Lecture 9: Matching Vector Codes

Sivakanth Gopi

October 23, 2019

In the last lecture, we have seen that Reed-Muller are LCCs and therefore also LDCs. But for constant q, the length of a q-query LCC based on Reed-Muller code is $\exp(k^{1/(q-1)})$ where k is the message length. In this lecture, we will construct Matching Vector Codes (MVCs) which are constant query LDCs of length $\exp(k^{o(1)})$. These codes are based on the so called Matching Vector Families (MVFs) which we will now define.

Matching Vector Families

Definition 1 (MVF). *Let $S \subset \mathbb{Z}_m \setminus \{0\}$ and let $\mathcal{F} = (\mathcal{U}, \mathcal{V})$ where $\mathcal{U} = (\mathbf{u}_1, \cdots, \mathbf{u}_k), \mathcal{V} = (\mathbf{v}_1, \cdots, \mathbf{v}_k)$ are lists of vectors $\mathbf{u}_i, \mathbf{v}_i \in \mathbb{Z}_m^d$. Then $\mathcal{F}$ is called an S-MVF over $\mathbb{Z}_m^d$ of size k (and dimension d) if $\forall\ i, j$,*

$$\langle \mathbf{u}_i, \mathbf{v}_j \rangle \begin{cases} = 0 & \text{if } i = j \\ \in S & \text{if } i \neq j \end{cases}$$

If S is omitted, it implies that $S = \mathbb{Z}_m \setminus \{0\}$.

Lemma 2. *If m is prime, then any MVF over $\mathbb{Z}_m^d$ must have size $k \leq 1 + d^{m-1}$.*

Given two vectors x, y of dimensions d_1, d_2 respectively, the tensor product $x \otimes y$ is a $d_1 d_2$-dimensional vector given by $(x \otimes y)_{ij} = x_i y_j$. $x^{\otimes \ell}$ denotes $x \otimes x \otimes \cdots \otimes x$ tensored ℓ times which will have dimension d_1^ℓ. Also note that $\langle x_1 \otimes y_1, x_2 \otimes y_2 \rangle = \langle x_1, x_2 \rangle \cdot \langle y_1, y_2 \rangle$.

Proof. Let $\mathcal{U} = (\mathbf{u}_1, \ldots, \mathbf{u}_k)$ and $\mathcal{V} = (\mathbf{v}_1, \ldots, \mathbf{v}_k)$ be the MVF. Consider the $k \times k$ matrix A given by $A_{ij} = \left\langle \mathbf{u}_i^{\otimes(m-1)}, \mathbf{v}_j^{\otimes(m-1)} \right\rangle = \langle \mathbf{u}_i, \mathbf{v}_j \rangle^{m-1}$. It is clear that $\text{rank}(A) \leq d^{m-1}$. By Femat's little theorem, A is equal to $J_k - I_k$ where J_k is the all ones matrix of size $k \times k$ and I_k is the identity matrix of size $k \times k$. Therefore $\text{rank}(A) \geq \text{rank}(I_k) - \text{rank}(J_k) = k - 1$. Combining both the bounds we get $k \leq 1 + d^{m-1}$. □

With a little more effort, we can extend Lemma 2 and show that for any prime power m, a MVF over $\mathbb{Z}_m^d$ must have size $k \leq d^{m-1}$. Thus for any constant prime power m, the size of a MVF can be only be polynomially larger than the dimension. Surprisingly, we can do much better if m is not a prime power!

Theorem 3 ([Gro99]). *Let $m = p_1 p_2 \cdots p_t$ where $p_1, p_2 \cdots, p_t$ are distinct primes with $t \geq 2$, then there exists an explicitly constructible S-MVF $\mathcal{F}$ in $\mathbb{Z}_m^d$ of size $k \geq \exp\left(\Omega\left(\frac{(\log d)^t}{(\log \log d)^{t-1}}\right)\right)$ for some set S of size $|S| = 2^t - 1$.*

We will prove this in a later lecture. In the special case when $p_1 = 2, p_2 = 3$, we have $m = 6$ (the smallest non-prime power) and the following corollary:

The set S in Theorem 3 can be described explicity as $S = \{a \in \mathbb{Z}_m : a \bmod p_i \in \{0,1\}\ \forall\, i \in [t]\} \setminus \{0\}$.

Corollary 4. *There is an explicitly constructible S-MVF $\mathcal{F}$ in $\mathbb{Z}_6^d$ of size $k \geq \exp\left(\Omega\left(\frac{(\log d)^2}{\log\log d}\right)\right)$ where $S = \{1,3,4\} \subset \mathbb{Z}_6$.*

LDCs from MVFs

We will now show how to get LDCs from MVFs. The codes we obtain from MVFs are called Matching Vector Codes (MVCs).

Theorem 5 ([Yeko7, Efro9])**.** *Suppose there exists a S-MVF over $\mathbb{Z}_m^d$ of size k. Let $\mathbb{F}$ be a finite field s.t. m divides $|\mathbb{F}| - 1$. Then there exists a (r, δ, η)-LDC $C : \mathbb{F}^k \to \mathbb{F}^n$ where $n = m^d, r = |S| + 1, \eta = \frac{1}{2} - r\delta$.*

In particular if we have $m = O(1)$ and $k = d^{\omega(1)}$, we get constant query LDCs of length $n = \exp(k^{o(1)})$ over constant size alphabet.

Construction of MVCs

Let $(\mathcal{U}, \mathcal{V})$ be a S-MVF over $\mathbb{Z}_m^d$ where $\mathcal{U} = (\mathbf{u}_1, \cdots, \mathbf{u}_k), \mathcal{V} = (\mathbf{v}_1, \cdots, \mathbf{v}_k)$. We will use the vectors in $\mathcal{U}$ while encoding and use the vectors in $\mathcal{V}$ while decoding. We will also need the following simple fact.

Fact 6. *If m divides $|\mathbb{F}| - 1$, then there exists $\gamma \in \mathbb{F}$ s.t. $\gamma^m = 1$ and $\gamma^i \neq 1$ for $0 < i < m$. (Such a γ is called an element of order m.)*

Proof. Let $q = |\mathbb{F}|$. We know that $\mathbb{F}^*$, the multiplicative group of non-zero elements of $\mathbb{F}$, is cylic. Suppose g is a generator for this group. Then $\gamma = g^{(q-1)/m}$ has order m. □

Fix some $\gamma \in \mathbb{F}$ of order m. Define the encoding $C : \mathbb{F}^k \to \mathbb{F}^n$ as follows. For $\mathbf{a} = (a_1, \cdots, a_k) \in \mathbb{F}^k$, define the function $F_{\mathbf{a}} : \mathbb{Z}_m^d \to \mathbb{F}$ as

Note that the function $F_{\mathbf{a}}$ is well defined because $\gamma^m = 1$.

$$F_{\mathbf{a}}(\mathbf{x}) = \sum_{i=1}^{k} a_i \gamma^{\langle \mathbf{x}, \mathbf{u}_i \rangle} \text{ where } \mathbf{x} = (x_1, \ldots, x_d) \in \mathbb{Z}_m^d.$$

The encoding $C(\mathbf{a})$ of a message $\mathbf{a} \in \mathbb{F}^k$ is the evaluation of $F_{\mathbf{a}}(\mathbf{x})$ at all points $\mathbf{x} \in \mathbb{Z}_m^d$, i.e.,

$$C(\mathbf{a}) = \langle F_{\mathbf{a}}(\mathbf{x}) \rangle_{\mathbf{x} \in \mathbb{Z}_m^d}.$$

Therefore the length of the encoding $n = m^d$.

Local decoding

Suppose we want to decode the message symbol a_τ (for some $\tau \in [k]$) given a corrupted codeword $f : \mathbb{Z}_m^d \to \mathbb{F}$ which is δ-close to $F_{\mathbf{a}} : \mathbb{Z}_m^d \to \mathbb{F}$. The local decoding algorithm is very similar to that of the Reed-Muller local decoding we have seen in the last class.

Let $r = |S| + 1$, note that $r \leq m$. Pick a random $\mathbf{z} \in \mathbb{Z}_m^d$ uniformly at random, and query f at r points on the line $\ell = \{\mathbf{z} + \lambda \mathbf{v}_\tau : 0 \leq \lambda \leq r - 1\}$. Since each point of the line ℓ is uniformly distributed over $\mathbb{Z}_m^d$, the probability that f agrees with $F_{\mathbf{a}}$ at every point we queried is at least $1 - r\delta$.

Now conditioned on this event, we have values of $F_{\mathbf{a}}(\mathbf{z} + \lambda \mathbf{v}_\tau)$ for $0 \leq \lambda \leq r - 1$. Let $p(\lambda)$ be the restriction of $F_{\mathbf{a}}$ to ℓ, then

$$\begin{aligned} p(\lambda) = F_{\mathbf{a}}(\mathbf{z} + \lambda \mathbf{v}_\tau) &= \sum_{i=1}^{k} a_i \gamma^{\langle \mathbf{z} + \lambda \mathbf{v}_\tau, \mathbf{u}_i \rangle} \\ &= \sum_{i=1}^{k} a_i \gamma^{\langle \mathbf{z}, \mathbf{u}_i \rangle + \lambda \langle \mathbf{v}_\tau, \mathbf{u}_i \rangle} \\ &= \sum_{\ell \in \{0\} \cup S} \left(\sum_{i : \langle \mathbf{v}_\tau, \mathbf{u}_i \rangle = \ell} a_i \gamma^{\langle \mathbf{z}, \mathbf{u}_i \rangle} \right) \gamma^{\lambda \ell}. \end{aligned}$$

Let $c_\ell = \sum_{i : \langle \mathbf{v}_\tau, \mathbf{u}_i \rangle = \ell} a_i \gamma^{\langle \mathbf{z}, \mathbf{u}_i \rangle}$, then we can write

$$p(\lambda) = c_0 + \sum_{\ell \in S} c_\ell \gamma^{\lambda \ell}.$$

Now comes the crucial observation. c_0 has only one term in its summation because $\langle \mathbf{v}_\tau, \mathbf{u}_i \rangle = 0$ iff $i = \tau$. Therefore $c_0 = a_\tau \gamma^{\langle z, \mathbf{u}_\tau \rangle}$. So to decode a_τ, it is enough to find c_0. Since we know the values of $p(\lambda)$ for r different values of λ, we get r linear equations in $|S| + 1 = r$ variables c_0 and $\{c_\ell : \ell \in S\}$. The linear systems of equations can be written as:

$$\begin{bmatrix} 1 & \dots & 1 & \dots \\ 1 & \dots & \gamma^\ell & \dots \\ 1 & \dots & (\gamma^\ell)^2 & \dots \\ \vdots & & \vdots & \\ 1 & \dots & (\gamma^\ell)^{r-1} & \dots \end{bmatrix} \begin{bmatrix} c_0 \\ \vdots \\ c_\ell \\ \vdots \end{bmatrix} = \begin{bmatrix} p(0) \\ p(1) \\ p(2) \\ \vdots \\ p(r-1) \end{bmatrix}. \tag{1}$$

This is invertible, because the coefficient matrix is a Vandermonde matrix. Therefore, we can solve find c_0 and from it we get $a_\tau = c_0 \gamma^{-\langle z, \mathbf{u}_\tau \rangle}$. This completes the proof of Theorem 5. □

Combining Theorem 5 and Theorem 3 we get the following corollary.

Corollary 7. *For every $t \geq 2$, there exists constants $C_t, D_t > 0$ depending only on t such that the following is true. There exists a $\left(2^t, \delta, \frac{1}{2} - 2^t\delta\right)$-LDC $C : \mathbb{F}^k \to \mathbb{F}^n$ where*

$$n = \exp\left(\exp\left(C_t(\log k)^{1/t}(\log\log k)^{1-1/t}\right)\right) \text{ and } |\mathbb{F}| \leq D_t.$$

Choosing $t = 2$, we get 4-query LDCs of length $n = \exp(\exp(\sqrt{\log k \log\log k})) = \exp(k^{o(1)})$ which is subexponential length. Note that 4-query LDCs that we can get from Reed-Muller codes have length $n = \exp(k^{1/3})$.

Though Corollary 7 gives constant query LDCs over some constant size alphabet, we can concatenate it with Hadamard code to get binary LDCs with the same query complexity (but worse error tolerance).

3-query LDCs of subexponential length

The smallest number of queries that Corollary 7 can give is 4. By being more careful, we can reduce the queries to just 3. We solved the linear system (1) just to find c_0. But we don't need to solve for all the variables, we just need solve for c_0. So we can hope to do it with fewer than $|S| + 1$ equations.

Let $g(x) \in \mathbb{F}[x]$ be a polynomial s.t. $g(\gamma^\ell) = 0$ for all $\ell \in S$ and $g(1) \neq 0$. Let $g(x) = \sum_{\lambda \in T} \alpha_\lambda x^\lambda$ for some coefficients $\alpha_\lambda = 1$. Such a polynomial is called an *S-decoding polynomial*. Suppose we know the values of $p(\lambda) = F_{\mathbf{a}}(\mathbf{z} + \lambda \mathbf{v}_\tau)$ for $\lambda \in T$. Then we can recover c_0 as:

$$\begin{aligned}
\sum_{\lambda \in T} \alpha_\lambda p(\lambda) &= \sum_{\lambda \in T} \alpha_\lambda \left(c_0 + \sum_{\ell \in S} c_\ell (\gamma^\ell)^\lambda\right) \\
&= c_0 \sum_{\lambda \in T} \alpha_\lambda + \sum_{\ell \in S} c_\ell \left(\sum_{\lambda \in T} \alpha_\lambda (\gamma^\ell)^\lambda\right) \\
&= c_0 g(1) + \sum_{\ell \in S} c_\ell g(\gamma^\ell) \\
&= c_0.
\end{aligned}$$

Therefore the number of queries is equal to the sparsity of the S-decoding polynomial. Note that trivially we can always get sparsity at most $|S| + 1$. But sometimes we can do better! For $m = 511 = 7 \cdot 73$ and $S = 1, 147, 365$ and $\mathbb{F} = \mathbb{F}_{2^9}$, there exists an S-decoding polynomial with sparsity 3 [Efr09]. This implies the existence of 3-query LDCs with subexponential length. It was later shown that any m of the form $m = 2^t - 1 = pq$, wher t, p, q are prime, has this property and $m = 511$ is the smallest such number [CFL+13].

Historical Note: Yekhanin introduced the idea of using MVFs to construct LDCs. But he was only working over prime m. By Lemma 2, he had to consider growing m to get subexponential size LDCs. And he constructed S-decoding polynomials with sparsity 3 for such growing m conditioned on their being infinitely many Mersenne primes (primes of the form $2^t - 1$). Efremenko showed that one can use non-prime m instead and used the construction of Grolmusz to finally construct 3-query subexponential LDCs unconditionally.

Open Problem 8. *Let $m = p_1 p_2 \dots p_t$ where $p_1, \dots, p_t$ are distinct primes and let $S = \{a : a \mod p_i \in \{0,1\}\ \forall i\} \setminus \{0\}$. What is the smallest sparsity of an S-decoding polynomial? For $t = 3$, can you get sparsity 3?*

It might be tempting to try to get an S-decoding polynomial with sparisty 2, that would give 2-query LDCs with subexponential length! But this is too good to be true. We can show that sparsity should be at least 3. But is there some deeper reason that this approach gets stuck at 3 queries? Yes! We will prove in the next few lectures that 2-query LDCs (over constant size alphabet) need to have exponential length.

In the next class we will prove Theorem 3 and see a lot of interesting open questions about MVFs.

References

[CFL+13] Yeow Meng Chee, Tao Feng, San Ling, Huaxiong Wang, and Liang Feng Zhang. Query-efficient locally decodable codes of subexponential length. *Computational Complexity*, 22(1):159–189, 2013.

[Efr09] Klim Efremenko. 3-query locally decodable codes of subexponential length. In *STOC*, pages 39–44, 2009.

[Gro99] Vince Grolmusz. Superpolynomial size set-systems with restricted intersections mod 6 and explicit ramsey graphs. *Combinatorica*, 20:2000, 1999.

[Yek07] Sergey Yekhanin. Towards 3-query locally decodable codes of subexponential length. In *Proceedings of the 39th annual ACM symposium on Theory of computing (STOC 2007)*, pages 266–274, 2007.