CS250/EE387 - LECTURE 16 - LOCALITY!

AGENDA

0. RECAP + WARMUP w/ $RM_q(2,r)$.
1. LOCALLY CORRECTABLE CODES
2. RM CODES as LCCs
3. HIGH-RATE LCCs [sketch]

GASTROPOD FACT.

Slugs (which have small, internal shells) are thought to have evolved from snails (with large external shells). In fact, if you expose snail eggs to platinum, they will develop without shells, and the thought is that the mechansm for this has to do w/ the evolutionary snail → slug mechanism.

In the bay area, even gastropod conversations devolve into a discussion of housing costs.

0 RECAP.

Last time, we saw how to correct binary RM codes.
This algorithm had a nice property: we recovered the symbols c_s one-at-a-time, by looking LOCALLY at different sets.

THE APPROACH

This was the approach from last lecture

Find a bunch of disjoint sets that can "vote" for each codeword symbol. Not too many of these sets will be corrupted if there are not too many errors... so the majority vote will be the correct value!

The algs we saw last time are "LOCAL" in the sense that

or rather "exploit locality"

(a) We recover one symbol at a time
(b) We do it by looking at small groups of other symbols.

All this should make us wonder:

> If we ONLY want one symbol of the codeword, can we get it in SUBLINEAR ($o(n)$) TIME???

A LOCALLY CORRECTABLE CODE allows us to do this.
(And RM codes are LCCs).

① LOCALLY CORRECTABLE CODES.

Let us try to formalize the question in the box.

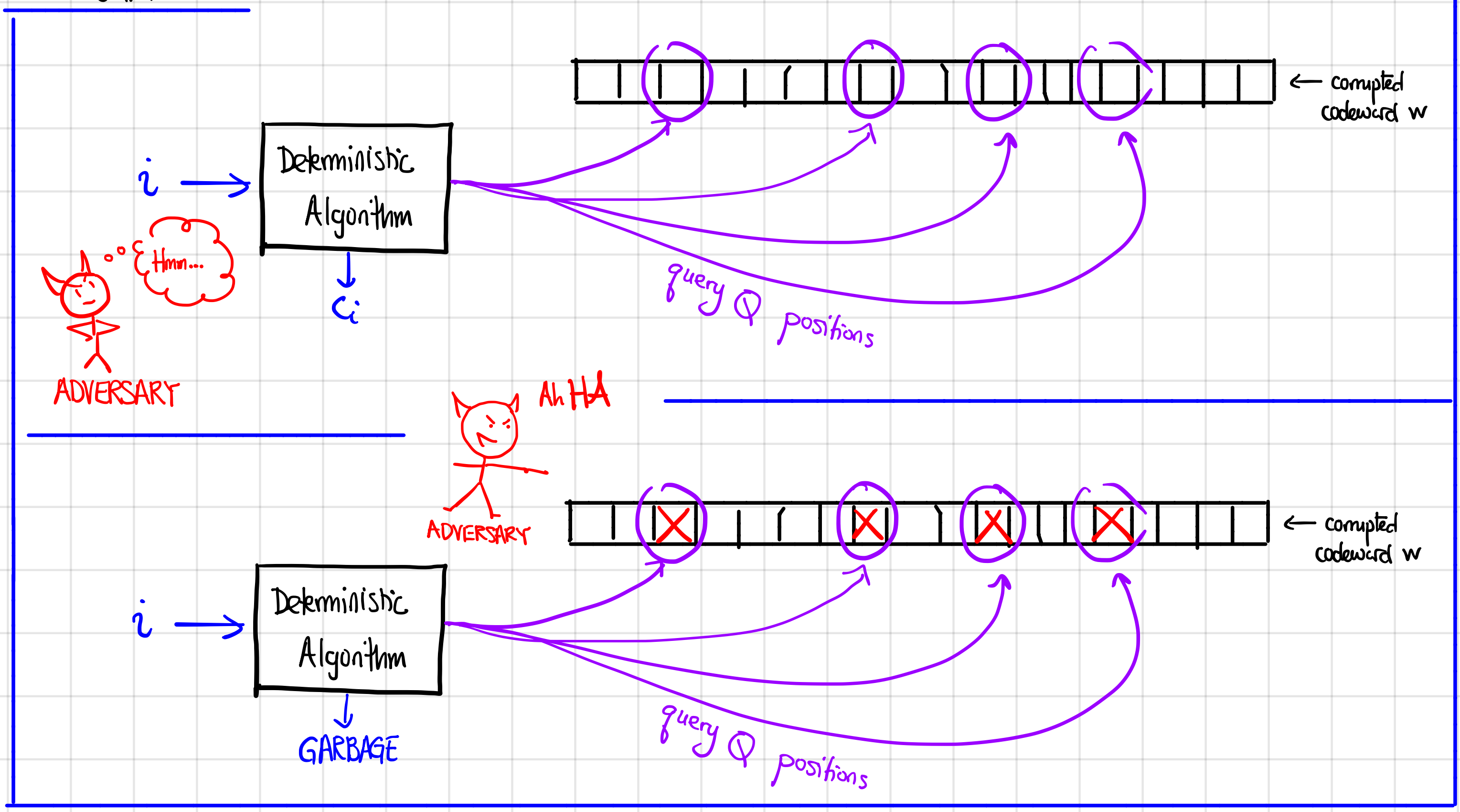
NOT THE CORRECT DEF.

> $\mathcal{C} \subseteq \mathbb{F}_q^n$ is a (δ, Q)- LOCALLY CORRECTABLE CODE (LCC) if there is an algorithm A so that the following holds:
>
> for all $w \in \mathbb{F}_q^n$ so that $\exists c \in \mathcal{C}$ s.t. $\Delta(c, w) \leq \delta n$, and $\forall i \in [n]$,
>
> $A^{(w)}(i)$ makes at most Q queries to w and returns c_i.
>
> input is i
>
> A has oracle access to w

Does this make sense? NO.

Sure, it parses, but if $Q = o(n)$ then this is a vacuous definition.

CARTOON:



That is, if the queries are deterministic, and $Q < \delta n$, then the adversary can COMPLETELY mess up the algorithm's view.

Instead, we will need to RANDOMIZE the queries if we want to deal with an adversary.

DEF. $\mathcal{C} \subseteq \mathbb{F}_q^n$ is a (δ, Q, γ)-LOCALLY CORRECTABLE CODE (LCC) if there is a randomized algorithm A so that the following holds:

for all $w \in \mathbb{F}_q^n$ so that $\exists c \in \mathcal{C}$ s.t. $\Delta(c, w) \leq \delta n$, and $\forall i \in [n]$,

- $A^{(w)}(i)$ makes at most Q queries to w

  input is i

  A has oracle access to w

- $A^{(w)}(i) = c_i$ with probability at least $1-\gamma$.

OTHER NOTIONS of LOCALITY:

- If you only want to recover a MESSAGE SYMBOL x_i instead of a CODEWORD SYMBOL c_i, it's called a LOCALLY DECODABLE CODE.

- If there's no adversary and you just want to be able to recover any symbol in SOME local way (not including that symbol) it's a LOCALLY REPAIRABLE CODE. (or LOCALLY RECOVERABLE CODE)

- Also: REGENERATING CODES, LOCALLY TESTABLE CODES, RELAXED LCCs, MAXIMALLY RECOVERABLE CODES, ...

Brief Lit. Review on LCC's.

Q	n (as a function of k)	Comments
2	$n = \Theta(2^k)$	Matching upper + lower bounds here.
3	$k^2 \le n \le \exp(\exp(\log\log(k)^{0.99}))$	The upper bd is actually an LDC
$O(\log(n))$	$k \le n \le \mathrm{poly}(k)$	
$O(n^\varepsilon)$	$k \le n \le (1+\alpha)k$ for any $\alpha > 0$	

Today we'll see how RM codes fit in, starting at $Q=2$ and ending at $Q = n^\varepsilon$.

2) RM codes as LCCs.

2A) $Q = 2$.

Last time, we saw the following alg for decoding the Hadamard Code:

ALG.

Input: $g : \mathbb{F}_2^m \to \mathbb{F}_2$ s.t. $\Delta(g,f) < 2^{m-2}$ for some $f \in RM_2(m,1)$

Output: f.

For $i = 1, \ldots, m$:

For each $\beta \in \mathbb{F}_2^m$:

Let $\hat{c}_i(\beta) = g(\beta) + g(\beta + e_i)$

Set $\hat{c}_i = \mathrm{MAJ}\{\hat{c}_i(\beta) : \beta \in \mathbb{F}_2^m\}$

Return $f(X_1, \ldots, X_m) = \sum_i \hat{c}_i X_i$.

We can easily modify this algorithm to be local:

ALG. Input: Query access to $g: \mathbb{F}_2^m \to \mathbb{F}_2$ s.t. $\Delta(g,f) < 2^{m-2}$ for some $f \in RM_2(m,1)$, and an index $i \in [m]$

Output: A guess for $f(e_i)$

Choose $\beta \in \mathbb{F}_2^m$ at random.
RETURN $g(\beta) + g(\beta + e_i)$

As stated that only recovers message symbols but it is easily modified to return any codeword symbol:

ALG. Input: Query access to $g: \mathbb{F}_2^m \to \mathbb{F}_2$ s.t. $\Delta(g,f) < 2^{m-2}$ for some $f \in RM_2(m,1)$, and an index $\alpha \in \mathbb{F}_2^m$

Output: A guess for $f(\alpha)$

Choose $\beta \in \mathbb{F}_2^m$ at random.
RETURN $g(\beta) + g(\beta + \alpha)$

CLAIM: $RM_2(m,1)$ is a $(\delta, 2, 1-2\delta)$-LCC for any $\delta < 1/4$.

proof. If $g(\beta) = f(\beta)$ and $g(\beta+\alpha) = f(\beta+\alpha)$ $(*)$

then $g(\beta) + g(\beta+\alpha) = f(\beta) + f(\beta+\alpha) = f(\alpha)$ since $\deg(f) = 1$.

Notice that it's ok if f has a constant term since it will cancel.

$(*)$ happens with probability $\geq 1 - 2\delta$, since

$\mathbb{P}\{g(\beta) \neq f(\beta)\} = \mathbb{P}\{g(\beta+\alpha) \neq f(\beta+\alpha)\} \leq \delta$, since β and $\alpha+\beta$ are both uniformly random.

(Notice that they are NOT jointly uniform, but each marginal is uniform).

GREAT! Now we have a 2-query LCC. But the rate is not great: $m/2^m$.

QUESTIONS:

1. Can we do better for $Q=2$?

   NO. See [Kerenadis+Wolf]

2. What if $Q=\omega(1)$?

   YES! Coming up next.

2B) $Q = \log(n)$

FIRST TRY: $RM_2(m,r)$, like we saw last time.
This gives us a local alg w/ $Q = 2^d$.

PROBLEM: If $Q=\omega(1)$, then the logic from above ("hope we avoid all the errors") fails badly. If Q is large, then WHP a δ-fraction of our queries will be corrupted.

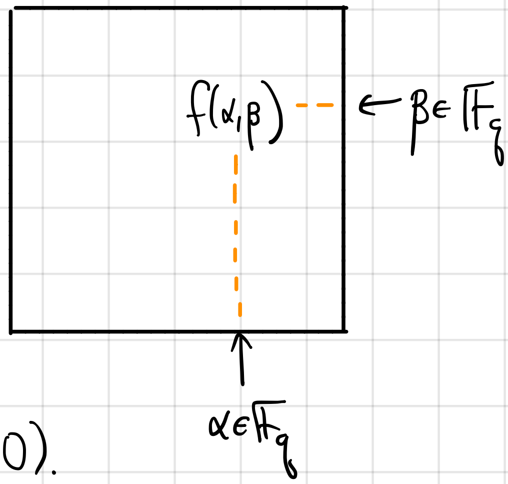
SECOND TRY: We'll still use an RM code, but over a bigger field.
This way, our queries will actually be robust to a little bit of noise.

For motivation, consider $RM_q(2,r)$.
That is, the codewords of $RM_q(2,r)$ are evaluations of bivariate polynomials

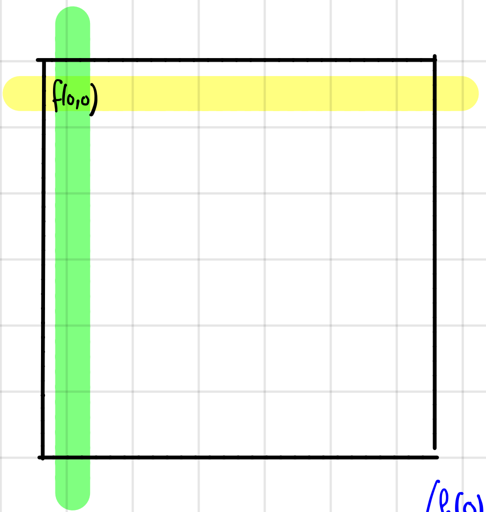
$$f(X,Y) = \sum_{i+j\le r} c_{i,j} X^i Y^j .$$

GOAL: Recover a single symbol (say, $f(\alpha,\beta)$) given query access to $g:\mathbb{F}_q^2 \to \mathbb{F}_q$ with $\Delta(g,f)\le\delta$.

We can think of codewords as $q \times q$ grids of evaluation points.



Suppose I want to recover $f(0,0)$.
As before, we want to find a bunch of LOCAL, LINEAR relationships involving $f(0,0)$.



$\leftarrow$ This row is $\left(f(0,0),\ f(0,\gamma),\ \ldots,\ f(0,\gamma^{q-1}) \right)$

$=: \left(g(0),\ g(\gamma),\ \ldots,\ g(\gamma^{q-1}) \right)$

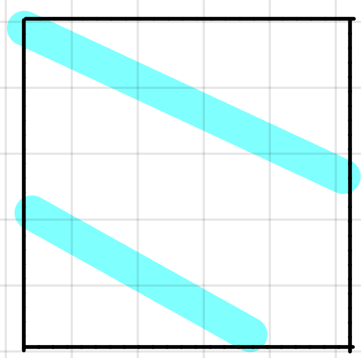
where $g(Y) := f(0,Y) = \sum_{i+j \le r} c_{ij}\, 0^i \cdot Y^j$

$= \sum_{j \le r} c_{0j} Y^j$

Similarly, this column is $\begin{pmatrix} h(0) \\ h(\gamma) \\ \vdots \\ h(\gamma^{q-1}) \end{pmatrix}$ where $h(X) = \sum_{j \le r} c_{j0} X^j$

Hey, those are univariate polynomials! (aka, RS codewords).

Moreover, the restriction of f to ANY line is an RS codeword!



Consider the line $L(Z) = \left(a_1 Z + b_1,\ a_2 Z + b_2 \right)$,
$a_i, b_i \in \mathbb{F}_q$.

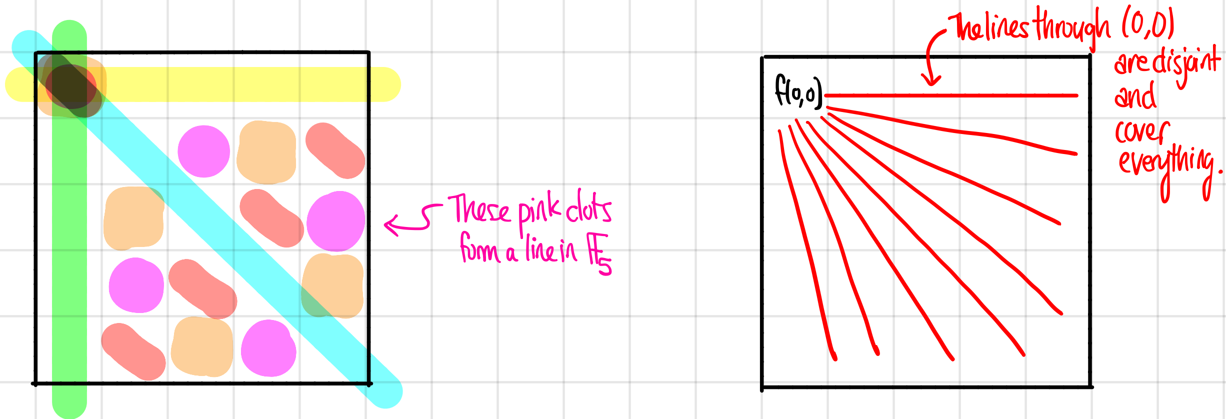
Then $f(L(Z)) = \sum_{i+j \le r} (a_1 Z + b_1)^i (a_2 Z + b_2)^j$

$=$ some degree $\le r$ polynomial in Z.

The lines through $f(0,0)$ have the properties we want:

- There are not too many (only q) points per line.
- Any two lines through $f(0,0)$ don't intersect anywhere else.

PICTURE(S):



Confusing but more accurate

Less accurate but hopefully more clear.

This inspires an algorithm:

ALG. (Let $r<q$ and suppose $\delta < \frac{1}{2}(1-r/q)$.)

Input: Query access to $g: \mathbb{F}_q^2 \to \mathbb{F}_q$ s.t. $\Delta(g,f) < \delta \cdot q^2$ for some $f \in RM_q(2,r)$.
and an index $(\alpha,\beta) \in \mathbb{F}_q^2$

Output: A guess for $f(\alpha,\beta)$.

Choose $(\sigma,\tau) \in \mathbb{F}_q^2 \setminus \{(0,0)\}$ at random, let $L(Z) = (\sigma \cdot Z + \alpha, \tau Z + \beta)$.

Query $g(L(\lambda))$ for all $\lambda \in \mathbb{F}_q$ and let $\tilde{h}(Z) := g(L(Z))$.

Use RS decoding to find an $h \in \mathbb{F}_q[Z]$, $\deg(h) \le r$, so that $\Delta(h, \tilde{h}) < \frac{q-r}{2}$

half the distance of the RS code.

RETURN $h(0)$.

CLAIM. For any $\delta > 0$, ALG is correct with prob. $\geq 1 - \left(\frac{2\delta q}{q-r-1}\right)$

Proof. The RS decoder will successfully find $h(z) = f(L(z))$ as long as the number of errors on $\{L(\lambda) : \lambda \in \mathbb{F}_q\}$ is $< \lfloor \frac{q-r}{2} \rfloor$, since $f(L(z)) \in RS_q(q, r+1)$.

$\mathbb{E}\{\text{ \#errors on a line }\} = \delta q$, so by Markov's inequality,

$$\mathbb{P}\left\{\text{ \#errors on a line } \geq \left\lfloor \frac{q-r}{2} \right\rfloor \right\} \leq \frac{\delta q}{\lfloor \frac{q-r}{2} \rfloor} \leq \frac{2\delta q}{q-r-1}$$

NOTE: If $\delta < \frac{1}{2}(1 - r/q) = \frac{1}{2}\,\mathrm{dist}(RM_q(2,r))$, then the failure probability above is interesting, otherwise it reads "with prob. ≥ 0."

For example:

$N = q^2$ here

COR. $RM_q(2, r = q/2) \subseteq \mathbb{F}_q^N$, is a $(Q = \sqrt{N}, \delta, 4\delta)$-LCC for any $\delta < \frac{1}{4}$. The rate is $\approx 1/8$ and the distance is $1/2$.

We can do EXACTLY the same thing with $m > 2$.

__LARGE-but-CONSTANT m:__

Then we get $Q = q = N^{1/m}$, since $N = q^m$.

However, as $m \uparrow$ then the rate $\downarrow$. Recall $R = \binom{q+m}{m} / q^m \leq \left(\frac{c}{m}\right)^m \to 0$ as $m \to \infty$.

But this does give us a constant-rate code w/ $Q = N^{1/100}$ (say).

__EVEN LARGER m:__

Choose $m = q/\log(q)$.

Then $N = q^{q/\log(q)} = 2^q$ so $Q = q = \log(N)$

But the rate is even worse, and in fact goes to 0 like $1/\mathrm{poly}(n)$.

This simple construction is the state-of-the-art for $\log(n)$ queries. Can you do better???

So far, we have seen how to use RM codes to get:

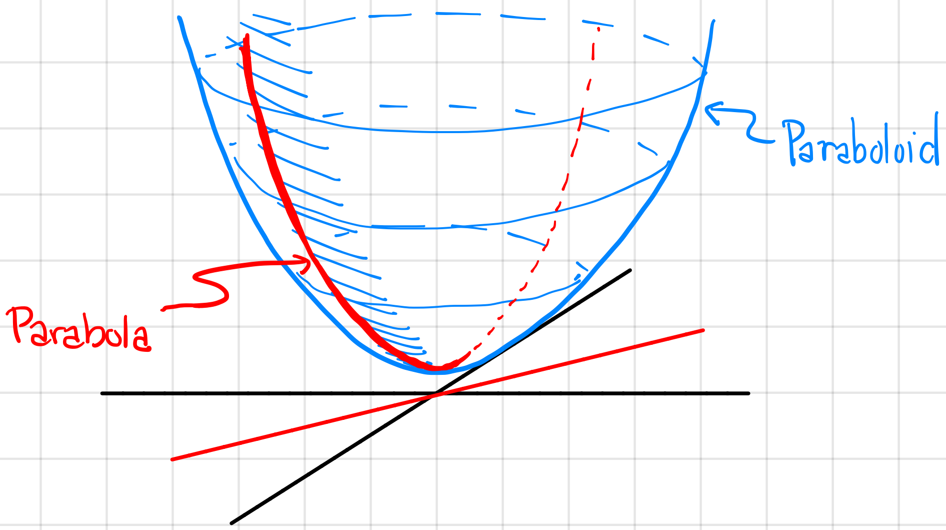
Q	n (as a function of k)	Code
2	$n = \Theta(2^k)$	$RM_2(m, 1)$
$\log(n)$	$n = \mathrm{poly}(k)$	$RM_q(m,r)$ for $m \approx q/\log(q)$, $q>r$
n^ε	$n = \Theta_\varepsilon(k)$	$RM_q(m,r)$ for $m = 1/\varepsilon$, $q>r$
$\sqrt{n}$	$n = 8k$	$RM_q(2, q/2)$

All of these have pretty low rate. Could we get an LCC with rate $\to 1$?

For $Q = n^\varepsilon$ (and even a bit smaller), the answer is YES.
There are several constructions. Here's a sketch of one based on RM codes.

2) HIGH-RATE LCCs.

The thing we needed from RM codes are that restrictions to lines are low-deg polys.



To make the rate better, we might try

$$\mathcal{C} = \left\{ (f(\alpha_1), \ldots, f(\alpha_{q^m})) : f \in \mathbb{F}_q[X], \text{ AND } \deg(f(L(z))) \le r \ \forall \text{ lines } L \right\}$$

This would be a win as long as $|\mathcal{C}| \geq |RM_q(m,r)|$, aka, as long as there are high-degree polynomials whose restrictions to lines are low-degree.

QUESTION. Does there exist a polynomial $f: \mathbb{F}_q^m \to \mathbb{F}_q$ of degree $> r$ so that, $\forall$ lines $L: \mathbb{F}_q \to \mathbb{F}_q^m$, $\deg(f(L(z))) \leq r$? (for $r < q-1$)

This means $\exists\, g(z)$ w/ $\deg(z) \leq r$ s.t. $g(\lambda) = f(L(\lambda))$ $\forall \lambda \in \mathbb{F}_q$

ANSWER.

Over $\mathbb{R}$ or $\mathbb{C}$: NO. (fun exercise!)

Over $\mathbb{F}_p$, for prime p: NO. (See [Rubinfeld-Sudan '96])

Over $\mathbb{F}_q$, and $q > 2 \cdot r$: NO. (" " ")

Over $\mathbb{F}_q$, and $q \approx r(1+\epsilon)$: YES, and there are LOTS of them. [Guo, Kopparty, Sudan '13]

EXAMPLE. Consider $f(X,Y) = X^2 Y^2$ over $\mathbb{F}_4$.

The degree of f is 4.
Any restriction of f to a line is equivalent to a polynomial of deg $\leq 3 = q-1$. (Not too helpful).

CLAIM $\forall$ lines $L: \mathbb{F}_4 \to \mathbb{F}_4^2$, $\deg(f(L(z))) \leq 2$.

pf. Say $L(z) = (\sigma Z + \alpha, \tau Z + \beta)$.

$$\begin{aligned} f(L(Z)) &= (\sigma Z + \alpha)^2 (\tau Z + \beta)^2 \\ &= (\sigma^2 Z^2 + \alpha^2)(\tau^2 Z^2 + \beta^2) && [(a+b)^2 = a^2 + b^2 \text{ in } \mathbb{F}_{2^t}] \\ &= \sigma^2 \tau^2 Z^4 + (\alpha^2 \tau^2 + \sigma^2 \beta^2) Z^2 + \alpha^2 \beta^2 && [\text{algebra}] \\ &\equiv (\alpha^2 \tau^2 + \sigma^2 \beta^2) Z^2 + \sigma^2 \tau^2 Z + \alpha^2 \beta^2. && [\alpha^4 = \alpha \text{ in } \mathbb{F}_4] \end{aligned}$$

That's just one example, but it turns out there are actually LOTS, enough so that

$$C = \{ (f(\alpha_1), \ldots, f(\alpha_{q^m})) : f \in \mathbb{F}_q[X], \text{ AND } \deg(f(L(z))) \leq r \ \forall \text{ lines } L \}$$

has $|C| \geq q^{(1-\epsilon)\cdot(q^m)}$, aka $\text{RATE}(C) \geq 1-\epsilon$.

C is called a "LIFTED CODE."

Thm (Guo, Kopparty, Sudan)

$\forall m > 0$, $q = 2^t$, $\forall \varepsilon > 0$, $\exists \varepsilon' > 0$ s.t. the set

$$S = \{ f : \mathbb{F}_q^m \to \mathbb{F}_q \mid f \text{ has degree } \leq (1-\varepsilon')\cdot q \text{ restrictions to ALL lines} \}$$

has $\dim(S) \geq (1-\varepsilon)\cdot q^m$

COR. $\forall \ \varepsilon, \alpha > 0$, $\exists \delta > 0$ and $\gamma > 0$ s.t. there exists a family of codes $C \subseteq \mathbb{F}_q^n$ so that C is a $(n^\alpha, \delta, \gamma)$-LCC of rate $1-\varepsilon$.

(One can do a bit better than this: see [Kopparty, Meir, Ron-Zewi, Saraf, 2015].)

RECAP:

- RM codes have nice local structure
- They are LCCs with $Q = 2$, $\log(n)$, $n^{1/100}$, although the rate gets bad.
- To get rate $1-\varepsilon$ with $Q = n^{1/100}$, we can "lift" RM codes.

- In general, there are TONS of open questions about LCCs!

QUESTIONS TO PONDER

1. Can you show that k must be at least $n^{2+\epsilon}$ for 3-query LCC's?
2. Can you beat RM codes for $Q = \log(n)$?
3. Can you do anything with the Hadamard code when $\delta > 1/4$?