \Rightarrow$ RS

- Thus, $RM: \{ c_{\bar{i}} \in \mathbb{F} \mid |\bar{i}| \leq d \} \longmapsto$
$$\{ P(x_1, \ldots, x_\ell) := \sum_{\bar{i}} c_{\bar{i}} \cdot \bar{x}^{\bar{i}} \mid x_1, \ldots, x_\ell \in \mathbb{F} \}.$$

Lemma 3: RM is an ecc of distance $1 - \frac{d}{|\mathbb{F}|}$.

Proof:

- As for RS, we have $\forall a \neq b$,
$$wt(RM(a-b)) = \Delta(RM(a), RM(b)) \cdot \overset{|\mathbb{F}|^\ell}{m}.$$

- By DeMillo et.al.'s lemma on zeros:
$$wt(RM(a-b)) / |\mathbb{F}^\ell| \geq 1 - \frac{d}{|\mathbb{F}|}.$$

$\square$

Concatenated code (Forney 1966)

- WH has a large m, while RS uses a non-binary alphabet. We want to remove both these drawbacks.

So we first apply RS & then WH.
to spread the bits around!

Defn: Let $\mathbb{F}$ be a finite field of size q, RS: $\mathbb{F}^n \to \mathbb{F}^m$, WH: $\{0,1\}^{\lg q} \to \{0,1\}^q$.

Then the concatenated code

$\underline{\text{WH}\circ\text{RS}}$: $\{0,1\}^{n \lg q} \to \{0,1\}^{mq}$ is :

1) View RS as a code from $\{0,1\}^{n \lg q}$, & WH as a code from $\mathbb{F}$. (Using a natural binary representation of the elements in $\mathbb{F}$.)

2) $\forall x \in \{0,1\}^{n \cdot \lg q}$,

$$\underline{\text{WH}\circ\text{RS}(x)} := \langle \text{WH}(\text{RS}(x)_i) \mid i \in [m] \rangle,$$

where $\text{RS}(x)_i \in \mathbb{F}$ is the ith symbol in $\text{RS}(x)$.

▷ WH∘RS is computable in time poly($|\mathbb{F}|$).

Lemma 4: WH∘RS is an ecc of distance $\frac{1}{2}\cdot\left(1-\frac{n-1}{m}\right)$.

Proof:

- Let $x \neq y \in \{0,1\}^{n\lg q}$. Then we know that the #distinct $\mathbb{F}$-elements in $RS(x), RS(y)$ is $\geq \left(1-\frac{n-1}{m}\right)$.
- If $x' \neq y' \in \mathbb{F}$ are in i-th place of $RS(x), RS(y)$, then $\Delta(WH(x'), WH(y')) \geq 1/2$.

$$\Rightarrow \Delta(WH\circ RS(x), WH\circ RS(y)) \geq \frac{1}{2}\cdot\left(1-\frac{n-1}{m}\right).$$

$\square$

- By the prime number theorems, $\forall k \geq 2$, $\exists$ prime $p \in [10k, 11k)$ Thus, we can work over the field $\mathbb{F} := \mathbb{F}_p$.

$\Rightarrow$ WH∘RS is an ecc that stretches a $\Theta(k\lg k)$-long message to length, say, $(10k\cdot 11k)$, with distance $\geq \frac{1}{2}\cdot\left(1-\frac{k}{10k}\right) = 0.45$.

$p \geq m \rightarrow$

▷ $\forall n \in \mathbb{N}$, $\exists$ poly-time computable ecc $E: \{0,1\}^n \rightarrow \{0,1\}^{n^2}$ that can sustain 22% errors.

Efficient decoding

- Can we find <u>the</u> unique x given a string y' "close" to $E(x)$?

- <u>Decoding WH</u> is trivial: Since WH length is 2^n, we can afford to scan the full space $\{0,1\}^n$ & find the unique x from y'.

Decoding RS

- <u>Setting</u>: Given a list $(a_1,b_1),\dots,(a_m,b_m) \in \mathbb{F}^2$ for which $\exists$ deg-d polynomial $G:\mathbb{F}\to\mathbb{F}$ s.t. $G(a_i)=b_i$ for t of the pairs.

- Since RS has distance $(1-\frac{d}{m})$, we are guaranteed the existence of a <u>unique</u> G, if $t > m-\frac{1}{2}(1-\frac{d}{m}).m = \frac{m+d}{2}$ & $m>d$.

- <u>Idea</u> - If t was m then we could have simply interpolated a deg-d G from the linear system $G(a_i) = b_i$, $i \in [m]$.

In the $t < m$ case we introduce an auxiliary error-locator polynomial $\mathcal{E}(X)$ of $\deg = \frac{m-d}{2}$ = # possible errors, & interpolate polynomials C & $\mathcal{E}$ from:

$$C(a_i) = b_i \cdot \mathcal{E}(a_i), \quad \forall i \in [m],$$

where $\deg(C) = d + \frac{m-d}{2} = \frac{m+d}{2}$.

<u>Theorem</u> (Berlekamp-Welch, 1986): $\exists$ poly$(m, \log|\mathbb{F}|)$-time algorithm to find G from $\{(a_i, b_i) | i\}$.

<u>Proof</u>:

- The algorithm is simply:

1) Find polynomials $C(X), \mathcal{E}(X)$ of degrees $\frac{m+d}{2}$, $\frac{m-d}{2}$ respectively s.t.

$$C(a_i) = b_i \cdot \mathcal{E}(a_i), \quad \forall i \in [m].$$

2) Output $C(X)/\mathcal{E}(X)$.

- The linear system, in Step 1, has m equations & $\left(1+\frac{m+d}{2}\right)+\left(1+\frac{m-d}{2}\right)=m+2$ unknowns.

- It has a <u>nonzero</u> solution because we can take $G(x)\cdot\underbrace{\left(\prod_{G(a_i)\neq b_i}(x-a_i)\right)}_{\mathcal{E}(x)}$ as $C(x)$.

- Let C & $\mathcal{E}$ be the solutions obtained in Step 1.

$$\Rightarrow \quad C(a_i)-G(a_i)\cdot\mathcal{E}(a_i)=0,$$

for $t>\frac{m+d}{2}$ of the i's.

- But, $\deg\left(C(x)-G(x)\cdot\mathcal{E}(x)\right)\leq\frac{m+d}{2}<t$

$$\Rightarrow \quad C=G\cdot\mathcal{E}$$

$$\Rightarrow \quad C/\mathcal{E}=G(x).$$

- Since we used deg-m polynomial arithmetic over $\mathbb{F}=\mathbb{F}_q$, it can be done in $\mathrm{poly}(m,\lg|\mathbb{F}|)$ time. $\square$

Decoding WH∘RS

Theorem: For $WH \circ RS : \{0,1\}^{n \lg q} \to \{0,1\}^{mq}$, $\exists$ poly(q)-time decoder, if the fraction of errors $< \frac{1}{4} \cdot \left(\frac{1}{2} - \frac{n-1}{2m}\right)$.

Notice the fall from RS by 1/4-th.

Proof:

- Let y' be "close" to $y = \langle WH(RS(x)_i) \mid i \in [m] \rangle$.
- The hypothesis implies that $\#\{i \mid WH(RS(x)_i) \text{ has} \geq q/4 \text{ errors}\} < \frac{m-n+1}{2}$.

$\Rightarrow$ WH-decoding will yield $\langle \tilde{y}_1, \ldots, \tilde{y}_m \rangle =: \tilde{y}$ with $\tilde{y}_i = RS(x)_i$ for $> \underbrace{\frac{m+n-1}{2}}_{= m - \frac{m-n+1}{2}}$ of the i's.

$\Rightarrow$ RS-decoding of $\tilde{y}$ yields the unique x. $\square$

- Thus, WH∘RS is a practical ecc that handles up to 11% of errors.

- For hardness amplification we need an even stronger kind of decoding:

Local Decoding

Defn: Let $E: \{0,1\}^n \to \{0,1\}^m$ be an ecc & $\rho \in (0,1)$.

Short: L_ρ → A local decoder for E handling ρ errors is an algorithm that:

Given $j \in [n]$ & oracle to y s.t. $\Delta(y, E(x)) < \rho$,

Outputs x_j with probability $\geq 2/3$

& poly($\lg m$)-time.

(Thus, when m is large, very few bits of y are needed to guess x_j !)

Theorem 1: $\forall \rho < 1/4$, WH-code has a L_ρ.

Proof:

- Idea - Querying the two positions - z & $z + e_j$ - suffices to guess x_j.

n-bit

the j-th bit 1 while others 0

- <u>Input</u>: $j \in [n]$, oracle $f: \{0,1\}^n \to \{0,1\}$ s.t.
$$\Pr_z[f(z) \neq x \odot z] \leq \rho.$$
[x is the unknown plaintext, $tt(f)$ is corrupted $E(x)$.]
- <u>Output</u>: $b \in \{0,1\}$ (Whp $b = x_j$).
- <u>Decoder</u>:
  1) Randomly pick $z \in \{0,1\}^n$.
  2) Let $e_j \in \{0,1\}^n$ be the string with 1 at the j-th place & 0 in the rest.
  3) Output $f(z) + f(z + e_j) \bmod 2$.

- Clearly, the time complexity is $poly(n) = poly(\lg m)$, as $m = 2^n$.

- <u>Analysis</u>: $\Pr_z[f(z) = x \odot z \wedge f(z+e_j) = x \odot (z+e_j)]$
$$\geq 1 - 2\rho > 1/2.$$
$\Rightarrow \Pr_z[f(z) + f(z+e_j) = x \odot e_j \bmod 2] > 1/2.$
$\Rightarrow \Pr_z[b = x_j] > 1/2.$
- This can be further boosted. $\square$

Local decoder for RM

- Recall RM: $\mathbb{F}^{\binom{\ell+d}{d}} \to \mathbb{F}^{|\mathbb{F}|^\ell}$ is of distance $\left(1-\frac{d}{|\mathbb{F}|}\right)$, $d < |\mathbb{F}| < \infty$.

- For local decoding it will be convenient to view RM as mapping $\binom{\ell+d}{d}$ evaluations of a polynomial f to its $|\mathbb{F}|^\ell$ evaluations.

Theorem 2: $\forall \rho \leq \frac{1}{6}\left(1-\frac{d+5}{|\mathbb{F}|-1}\right)$, RM-code has a LD_ρ.

Proof:

- Idea – The degree-d polynomial f is unknown & we want to evaluate it at, say, $x \in \mathbb{F}^\ell$.

  Pick a random line L_x through x, evaluate f on each point in L_x, & use RS-decoder to learn $f|_{L_x}$.

  (This is a generalization of WH local decoder)

- Input: $x \in \mathbb{F}^\ell$, oracle $\tilde{f} : \mathbb{F}^\ell \to \mathbb{F}$ that agrees with some ℓ-variate d-deg f on $\geq 1-\rho$ points.

- <u>Output:</u> $\alpha \in \mathbb{F}$ [whp $\alpha = f(x)$].
- <u>Decoder:</u>
  1) Pick a random $z \in \mathbb{F}^\ell$ & define "line" $L_x := \{x + tz \mid t \in \mathbb{F}\}$.
  2) Query $\tilde{f}$ on L_x, i.e. collect the pairs $\{(t, f(x+tz)) \mid t \in \mathbb{F}\} =: \tilde{f}(L_x)$.
  3) Via RS-decoder, on $\tilde{f}(L_x)$, find a degree $\leq d$ polynomial $\tilde{Q}: \mathbb{F} \to \mathbb{F}$ s.t. $\tilde{Q}(t) = \tilde{f}(x+tz)$ for the largest number of t's.
  4) Output $\tilde{Q}(0)$.

- Clearly, the time complexity is $\mathrm{poly}(\ell, d, |\mathbb{F}|)$.
- <u>Analysis:</u>
  - RS decoder tries to reconstruct $f(x+tz) =: Q(t)$, which has deg $\leq d$ & is univariate.
  - For the decoder to find Q we need the guarantee, $\Pr_z\left[\#t, \text{ with } Q(t) \neq \tilde{f}(x+tz), \text{ is} < \frac{|\mathbb{F}|-d}{2}\right] \geq 2/3$.

· For that we compute the expectation:

$$\mathop{\mathrm{Exp}}_{z}\left[\#\{t\in\mathbb{F} \mid f(x+tz)\neq \tilde{f}(x+tz)\}\right] \leq$$

$$1+\sum_{t\in\mathbb{F}^*}\Pr_{z}\left[f(x+tz)\neq\tilde{f}(x+tz)\right]\leq 1+\rho(|\mathbb{F}|-1).$$

· Thus, by Markov's inequality:

$$\Pr_{z}\left[\#\{t\in\mathbb{F} \mid Q(t)\neq\tilde{Q}(t)\}\geq\frac{|\mathbb{F}|-d}{2}\right]$$

$$\leq 1+\rho(|\mathbb{F}|-1)\Big/\frac{|\mathbb{F}|-d}{2}\leq\frac{1+\frac{1}{6}(|\mathbb{F}|-d-6)}{(|\mathbb{F}|-d)/2}=\frac{1}{3}.$$

· Thus, with $\text{prob}_z\geq\frac{2}{3}$, Step-3 produces

$$\tilde{Q}(t)=Q(t)=f(x+tz).$$

$$\Rightarrow\ \tilde{Q}(0)=f(x).$$

$\square$

Local decoder for concatenated codes

- Let $E_1:\{0,1\}^n\to\Sigma^m$ resp. $E_2:\Sigma\to\{0,1\}^k$ be ecc's with local decoders of q_1 resp. q_2 queries handling ρ_1 resp. ρ_2 errors.

  [Like RM we assume that $q_1\geq|\Sigma|$.]

Theorem 3: $\exists\ O(q_1 q_2 \cdot \lg q_1 \cdot \lg|\Sigma|)$-query $Ld_{\rho_1\rho_2}$ for the ecc $E := E_2 \circ E_1 : \{0,1\}^n \to \{0,1\}^{mk}$.

Proof:

- Idea – Break y into blocks of size k. On a block call E_2's Ld_{ρ_2} $(\lg|\Sigma|)$-times. Finally, call E_1's Ld_{ρ_1} on several decoded blocks.

- Input: Index $i \in [n]$ & oracle access to $y \in \{0,1\}^{mk}$ s.t. $\exists x \in \{0,1\}^n$, $\Delta(y, E_2 \circ E_1(x)) < \rho_1\rho_2$.
- Output: $b \in \{0,1\}$ (whp $b = x_i$).
- Decoder:
  1) View y as m blocks each of k-bits. [It is a corrupted $\langle E_2(E_1(x)_j) \mid j \in [m] \rangle$.]
  2) To find the j-th symbol $E_1(x)_j$ we call E_2's Ld_{ρ_2} on the j-th block of y.
     We do this $\lg|\Sigma|$ times to recover the full $E_1(x)_j$.
  3) We repeat this $(50 \cdot \lg q_1)$ times so that the probability of not decoding $E_1(x)_j$ is

$< 1/10q_1$ [Hint: Chernoff bound & then the union bound yields $1/q_1^2 \times \lg|\Sigma| < \frac{1}{10q_1}$, as $q_1 \geq |\Sigma|$.]

4) Since $< \rho_1$ of the blocks in y can be at distance $> \rho_2$ from the respective <u>true</u> block, we use E_1's Ld_{ρ_1} to query q_1 blocks.

With probability $> 1 - \frac{1}{10q_1} \times q_1 = 0.9$

the q_1 answers to E_1's Ld_{ρ_1} are <u>consistent</u> with that of a string that is ρ_1-close to $E(x)$.

$\Rightarrow$ E_1's Ld_{ρ_1} outputs x_j with probability

$\geq 0.9 - 1/3 > 1/2$

& queries $= O(q_2 \cdot \lg|\Sigma| \cdot \lg q_1 \cdot q_1)$.

$\square$

<u>Corollary:</u> For WHORM local decoder the #queries is $O(q \cdot \lg^2 q)$ handling up to

$\frac{1}{6} \cdot \left(1 - \frac{d+5}{q-1}\right) \cdot \frac{1}{4}$ errors, where $q = |\mathbb{F}|$.

(codelength $\approx q^\ell$)

(msg-length $\approx \binom{d+\ell}{\ell}$)

- Our final goal is to show: If f is a worst-case hard function & E a locally decodable code, then $E \circ tt(f)$ is the truth-table of an average-case hard function g.

- For average-case hardness of g we would need an E that is locally decodable up to $(1/2 - \delta)$ errors!

  This type of decodability cannot be unique.

- So, we relax unique decodability to that of finding a list.

Theorem (Johnson bound 1962): If $E: \{0,1\}^n \to \{0,1\}^m$ is an ecc with distance $\geq (\frac{1}{2} - \varepsilon)$ then $\forall x \in \{0,1\}^m$ & $\delta \overset{>0}{\geq} \sqrt{\varepsilon}$, $\exists \leq 1/2\delta^2$ codewords $y_1, \dots, y_\ell$ s.t. $\Delta(x, y_i) \leq 1/2 - \delta$, $\forall i \in [\ell]$.

Proof: • Idea - We reduce the notion of distance to that of inner-product & use linear algebra.

• Let $\Delta(x, y_i) \leq 1/2 - \delta$, $\forall i \in [\ell]$.

• Define $z_1, \ldots, z_\ell \in \{-1, 1\}^m$ s.t.

$$\underline{z_{i,k}} = \begin{cases} 1, & \text{if } x_k = y_{i,k} \\ -1, & \text{else.} \end{cases}$$

• $\Delta(x, y_i) \leq 1/2 - \delta \Rightarrow$

(1) --- $$\sum_{k \in [m]} z_{i,k} \geq \left(\tfrac{1}{2} + \delta\right)m - \left(\tfrac{1}{2} - \delta\right)m = 2\delta m.$$

counting 1's -1's

• $\Delta(y_i, y_j) \geq 1/2 - \varepsilon \Rightarrow$

(2) --- $$\langle z_i, z_j \rangle = \sum_k z_{i,k} \cdot z_{j,k} \leq \left(\tfrac{1}{2} + \varepsilon\right)m - \left(\tfrac{1}{2} - \varepsilon\right)m$$

$$= 2\varepsilon m \leq 2\delta^2 m.$$

• Let $w := \sum_{i \in [\ell]} z_i$.

$$\Rightarrow \langle w, w \rangle = \sum_{i \in [\ell]} \langle z_i, z_i \rangle + \sum_{i \neq j} \langle z_i, z_j \rangle$$

By (2)

(3) --- $$\leq \sum_i m + \sum_{i \neq j} 2\delta^2 m \leq \ell m + 2\delta^2 \ell^2 m.$$

- Also, by (1): $\sum_{k=1}^{m} w_k = \sum_{\substack{k\in[m]\\ i\in[\ell]}} z_{i,k} \geq 2\delta \ell m$.

  By Cauchy-Schwarz's, $\sum w_k^2 \geq (\sum w_k)^2/m$,

  $\Rightarrow \langle w, w\rangle \geq (2\delta m \ell)^2/m = 4\delta^2\ell^2 m$.

- This means, by (3), that:

  $4\delta^2\ell^2 m \leq \langle w, w\rangle \leq \ell m + 2\delta^2\ell^2 m$

  $\Rightarrow \quad 2\delta^2 \ell \leq 1$

$\because \delta > 0$ $\quad \Rightarrow \quad \ell \leq 1/2\delta^2 \leq 1/2\varepsilon$. $\square$

- Thus, there are not too many codewords $(\frac{1}{2} - \sqrt{\varepsilon})$-close to x if the distance of the code is $(\frac{1}{2} - \varepsilon)$.

  Can we compute this list efficiently? & locally?

- We will see that both the answers are yes!

List Decoding RS

Theorem (Sudan '95): $\exists$ randomized poly-time algorithm that given $\{(a_i, b_i) \in \mathbb{F}^2 \mid i \in [m]\}$, returns the list of <u>all</u> degree $\leq d$ polynomials G s.t. $\#\{i \in [m] \mid G(a_i) = b_i\} > \sqrt{2dm}$.

[i.e. for distance $> 1 - \frac{d-1}{m}$, the list decoder handles $< 1 - \sqrt{\frac{2d}{m}}$ errors.]

Proof:

- Idea – Instead of interpolating a univariate, work with a <u>bivariate</u> polynomial.

1) Find a nonzero $Q(x,y) \in \mathbb{F}[x,y]$ s.t.

$$Q(a_i, b_i) = 0, \quad \forall i \in [m] \quad \&$$

$(1,d)$-weighted-deg $(Q) \leqslant \sqrt{2dm} =: t$.

[$^{*}:= \max\{i + dj \mid$ monomial $x^i y^j$ is in the support of $Q\}$.]

[The maximum number of monomials that Q can have $= \sum_{0 \leq j \leq t/d} (1 + t - dj)$

$$= (1+t)\cdot(1+\lfloor t/d \rfloor) - \frac{d}{2}\cdot\left\lfloor \frac{t}{d} \right\rfloor (1+\lfloor t/d \rfloor)$$

$$\geq (1+t-\tfrac{d}{2}\cdot\lfloor t/d \rfloor)\cdot(1+\lfloor t/d \rfloor)$$

$$\geq (1+t/2)\cdot t/d = t/d + t^2/2d$$

$$> m.$$

Since #eqns. < #unknowns, the homogeneous linear system can be solved to get a $Q(X,Y)$ in Step-1.]

2) Factor $Q(X,Y)$ using any efficient polynomial factoring algorithm (over finite $\mathbb{F}$).

3) For factors of the form $Y-P(X)$, where $\deg P \leq d$ & $\#\{i\in[m] \mid P(a_i)=b_i\} > t$, OUTPUT $P(X)$.

[Any $\deg\leq d$ $G(X)$ that "fits" $>t$ points yields: $\begin{cases} \deg Q(X,G(X)) \leq t, \ \& \\ Q(X,G(X))=0 \text{ on } >t \text{ distinct } a_i\text{'s}. \end{cases}$

$\Rightarrow Q(X,G(X))=0 \Rightarrow Y-G(X) \mid Q(X,Y).$]

$\square$

Corollary: RS, of distance $1-\frac{d-1}{m}$, has a list decoder handling $< 1-\sqrt{2d/m}$ errors & list-size $\leq \sqrt{2m/d}$.

Proof:

- In the proof above, the list-size is $\leq \deg_Y Q \leq t/d = \sqrt{2m/d}$. $\square$

Local List Decoding

Defn: Let $E: \{0,1\}^n \to \{0,1\}^m$ be an ecc & let $\varepsilon := 1/2 - \rho$ for $\rho \in (0, 1/2)$.

An algorithm $\mathcal{D}$ is a <u>local list decoder for E handling ρ errors</u>, if $\forall x \in \{0,1\}^n$, $\forall y \in \{0,1\}^m$ with $\Delta(E(x), y) \leq \rho$

i_0 is the advice (location of x in correct list) $\rightarrow$ $\exists i_0 \in [\text{poly}(n/\varepsilon)]$ s.t.

On inputs $\langle i_0, j, \text{ oracle } y \rangle$, $\mathcal{D}$ runs for $\text{poly}(\lg m, n/\varepsilon)$-time & outputs x_j with probability $\geq 2/3$.

(or x?)

Local list decoding WH

Theorem 1 (Goldreich-Levin, '89): Let $WH: \{0,1\}^n \to \{0,1\}^{2^n}$ & $f: \{0,1\}^n \to \{0,1\}$ be an oracle s.t. $\exists x \in \{0,1\}^n$, $\Pr_z[f(z) = WH(x)_z] \geq \frac{1}{2} + \frac{\varepsilon}{2}$.

Let $\mathcal{L}_f$ be the list of such x's.

$\exists$ poly(n/ε)-time randomized algorithm to find $\mathcal{L}_f := \{x \mid \Delta(f, WH(x)) \leq \frac{1}{2} - \frac{\varepsilon}{2}\}$.

Proof:

- Idea – Since the corruption in f is close to $1/2$, we do not get x_i in two queries. Instead we will make a single query & the other answer we will "guess".

  To reduce the error/time in finding all $x = x_1 \cdots x_n$ we will use correlated but pairwise-independent queries.

- Fix $m = \lceil 200n/\varepsilon^2 \rceil$, $k := \lceil \lg(m+1) \rceil$.
- Randomly pick "points" $s_1, \ldots, s_k \in \{0,1\}^n$ & "guesses" $\sigma_1, \ldots, \sigma_k \in \{0,1\}$.

[Hope: $\exists x \in \mathcal{L}_f, \forall i \in [k], \sigma_i = x \odot s_i$.]

- [Lots of pairwise-independent points]

$\forall \phi \neq T \subseteq [k]$, $s_T := \bigoplus_{i \in T} s_i$ (bit-wise XOR)

& $\sigma_T := \bigoplus_{i \in T} \sigma_i$.

- Compute $\forall i \in [n]$, $x_i = \mathrm{maj}_T \{ \sigma_T \oplus f(s_T + e_i) \}$,

where e_i is all 0's

except 1 at the i-th position.

- OUTPUT $x_1 \ldots x_n$ [Hope: it is in $\mathcal{L}_f$.]
- Repeat the above $1000n/\varepsilon^4$ times.

<u>Analysis</u>: Let us consider a fixed $x \in \mathcal{L}_f$ & compute the probability of outputing it.

- $\Pr[\forall i \in [k], \sigma_i = x \odot s_i] = 2^{-k} \geq 1/2m$.

- We now assume $\forall i \in [k]$, $\sigma_i = x \odot s_i$.
- Define $\forall \phi \neq T \subseteq [k]$, $\forall j \in [n]$,

$$Z_{T,j} := \begin{cases} 1, & \text{if } f(s_T + e_j) = x \odot (s_T + e_j). \\ 0, & \text{else}. \end{cases}$$

- Define $Z_j := \sum_{\phi \neq T \subseteq [k]} Z_{T,j}$, $\forall j \in [n]$.

$$\Rightarrow E[Z_j] = \sum_T \Pr[f(s_T + e_j) = x \odot (s_T + e_j)]$$
$$\geq m \cdot \left(\frac{1}{2} + \frac{\varepsilon}{2}\right). \quad \text{[linearity of expectation]}$$

- For a fixed j: $Z_{T,j}$'s are <u>not</u> independent but are <u>pairwise independent</u>.

So, instead of Chernoff, we apply the <u>Chebychev's inequality</u>:

$$\Pr[|R - E(R)|^2 \geq k^2 \cdot \mathrm{var}(R)] \leq 1/k^2.$$

$$\Rightarrow \Pr[Z_j \leq m/2] = \Pr[|Z_j - E[Z_j]| \geq m\varepsilon/2]$$
$$\leq \mathrm{var}(Z_j) / \left(\frac{m\varepsilon}{2}\right)^2 \underset{*}{=} \sum_T \mathrm{var}(Z_{T,j}) / \left(\frac{m\varepsilon}{2}\right)^2$$
$$\leq m / \left(\frac{m\varepsilon}{2}\right)^2 = 4/m\varepsilon^2 \leq 1/50n.$$

* needs pairwise-independence

- Overall, $\Pr[\text{"}x_1 \cdots x_n\text{"} = x] \geq \frac{1}{2m} \cdot \left(1 - \frac{1}{50n}\right)^n$
$> e^{-1/50}/m.$

output ↑ ↑ in $\mathcal{L}_f$

$\Rightarrow$ Repeating the above algorithm $2m$ times would yield the x whp.

Further, repeating it $(2m \cdot 2/\varepsilon^2)$ times would yield the list $\mathcal{L}_f$ with probability $\geq 2/3$. $\square$

Local list decoding RM

- Recall that RM "maps" $\binom{\ell+d}{d}$ evaluations of a d-deg ℓ-variate polynomial $P(\bar{x})$ to all $|\mathbb{F}|^\ell$ evaluations.

Our goal is to output $P(x)$, given an $x \in \mathbb{F}^\ell$, an oracle to a corrupted RM∘P & an advice. Let $q := |\mathbb{F}|$.

Theorem 2 (Sudan, Trevisan, Vadhan, 1999): RM has a local list decoder handling $1-10\sqrt{d/q}$ errors.

(Compare: RS list decoder handled $1-\sqrt{d/q}$ errors.)

Proof: • Idea — Given $x \in \mathbb{F}^\ell$ & an oracle f to a corrupted RMoP, randomly pick an $r \in \mathbb{F}$.

The advice is $(x_0, y_0) \in \mathbb{F}^{\ell+1}$ s.t. $P(x_0) = y_0$.

Let L_{x,x_0} be a random cubic curve passing through $(0, x)$ & $(r, x_0) \in \mathbb{F}^{\ell+1}$.

(L_{x,x_0} has the points $\{q(t) := (q_1(t), \ldots, q_\ell(t)) \mid t \in \mathbb{F}\}$ in $\mathbb{F}^\ell$, where q_i's are cubics.)

Query f on L_{x,x_0} & run RS list decoder to find a unique $g(t) = P \circ q(t)$ s.t. $g(r) = y_0$.

Output $g(0)$ $(= P(x)$ whp$)$.

• We will give a decoder that works for "most" of the input $x \in \mathbb{F}^\ell$.

This suffices as one can later use the "querying on a line" idea (of the RM local decoder) to make the above work $\forall x \in \mathbb{F}^\ell$.

Input: 1) Oracle f s.t. $\Pr_{x \in \mathbb{F}^\ell}[f(x) = P(x)] > 10\sqrt{d/q}$, and $|\mathbb{F}| > d^4$.

unknown poly. d-deg ℓ-var.

2) advice $(x_0, y_0) \in \mathbb{F}^\ell \times \mathbb{F}$. $[P(x_0)=y_0]$

3) $x \in \mathbb{F}^\ell$.

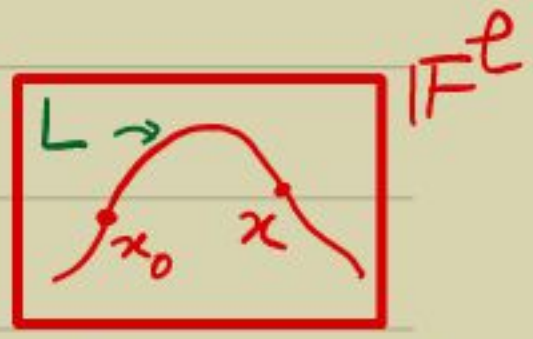
<u>Output:</u> $y \in \mathbb{F}$ [whp $\exists!\, P$ s.t. $P(x)=y$.]

<u>Decoder</u>:

1) Pick a random $r \in \mathbb{F}$ & monic cubics $q_i(t)$, $i \in [\ell]$, s.t. $q(t) := (q_1(t), \ldots, q_\ell(t))$ satisfies:

$$q(0) = x \ \& \ q(r) = x_0.$$

Define $L = L_{x,x_0} := \{q(t) \mid t \in \mathbb{F}\}$.



2) Query f on L to obtain $S := \{(t, f \circ q(t)) \mid t \in \mathbb{F}\}$.

3) Run RS list decoder on S to find the <u>list</u> $g_1, \ldots, g_k$ of all deg-$3d$ polynomials that agree on $\geq 8\sqrt{dq}$ pairs in S.

4) If $\exists$ <u>unique</u> i s.t. $g_i(r) = y_0$ then OUTPUT $g_i(0)$. Else FAIL.

<u>Analysis</u>: • Hypothesis on f implies that f agrees with a P on $\geq 8\sqrt{dq}$ points in L with probability ≥ 0.99.

[Hint: Since points other than $\{x, x_0\}$ on the <u>random</u> L are pairwise independent, we can use Chebyshev's inequality.]

• Assuming this agreement of f with P on L: the list-size $k \leq 8\sqrt{dq}/3d$. ← degree

$$\Rightarrow \Pr_r[\exists!\, i,\ g_i(r) = y_0] > 1 - k \cdot \Pr[g_i(r) = g_j(r)]$$
$$\geq 1 - k \cdot 3d/q \geq 1 - 8\sqrt{d/q} > 0.99$$

$\Rightarrow$ Overall, the decoder has success probability $> 0.99^2 > 0.98$ & the time-complexity is $\text{poly}(q, \ell)$. $\square$

<u>Remark</u>: The above shows that "most" (x_0, y_0) are an advice for x. Thus, $\exists (x_0, y_0)$ that works for "most" x.

Local list decoding WH∘RM

Theorem 3 (STV '99): $E_1 : \{0,1\}^n \to \Sigma^m$ resp. $E_2 : \Sigma \to \{0,1\}^k$ are ecc with local list decoders using advice from index-sets I_1 resp. I_2 & handling $1-\varepsilon_1$ errors resp. $1/2 - \varepsilon_2$ errors.

Then $E_2 \circ E_1$ has a local list decoder using advice from $I_1 \times I_2$ that handles $(1-\varepsilon_1|I_2|)\cdot(\frac{1}{2}-\varepsilon_2)$ errors.

needed to get an $i_2 \in I_2$ & then do ℓd_2

Proof sketch:

- Idea – Similar to that of their local decoder. $\square$

- From this we now deduce that for a worst-case hard f, $WH \circ RM \circ tt(f)$ is the truth-table of an average-case hard function.

Hardness amplification

Theorem (Impagliazzo-Wigderson '97; STV '99): Let $f \in \mathbb{E}$ be s.t. $H_{wors}(f) \geq S(n)$ for some $S: \mathbb{N} \to \mathbb{N}$. Then $\exists\, g \in \mathbb{E}$, $c \in \mathbb{R}_{>0}$ s.t. $H_{avg}(g) \geq S(n/c)^{1/c}$ for all sufficiently large n.

Proof:

- For large n, consider the $tt(f|_{\{0,1\}^n}) =: f_n \in \{0,1\}^N$, $N := 2^n$.
- Encode f_n to $g_{n'} = \mathrm{WHoRM}(f_n) \in \{0,1\}^{N'}$, where $N' = 2^{n'} = 2^{O(n)} = \mathrm{poly}(N)$.

  We will take $n' = 5n$ & see $g_{n'}$ as the truth-table of a function g on $\{0,1\}^{n'}$.

▷ Since $f \in \mathbb{E}$ & $g_{n'}$ is a $2^{n'}$-length string, we deduce that $g \in \mathbb{E}$.

- Parameters of RM: Fix a small $\delta \in \mathbb{R}_{>0}$.

Assume $n^2 \leq q \leq N$

$\to$ $q = |\mathbb{F}| = S(n)^{\delta} =: S^{\delta}$, $d = \sqrt{q}$, $\ell = \log_d N^2$.

$\Rightarrow$ RM-codeword length is $\binom{d+\ell}{\ell} \geq \left(1+\frac{d}{\ell}\right)^{\ell} \geq N$.

$\Rightarrow$ WH∘RM can encode f_n.

- Also, $N' = 2^{n'} = q^{\ell} \cdot q \leq N^4 \cdot N = N^5 = 2^{5n}$.

- Local list decoder: Let us use an lld for WH handling $\left(\frac{1}{2} - S^{-\delta/9}\right)$ errors & an lld for RM handling $\left(1 - 10 \cdot \sqrt{d/q}\right)$ errors.

$\Rightarrow$ lld for WH∘RM handles errors

$$\left(\frac{1}{2} - S^{-\delta/9}\right) \cdot \left(1 - 10 \cdot S^{-\delta/4} \cdot \frac{1}{2 \cdot (S^{-\delta/9})^2}\right)$$

$$\geq \left(\frac{1}{2} - S^{-\delta/9}\right) \cdot \left(1 - \frac{5}{S^{\delta/36}}\right) \geq \frac{1}{2} - S^{-\delta/10}.$$

- List-size $= O\left((S^{\delta/9})^2 \cdot \sqrt{q/d}\right) = O(\sqrt{q}) = O(\sqrt{N})$.

- If $\exists$ circuit $G_{n'} \in$ size$(S(n)^{\delta/10})$ computing g on $\frac{1}{2} + S^{-\delta/10}$ of the inputs in $\{0,1\}^{n'}$, then (WH∘RM-lld) $\circ\, G_{n'}$, with advice hardwired, yields a circuit of size $S^{c\delta} \cdot S^{\delta/10}$ computing f on $\{0,1\}^n$.

say, q^c is the complexity of WH∘RM

$\Rightarrow$ $\delta < (c + 0.1)^{-1}$ gives a contradiction.

$\Rightarrow g|_{\{0,1\}^{n'}}$ is $S(n'/5)^{\delta/10}$ - average-case hard. $\square$