Local decoding of Reed Muller codes

The RM codes over a field $\mathbb{F}_q$ is obtained as follows:

The message is of length $\binom{l+d}{d}$ where the bits of the message are considered as the coefficients of an l-variate polynomial of degree $\leq d$ over $\mathbb{F}_q$

Let $P(x_1, \ldots, x_n) = \sum_{i_1+i_2+\cdots+i_l \leq d} C_{i_1, i_2, \ldots, i_l} x_1^{i_1} x_2^{i_2} \cdots x_l^{i_l}$

The RM encoding is obtained by evaluating P at all points in $\mathbb{F}_q^l$.

ie the codeword is the string

$$\langle P(\alpha_1, \alpha_2, \ldots, \alpha_l) \rangle_{\alpha_1, \ldots, \alpha_l \in \mathbb{F}_q}$$

S-Z-DeM-L : Let $P(x_1, x_2, \ldots, x_n)$ be an n-variate non-zero polynomial of degree d over $\mathbb{F}$, where $d \leq |\mathbb{F}|$. Then, we have

$$\Pr_{\alpha_1, \ldots, \alpha_n \in_r \mathbb{F}} \left[P(\alpha_1, \ldots, \alpha_n) = 0 \right] \leq \frac{d}{|\mathbb{F}|}$$

The lemma above shows that the distance of the RM code is $\geq \left(1 - \frac{d}{q}\right) q^l$.

Now, let's look at the local decoding procedure for RM codes. Before that, we make a small change in the representation of our RM code. We will think of the message $x \in \mathbb{F}_q^{\binom{l+d}{d}}$ as the evaluation of an l-variate degree $\leq d$ polynomial at some $\binom{l+d}{d}$.

points, chosen in some arbitrary way. The polynomial $P(x_1, \dots, x_n)$ is then obtained by standard interpolation.

Let $y \in \mathbb{F}_q^l$ be the index of the msg. we want to recover, and assume that it has errors in atmost ρ fraction of points (we will fix ρ later).

i.e. $\Pr_{z \in_r \mathbb{F}_q^l} [f(x) \neq P(x)] \le \rho$, where f is the function that gives oracle access to the codeword

& P is the l-variate degree $\le d$ polynomial we are interested in.

The desired output is $x = P(y)$.

The algorithm proceeds as follows (we will describe the alg. & analysis together)

Choose $w \in_r \mathbb{F}_q^l$, and look at the line $\{y + t \cdot w \mid t \in \mathbb{F}_q\}$

This is a random line passing through y in $\mathbb{F}_q^l$.

The polynomial $P(y+tw)$ is a univariate polynomial of degree $\le d$ over $\mathbb{F}_q$

Since w is random, the points on this line are uniformly distributed. So, for ρ fraction of points $f(y+tw) \neq P(y+tw)$ in expectation. Therefore, by Markov's inequality, w.p $\geq 2/3$, the number of points such that f and P differ is at most $3\rho q$. If $\rho < \left(1 - \frac{d}{q}\right) \cdot \frac{1}{6}$, then, w.p $\geq 2/3$, the number of errors is at most $\left(1 - \frac{d}{q}\right) \cdot \frac{2}{2}$ and we use the RS decoder to obtain the univariate polynomial. $P(y+tw)$. Substituting $t=0$ gives $P(y)$.

Private information retrieval

- Want to query a database for some information.
- You want the answer, but don't want the database to know about it.

An r-server PIR protocol is a 3-tuple of algorithms (Q, A, ζ). We assume that each algorithm is given k (the length of the database) as an advice.

- At the beginning, the user obtains a random string rand
- Then it runs $Q(i, \text{rand})$ to obtain queries $q_1, q_2, \ldots, q_r$.
- For each $j \in [r]$, the user sends q_j to server S_j which returns an answer $A(j, x, q_j) = a_j$.
- Then the user obtains $b = \zeta(i, \text{rand}, a_1, a_2, \ldots, a_r)$

This protocol is a PIR protocol if the following holds.

1. For any k, x & $i \in [k]$, $b = x_i$ w.p. 1

2. Each server learns no information about i on its own. The distribution of q_i are identical for every $j \in [k]$.

Smooth decoders $\Longrightarrow$ PIR schemes.