Matching Vector Codes *

Zeev Dvir[†]
IAS
zeev.dvir@gmail.com

Parikshit Gopalan
MSR SVC
parik@microsoft.com

Sergey Yekhanin
MSR SVC
yekhanin@microsoft.com

Abstract

An (r, δ, ϵ)-locally decodable code encodes a k-bit message x to an N-bit codeword $C(x)$, such that for every $i \in [k]$, the i-th message bit can be recovered with probability $1 - \epsilon$, by a randomized decoding procedure that queries only r bits, even if the codeword $C(x)$ is corrupted in up to δN locations.

Recently a new class of locally decodable codes, based on families of vectors with restricted dot products has been discovered. We refer to those codes as Matching Vector (MV) codes. Several families of $(r, \delta, \Theta(r\delta))$-locally decodable MV codes have been obtained. While codes in those families were shorter than codes of earlier generations, they suffered from having large values of $\epsilon = \Omega(r\delta)$. Codes with constant query complexity could only tolerate tiny amounts of error, and no MV codes of super-constant number of queries capable of tolerating a constant fraction of errors were known to exist.

In this paper we develop a new view of matching vector codes and uncover certain similarities between MV codes and classical Reed Muller codes. Our view allows us to obtain a deeper insight into power and limitations of MV codes. Specifically,

1. We show that existing families of MV codes can be enhanced to tolerate a nearly 1/8 fraction of errors, independent of the number of queries. Such enhancement comes at a price of a moderate increase in the number of queries;
2. Our construction yields the first families of matching vector codes of super-constant query complexity that can tolerate a constant fraction of errors. Our codes are shorter than Reed Muller LDCs for all values of $r \leq \log k/(\log \log k)^c$, for some constant c;
3. We show that any MV code encodes messages of length k to codewords of length at least $k2^{\Omega(\sqrt{\log k})}$. Therefore MV codes do not improve upon Reed Muller LDCs for $r \geq (\log k)^{\Omega(\sqrt{\log k})}$.

[*] An early version of this work has appeared as an ECCC report [Gop09].

[†] Research partially supported by NSF grants CCF-0832797 and DMS-0835373.

1 Introduction

Classical error-correcting codes allow one to encode a k-bit message x into an N-bit codeword $C(x)$, in such a way that x can still be recovered even if $C(x)$ gets corrupted in a number of coordinates. The disadvantage of classical error-correction is that one needs to read all or most of the (corrupted) codeword to recover any information about x. Suppose that one is only interested in recovering one or a few bits of x. In this case, more efficient schemes are possible allowing one to read only a small number of code positions. Such schemes are known as Locally Decodable Codes (LDCs). Locally decodable codes allow reconstruction of an arbitrary bit x_i, from looking only at $r << N$ randomly chosen coordinates of $C(x)$. While initial applications of locally decodable codes have been to data transmission and storage, they have found applications in other areas such as complexity theory and cryptography. See the surveys [Yek10, Yek07, Tre04, Gas04] for more information. Below is a slightly informal definition of LDCs:

An (r, δ, ϵ)-locally decodable code encodes k-bit messages x to N-bit codewords $C(x)$, such that for every $i \in [k]$, the bit x_i can be recovered with probability $1 - \epsilon$, by a randomized decoding procedure that makes only r queries, even if the codeword $C(x)$ is corrupted in up to δN locations.

We would like to have LDCs that have small values of r, N and ϵ and a large value of δ. However typically the parameters are not regarded as equally important. In applications of LDCs to data transmission and storage one wants δ to be a large constant, (ideally close to 1/4), and the codeword length N to be small. At the same time the exact number of queries r is not very important provided that it is much smaller than k. Similarly the exact value of $\epsilon < 1/2$ is not important since one can easily amplify ϵ to be close to 0, by running the decoding procedure few times and taking a majority vote. In applications of LDCs in cryptography one thinks of $\delta > 0$ and $\epsilon < 1/2$ as constants whose values are of low significance and focuses on the trade-off between r and N, with emphasis on very small values of r such as $r = 3$ or $r = 4$.

1.1 Three generations of locally decodable codes

The notion of locally decodable codes was explicitly discussed in various places in the early 1990s, most notably in [BFLS91, Sud92, PS94]. Katz and Trevisan [KT00] were the first to provide a formal definition of LDCs (see also [STV01]) and prove lower bounds on their length. Their bounds were improved in [GKST02, KdW04] where a tight (exponential) lower bound for the length of 2-query LDCs was obtained. Further lower bounds on the length of LDCs were obtained in [WdW05, Woo07, DJK$^+$02, Oba02]. The best lower bounds for the length of r-query LDCs currently have the form $\tilde{\Omega}\left(n^{1+1/(\lceil r/2 \rceil - 1)}\right)$ [Woo07]. They are very far form matching the best upper bounds.

One can informally classify the known families of locally decodable codes into three generations based on the technical ideas that underlie them. The first generation captures codes based on the idea of (low-degree) multivariate polynomial interpolation. All such codes [BFLS91, KT00, BIK05, CGKS98, Amb97, Ito99] are (directly or indirectly) based on classical (generalized) Reed Muller (RM) codes [MS77, vL82]. The code consists of evaluations of low degree polynomials in $\mathbb{F}_q[z_1, \ldots, z_n]$, at all points of $\mathbb{F}_q^n$, for some finite field $\mathbb{F}_q$. The decoder recovers the value of the unknown polynomial at a point by shooting a line in a random direction and decoding along it using noisy polynomial interpolation [BF90, Lip90, STV01]. The method behind these constructions is very general. It yields locally decodable codes of all possible query complexities, (i.e., one can choose r to be an arbitrary non-decreasing function of k) that tolerate a constant fraction of errors.

(We say that an r-query code C *tolerates* δ fraction of errors if C is (r, δ, ϵ)-locally decodable for some $\epsilon < 1/2$.)

The second generation of LDCs [BIKR02, WY05] combined the earlier ideas of polynomial interpolation with a clever use of recursion to show that Reed-Muller type codes are not the shortest possible for constant values of query complexity $r \geq 3$. Codes of the second generation are $(r, \delta, \Theta(r\delta))$-locally decodable. Thus the fraction of noise handled by these codes decays linearly with r. No LDCs of the second generation with $r = \omega(1)$ and $\delta = \Omega(1)$ are known to exist.

The latest (third) generation of LDCs was initiated in [Yek08] and developed further in [Rag07, KY09, Efr09, IS08]. New codes are obtained through an argument involving a mixture of combinatorial and algebraic ideas, where the key ingredient is a design of a large family of low dimensional (matching) vectors with constrained dot products. In what follows we refer to LDCs of the third generation as Matching Vector (MV) codes. Several families of $(r, \delta, \Theta(r\delta))$-locally decodable MV codes have been obtained. While codes in those families were dramatically shorter than codes of earlier generations, similarly to codes of [BIKR02, WY05] they suffered from having large values of $\epsilon = \Omega(r\delta)$. Codes with constant query complexity could only tolerate tiny amounts of error, and no MV codes with $r = \omega(1)$ capable of tolerating a constant fraction of errors were known to exist.

1.2 Our results

In this work we develop a new view of matching vector codes and uncover certain similarities between MV codes and classical Reed Muller codes. Our view allows us to obtain a deeper insight into the power and limitations of MV codes.

1. We show that existing families of MV codes can be enhanced to tolerate a nearly 1/8 fraction of errors, independent of the value of r, at a price of a moderate increase in the number of queries[1]. Specifically, for every constant $t \geq 2$, we obtain a family of $\left(t^{O(t)}, \delta, 4\delta(1 + O(1/\ln t))\right)$-locally decodable codes of length essentially identical to the length of the shortest known $\left(2^{O(t)}, \delta, 2^{O(t)}\delta\right)$-LDCs of [Efr09, IS08]. These codes encode messages of length k into codewords of length $\exp\exp\left((\log k)^{1/t}(\log\log k)^{1-1/t}\right)$.

2. We obtain the first families of matching vector codes of super-constant query complexity that can tolerate a constant fraction of errors, close to 1/8. Our codes are shorter than Reed Muller LDCs for all values of $r \leq \log k/(\log\log k)^c$, for some constant c.

3. We obtain a new family of MV codes, that matches the parameters of RM codes for $r = \Theta(\log k \log\log k)$.

4. The parameters of an MV code are determined by the parameters of the underlying family of matching vectors. We obtain bounds on the parameters of such families and conclude that any MV code encodes messages of length k to codewords of length at least $k2^{\Omega(\sqrt{\log k})}$. Therefore MV codes do not improve upon Reed Muller locally decodable codes for $r \geq (\log k)^{\Omega(\sqrt{\log k})}$.

1.3 Our techniques

Our constructions are centered around a new view of MV codes that fleshes out some intrinsic similarities between MV codes and RM codes. In our view an MV code consists of a linear subspace

[1] It is interesting to contrast our work with the work of Woodruff [Woo08] who obtained a non-linear transformation that (in certain circumstances) allows one to reduce LDC codeword length at a price of a *loss* in the value of δ.

of polynomials in $\mathbb{F}_q[z_1, \ldots, z_n]$, evaluated at all points of $\mathbb{C}_m^n$, where $\mathbb{C}_m$ is a certain multiplicative subgroup of $\mathbb{F}_q^*$. The decoding algorithm is similar to traditional local decoders for RM codes. The decoder shoots a line in a certain direction and decodes along it. The difference is that the monomials which are used are not of low-degree, they are chosen according to a matching family of vectors. (Two collections of vectors $\mathcal{U}, \mathcal{V} \subseteq \mathbb{Z}_m^n$ form a matching family if for every $\mathbf{u}_i \in \mathcal{U}$ there is a unique $\mathbf{v}_i \in \mathcal{V}$ such that $(\mathbf{u}_i, \mathbf{v}_i) = 0$, while other dot products $(\mathbf{u}_j, \mathbf{v}_i)$ belong to a small set $S \subseteq \mathbb{Z}_m \setminus \{0\}$.) Further, the lines for decoding are *multiplicative*.

Constructions of locally decodable codes from matching vectors have previously been considered in [Yek08] and [Rag07, Efr09, IS08]. In this work we show that if the family of matching vectors underling the MV code is *bounded* (meaning that dot products between all vectors $\mathbf{u} \in \mathcal{U}$ and $\mathbf{v} \in \mathcal{V}$ are small in $\mathbb{Z}_m$ with respect to the natural total ordering); then the restriction of a codeword of the MV code to a multiplicative line yields an evaluation of a low degree polynomial. Therefore one can apply existing techniques for noisy polynomial interpolation in the decoding process and tolerate a large fraction of errors. We further argue that the currently best known families of matching vectors (due to Grolmusz [Gro00]) can be turned into bounded, as well as obtain a new bounded matching family.

We also initiate a systematic study of families of matching vectors and prove upper bounds on their sizes. For the case when $m = p$ is a prime, our bounds are obtained by using the expansion of hyperplanes in $\mathbb{Z}_p^n$ when viewed as a collection of points. This bound beats the known bounds when the dimension n is small. Our bounds for composites are obtained via reductions to the prime case. These bounds in turn imply that any matching vector code must stretch messages of length k to codewords of length $k2^{\Omega(\sqrt{\log k})}$ for large enough k, regardless of the query complexity.

1.4 Outline

We start section 3 with formal definitions of locally decodable codes and matching families of vectors. We introduce the concept of a bounded matching family and show how any such family yields an LDC tolerating a large fraction of errors. In section 4 we present two constructions of bounded matching families. In section 5 we put the results of sections 3 and 4 together to obtain new upper bounds on the length of MV codes. In section 6 we obtain a collection of upper bounds on the size of matching families of vectors. In section 7 we translate the results of section 6 into lower bounds on the length of MV codes.

2 Notation

We use the following standard mathematical notation:

- $[k] = \{1, \ldots, k\}$;
- $\mathbb{F}_q$ is a finite field of q elements. $\mathbb{F}_q^*$ is the multiplicative group of $\mathbb{F}_q$;
- For a polynomial $f \in \mathbb{F}_q[z_1, \ldots, z_h]$ we denote by supp(f) the set of monomials with non zero coefficients in f, where a monomial $z_1^{e_1} \cdots z_h^{e_h}$ is identified with the integer h-tuple $(e_1, \ldots, e_h)$;
- $\mathbb{Z}_m$ is the ring of integers modulo an integer m. $\mathbb{Z}_m^*$ is the set of invertible elements of $\mathbb{Z}_m$;
- $d(\mathbf{x}, \mathbf{y})$ denotes the Hamming distance between vectors $\mathbf{x}$ and $\mathbf{y}$;

- $(\mathbf{u}, \mathbf{v})$ stands for the dot product of vectors $\mathbf{u}$ and $\mathbf{v}$;
- For a vector $\mathbf{w} \in \mathbb{Z}_m^n$ and an integer $l \in [n]$ let $\mathbf{w}(l)$ denote the l-th coordinate of $\mathbf{w}$;
- A D-evaluation of a function f defined over D, is a vector of values of f at all points of D.
- We write $\exp(x)$ to denote $2^{O(x)}$.

3 Matching vector codes: the framework

In this section we formally define locally decodable codes and matching families of vectors. We review the existing construction of LDCs from matching families, casting it in a new language that makes explicit certain intrinsic similarity between MV codes and RM codes. We then introduce the concept of a *bounded* matching family and show how MV codes based on these families can be decoded from large amounts of error.

Definition 1 *A q-ary code $C : \mathbb{F}_q^k \to \mathbb{F}_q^N$ is said to be (r, δ, ϵ)-locally decodable if there exists a randomized decoding algorithm $\mathcal{A}$ such that*

1. *For all $\mathbf{x} \in \mathbb{F}_q^k$, $i \in [k]$ and $\mathbf{y} \in \mathbb{F}_q^N$ such that $d(C(\mathbf{x}), \mathbf{y}) \leq \delta N$: $Pr[\mathcal{A}^{\mathbf{y}}(i) = \mathbf{x}(i)] \geq 1 - \epsilon$, where the probability is taken over the random coin tosses of the algorithm $\mathcal{A}$.*
2. *$\mathcal{A}$ makes at most r queries to $\mathbf{y}$.*

A locally decodable code is called linear if C is a linear transformation over $\mathbb{F}_q$. Our constructions of locally decodable codes are linear. While our main interest is in binary codes we deal with codes over larger alphabets as well.

Definition 2 *Let $S \subseteq \mathbb{Z}_m \setminus \{0\}$. We say that subsets $\mathcal{U} = \{\mathbf{u}_1, \ldots, \mathbf{u}_k\}$ and $\mathcal{V} = \{\mathbf{v}_1, \ldots, \mathbf{v}_k\}$ of vectors in $\mathbb{Z}_m^n$ form an S-matching family if the following two conditions are satisfied:*

- *For all $i \in [k]$, $(\mathbf{u}_i, \mathbf{v}_i) = 0$;*
- *For all $i, j \in [k]$ such that $i \neq j$, $(\mathbf{u}_j, \mathbf{v}_i) \in S$.*

We now show how one can obtain an MV code out of a matching family. We start with some notation.

- We assume that q is a prime power, m divides $q - 1$, and denote a subgroup of $\mathbb{F}_q^*$ of order m by $\mathbb{C}_m$;
- We fix some generator g of $\mathbb{C}_m$;
- For $\mathbf{w} \in \mathbb{Z}_m^n$, we define $g^{\mathbf{w}} \in \mathbb{C}_m^n$ by $(g^{\mathbf{w}(1)}, \ldots, g^{\mathbf{w}(n)})$;
- For $\mathbf{w}, \mathbf{v} \in \mathbb{Z}_m^n$ we define the multiplicative line $M_{\mathbf{w},\mathbf{v}}$ through $\mathbf{w}$ in direction $\mathbf{v}$ to be the multi-set
$$M_{\mathbf{w},\mathbf{v}} = \left\{ g^{\mathbf{w}+\lambda\mathbf{v}} \mid \lambda \in \mathbb{Z}_m \right\}; \tag{1}$$
- For $\mathbf{u} \in \mathbb{Z}_m^n$, we define the monomial $\mathrm{mon}_{\mathbf{u}} \in \mathrm{\mathbb{F}_q[z_1, \ldots, z_n]}$ by
$$\mathrm{mon}_{\mathbf{u}}(\mathrm{z_1, \ldots, z_n}) = \prod_{\ell \in [\mathrm{n}]} \mathrm{z}_\ell^{\mathbf{u}(\ell)}. \tag{2}$$

3.1 The general encoding/decoding framework

Observe that for any $\mathbf{w}, \mathbf{u}, \mathbf{v} \in \mathbb{Z}_m^n$ and $\lambda \in \mathbb{Z}_m$ we have

$$\mathrm{mon}_{\mathbf{u}}\left(g^{\mathbf{w}+\lambda\mathbf{v}}\right) = g^{(\mathbf{u},\mathbf{w})}\left(g^{\lambda}\right)^{(\mathbf{u},\mathbf{v})}. \tag{3}$$

The formula above implies that the $M_{\mathbf{w},\mathbf{v}}$-evaluation of a monomial $\mathrm{mon}_{\mathbf{u}}$ is a $\mathbb{C}_m$-evaluation of a (univariate) monomial

$$g^{(\mathbf{u},\mathbf{w})}y^{(\mathbf{u},\mathbf{v})} \in \mathbb{F}_q[y]. \tag{4}$$

This observation is the foundation of our decoding algorithms. We now specify the encoding procedure and outline the main steps taken by all decoding procedures described later on (propositions 3, 7, and 8). Let $\mathcal{U}, \mathcal{V}$ be an S-matching family in $\mathbb{Z}_m^n$.

Encoding: We encode a message $(\mathbf{x}(1), \ldots, \mathbf{x}(k)) \in \mathbb{F}_q^k$ by the $\mathbb{C}_m^n$-evaluation of the polynomial

$$F(z_1, \ldots, z_n) = \sum_{j=1}^{k} \mathbf{x}(j) \cdot \mathrm{mon}_{\mathbf{u}_\mathrm{j}}(\mathrm{z}_1, \ldots, \mathrm{z}_\mathrm{n}). \tag{5}$$

Notice that $F = F_{\mathbf{x}}$ is a function of the message $\mathbf{x}$ (we will omit the subscript and treat $\mathbf{x}$ as fixed throughout this section).

Basic decoding: The input to the decoder is a (corrupted) $\mathbb{C}_m^n$-evaluation of F and an index $i \in [k]$.

1. The decoder picks $\mathbf{w} \in \mathbb{Z}_m^n$ uniformly at random;
2. The decoder recovers the noiseless restriction of F to $M_{\mathbf{w},\mathbf{v}_i}$. To accomplish this the decoder may query the (corrupted) $M_{\mathbf{w},\mathbf{v}_i}$-evaluation of F at m or fewer locations.

To see that noiseless $M_{\mathbf{w},\mathbf{v}_i}$-evaluation of F uniquely determines x_i note that by formulas (3), (4) and (5) the $M_{\mathbf{w},\mathbf{v}_i}$-evaluation of F is a $\mathbb{C}_m$-evaluation of a polynomial

$$f(y) = \sum_{j=1}^{k} \mathbf{x}(j) \cdot g^{(\mathbf{u}_j,\mathbf{w})}y^{(\mathbf{u}_j,\mathbf{v}_i)} \in \mathbb{F}_q[y]. \tag{6}$$

Further observe that the properties of the S-matching family $\mathcal{U}, \mathcal{V}$ and (6) yield

$$f(y) = \mathbf{x}(i) \cdot g^{(\mathbf{u}_i,\mathbf{w})} + \sum_{s\in S}\left(\sum_{j\ :\ (\mathbf{u}_j,\mathbf{v}_i)=s} \mathbf{x}(j) \cdot g^{(\mathbf{u}_j,\mathbf{w})}\right) y^s. \tag{7}$$

It is evident from the above formula that

$$\mathbf{x}(i) = f(0)/g^{(\mathbf{u}_i,\mathbf{w})} \tag{8}$$

and that $\mathrm{supp}(f) \subseteq S \cup \{0\}$.

We now describe several local decoders that follow the general paradigm outlined above.

3.2 The simplest decoder

The proposition below gives the simplest local decoder. In the current form it has first appeared in [Efr09]. Earlier versions can be found in [Yek08, Rag07].

Proposition 3 *Let $\mathcal{U}, \mathcal{V}$ be a family of S-matching vectors in $\mathbb{Z}_m^n$, $|\mathcal{U}| = |\mathcal{V}| = k$, $|S| = s$. Suppose $m \mid q-1$, where q is a prime power; then there exists a q-ary linear code encoding k-long messages to m^n-long codewords that is $(s+1, \delta, (s+1)\delta)$-locally decodable for all δ.*

Proof: The encoding procedure has already been specified by formula (5). To recover the value $\mathbf{x}(i)$

1. The decoder picks $\mathbf{w} \in \mathbb{Z}_m^n$ at random, and queries the (corrupted) $M_{\mathbf{w},\mathbf{v}_i}$-evaluation of F at $(s+1)$ consecutive locations $\left\{ g^{\mathbf{w}+\lambda\mathbf{v}} \mid \lambda \in \{0, \ldots, s\} \right\}$ to obtain values $c_0, \ldots, c_s$.

2. The decoder recovers the unique sparse univariate polynomial $h(y) \in \mathbb{F}_q[y]$ with supp(h) $\subseteq S \cup \{0\}$ such that for all $\lambda \in \{0, \ldots, s\}$, $h(g^\lambda) = c_\lambda$. (The uniqueness of $h(y)$ follows from standard properties of Vandermonde matrices.)

3. Following the formula (8) the decoder returns $h(0)/g^{(\mathbf{u}_i,\mathbf{w})}$.

The discussion above implies that if all $(s+1)$ locations queried by the decoder are not corrupted then $h(y)$ is indeed the noiseless restriction of F to $M_{\mathbf{w},\mathbf{v}_i}$, and decoder's output is correct. It remains to note that each individual query of the decoder goes to a uniformly random location and apply the union bound. ■

Remark 4 In the proposition above we interpolate the polynomial $h(y)$ to recover its free coefficient. In certain cases (relying on special properties of the integer m and the set S) it may be possible to recover the free coefficient in ways that do not require complete interpolation and thus save on the number of queries. This general idea has been used in [Yek08], [Efr09] for the case of three-query codes, and in [IS08].

3.3 Improved decoding using bounded matching families

We now introduce the concept of a bounded matching family of vectors and show how MV codes based on bounded matching families can be decoded from large amounts of error. In what follows we identify $\mathbb{Z}_m$ with the subset $\{0, \ldots, m-1\}$ or real numbers. This imposes a total ordering on $\mathbb{Z}_m$, $0 < 1 < \ldots < m-1$ and allows us to compare elements of $\mathbb{Z}_m$ with reals.

Definition 5 *Let b be a positive real. A set $S \subseteq \mathbb{Z}_m$ is b-bounded if for all $s \in S$, $s < b$.*

Definition 6 *Let b be a positive real. An S-matching family $\mathcal{U}, \mathcal{V}$ in $\mathbb{Z}_m^n$ is b-bounded if the set S is b-bounded.*

Proposition 7 *Let σ be a positive real. Let $\mathcal{U}, \mathcal{V}$ be a σm-bounded family of S-matching vectors in $\mathbb{Z}_m^n$, $|\mathcal{U}| = |\mathcal{V}| = k$. Suppose $m \mid q-1$, where q is a prime power; then there exists a q-ary linear code encoding k-long messages to m^n-long codewords that is $(m, \delta, 2\delta/(1-\sigma))$-locally decodable for all δ.*

Proof: The encoding procedure has already been specified by formula (5). To recover the value $\mathbf{x}(i)$

1. The decoder picks $\mathbf{w} \in \mathbb{Z}_m^n$ at random, and queries every point of the (corrupted) $M_{\mathbf{w},\mathbf{v}_i}$-evaluation of F at all m locations $\left\{g^{\mathbf{w}+\lambda\mathbf{v}} \mid \lambda \in \mathbb{Z}_m\right\}$ to obtain values $c_0, \ldots, c_{m-1}$.
2. The decoder recovers the univariate polynomial $h(y) \in \mathbb{F}_q[y]$ of degree less than σm such that for all but at most $(m - \sigma m)/2$ values of $\lambda \in \mathbb{Z}_m$, $h(g^\lambda) = c_\lambda$. If such an h does not exist the decoder encounters a failure, and returns 0. Note that $\deg h < \sigma m$ implies that $h(y)$ is unique, if it exists. The search for $h(y)$ can be done efficiently using the Berlekamp-Welch algorithm [MS77].
3. Following the formula (8) the decoder returns $h(0)/g^{(\mathbf{u}_i,\mathbf{w})}$.

The discussion above implies that if the $M_{\mathbf{w},\mathbf{v}_i}$-evaluation of F is corrupted in at most $(m - \sigma m)/2$ locations, then $h(y)$ is indeed the noiseless restriction of F to $M_{\mathbf{w},\mathbf{v}_i}$, and the decoder's output is correct. It remains to note that each individual query of the decoder goes to a uniformly random location and thus by Markov's inequality the probability that more than $(m - \sigma m)/2$ of decoder's queries go to corrupted locations is at most $2\delta/(1 - \sigma)$. ■

3.4 Further improvement for small and bounded S

The improved decoding described in the previous section did not use any information on the size of the set S (only the fact that all elements in S are bounded). We now show that, in the case when $|S|$ is bounded *and* small (and $\ln q$ is small relative to m), one can get an even better result.

Proposition 8 *Let σ be a positive real. Let $\mathcal{U}, \mathcal{V}$ be a σm-bounded family of S-matching vectors in $\mathbb{Z}_m^n$, $|\mathcal{U}| = |\mathcal{V}| = k$, $|S| = s$. Suppose $m \mid q - 1$, where q is a prime power; then there exists a q-ary linear code encoding k-long messages to m^n-long codewords that is (r, δ, ϵ)-locally decodable for all α, δ, where*

$$r = \lceil (s + 2) \ln q / \alpha^2 \rceil,$$

$$\epsilon = \frac{2\delta}{(1 - \sigma - \alpha)}.$$

Proof: Our decoding algorithm is similar to the one of proposition 7. The saving in the number of queries comes from the fact that the decoder does not query all points on the multiplicative line but rather partitions the line into classes, and queries all points within a certain class. Our proof consists of two parts. Firstly, we establish the existence of an appropriate partition. Secondly, we present the decoding algorithm. We start with some notation. Let $\alpha > 0$ be fixed.

- Let $L \subseteq \mathbb{F}_q[y]$ be the linear space of polynomials whose support is contained in $\{0\} \cup S$;
- Let $T \subseteq \mathbb{Z}_m$. We say that T is α-regular, if for all $h \in L$ we have

$$\left| T \cap \left\{ \lambda \in \mathbb{Z}_m \mid h(g^\lambda) = 0 \right\} \right| < (\sigma + \alpha)|T|; \tag{9}$$

- Let $t \leq m$ be a fixed positive integer. Let π be a partition of $\mathbb{Z}_m$ into $p = \lfloor m/t \rfloor$ classes where each class is of size t or more
$$\mathbb{Z}_m = \bigsqcup_{\ell=1}^{p} \pi_\ell; \tag{10}$$
- We say that π is α-regular, if for each $\ell \in [p]$, π_ℓ, is α-regular.

We now argue that for a sufficiently large t, there exists a partition π satisfying (10) that is α-regular. Fix an arbitrary non-zero polynomial $h \in L$. Let $W = \left\{\lambda \in \mathbb{Z}_m \mid h(g^\lambda) = 0\right\}$. Clearly, $|W| < \sigma m$. Fix $t' \geq t$ and pick a set $T \subseteq \mathbb{Z}_m$ of size exactly t' uniformly at random.

$$\begin{gathered}\Pr\left[|T \cap W| \geq (\sigma + \alpha)t'\right] = \Pr\left[|T \cap W| - \sigma t' \geq \alpha t'\right] \leq \\ \Pr\left[|T \cap W| - \mathbb{E}(|T \cap W|) > \alpha t'\right] \leq \exp(-2\alpha^2 t),\end{gathered} \tag{11}$$

where the last inequality follows from [DP09, theorem 5.3].

Now let $t = \lceil (s+2) \ln q / 2\alpha^2 \rceil$. If $t > m$; then the proposition trivially follows from the proposition 7. We assume $t \leq m$ and pick π to be a random partition satisfying (10). Clearly, no class in π has size more than $2t - 1$. Relying on (11), the union bound, and $m/t < q$ we conclude that π is α-regular with positive probability since

$$(m/t)(q^{(s+1)} - 1) < e^{2\alpha^2 t}. \tag{12}$$

Fix an α-regular π. We are now ready to define the code. The encoding procedure has already been specified by formula (5). To recover the value $\mathbf{x}(i)$

1. The decoder picks $\mathbf{w} \in \mathbb{Z}_m^n$ and $\ell \in [p]$ uniformly at random, and queries points of the (corrupted) $M_{\mathbf{w},\mathbf{v}_i}$-evaluation of F at $|\pi_\ell|$ locations $\left\{g^{\mathbf{w}+\lambda\mathbf{v}} \mid \lambda \in \pi_\ell\right\}$ to obtain values $\{c_\lambda \mid \lambda \in \pi_\ell\}$.
2. The decoder recovers the univariate polynomial $h(y) \in \mathbb{F}_q[y]$ with $\mathrm{supp}(h) \subseteq \{0\} \cup S$ such that for all but at most $(1 - \sigma - \alpha)|\pi_\ell|/2$ values of $\lambda \in \pi_\ell$, $h(g^\lambda) = c_\lambda$. If such an h does not exist the decoder encounters a failure, and returns 0. Note that the properties of π imply that $h(y)$ is unique, if it exists.
3. Following the formula (8) the decoder returns $h(0)/g^{(\mathbf{u}_i,\mathbf{w})}$.

The discussion above implies that if at most $(1 - \sigma - \alpha)|\pi_l|/2$ locations queried by the decoder are corrupted; then $h(y)$ is indeed the noiseless restriction of F to $M_{\mathbf{w},\mathbf{v}_i}$, and the decoder's output is correct. It remains to note that each individual query of the decoder goes to a uniformly random location and thus by Markov's inequality the probability that more than $(1 - \sigma - \alpha)|\pi_l|/2$ queries go to corrupted locations is at most $2\delta/(1 - \sigma - \alpha)$, and to observe that the total number of queries is at most $2t - 1$. ■

3.5 From q-ary to binary codes

Propositions 7 and 8 yield non-binary locally decodable codes. As we remarked earlier our main interest is in binary LDCs. The next lemma extends proposition 7 to produce binary codes. The idea behind the proof is fairly standard and involves concatenation with a good binary error correcting code.

Lemma 9 *Let σ be a positive real. Let $\mathcal{U}, \mathcal{V}$ be a σm-bounded family of S-matching vectors in $\mathbb{Z}_m^n$, $|\mathcal{U}| = |\mathcal{V}| = k$. Suppose $m \mid q-1$, where $q = 2^b$. Further suppose that there exists a binary linear code C_{inner} of distance μB encoding b-bit messages to B-bit codewords; then there exists a binary linear code C encoding kb-bit messages to $m^n B$-bit codewords that is $(mB, \delta, 2\delta/(\mu - \mu\sigma))$-locally decodable for all δ.*

Proof: Observe that the condition of the lemma is strictly stronger than the condition of proposition 7. Thus the implication of proposition 7 holds. Let C_{outer} be the 2^b-ary $(m, \delta, 2\delta/(1-\sigma))$-LDC encoding k-long messages to m^n-long codewords. We define the code C to be the concatenation [MS77, vL82] of C_{outer} and C_{inner}. In order to decode a single bit, the decoder recovers the symbol of the large alphabet that the bit falls into.

Recall that in order to recover a single coordinate of the message the decoder of C_{outer} queries the corrupted encoding at m points of a multiplicative line, and than solves the Reed Solomon decoding problem (i.e., finds the unique univariate polynomial of degree less than σm that agrees with the observed sequence of values in all but at most $(1-\sigma)m/2$ locations).

The decoder of C acts similarly. Firstly, it entirely reads m corrupted B-bit codewords of the inner code that store (encoded) coordinate values of the outer code along a randomly chosen multiplicative line. Secondly, it decodes a binary code that is a concatenation of a Reed Solomon code of degree less then σm over $\mathbb{F}_{2^b}$ and a binary code C_{inner} up to half of its minimum distance. Decoding is correct provided that the total number of errors in mB locations read is at most $(1-\sigma)\mu mB/2$. Decoding can be done efficiently provided that C_{inner} has an efficient decoder.

It remains to note that each individual query of the decoder goes to a uniformly random location and thus by Markov's inequality the probability that more than $(1-\sigma)\mu mB/2$ of decoder's queries go to corrupted locations is at most $2\delta/(\mu - \mu\sigma)$. ■

Proposition 7 allows one to obtain LDCs over large alphabets that tolerate δ up to $1/4$. Lemma 9 allows one to obtain *binary* LDCs that tolerate δ up to $1/8$.

4 Matching vectors: constructions

In this section we present two constructions of bounded matching families of vectors (lemmas 13 and 14). Later in section 5 we use the first family of obtain MV codes that improve upon LDCs of [Efr09, IS08] in terms of the amount of noise that that can tolerate, and improve upon classical r-query RM LDCs in terms of codeword length for all $r \leq \log k/(\log\log k)^c$. We use the second family to obtain codes that (roughly) match RM LDCs for $r = \Theta(\log k \log\log k)$. Our first construction is based on an existing matching family due to Grolmusz [Gro00]. We argue that an appropriate scaling turns Grolmusz's family into bounded.

Definition 10 (Canonical set) *Let $m = \prod_{i=1}^t p_i$ be a product of distinct primes. The* canonical set *in $\mathbb{Z}_m$ is the set of all non-zero s such that for every $i \in [t]$, $s \in \{0,1\} \bmod p_i$.*

Basic parameters of Grolmusz's family are given by the following lemma. The lemma (as stated below) is new. The proof is modeled along the lines of Grolmusz's construction of a set system with restricted intersections modulo composites [Gro00, Gro02]. We defer the proof to appendix.

Lemma 11 *Let $m = \prod_{i=1}^{t} p_i$ be a product of distinct primes. Let w be a positive integer. Let $\{e_i\}$, $i \in [t]$ be integers such that for all i, we have $p_i^{e_i} > w^{1/t}$. Let $d = \max_i p_i^{e_i}$, and $h \geq w$ be arbitrary. Let S be the canonical set; then there exists an $\binom{h}{w}$-sized family of S-matching vectors in $\mathbb{Z}_m^n$, where $n = \binom{h}{\leq d}$.*

We now argue that a canonical set can be turned into a bounded one via scaling by an invertible element.

Lemma 12 *Let $m = \prod_{i=1}^{t} p_i$ be a product of distinct primes. Let S be the canonical set in $\mathbb{Z}_m$. There exists an $\alpha \in \mathbb{Z}_m^*$ such that the set αS is σm-bounded for any $\sigma > \sum_{i \in [t]} 1/p_i$.*

Proof: We start with some notation.

- For every $i \in [t]$, define the integer $\hat{p_i} = m/p_i$;
- Let $\alpha \in \mathbb{Z}_m^*$ be the unique element such that for all $i \in [t], \alpha = \hat{p_i} \bmod p_i$.

Observe that for any $i, j \in [t]$,

$$\left(\alpha^{-1}\hat{p_i}\right) \bmod p_j = \begin{cases} 1, & \text{if i=j;} \\ 0, & \text{otherwise.} \end{cases}$$

Let $s \in S$ be arbitrary. Set $I = \{i \in [t] \mid p_i \text{ does not divide s}\}$. Observe that $s = \alpha^{-1} \sum_{i \in I} \hat{p_i}$. Therefore

$$\alpha s = \sum_{i \in I} \hat{p_i} \leq m \sum_{i \in [t]} 1/p_i.$$

■

The lemma above implies that any S-matching family $\mathcal{U}, \mathcal{V}$ where S is the canonical set can be turned into a bounded one (by scaling all vectors in $\mathcal{V}$ by an invertible element). Combining it with lemma 11 we obtain

Lemma 13 *Let $m = \prod_{i=1}^{t} p_i$ be a product of distinct primes. Let w be a positive integer. Let $\{e_i\}$, $i \in [t]$ be integers such that for all i, we have $p_i^{e_i} > w^{1/t}$. Let $d = \max_i p_i^{e_i}$, and $h \geq w$ be arbitrary. Then there exists an $\binom{h}{w}$-sized σm-bounded family of matching vectors in $\mathbb{Z}_m^n$, where $n = \binom{h}{\leq d}$ and σ is an arbitrary real number larger than $\sum_{i \in [t]} 1/p_i$.*

The following lemma gives another (simple) family of bounded matching vectors.

Lemma 14 *Let m and n be arbitrary positive integers such that n is even and $m > n/2$. There exists an $(n/2)$-bounded $\binom{n}{n/2}$-sized family of matching vectors in $\mathbb{Z}_m^n$.*

Proof: Let $\mathcal{U}$ be the family of all vectors in $\{0,1\}^n$ that have weight exactly $n/2$. For every vector $\mathbf{u}_i \in \mathcal{U}$ set $\mathbf{v}_i$ to be its complement, (i.e., $\mathbf{v}_i$ is the unique binary vector such that $d(\mathbf{u}_i, \mathbf{v}_i) = n$.) It is not hard to see that the family $\mathcal{U}, \mathcal{V}$ is indeed $n/2$-bounded. ■

5 Upper bounds for MV codes

In this section we combine the results of the previous sections to derive upper bounds on MV codes. A combination of lemma 9 and lemma 13 yields

Lemma 15 *Let $m = \prod_{i=1}^{t} p_i$ be a product of distinct primes. Let w be a positive integer. Suppose integers $\{e_i\}$, $i \in [t]$ are such that for all i, we have $p_i^{e_i} > w^{1/t}$. Let $d = \max_i p_i^{e_i}$, and $h \geq w$ be arbitrary. Let σ is an arbitrary real number larger than $\sum_{i \in [t]} 1/p_i$. Suppose $m \mid q-1$, where $q = 2^b$. Further suppose that there exists a binary code C_{inner} of distance μB encoding b-bit messages to B-bit codewords; then there exists a binary linear code C encoding $\binom{h}{w} \cdot b$-bit messages to $m^{\binom{h}{\leq d}} \cdot B$-bit codewords that is $(mB, \delta, 2\delta/(\mu - \mu\sigma))$-locally decodable for all δ.*

In what follows we estimate asymptotic parameters of our codes.

Lemma 16 *There exists $c > 1$ such that for all integers $t \geq 2$ and $k \geq 2^{c2^t}$ there exists a binary linear code encoding k-bit messages to*

$$N = \exp\exp\left((\log k)^{1/t}(\log\log k)^{1-1/t} t \ln t\right)$$

-bit codewords that is $\left(t^{O(t)}, \delta, 4\delta(1 + O(1/\ln t))\right)$-locally decodable for all δ.

Proof: The proof follows by appropriately setting the parameters in lemma 15.

1. By [Sho05, theorem 5.7] there exists a universal constant c' such that the range $[(c'/2)t\ln t, c't\ln t]$ contains at least t distinct odd primes $p_1, \ldots, p_t$;

2. Note that $\sum_{i \in [t]} 1/p_i = O(1/\ln t)$;

3. Set $m = \prod_{i \in [t]} p_i$. Clearly, $m = t^{O(t)}$;

4. Set b to be the smallest positive integer such that $m \,\big|\, 2^b - 1$ Clearly, $b = t^{O(t)}$. Set $q = 2^b$;

5. A standard greedy argument (that is used to prove the classical Gilbert-Varshamov bound [MS77, vL82]) implies that there is a universal constant c'' such that for all integers $s \geq 1$, there exists a binary linear code of distance $(1/2 - c''/\sqrt{s})s^2$ encoding s-bit messages to s^2-bit codewords. We set C_{inner} to be a binary linear code that encodes b-bit messages to $B = t^{O(t)}$-bit codewords and has distance μB, for $\mu \geq \left(1/2 - c''/t^{\Omega(t)}\right)$;

6. We now assume that there exists a positive integer w which is a multiple of t such that $k = w^{w/t}$. Clearly, we have $w = \Theta(t \log k / \log\log k)$;

7. Following lemma 15 for every $i \in [t]$, let e_i be the smallest integer such that $p_i^{e_i} > w^{1/t}$. Let $d = \max_i p_i^{e_i}$. Clearly, $d = O(w^{1/t} t \ln t)$;

8. Set $h = w^{1+1/t}$;

9. $k = w^{w/t} \geq 2^{c2^t}$ yields $w^{1/t} > 2$ and $h > 2w$. Therefore $\binom{h}{w} b \geq (h/w)^w \geq k$;

10. We set the constant c large enough to ensure that (independent of t) we have $h > 2d$. This implies $\binom{h}{\leq d} \leq (eh/d)^d$. We set $N = m^{\binom{h}{\leq d}} B \leq t^{O(t)+x}$, where $x = O(t)(ew)^{O(w^{1/t} t \ln t)}$;

11. We combine lemma 15 with inequalities that we proved above and make basic manipulations to obtain a binary linear code encoding k-bit messages to $\exp\exp\left((\log k)^{1/t}(\log\log k)^{1-1/t} t \ln t\right)$-bit codewords that is $\left(t^{O(t)}, \delta, 4\delta(1 + O(1/\ln t))\right)$-locally decodable for all δ.

12. Finally, we note that the assumption about $k = w^{w/t}$, for some w can be safely dropped. If k does not have the required shape, we pad k-bit messages with zeros to get messages of length k', where k' has the shape $w^{w/t}$ and then apply the procedure above. One can easily check that such padding requires a sub-quadratic blow up in the message length and therefore does not affect asymptotic parameters.

■

Setting t to be a constant in lemma 16 yields

Theorem 17 *For every integer $t \geq 2$ and all sufficiently large integers k, there exists a binary linear code encoding k-bit messages to* $\exp\exp\left((\log k)^{1/t}(\log\log k)^{1-1/t}\right)$*-bit codewords that is* $\left(t^{O(t)}, \delta, 4\delta(1 + O(1/\ln t))\right)$*-locally decodable for all δ.*

For every constant $t \geq 2$, theorem 17 gives a family of $\left(t^{O(t)}, \delta, 4\delta(1 + O(1/\ln t))\right)$-locally decodable codes of length essentially identical to the length of the shortest known $\left(2^{O(t)}, \delta, 2^{O(t)}\delta\right)$-locally decodable codes of [Efr09, IS08]. Our codes can tolerate much larger amounts of noise, (i.e., for large values of t our codes tolerate approximately $1/8$ fraction of errors, while the fraction of errors tolerated by codes from earlier work drops to zero rapidly.) The improvement comes at a price of a moderate increase in the number of queries.

The following theorem gives asymptotic parameters of our codes in terms of r and k.

Theorem 18 *For every large enough integer r and every k, such that $k > 2^r$ there exists a binary linear code encoding k-bit messages to*

$$\exp\exp\left((\log k)^{O(\log\log r/\log r)}(\log\log k)^{1-\Omega(\log\log r/\log r)} \log r\right) \qquad (13)$$

bit codewords that is $(r, \delta, 4\delta(1 + O(1/\ln\ln r)))$-locally decodable for all δ.

Proof: The proof follows by setting parameters in lemma 16. Set t to be the largest integer such that $t^{O(t)} \leq r$, where the constant in O-notation is the same as the one in lemma 16. Assuming r is sufficiently large we have $t = \Theta(\log r/\log\log r)$. One can also check that $k > 2^r$ implies that the pre-condition of lemma 16 is satisfied. An application of the lemma concludes the proof. ■

5.1 MV codes over characteristic zero

We remark here that the entire construction and analysis of MV codes described in the preceding sections (apart from the parts dealing with reduction to the binary case) work also if the underlying field, $\mathbb{F}_q$, is replaced with the complex number field $\mathbb{C}$. The only property we used in $\mathbb{F}_q$ is that it contains an element of order m, which trivially holds over $\mathbb{C}$ for every m. This implies the existence of *linear* LDC's with essentially the same parameters as above also over the complex numbers (the definition of LDC's over an arbitrary field is the same as for finite fields, we simply allow the decoder to preform field arithmetic operations on its inputs). Once one has linear a code over the complex numbers, it is straight forward to get a code over the reals by writing each complex number as a pair of real numbers.

We find this interesting since, previous to MV codes, there were no known constructions of LDC's over characteristic zero (apart from trivial 2-query codes of exponential stretch). In fact, there are no known constructions, apart for MV codes, even over finite fields with very large characteristic (RM codes require that the characteristic will be at most the number of queries). Even though this might seem like an esoteric setting, LDC's over characteristic zero did came up in some recent works in connection to arithmetic circuit complexity [DS06, Dvi09].

5.2 Comparison to Reed Muller codes

Theorem 18 yields the first family of locally decodable codes (other than RM codes) that have super-constant query complexity and tolerate a constant fraction of errors. In this section we provide a comparison between RM codes and our codes.

A Reed Muller locally decodable code [KT00, Tre04, Yek10] is specified by three integer parameters. Namely, a prime power (alphabet size) q, number of variables n, and the degree $d < q-1$. The q-ary code consists of $\mathbb{F}_q^m$-evaluations of all polynomials in $\mathbb{F}_q[z_1, \ldots, z_n]$ of total degree at most d. Such code encodes $k = \binom{n+d}{d}$-long messages to q^n-long codewords. Provided that $d < \sigma(q-1)$, the code is $(q-1, \delta, 2\delta/(1-\sigma))$-locally decodable for all δ. If q is a power of 2 non-binary RM LDCs can be turned into binary via concatenation (in a manner similar to the one used in lemma 9). If one does concatenation with an asymptotically good code of relative distance μ one gets a binary linear code encoding k-bit messages to N-bit codewords that is $(r, \delta, 2\delta/(\mu - \mu\sigma))$-locally decodable for all δ, where

$$k = \binom{n+d}{d} \log q, \quad N = \Theta(q^n \log q), \quad r = \Theta(q \log q). \tag{14}$$

One can get various asymptotic families of RM LDCs by specifying an appropriate relation between n and d and letting these parameters grow to infinity. Increasing d relative to m yields shorter codes of larger query complexity.

Example 19 Setting $d = n$, $q = cn$ (for a constant c), and letting n grow while concatenating with asymptotically good binary codes of relative distance μ one gets a family of binary LDCs that encode k-bit messages to $k^{\Theta(\log \log k)}$-bit codewords and are $(\Theta(\log k \log \log k), \delta, 2\delta/(\mu - 2\mu/c))$-locally decodable for all δ.

We now argue that RM LDCs are inferior to codes of theorem 18 (with respect to codeword length) for all $r \leq \log k/(\log \log k)^c$, where c is a universal constant. To arrive at such a conclusion we need a lower bound on the length of RM LDCs. Let d, n, and q be such that formulas (14) yield an r-query LDC, where r belongs to the range of our interest. We necessarily have $d \leq n$ (otherwise $r > \log k$). Thus

$$k = \binom{n+d}{d} \log q \leq (en/d)^d \log q \leq n^{O(d)}, \tag{15}$$

and $n \geq k^{\Omega(1/d)}$. Therefore writing $\exp(x)$ to denote $2^{\Omega(x)}$, we have

$$N \geq \exp\exp\left(\log k/d\right) \geq \exp\exp\left(\log k/r\right). \tag{16}$$

Note that when r is a constant then already 3-query codes of [Efr09] improve substantially upon (16). To conclude the argument one needs to verify that there exists a constant c such that for every nondecreasing function $r(k)$, where $r(k)$ grows to infinity, and satisfies $r(k) \leq \log k/(\log\log k)^c$, for all sufficiently large k the right hand side of (16) evaluates to a larger value than (13).

Remark 20 It is interesting to observe that while MV codes of theorem 18 improve upon RM LDCs only for $r \leq \log k/(\log\log k)^c$, one can get MV codes that (asymptotically) match RM LDCs of example 19 combining lemma 14 (where m has the shape $2^b - 1$) with lemma 9.

6 Matching Vectors: limitations

Let $k(m,n)$ denote the size of the largest family of S-matching vectors in $\mathbb{Z}_m^n$ where we allow S to be an arbitrary subset of $\mathbb{Z}_m \setminus \{0\}$. The rate of any locally decodable code obtained via propositions 3, 7, or 8 is at most $k(m,n)/m^n$. Our goal in this section is to prove upper bounds on $k(m,n)$. In section 7 we translate these bounds into lower bounds on the length of MV codes. We start by bounding $k(m,n)$ in the prime case.

6.1 The Prime Case

We present two bounds for the prime case, the first is based on the linear algebra method [BF92] and is better for small p. It is tight when $p = 2$. Our second bound beats the linear algebra bound when p is large compared to n, it is tight for $n = 2$.

Theorem 21 *For any positive integer n and any prime p, we have*

$$k(p,n) \leq 1 + \binom{n+p-2}{p-1}.$$

Proof: Let $\mathcal{U} = \{\mathbf{u}_1, \ldots, \mathbf{u}_k\}$, $\mathcal{V} = \{\mathbf{v}_1, \ldots, \mathbf{v}_k\}$ be a family of S-matching vectors of $\mathbb{F}_p^n$, for some $S \subseteq \mathbb{F}_p^*$. For each $i \in [k]$, we consider the polynomial

$$P_i(z_1, \ldots, z_n) = 1 - \left(\sum_{l=1}^{n} \mathbf{v}_i(l) \cdot z_l\right)^{p-1}$$

in the ring $\mathbb{F}_p[z_1, \ldots, z_n]$. It is easy to see that $P_i(\mathbf{u}_i) = 1$, whereas $P_i(\mathbf{u}_j) = 0$ for all $j \neq i$. This implies that the k polynomials $\{P_i\}_{i=1}^k$ are linearly independent. But these polynomials all lie in an $\mathbb{F}_p$ vector-space of dimension $1 + \binom{n+p-2}{p-1}$, since they are spanned by the monomial 1 and all monomials of degree exactly $p-1$ in $z_1, \ldots, z_n$. ■

Remark 22 Let n be a positive integer and p be a prime. Suppose $n \geq p$, then

$$k(p,n) \geq \binom{n-1}{p-1}.$$

Proof: Take the $\mathbf{u}_i$s to be all vectors in $\{0,1\}^n$ of weight p where the first co-ordinate is 1, and let $\mathbf{u}_i = \mathbf{v}_i$. A simple calculation shows that $(\mathbf{u}_i, \mathbf{v}_i) = 0 \bmod p$, whereas for $i \neq j$ we have $(\mathbf{u}_j, \mathbf{v}_i) \in \{1, \ldots, p-1\}$. ■

If p is a constant, the first bound above nearly matches the bound in Theorem 21.

Our second bound comes from translating the problem of constructing matching vectors into a problem about points and hyperplanes in projective space. The $n-1$ dimensional projective geometry $\mathrm{PG}(\mathbb{F}_p, n-1)$ over $\mathbb{F}_p$ consist of all points in $\mathbb{F}_p^n \setminus \{0^n\}$ under the equivalence relation $\lambda\mathbf{v} \equiv \mathbf{v}$ for $\lambda \in \mathbb{F}_p^*$. Projective hyperplanes are specified by vectors $\mathbf{u} \in \mathbb{F}_p^n \setminus \{0^n\}$ under the equivalence relation $\lambda\mathbf{u} \equiv \mathbf{u}$ for $\lambda \in \mathbb{F}_p^*$; such a hyperplane contains all points $\mathbf{v}$ where $(\mathbf{u}, \mathbf{v}) = 0$.

We define a bipartite graph $H(U, V)$ where the vertices on the left correspond to all hyperplanes in $\mathrm{PG}(\mathbb{F}_p, n-1)$, vertices on the right correspond to all points in $\mathrm{PG}(\mathbb{F}_p, n-1)$ and $\mathbf{u}$ and $\mathbf{v}$ are adjacent if $(\mathbf{u}, \mathbf{v}) = 0$. For $X \subseteq U$ and $Y \subseteq V$, we define $N(X)$ and $N(Y)$ to be their neighborhoods. We use $N(\mathbf{u})$ for the neighborhood of $\mathbf{u}$.

Definition 23 *Let n be a positive integer and p be a prime. Let U be the set of hyperplanes in* $\mathrm{PG}(\mathbb{F}_p, n-1)$. *We say that a set $X \subseteq U$ satisfies the* unique neighbor property *if for every $\mathbf{u} \in X$, there exists $\mathbf{v} \in N(\mathbf{u})$ such that $\mathbf{v}$ is not adjacent to $\mathbf{u}'$ for any $\mathbf{u}' \in X \setminus \{\mathbf{u}\}$.*

Lemma 24 *Let n be a positive integer and p be a prime. Let U be the set of hyperplanes in* $\mathrm{PG}(\mathbb{F}_p, n-1)$. *There exists a set $X \subseteq U$, $|X| = k$ satisfying the unique neighbor property if and only if there exists a k-sized family of $\mathbb{Z}_p^*$-matching vectors in $\mathbb{Z}_p^n$.*

Proof: Assume that $X = \{\mathbf{u}_1, \ldots, \mathbf{u}_k\}$ satisfies the unique neighbor property. Let $Y = \{\mathbf{v}_1, \ldots, \mathbf{v}_k\}$ be such that $\mathbf{v}_i$ is a unique neighbor of $\mathbf{u}_i$. This implies that $(\mathbf{u}_i, \mathbf{v}_i) = 0$ and $(\mathbf{u}_j, \mathbf{v}_i) \neq 0$ for $i \neq j$. Thus X, Y gives a $\mathbb{Z}_p^*$-matching vector family in $\mathbb{Z}_p^n$.

For the converse, let us start with a k-sized matching vector family $\mathcal{U}, \mathcal{V}$ in $\mathbb{Z}_p^n$. In case $k = 1$ the lemma holds trivially. We claim that if $k \geq 2$; then $\mathbf{u} \in \mathcal{U}$ implies that $\lambda\mathbf{u} \notin \mathcal{U}$ for any $\lambda \in \mathbb{F}_p^* \setminus \{1\}$. This is true since $(\mathbf{u}, \mathbf{v}) = 0$ implies $(\lambda\mathbf{u}, \mathbf{v}) = 0$, which would violate the definition of a matching vector family. Thus we can associate each $\mathbf{u} \in \mathcal{U}$ with a distinct hyperplane in $\mathrm{PG}(\mathbb{F}_p, n-1)$. Similarly, we can associate every $\mathbf{v} \in \mathcal{V}$ with a distinct point in $\mathrm{PG}(\mathbb{F}_p, n-1)$. It is easy to see that $\mathbf{v}_i$ is a unique neighbor of $\mathbf{u}_i$, hence the set $\mathcal{U}$ satisfies the unique neighbor property. ■

Corollary 25 *Let n be a positive integer and p be a prime. Let U be the set of hyperplanes in* $\mathrm{PG}(\mathbb{F}_p, n-1)$. *The size of the largest set $X \subseteq U$ that satisfies the unique neighbor property is exactly $k(p, n)$.*

The expansion of the graph $H(U, V)$ was analyzed by Alon using spectral methods [Alo86, theorem 2.3]. We use the rapid expansion of this graph to bound the size of the largest matching vector family.

Lemma 26 *Let $n \geq 2$ be an integer and p be a prime. Let U (V) be the set of hyperplanes (points) in* $\mathrm{PG}(\mathbb{F}_p, n-1)$. *Let $u = \frac{p^n-1}{p-1} = |U| = |V|$. For any nonempty set $X \subseteq U$ with $|X| = x$,*

$$|N(X)| \geq u - u^{\frac{n}{n-1}}/x. \tag{17}$$

Lemma 27 *Let n be a positive integer and p be a prime; then*

$$k(p,n) \leq 4p^{n/2} + 2. \tag{18}$$

Proof: If $n = 1$, inequality (18) holds trivially. We assume $n \geq 2$. Let $\mathcal{U} \subseteq U$, $\mathcal{V} \subseteq V$ be a matching family of size $k(p,n)$. Pick $X \subseteq \mathcal{U}$ of size $x > 0$. By (17),

$$|N(X)| \geq u - u^{\frac{n}{n-1}}/x.$$

Since every point in $\mathcal{U} \setminus X$ must contain a unique neighbor from $V \setminus N(X)$, we have

$$|\mathcal{U} \setminus X| \leq |V \setminus N(X)| \leq \frac{u^{\frac{n}{n-1}}}{x} \;\Rightarrow\; |\mathcal{U}| \leq \frac{u^{\frac{n}{n-1}}}{x} + x. \tag{19}$$

Note that the inequality in the right hand side of (19) holds for all positive integers x. Picking $x = \left\lceil u^{\frac{n}{2(n-1)}} \right\rceil$ gives

$$|\mathcal{U}| \leq 2\left\lceil u^{\frac{n}{2(n-1)}} \right\rceil \leq 2\left(\frac{p^n}{p-1}\right)^{\frac{n}{2(n-1)}} + 2 = 2\left(\frac{p}{p-1}\right)^{\frac{n}{2(n-1)}} p^{n/2} + 2 \leq 4p^{n/2} + 2,$$

where the last inequality is a simple calculation. ■

For $n = 2$, one can construct matching vector families of size $\Omega(p)$. For instance, one can choose $\mathcal{U}$ to be the family of all vectors of the form $(1, x)$ and $\mathcal{V}$ to be the family of all vectors of the form $(x, -1)$ for all $x \in \mathbb{Z}_p$. Therefore one cannot replace $\frac{1}{2}$ in the exponent of (18) with a smaller constant.

6.2 The Prime Power Case

Lemma 28 *Let n be a positive integer, p be a prime and $e \geq 2$. We have*

$$k(p^e, n) \leq p^{(e-1)n} k(p, n+1).$$

Proof: Assume for contradiction that we have a matching family $\mathcal{U} = \{\mathbf{u}_1, \ldots, \mathbf{u}_k\}, \mathcal{V} = \{\mathbf{v}_1, \ldots, \mathbf{v}_k\}$ of size $k > p^{(e-1)n}k(p, n+1)$ in $\mathbb{Z}_{p^e}^n$. For every $i \in [k]$, write $\mathbf{u}_i = \mathbf{u}_i' + p^{e-1}\mathbf{u}_i''$ where $\mathbf{u}_i' \in \mathbb{Z}_{p^{e-1}}^n$ and $\mathbf{u}_i'' \in \mathbb{Z}_p^n$. By the pigeonhole principle, there are $k' > k(p, n+1)$ values of i which give the same vector $\mathbf{u}_i' \in \mathbb{Z}_{p^{e-1}}^n$, assume for convenience that the corresponding vectors in $\mathcal{U}$ are $\mathbf{u}_1, \ldots, \mathbf{u}_{k'}$ with matching vectors $\mathbf{v}_1, \ldots, \mathbf{v}_{k'}$. We will use these vectors to construct a matching vector family of size $k' > k(p, n+1)$ in $\mathbb{Z}_p^{n+1}$, which gives a contradiction.

For each $i \in [k']$, we extend $\mathbf{u}_i''$ to a vector $\bar{\mathbf{u}}_i$ by appending 1 in the last coordinate. For every $i \in [k']$, write $\mathbf{v}_i = \mathbf{v}_i' + p\mathbf{v}_i''$ where $\mathbf{v}_i' \in \mathbb{Z}_p^n$ and $\mathbf{v}_i'' \in \mathbb{Z}_{p^{e-1}}^n$. We extend $\mathbf{v}_i'$ to a vector $\bar{\mathbf{v}}_i$ by appending $(\mathbf{u}_i', \mathbf{v}_i)/p^{e-1} \in \mathbb{Z}_p$ in the last coordinate (we will show that this ratio is in fact integral).

We claim that for all $i \in [k']$, $(\bar{\mathbf{u}}_i, \bar{\mathbf{v}}_i) = 0 \bmod p$. To see this, observe that

$$(\bar{\mathbf{u}}_i, \bar{\mathbf{v}}_i) = (\mathbf{u}_i'', \mathbf{v}_i') + (\mathbf{u}_i', \mathbf{v}_i)/p^{e-1}. \tag{20}$$

But we have

$$(\mathbf{u}_i, \mathbf{v}_i) = (\mathbf{u}_i', \mathbf{v}_i) + p^{e-1}(\mathbf{u}_i'', \mathbf{v}_i) \equiv (\mathbf{u}_i', \mathbf{v}_i) + p^{e-1}(\mathbf{u}_i'', \mathbf{v}_i') = 0 \bmod p^e$$

From this we conclude that $(\mathbf{u}_i', \mathbf{v}_i) \equiv 0 \bmod p^{e-1}$, and that $(\mathbf{u}_i'', \mathbf{v}_i') + (\mathbf{u}_i', \mathbf{v}_i)/p^{e-1} = 0 \bmod p$. From equation (21), we conclude that $(\bar{\mathbf{u}}_i, \bar{\mathbf{v}}_i) = 0 \bmod p$. Next we claim that $(\bar{\mathbf{u}}_j, \bar{\mathbf{v}}_i) \neq 0 \bmod p$ for $i \neq j \in [k']$. We have

$$(\bar{\mathbf{u}}_j, \bar{\mathbf{v}}_i) = (\mathbf{u}_j'', \mathbf{v}_i') + (\mathbf{u}_i', \mathbf{v}_i)/p^{e-1} \tag{21}$$

But, since $\mathbf{u}_i' = \mathbf{u}_j'$, we also have

$$(\mathbf{u}_j, \mathbf{v}_i) = (\mathbf{u}_j', \mathbf{v}_i) + p^{e-1}(\mathbf{u}_j'', \mathbf{v}_i) \equiv (\mathbf{u}_i', \mathbf{v}_i) + p^{e-1}(\mathbf{u}_j'', \mathbf{v}_i') \not\equiv 0 \bmod p^e$$

which implies that $(\mathbf{u}_j'', \mathbf{v}_i') + (\mathbf{u}_i', \mathbf{v}_i)/p^{e-1} \not\equiv 0 \bmod p$. This shows that the vectors $\{\bar{\mathbf{u}}_j\}_{j=1}^{k'}$, $\{\bar{\mathbf{v}}_i\}_{i=1}^{k'}$ give a matching vector family of size $k' > k(p, n+1)$, which is a contradiction. ■

6.3 The Composite Case

Lemma 29 *Let m, n, and q be arbitrary positive integers such that $q|m$ and $(q, m/q) = 1$; then*

$$k(m,n) \leq (m/q)^n \, k(q,n).$$

Proof: Let us write $m/q = r$. Let $\mathcal{U} = \{\mathbf{u}_1, \ldots, \mathbf{u}_k\}$, $\mathcal{V} = \{\mathbf{v}_1, \ldots, \mathbf{v}_k\}$ be a family of S-matching vectors of $\mathbb{Z}_m^n$, for some $S \subseteq \mathbb{Z}_m \setminus \{0\}$. For each vector $\mathbf{u} \in \mathbb{Z}_m^n$ we can define the vectors $\mathbf{u}' \equiv \mathbf{u} \bmod q \in \mathbb{Z}_q^n$ and $\mathbf{u}'' \equiv \mathbf{u} \bmod r \in \mathbb{Z}_r^n$. From the definition of a matching vector family, we have that

- For all $i \in [k]$, $(\mathbf{u}_i', \mathbf{v}_i') = 0$ and $(\mathbf{u}_i'', \mathbf{v}_i'') = 0$;
- For all $i, j \in [k]$ such that $i \neq j$, $(\mathbf{u}_j', \mathbf{v}_i') \neq 0$ or $(\mathbf{u}_j'', \mathbf{v}_i'') \neq 0$.

Assume $k > (m/q)^n \, k(q,n)$. By the pigeonhole principle, there exists a vector $\mathbf{u} \in \mathbb{Z}_r^n$ such that $\mathbf{u}_j'' = \mathbf{u}$ holds for $k' > k(q,n)$ values of $j \in [k]$. Let us assume that these values are $1, \ldots, k'$. Note that for any $i, j \in [k']$ we have $(\mathbf{u}_j'', \mathbf{v}_i'') = (\mathbf{u}_i'', \mathbf{v}_i'') = 0$. Hence, by the definition of a matching family, we must have

- For all $i \in [k']$, $(\mathbf{u}_i', \mathbf{v}_i') = 0$;
- For all $i, j \in [k']$ such that $i \neq j$, $(\mathbf{u}_j', \mathbf{v}_i') \neq 0$.

Thus vectors $\{\mathbf{u}_1', \ldots, \mathbf{u}_{k'}'\}$ and $\{\mathbf{v}_1', \ldots, \mathbf{v}_{k'}'\}$ form a matching family mod q of size larger than $k(q,n)$ which gives a contradiction. ■

Theorem 30 *Let m and n be arbitrary positive integers. Suppose p is a prime divisor of m; then*

$$k(m,n) \leq 5 \frac{m^n}{p^{(n-1)/2}}.$$

Proof: Let p^e be the largest power of p which divides m. By lemmas 29, 28 and 27, we get

$$k(m,n) \leq \left(\frac{m}{p^e}\right)^n p^{(e-1)n} \left(4p^{(n+1)/2} + 2\right) \leq 5 \frac{m^n}{p^{(n-1)/2}}$$

■

The above bound is weak when n and p are constants, for instance it is meaningless for $n = 1$. We give another bound below which handles the case of small m. We start with the case when $n = 1$.

Lemma 31 *Let $m \geq 2$ be an arbitrary positive integer; then*

$$k(m, 1) \leq m^{O(1/\log\log m)} = m^{o_m(1)}.$$

Proof: Let $\mathcal{U} = \{\mathbf{u}_1, \dots, \mathbf{u}_k\}, \mathcal{V} = \{\mathbf{v}_1, \dots, \mathbf{v}_k\}$ be a family of $\mathbb{Z}_m \backslash \{0\}$-matching vectors in $\mathbb{Z}_m^1$. We treat every vector $\mathbf{u} \in \mathcal{U}$ as an integer and observe that for any $i \neq j \in [k]$, $\gcd(\mathbf{u}_i, m) \neq \gcd(\mathbf{u}_j, m)$. (Otherwise $(\mathbf{u}_i, \mathbf{v}_i) = 0$ would yield $(\mathbf{u}_j, \mathbf{v}_i) = 0$.) An application of a standard upper bound on the number of distinct divisors of an integer [HW85] concludes the proof. ■

We now proceed to the case of general n.

Theorem 32 *Let m and n be arbitrary positive integers; then*

$$k(m, n) \leq m^{n-1+o_m(1)}.$$

Proof: Given a vector $\mathbf{u} \in \mathbb{Z}_m^n$, we define the $\mathbb{Z}_m$-orbit of $\mathbf{u}$ to be the set of all vectors that can be written as $\lambda\mathbf{u}$ for $\lambda \in \mathbb{Z}_m$. Unlike over $\mathbb{Z}_p$, these orbits are no longer disjoint. We claim that all of $\mathbb{Z}_m^n$ can be covered by no more $\frac{m^n}{\phi(m)}$ orbits, and that each such orbit can contribute at most $k(m, 1)$ vectors to $\mathcal{U}$.

Let $U \subseteq \mathbb{Z}_m^n$ denote the set of all vectors $\mathbf{u}$ such that the GCD of all coordinates of $\mathbf{u}$ is 1. Any vector $\mathbf{u}' \in \mathbb{Z}_m^n$ can be written it as $\lambda\mathbf{u}$ for $\mathbf{u} \in U$ and $\lambda \in \mathbb{Z}_m$. Thus the orbits of vectors in U cover all of $\mathbb{Z}_m^n$. For $\mathbf{u}, \mathbf{u}' \in U$, we say that $\mathbf{u}' \equiv \mathbf{u}''$ if $\mathbf{u}''$ lies in the $\mathbb{Z}_m$ orbit of $\mathbf{u}'$. It is easy to see that this is indeed an equivalence relation on U, which divides U into equivalence classes of size $\phi(m)$. Thus if we pick $U' \subseteq U$ which contains a single representative of each equivalence class, then the orbits of U' contain all of $\mathbb{Z}_m^n$. Thus we have $|U'| = \frac{|U|}{\phi(m)} \leq \frac{m^n}{\phi(m)}$.

Now consider the orbit of any vector $\mathbf{u}$. Assume that it contributes the vector $\mathbf{u}_1 = \lambda_1\mathbf{u}, \dots, \lambda_t\mathbf{u}$ to $\mathcal{U}$ where $\lambda_i \in \mathbb{Z}_m$. Assume that the matching vectors in $\mathcal{V}$ are $\mathbf{v}_1, \dots, \mathbf{v}_t$. Then it is easy to see that $\mathcal{U}' = \{\lambda_1, \dots, \lambda_t\}$ and $\mathcal{V}' = \{(\mathbf{u}, \mathbf{v}_1), \dots, (\mathbf{u}, \mathbf{v}_t)\}$ are a matching vector family in one dimension, so that $t \leq k(m, 1)$. Thus we conclude that

$$k(m, n) \leq \frac{m^n}{\phi(m)} k(m, 1) \leq m^{n-1+o_m(1)}.$$

using a standard lower bound on $\phi(m)$ [HW85] and lemma 31. ■

7 Lower bounds for MV codes

We now translate the bounds on matching vector families from the previous section to bounds on the encoding length of matching vector codes. We argue that any family of (non-binary) matching vector codes, (i.e., codes that for some m and n, encode $k(m, n)$-long messages to m^n-long codewords) has an encoding blow-up of at least $2^{\Omega(\sqrt{\log k})}$.

Theorem 33 *Consider an infinite family of Matching Vector codes $C_\ell : \mathbb{F}_q^k \to \mathbb{F}_q^N$ for $\ell \in \mathbb{N}$, where $k = k(\ell)$ and $N = N(\ell)$ go to infinity with ℓ. For large enough ℓ, we have*

$$k \leq \frac{N}{2^{0.4\sqrt{\log N}}} \quad \Rightarrow \quad N \geq k 2^{0.4\sqrt{\log k}}.$$

Proof: For each ℓ, we have a family of matching vectors in $\mathbb{Z}_m^n$ where m, n depend on ℓ. We have $N = m^n$ while $k \leq k(m,n)$. First assume that $n > \sqrt{\log N}$. Then by theorem 30 with p a prime divisor of m, we have

$$k \leq \frac{5m^n}{p^{(n-1)/2}} \leq \frac{5N}{2^{0.5\sqrt{\log N}-1/2}} \leq \frac{N}{2^{0.4\sqrt{\log N}}}$$

where the last inequality holds for large enough N, and hence for all large ℓ. Hence assume that $n \leq \sqrt{\log N}$ so that $m \geq 2^{\sqrt{\log N}}$. As ℓ goes to infinity, N and hence m go to infinity. So for large enough ℓ, Theorem 32 gives $k(m,n) \leq m^{n-1+o_m(1)} \leq m^{n-0.9}$. Hence

$$k \leq \frac{m^n}{m^{0.9}} \leq \frac{N}{2^{0.9\sqrt{\log N}}}.$$

Thus $k \leq \frac{N}{2^{0.4\sqrt{\log N}}}$ for large enough ℓ. This implies that $N \geq k2^{0.4\sqrt{\log k}}$ for large enough ℓ. ■

One can generalize theorem 33 to get a similar statement for binary MV codes (i.e., codes obtained by a concatenation of a non-binary MV code with an asymptotically good binary code).

Theorem 34 *Let $\{m_\ell\}$ and $\{n_\ell\}$, $\ell \in \mathbb{N}$ be two arbitrary sequences of positive integers, such that ${m_\ell}^{n_\ell}$ monotonically grows to infinity. Consider an infinite family of binary codes $C_\ell : \mathbb{F}_2^{k_\ell} \to \mathbb{F}_2^{N_\ell}$ for $\ell \in \mathbb{N}$, where each code C_ℓ is obtained via a concatenation of an MV code encoding $k(m_\ell, n_\ell)$-long messages to $m_\ell^{n_\ell}$-long codewords over $\mathbb{F}_{q_\ell}$, (here $q_\ell = 2^t$ is the smallest such that $m_\ell \mid 2^t - 1$) with an asymptotically good binary code of some fixed rate; then for large enough ℓ the relative redundancy of C_ℓ is at least $2^{\Omega(\sqrt{\log k_\ell})}$.*

Proof: Pick a sufficiently large value of ℓ. Consider two cases

- $n_\ell \neq 1$. Observe that the argument in the end of section 6.1 yields $k(m_\ell, n_\ell) \geq m_\ell$. Next note that by theorem 33 relative redundancy of the non-binary code is at least $2^{\Omega\left(\sqrt{\log k(m_\ell,n_\ell)}\right)}$, and the concatenation with a binary code can only increase relative redundancy. Finally note that the dimension k_ℓ of the binary code is at most $k(m_\ell, n_\ell)m_\ell \leq k^2(m_\ell, n_\ell)$. Thus $2^{\Omega\left(\sqrt{\log k(m_\ell,n_\ell)}\right)} \geq 2^{\Omega(\sqrt{\log k_\ell})}$, for an appropriately chosen constant in Ω notation.
- $n_\ell = 1$. Set $k' = k(m_\ell, n_\ell)$. Be lemma 31, $k' = m_\ell^{o(1)}$. Note that $k_\ell = k't$ and $N_\ell = \Omega(mt)$, for some $t \leq m$. These conditions yield $N_\ell \geq \Omega\left(k_\ell^{3/2}\right)$.

■

7.1 Comparison with RM LDCs

Here we observe that it is possible to construct binary RM LDCs that have a blow-up of $2^{O(\sqrt{\log k})}$ and query complexity of $(\log k)^{O(\sqrt{\log k})}$. By formula (14) the relative redundancy of any RM LDC specified by parameters n, d and q is given by

$$k/N \leq O\left(\binom{n+d}{d}/q^n\right).$$

We assume that $n < d$; then $\binom{n+d}{d} \leq (2ed/n)^n$. Therefore (relying of $d \leq q$) we get

$$k/N \leq O((2e/n)^n).$$

Thus to have relative redundancy smaller than $2^{O(\sqrt{\log k})}$ it suffices to have

$$n = O\left(\sqrt{\log k}/\log\log k\right). \tag{22}$$

Given k we choose m to be the largest integer satisfying (22). Next we choose d to be the smallest integer satisfying $k \leq \binom{n+d}{d} \log q$. One can easily check that this yields $d = (\log k)^{O(\sqrt{\log k})}$, giving an RM LDC with desired parameters.

8 Conclusions

In this work we developed a new view of matching vector codes and uncovered certain similarities between MV codes and classical Reed Muller codes. Our view allowed us to obtain a deeper insight into the power and limitations of MV codes. We showed that similarly to Reed Muller codes MV codes constitute a rich code class containing codes of both constant and growing query complexities, capable of tolerating large amounts of noise. We showed that for query complexity $r \leq \log k/(\log\log k)^{O(1)}$ MV codes are superior to RM LDCs and for $r \geq (\log k)^{\Omega(\sqrt{\log k})}$ MV codes are inferior to RM codes. There are many questions that are left open by our work. We elaborate on two of them.

- It is very interesting to see if one can get MV codes that improve upon RM codes for values of r in the range $\log k/(\log\log k)^{O(1)} \leq r \leq (\log k)^{\Omega(\sqrt{\log k})}$. This calls for constructions of bounded matching families in $\mathbb{Z}_m^n$, where m is comparable to (or larger than) n.
- Our proposition 7 yields MV LDCs over large alphabets that tolerate δ up to $1/4$. Lemma 9 yields binary MV LDCs that tolerate δ up to $1/8$. The same critical values of δ apply to RM codes when one employs a local decoder that shoots a line and decodes along it [STV01]. However in case of RM LDCs one can use the local decoder of [GS92], that decodes along degree two curves rather than lines, and tolerates δ up to $1/2$ over large alphabets and δ up to $1/4$ over $\mathbb{F}_2$. It is interesting to see if one can get an analogous improvement in the case of MV codes.

References

[Alo86] Noga Alon. Eigenvalues, geometric expanders, sorting in rounds, and Ramsey theory. *Combinatorica*, 6:207–219, 1986.

[Amb97] Andris Ambainis. Upper bound on the communication complexity of private information retrieval. In *32th International Colloquium on Automata, Languages and Programming (ICALP)*, volume 1256 of Lecture Notes in Computer Science, pages 401–407, 1997.

[BBR94] David A. Barrington, Richard Beigel, and Steven Rudich. Representing Boolean functions as polynomials modulo composite numbers. *Computational Complexity*, 4:67–382, 1994.

[BF90] Donald Beaver and Joan Feigenbaum. Hiding instances in multioracle queries. In *7th International Symposium on Theoretical Aspects of Computer Science (STACS)*, volume 415 of Lecture Notes in Computer Science, pages 37–48, 1990.

[BF92] Laszlo Babai and Peter Frankl. *Linear Algebra Methods in Combinatorics, Preliminary version 2.* University of Chicago, 1992.

[BFLS91] Laszlo Babai, Lance Fortnow, Leonid Levin, and Mario Szegedy. Checking computations in polylogarithmic time. In *23th ACM Symposium on Theory of Computing (STOC)*, pages 21–31, 1991.

[BIK05] Amos Beimel, Yuval Ishai, and Eyal Kushilevitz. General constructions for information-theoretic private information retrieval. *Journal of Computer and System Sciences*, 71:213–247, 2005.

[BIKR02] Amos Beimel, Yuval Ishai, Eyal Kushilevitz, and Jean-Francios Raymond. Breaking the $O\left(n^{1/(2k-1)}\right)$ barrier for information-theoretic private information retrieval. In *43rd IEEE Symposium on Foundations of Computer Science (FOCS)*, pages 261–270, 2002.

[CGKS98] Benny Chor, Oded Goldreich, Eyal Kushilevitz, and Madhu Sudan. Private information retrieval. *Journal of the ACM*, 45:965–981, 1998.

[DJK$^+$02] A. Deshpande, R. Jain, T. Kavitha, S. Lokam, and J. Radhakrishnan. Better lower bounds for locally decodable codes. In *20th IEEE Computational Complexity Conference (CCC)*, pages 184–193, 2002.

[DP09] Devdatt P. Dubashi and Alessandro Panconesi. *Concentration of measure for the analysis of algorithms.* 2009.

[DS06] Z. Dvir and A. Shpilka. Locally decodable codes with 2 queries and polynomial identity testing for depth 3 circuits. *SIAM J. on Computing*, 36(5):1404–1434, 2006. (Extended abstract appeared in STOC '05).

[Dvi09] Z. Dvir. On matrix rigidity and locally self-correctable codes. Manuscript, 2009.

[Efr09] Klim Efremenko. 3-query locally decodable codes of subexponential length. In *41st ACM Symposium on Theory of Computing (STOC)*, pages 39–44, 2009.

[Gas04] William Gasarch. A survey on private information retrieval. *The Bulletin of the EATCS*, 82:72–107, 2004.

[GKST02] Oded Goldreich, Howard Karloff, Leonard Schulman, and Luca Trevisan. Lower bounds for locally decodable codes and private information retrieval. In *17th IEEE Computational Complexity Conference (CCC)*, pages 175–183, 2002.

[Gop06] Parikshit Gopalan. *Computing with Polynomials over Composites.* PhD thesis, Georgia Institute of Technology, 2006.

[Gop09] Parikshit Gopalan. A note on Efremenko's locally decodable codes. Electronic Colloquium on Computational Complexity (ECCC) TR09-069, 2009.

[Gro00] Vince Grolmusz. Superpolynomial size set-systems with restricted intersections mod 6 and explicit Ramsey graphs. *Combinatorica*, 20:71–86, 2000.

[Gro02] Vince Grolmusz. Constructing set-systems with prescribed intersection sizes. *Journal of Algorithms*, 44:321–337, 2002.

[GS92] Peter Gemmel and Madhu Sudan. Highly resilient correctors for polynomials. *Information processing letters*, 43:169–174, 1992.

[HW85] G.H. Hardy and E.M. Wright. *An Introduction to the Theory of Numbers.* Clarendon Press, Oxford, 1985.

[IS08] Toshiya Itoh and Yasuhiro Suzuki. New constructions for query-efficient locally decodable codes of subexponential lentgh. Arxiv 0810.4576, October 2008.

[Ito99] Toshiya Itoh. Efficient private information retrieval. *IEICE Trans. Fund. of Electronics, Commun. and Comp. Sci.*, E82-A:11–20, 1999.

[KdW04] Iordanis Kerenidis and Ronald de Wolf. Exponential lower bound for 2-query locally decodable codes via a quantum argument. *Journal of Computer and System Sciences*, 69:395–420, 2004.

[KT00] Jonathan Katz and Luca Trevisan. On the efficiency of local decoding procedures for error-correcting codes. In *32th ACM Symposium on Theory of Computing (STOC)*, pages 80–86, 2000.

[KY09] Kiran S. Kedlaya and Sergey Yekhanin. Locally decodable codes from nice subsets of finite fields and prime factors of Mersenne numbers. *SIAM Jounrnal on Computing*, 38:1952–1969, 2009.

[Lip90] Richard Lipton. Efficient checking of computations. In *7th International Symposium on Theoretical Aspects of Computer Science (STACS)*, volume 415 of Lecture Notes in Computer Science, pages 207–215, 1990.

[MS77] F.J. MacWilliams and N.J.A. Sloane. *The theory of error correcting codes.* 1977.

[Oba02] Kenji Obata. Optimal lower bounds for 2-query locally decodable linear codes. In *6th International Workshop on Randomization and Computation (RANDOM)*, volume 2483 of Lecture Notes in Computer Science, pages 39–50, 2002.

[PS94] Alexander Polishchuk and Daniel Spielman. Nearly-linear size holographic proofs. In *26th ACM Symposium on Theory of Computing (STOC)*, pages 194–203, 1994.

[Rag07] Prasad Raghavendra. A note on Yekhanin's locally decodable codes. Electronic Colloquium on Computational Complexity (ECCC) TR07-016, 2007.

[Sho05] V. Shoup. *A computational introduction to number theory and algebra.* 2005.

[STV01] Madhu Sudan, Luca Trevisan, and Salil Vadhan. Pseudorandom generators without the XOR lemma. 62:236–266, 2001.

[Sud92] Madhu Sudan. *Efficient checking of polynomials and proofs and the hardness of approximation problems.* PhD thesis, University of California at Berkeley, 1992.

[Sud09] Madhu Sudan. Personal Communication, 2009.

[Tre04] Luca Trevisan. Some applications of coding theory in computational complexity. *Quaderni di Matematica*, 13:347–424, 2004.

[vL82] J.H. van Lint. *Introduction to Coding Theory.* 1982.

[WdW05] Stephanie Wehner and Ronald de Wolf. Improved lower bounds for locally decodable codes and private information retrieval. In *32nd International Colloquium on Automata, Languages and Programming (ICALP)*, volume 3580 of Lecture Notes in Computer Science, pages 1424–1436, 2005.

[Woo07] David Woodruff. New lower bounds for general locally decodable codes. Electronic Colloquium on Computational Complexity (ECCC) TR07-006, 2007.

[Woo08] David Woodruff. Corruption and recovery-efficient locally decodable codes. In *International Workshop on Randomization and Computation (RANDOM)*, pages 584–595, 2008.

[WY05] David Woodruff and Sergey Yekhanin. A geometric approach to information theoretic private information retrieval. In *20th IEEE Computational Complexity Conference (CCC)*, pages 275–284, 2005.

[Yek07] Sergey Yekhanin. *Locally decodable codes and private information retrieval schemes.* PhD thesis, Massachusetts Institite of Technology, 2007.

[Yek08] Sergey Yekhanin. Towards 3-query locally decodable codes of subexponential length. *Journal of the ACM*, 55:1–16, 2008.

[Yek10] Sergey Yekhanin. Locally decodable codes. *Foundations and trends in theoretical computer science*, 2010. to appear.

9 Appendix

Our goal here is to prove the following

Lemma 11 *Let $m = \prod_{i=1}^{t} p_i$ be a product of distinct primes. Let w be a positive integer. Suppose integers $\{e_i\}$, $i \in [t]$ are such that for all i, we have $p_i^{e_i} > w^{1/t}$. Let $d = \max_i p_i^{e_i}$, and $h \geq w$ be arbitrary. Let S be the canonical set modulo m; then there exists an $\binom{h}{w}$-sized family of S-matching vectors in $\mathbb{Z}_m^n$, where $n = \binom{h}{\leq d}$.*

Our construction of the matching family is modeled along the lines of Grolmusz's construction of a set system with restricted intersections modulo composites [Gro00, Gro02]. His construction uses the low-degree OR representations of Barrington et al. [BBR94]. However, we will use Lemma 36 to bypass the set system and go directly to the matching family from polynomials. In addition to being more direct, this also gives a slightly larger collection of vectors.

Definition 35 *Let $S \subseteq \mathbb{Z}_m \backslash \{0\}$. We say that a set of polynomials $\mathcal{F} = \{f_1, \ldots, f_k\} \subseteq \mathbb{Z}_m[z_1, \ldots, z_h]$ and a set of points $\mathcal{X} = \{\mathbf{x}_1, \ldots, \mathbf{x}_k\} \subseteq \mathbb{Z}_m^h$ form a polynomial S-matching family of size k if*

- *For all $i \in [k]$, $f_i(\mathbf{x}_i) = 0$;*
- *For all $i, j \in [k]$ such that $i \neq j$, $f_j(\mathbf{x}_i) \in S$.*

Let $\mathcal{F}, \mathcal{X}$ be a k-sized polynomial matching family. For $i \in [k]$, let $\mathrm{supp(f_i)}$ denote the set of monomials in the support of the polynomial f_i. We define $\mathrm{supp}(\mathcal{F}) = \bigcup_{\mathrm{i=1}}^{\mathrm{k}} \mathrm{supp(f_i)}$ and $\dim(\mathcal{F}) = |\mathrm{supp}(\mathcal{F})|$. The following lemma was observed by Sudan [Sud09].

Lemma 36 *An k-sized polynomial S-matching family $\mathcal{F}, \mathcal{X}$ over $\mathbb{Z}_m$ yields a k-sized S-matching family $\mathcal{U}, \mathcal{V}$ in $\mathbb{Z}_m^n$, where $n = \dim(\mathcal{F})$.*

Proof: Let $\mathrm{mon_1}, \ldots, \mathrm{mon_n}$ be the set of monomials in $\mathrm{supp}(\mathcal{F})$. For every $j \in [k]$ we have

$$f_j(z_1 \ldots, z_h) = \sum_{l=1}^{n} c_{jl} \mathrm{mon_l}.$$

We define the vector $\mathbf{u}_j$ to be the n-dimensional vector of coefficients of the polynomial f_j. Similarly, for $i \in [k]$, we define the vector $\mathbf{v}_i$ to be the vector of evaluations of monomials $\mathrm{mon_1}, \ldots, \mathrm{mon_n}$ at the point $\mathbf{x}_i$. It is easy to check that for all $i, j \in [k]$, $(\mathbf{u}_j, \mathbf{v}_i) = f_j(\mathbf{x}_i)$ and hence the sets $\mathcal{U}, \mathcal{V}$ indeed form an S-matching family. ■

In what follows we assume that parameters $m, t, \{p_i\}_{i \in [t]}, \{e_i\}_{i \in [t]}, w, h$, and the set S satisfy the precondition of lemma 11. Theorem 2.16 in [Gop06] yields

Lemma 37 *For every $i \in [t]$, there is an explicit multilinear polynomial $f_i(z_1, \ldots, z_h) \in \mathbb{Z}_{p_i}[z_1, \ldots, z_h]$ where $\deg(f_i) \leq p_i^{e_i} - 1$ such that for $\mathbf{x} \in \{0,1\}^h$, we have*

$$f_i(\mathbf{x}) \equiv \begin{cases} 0 \bmod p_i, & \textit{if } \sum_{l=1}^{h} \mathbf{x}(l) \equiv w \bmod p_i^{e_i}, \\ 1 \bmod p_i, & \textit{otherwise.} \end{cases}$$

Corollary 38 *There is an explicit multilinear polynomial $f(z_1, \ldots, z_h) \in \mathbb{Z}_m[z_1, \ldots, z_n]$ such that for all $\mathbf{x} \in \{0,1\}^h$, we have*

$$f(\mathbf{x}) = \begin{cases} 0 \bmod m, & \textit{if } \sum_{l=1}^{h} \mathbf{x}(l) = w, \\ s \bmod m, \text{ for s} \in \text{S}, & \textit{if } \sum_{l=1}^{h} \mathbf{x}(l) < w, \end{cases}$$

where coordinates of $\mathbf{x}$ are summed as integers.

Proof: Define the polynomial f so that for all $i \in [t]$, $f(z_1, \ldots, z_h) \equiv f_i(z_1, \ldots, z_h) \bmod p_i$. We claim that it satisfies the above requirement. Observe that by the Chinese remainder theorem

$$f(\mathbf{x}) = 0 \bmod m \quad \text{iff} \quad \text{for all } i \in [t], \sum_{l=1}^{h} \mathbf{x}(l) \equiv w \bmod p_i^{e_i}.$$

This is equivalent to saying that

$$\sum_{l=1}^{h} \mathbf{x}(l) \equiv w \bmod \prod_i p_i^{e_i}.$$

Note that for all $i \in [t]$, $p_i^{e_i} > w^{1/t}$. Hence $m = \prod_i p_i^{e_i} > w$. Thus whenever the integer sum $\sum_{l=1}^{h} \mathbf{x}(l) < w$, we have $\sum_{l=1}^{h} \mathbf{x}(l) \not\equiv w \bmod m$, which proves the claim. ■

Proof of lemma 11: For every $T \subseteq [h]$ of size w, define the polynomial f_T wherein the polynomial f from corollary 38, we set $z_j = 0$ for $j \notin T$ (but z_j stays untouched for $j \in T$). Define $\mathbf{x}_T \in \{0,1\}^h$ to be the indicator of the set T. Viewing vectors $\mathbf{x} \in \{0,1\}^h$ as indicator vectors $\mathbf{x}_L$ for sets $L \subseteq [h]$, it is easy to check that for all $T, L \in [h]$, $f_T(\mathbf{x}_L) = f(\mathbf{x}_{L \cap T})$. Combining this with Corollary 38 gives

- For all $T \subseteq [h]$, where $|T| = w$, $f_T(\mathbf{x}_T) = f(\mathbf{x}_T) \equiv 0 \bmod m$,
- For all $T \neq L \subseteq [h]$, where $|T| = |L| = w$, $f_T(\mathbf{x}_L) = f(\mathbf{x}_{L \cap T}) \in S \bmod m$,

where the second bullet follows from the observation that $|L \cap T| \leq w - 1$. Thus the set of polynomials $\mathcal{F} = \{f_T\}_{T \subseteq [h], |T| = w}$ and points $\mathcal{X} = \{\mathbf{x}_T\}_{T \subseteq [h], |T| = w}$ form a polynomial S-matching family.

It is clear that $k = |\mathcal{F}| = \binom{h}{w}$. To bound n, we note that $\deg(f) \leq d$ and f is multilinear. Thus we can take $\mathrm{supp}(\mathcal{F})$ to be the set of all multilinear monomials in $z_1, \ldots, z_h$ of degree at most d. Thus $\dim(\mathcal{F}) = \binom{h}{\leq d}$. ■