In a previous class, we saw Reed-Muller locally decodable codes, with constant queries, where the encoding map sends: $k$ bits $\rightarrow 2^{k^c}$ bits. We then saw matching vector codes that improve this to $k \rightarrow 2^{2(\log k)^c}$ bits.

This lecture: Locally decodable code with constant rate, i.e. $k \rightarrow O(k)$ bits. As we mentioned earlier, it is known that constant rate cannot coexist with constant query local decoding.

1 Review of RM codes with constant rate

First, we consider RM code with the following parameters: code words are $m$ variables polynomials in $\mathbb{F}_q$ with degree at most $d$. So we have:

dim = number of monomials with total degree $d$ in $m$ variables = $\binom{d+m}{m}$

length = $q^m$.

$\delta = distance = 1 - \frac{d}{q}$ (i.e. $d = (1-\delta)q$)

Rate = $\left(\frac{d+m}{m}\right) \cdot \frac{1}{q^m} = \frac{d}{m!} \cdot \frac{1}{q^m}(1 + o(1)) = (1 + o(1)) \frac{1}{m!} (1 - \delta)^m$ Here think $m$ is constant.

We know this is locally decodable with $q = O(\frac{k}{1/m})$ queries. So we can see the rate is less than $\frac{1}{2}$ for any instantiation of RM codes. Now our question is: can we have locally decodable codes with $R > \frac{1}{2}$? (Discuss in class: maybe we can try RM code with other evaluate set. But it is not obvious to see whether it works or not.)

2 Multiplicity Code

Here we still works in $\mathbb{F}_q$, first set $m = 2$. Different from RM code, here we evaluate both values and derivatives on $\mathbb{F}_q^m$. This allows us to increase the degree parameter $d$ up to $2q$.

Message := $\{p(x,y) \in \mathbb{F}_q[x,y], \deg(p) \leq d\}$

Codeword of $p$ := Evaluate for each $(x,y) \in \mathbb{F}_q^2 (p(x,y), \frac{dp}{dx}(x,y), \frac{dp}{dy}(x,y))$.

(We may need Hasse derivative if it’s necessary)

Alphabet = $\Sigma = \mathbb{F}_q^3$.

Under these settings:

Number of Codewords = $q^\# \ of \ monomials \ of \ total \ deg \ d = q^{\binom{d+m}{m}}$

Rate = $\frac{\log_3(\# \ of \ codewords)}{q^2} = \frac{1/3(\binom{d+m}{m})}{q^2} = \frac{1}{3} \cdot \frac{d^2}{2} \cdot \frac{1}{2} = \frac{1}{6} \cdot \frac{d^2}{q^2}$

Why does this code have distance? For this we need a lemma.
Lemma 1. Multiplicity Schwartz-Zippel Lemma

Let \( p(x_1, \ldots, x_m) \in \mathbb{F}_q[x_1, \ldots, x_m] \) be non-zero polynomial with \( \deg(p) \leq d \). Then,

\[
\sum_{a \in S^m} \text{mult}(p, a) \leq d \cdot |S|^{m-1}
\]

Or equivalently,

\[
\mathbb{E}_{a \in S^m} \text{mult}(p, a) \leq \frac{d}{|S|}
\]

Using this Lemma we can bound the distance of the code. Let \( p \neq \tilde{p} \) be polynomial of \( \deg \leq d \), \( c \) and \( \tilde{c} \) be codewords. If \( c \) and \( \tilde{c} \) agrees in coordinate \((x, y)\). Then

\[
p(x, y) = \tilde{p}(x, y), \quad \frac{\partial p}{\partial x}(x, y) = \tilde{p}(x, y), \quad \frac{\partial p}{\partial y}(x, y) = \tilde{p}(x, y)
\]

Let \( Q = p - \tilde{p} \), then \( Q(x, y) = \frac{\partial Q}{\partial x}(x, y) = \frac{\partial Q}{\partial y}(x, y) = 0 \Rightarrow \text{mult}(p, (x, y)) \geq 2 \)

So every agreement of \( c, \tilde{c} \) contributes 2 to \( \sum_{a \in S^2} \text{mult}(p, a) \). By Multiplicity Schwartz-Zippel Lemma, number of agreements \( \leq \frac{dq}{2} \). Which means distance \( \geq 1 - \left( \frac{dq}{2q} \right) = 1 - \frac{d}{2q} \). Fix the distance to be \( \delta \), so we have \( d = 2(1 - \delta)q \), \( \text{Rate} = \frac{1}{6} \cdot d^2q^2 = \frac{2}{3}(1 - \delta)^2 \).

Now there are two things remains for us to do:
1. Show this code is locally decodable.
2. Prove the Multiplicity Schwartz-Zippel Lemma.

3 Local Decoding

Given \( r : \mathbb{F}_2^2 \to \Sigma \), s.t. \( \Delta(r, \text{codeword of } p) \leq \epsilon \). We want to recover \( (p(a), \frac{\partial p}{\partial x}(a), \frac{\partial p}{\partial y}(a)) \).

Why this is enough? Here instead of view the polynomial itself as message, we find a set of points that the values on these points uniquely defines the polynomial. View those values as our message (just as in the Reed-Muller case).

Pick \( b \in \mathbb{F}_q^2 \) uniformly at random, let \( L = \bar{a} + \bar{b}T \). Query \( r(\bar{a} + \bar{b}t) \) for all \( t \in \mathbb{F}_q^* \). Plan: To find \( Q(T) \). An easy observation is we can access ”noisy” version of \( Q(t) \) and \( \frac{\partial Q}{\partial T}(t) \) for \( t \in \mathbb{F}_q \).

\[
Q(T) = p(a_1 + b_1T, a_2 + b_2T), \quad \frac{\partial Q}{\partial T}(T) = \frac{\partial p}{\partial x} \cdot b_1 + \frac{\partial p}{\partial y} \cdot b_2.
\]

Now the problem are converted to the subproblem: Decoding univariable multiplicity codes. Given \( r : \mathbb{F}_q \to \mathbb{F}_2^2 \), find the unique polynomial \( Q(T) \in \mathbb{F}_q[T] \) of \( \deg \leq d \), s.t.

\[
|\{t \in \mathbb{F}_q \text{ s.t. } (Q(t), \frac{\partial Q}{\partial T}(t)) \neq r(t)\}| \leq \frac{1}{2} (1 - \frac{d}{2q})
\]

Left as an exercise, just like decoding R-S code (via Berlekamp Welch) w.h.p. over choice of line, we can recover \( Q(T) = p(a + bT) \). Which means we have:

\[
p(a) = Q(0), \quad b_1 \frac{\partial p}{\partial x}(a) + b_2 \frac{\partial p}{\partial y}(a) = \frac{\partial Q}{\partial T}(0)
\]
So we can see \[ \sum \delta \] from \[ \delta \] given codeword \( \Sigma = \mathbb{F}_q^{s+1} \), \( \deg = d = (1 - \delta) sq \).

Rate \( = \frac{s}{s+1} (1 - \delta)^2 \), \( dist = \delta \).

Local decoding using \( S \) lines, handles error fraction \( = \frac{\delta}{10} \cdot \frac{1}{s} \).

Instead, there is a variant of this decoding algorithm that decodes with 10 lines - this lets us decode from \( \frac{\delta}{10} \) fraction of errors for all \( s \).

So query complexity = \( 10sq = O(s \sqrt{k}) \).

Thus if we take \( m \) to be a big constant, \( \delta \) to be a very tiny constant \( \ll 1/m \), and \( s \) to be a massive constant \( \gg m^2 \), we get codes with rate nearly 1, and decodable with \( k^\epsilon \) queries from a constant \( (= \frac{\delta}{20}) \) fraction errors.

Today it is known how to reduce the query complexity to around \( 2\sqrt{\log k} \). This is done by combining multiplicity codes with subconstant relative distance with the Alon-Edmonds-Luby distance amplification trick to bring the relative distance back up to constant (one needs to understand how AEL interacts with local decoding here).

### 4 Proof of Multiplicity Schwartz-Zippel Lemma

Here we only show the easy case where \( m = 2 \). Our strategy is fix \( x = a \) and count the roots.

Assume that

\[ p(x, y) = p_0(x) + p_1(x)y + ... + p_t(x)y^t \]

such that \( \deg(p_i(x)) \leq d - i \), \( p_t(x) \neq 0 \).

Claim: For any \( a \in S \), \( \sum_{b \in S} \text{mult}(p, (a, b)) \leq \text{mult}(p_t, a) \cdot |S| + t \).

Claim \( \Rightarrow \) MSZ lemma:

\[
\sum_{a \in S} \sum_{b \in S} \text{mult}(P, (a, b)) \leq \left( \sum_{a \in S} \text{mult}(p_t, a) \right) \cdot |S| + t|S| \\
\leq \deg(p_t) \cdot |S| + t|S| \leq d|S|
\]

Proof of Claim:

Let \( M = \text{mult}(p_t, a) \), we have: \( \frac{\partial^i}{\partial x^i} p_t(a) = 0 \), \( \forall i < M \), \( \frac{\partial^M}{\partial x^M} p_t(a) \neq 0 \).

\[
\frac{\partial^i}{\partial x^i \partial y^j} p(x, y) = p_0^{(i)}(x) + ... + p_t^{(i)}(x)(y^t)^{(j)}
\]

Consider \( p^{(M,0)}(x, y) = +... + p_t^{(M)}(x)y^t \). Let \( Q(y) = p^{(M,0)}(a, y) = +... + p_t^{(M)}(a)y^t \). Recall that \( p_t^{(M)}(a) \neq 0 \).

So we can see \( \sum_{b \in S} \text{mult}(Q, b) \leq t \). So we only need to show:

\[ \text{mult}(p, (a, b)) \leq \text{mult}(Q, b) + M \]
which is easy to verify directly using the fact that $p^{(M,j)}(a, b) = Q^{(j)}(b)$. 
