Matching Vector Codes

Zeev Dvir
Princeton University
zeev.dvir@gmail.com

Parikshit Gopalan
MSR - Silicon Valley.
parik@microsoft.com

Sergey Yekhanin
MSR - Silicon Valley.
sergey@microsoft.com

***Abstract*—A locally decodable code encodes a message by a codeword, such that even if the codeword is corrupted by noise, each message bit can be recovered with high probability by a randomized decoding procedure that reads only few bits of the codeword.**

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. In this work we develop a new view of MV codes and uncover certain similarities between them and classical Reed Muller codes. Our view allows us to obtain a deeper insight into the power and limitations of MV codes. We use it to construct codes that can tolerate more errors or are shorter than previously known codes for certain parameter settings. We also show super-linear lower bounds on the codeword length of any MV code.

***Keywords*-locally decodable codes; matching vectors; Reed Muller codes;**

I. 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 [37], [31], [16] 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.

Zeev Dvir's research was partially supported by NSF grants CCF-0832797 and DMS-0835373.

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 $r = 4$.

A. 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 [2], [29], [26]. Katz and Trevisan [21] were the first to provide a formal definition of LDCs (see also [30]) and prove lower bounds on their length. Their bounds were improved in [17], [23] 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 [32], [33]. The best lower bounds for the length of r-query LDCs currently have the form $\tilde{\Omega}\left(n^{1+1/(\lceil r/2 \rceil - 1)}\right)$ [33]. 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 [2], [21], [6], [9] are (directly or indirectly) based on classical (generalized) Reed Muller (RM) codes [25]. 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 [5], [24], [30]. 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 [7], [35] 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 [36] and developed further in [27], [22], [13], [20]. 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. Recently an important progress in constructions of LDCs of the third generation has been accomplished in [13] where the first constructions of codes from matching vectors modulo composites (rather than primes) were considered. In what follows we refer to LDCs of the third generation as Matching Vector (MV) codes.

To date 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 [7], [35] they suffered from having large values of $\epsilon = \Omega(r\delta)$. Thus as the number of queries increased, the length N became smaller as a function of k, but at the price of a reduction in the error-rate that the code could handle. 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.

The reason that previous constructions all gave $\epsilon = \Omega(r\delta)$ lay in the reliance on the *smoothness* of the decoder to prove its correctness. The proofs proceeded by showing that each of the r queries made by the decoder is smooth, meaning that it distributed (close to) uniformly over the bits of the codeword. By the union bound, if a δ fraction is corrupted, then we are unlikely to query any of these locations. This argument clearly will not work once the error rate exceeds $1/r$. Indeed a recent result [15] shows that for 3-query LDCs, correcting more than $1/3$ fraction of errors requires exponential length.

B. Our results

In this work we develop a new view of MV codes and uncover certain similarities between them 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 binary $\left(t^{O(t)}, \delta, 4\delta(1 + O(1/\ln t))\right)$-locally decodable codes of length essentially identical to the length of the currently shortest known $\left(2^{O(t)}, \delta, 2^{O(t)}\delta\right)$-LDCs of [13], [20]. 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 (binary) 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) The parameters of an MV code are determined by the parameters of the underlying family of matching vectors. We obtain new upper and lower 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 $r \geq (\log k)^{\Omega(\sqrt{\log k})}$.

C. 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 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*, a notion that we will define shortly.

Constructions of locally decodable codes from matching vectors have previously been considered in [36], [27], [13], [20]. In this work we show that if the family of matching vectors underlying 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 show how the

[1]It is interesting to contrast our work with the work of Woodruff [34] 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.

currently best known families of matching vectors (due to Grolmusz [19]) can be turned into bounded families. We also give a simple construction of bounded families of matching vectors.

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 classical linear-algebra based bound 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.

D. Subsequent work

After the initial publication [18], [11] of our work Ben-Aroya et al. [8] have independently rediscovered some of the ideas that we use. See [8, section Related work] for an accurate account of the relation between the papers. Ben-Aroya et al. have extended some our results. Specifically, we show how one can locally decode binary MV codes from nearly $1/8$ fraction of errors ($1/4$ for codes over large alphabets). Ben-Aroya et. al [8] show how one can decode binary MV codes from nearly $1/4$ fraction of errors ($1/2$ for codes over large alphabets). They also consider local list decoding of MV codes.

The idea behind the improved unique-decoder of [8] is that if we take a sufficiently large (but constant) number of multiplicative lines; then the average agreement with the codeword along a line is likely to exceed $1/2$. We run our decoder (proposition 7) along every line we picked. Each invocation returns a candidate value of the desired message symbol. Ben-Aroya et. al [8] show that if we use Forney's GMD decoding technique [14] where one assigns weights to each output of the decoder based on the number of errors along the corresponding line; then the symbol with the largest weight is with high probability the correct symbol.

E. Outline

We start section III 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 IV we present two constructions of bounded matching families. In section V we put the results of sections III and IV together to obtain new upper bounds on the length of MV codes. In section VI we obtain a collection of upper bounds on the size of matching families of vectors. In section VII we translate the results of section VI into lower bounds on the length of MV codes.

II. 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 $\mathrm{supp(f)}$ 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 $2^{O(x)}$.

III. 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 families $\mathcal{U} = \{\mathbf{u}_1, \ldots, \mathbf{u}_k\}$ 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 $\left(g^{\mathbf{w}(1)}, \ldots, g^{\mathbf{w}(n)}\right)$;
- 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 $\mathrm{mon}_{\mathbf{u}} \in \mathbb{F}_{\mathrm{q}}[\mathrm{z}_1, \ldots, \mathrm{z}_{\mathrm{n}}]$ by

$$\mathrm{mon}_{\mathbf{u}}(\mathrm{z}_1, \ldots, \mathrm{z}_{\mathrm{n}}) = \prod_{\ell \in [\mathrm{n}]} \mathrm{z}_\ell^{\mathbf{u}(\ell)}. \tag{2}$$

A. 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 and 7). 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 $\mathbf{x}(i)$ 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

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

It is evident from the above formula that $\mathrm{supp}(f) \subseteq S \cup \{0\}$ and

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

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

B. The simplest decoder

The proposition below gives the simplest local decoder. In the current form it has first appeared in [13]. Earlier versions can be found in [36], [27].

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 $\left\{ g^{\mathbf{w}+\lambda\mathbf{v}_i} \mid \lambda \in \{0, \ldots, s\} \right\}$ 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 $\mathrm{supp(h)} \subseteq S \cup \{0\}$ 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 [36], [13] for the case of three-query codes, and in [20]. In the full version of the paper [11] we present some new general sufficient conditions that allow for such a saving.

C. 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 $\{0, \ldots, m-1\}$ of 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.

The proposition below gives the first local decoder for MV codes that is capable of tolerating large amounts of error. Our constructions of MV codes in section V rely on it.

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 (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 $\left\{g^{\mathbf{w}+\lambda\mathbf{v}_i} \mid \lambda \in \mathbb{Z}_m\right\}$ 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 [25].
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 $2\delta/(1-\sigma)$. ∎

D. 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). In the full version of the paper [11] we show that, in the case when $|S|$ is small (and $\ln q$ is small relative to m), one can get an even better result.

E. From q-ary to binary codes

Proposition 7 yields 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. We defer the proof to the full version of the paper.

Lemma 8: Let σ be a positive real. Let $\mathcal{U}, \mathcal{V}$ be a σm-bounded family of S-matching vectors in $\mathbb{Z}_m^n$, $|\mathcal{U}| = |\mathcal{V}| = k$. 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.

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

IV. Matching vectors: constructions

In this section we present constructions of bounded matching families of vectors. Our first construction (lemma 12) is based on an existing matching family due to Grolmusz [19]. We argue that an appropriate scaling turns Grolmusz's family into a bounded family. Later in section V we use this construction to obtain MV codes that improve upon LDCs of [13], [20] in terms of the amount of noise that they can tolerate, and improve upon classical r-query RM LDCs in terms of codeword length for all $r \leq \log k/(\log\log k)^c$. Our second construction (lemma 15) is self-contained. It improves on the first construction for large values of m, and yields MV codes that match RM LDCs for certain values of $r > \log k/(\log\log k)^c$.

Definition 9 (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 construction follows the lines of Grolmusz's construction of a set system with restricted intersections modulo composites [19], but with some differences. We use an approach suggested by Sudan to go directly from polynomials to matching vectors without constructing set-systems, which gives a slight improvement in parameters. The proof can be found in the full version of the paper [11].

Lemma 10: 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 11: 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 argument above shows 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). Note that such scaling does not change the set $\mathcal{U}$, and hence the corresponding MV code. It also does not change the set of points queried by the decoder (of proposition 7), since for an invertible $\alpha \in \mathbb{Z}_m$, and an arbitrary $\mathbf{v} \in \mathbb{Z}_m^n$ multiplicative lines in the directions $\mathbf{v}$ and $\alpha\mathbf{v}$ are the same. Combining lemma 11 with lemma 10 we obtain

Lemma 12: 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 $\sum_{i \in [t]} 1/p_i$.

In fact one can show that the scaling above is the optimal scaling of the canonical set, in the sense that it minimizes the size of the maximum element.

A. Simple construction of matching vectors

In this section we give an elementary construction of a bounded family of matching vectors. The construction works for both prime and composite moduli. The family improves upon the family of lemma 12 for large values of m. The proofs appear in the full version of the paper. In what follows we use $\mathbb{Z}_{\geq 0}$ to denote the set of non-negative integers.

Definition 13: Let $b(m', n)$ denote the number of vectors $\mathbf{w} \in \mathbb{Z}_{\geq 0}^n$ such that $\|\mathbf{w}\|_2^2 = m'$.

Thus $b(m', n)$ counts the number of integer points on the surface of the n-dimensional ball of radius $\sqrt{m'}$ in the positive orthant.

Lemma 14: Let $m' < m$ and $n \geq 2$ be arbitrary positive integers. There exists a $b(m', n-1)$-sized $(m'+1)$-bounded family of matching vectors in $\mathbb{Z}_m^n$.

The lemma below follows by combining lemma 14 with some crude lower bounds for $b(m', n-1)$.

Lemma 15: Let $m' < m$ and $n \geq 2$ be arbitrary positive integers. There exists a k-sized $(m'+1)$-bounded family of matching vectors in $\mathbb{Z}_m^n$, where

$$k = \frac{1}{m'+1}\left(\frac{m'}{n-1}\right)^{(n-1)/2} \quad \text{for } m' \geq n, \tag{9}$$

$$k = \binom{n-1}{m'} \quad \text{for } m' < n. \tag{10}$$

V. 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 8 and lemma 12 yields

Lemma 16: 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.

We now estimate asymptotic parameters of our codes. The lemma below follows by appropriately setting the parameters in lemma 16. We defer the proof to the full version of the paper [11].

Lemma 17: For all integers $t \geq 2$ and $k \geq 2^t$ 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.

Setting t to be a constant in lemma 17 yields

Theorem 18: For every $t \geq 2$ and all sufficiently large 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 18 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 [13], [20]. 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. See [11] for a proof.

Theorem 19: For every large enough integer r and every $k \geq r$, there exists a binary linear code encoding k-bit messages to

$$\exp\exp\Big((\log k)^{O(\log\log r/\log r)} \cdot (\log\log k)^{1-\Omega(\log\log r/\log r)} \log r\Big) \tag{11}$$

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

A. 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* LDCs with essentially the same parameters as above also over the complex numbers (the definition of LDCs 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 straightforward 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 from MV codes, even over finite fields with very large characteristic (RM codes require that the characteristic will be at most the codeword length). Even though this might seem like an esoteric setting, LDCs over characteristic zero did came up in some recent works in connection to arithmetic circuit complexity [12], [10].

B. Comparison to Reed Muller codes

Theorem 19 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 [21], [31], [37] 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 $\mathbb{F}_q^n$-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 8). 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

$$k = \binom{n+d}{d} \log q,\ N = \Theta(q^n \ln q),\ r = \Theta(q \ln q). \quad (12)$$

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 n yields shorter codes of larger query complexity.

Example 20: 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 19 (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 (12) 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

$$k = \binom{n+d}{d} \log q \leq (en/d)^d \log q \leq n^{O(d)}, \quad (13)$$

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

$$N \geq \exp\exp(\log k/d) \geq \exp\exp(\log k/r). \quad (14)$$

Note that when r is a constant then already 3-query codes of [13] improve substantially upon (14). 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 (14) evaluates to a larger value than (11).

Remark 21: It is interesting to observe that while MV codes of theorem 19 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 20 combining lemma 15 (where m has the shape $2^b - 1$, $n = m + 1$ and $m' = n/2$) with lemma 8.

VI. 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 and 7 is at most $k(m, n)/m^n$. Our goal in this section is to establish upper bounds on $k(m, n)$ (all proofs are deferred to the full version [11].) In section VII we translate these bounds into lower bounds on the length of MV codes.

There is a large body of work in combinatorics on the closely related problem of set-systems with restricted modular intersections. The problem there is to bound the size of the largest set family $\mathcal{F}$ on $[n]$, where the sets in $\mathcal{F}$ have cardinality 0 modulo some integer m, while their intersections have non-zero cardinality modulo m. The classical result in this area is the modular Ray-Chaudhuri-Wilson theorem [3] showing that when m is a prime (or a prime

power), an upper bound of $n^{O(m)}$ holds. It is known that such a bound does not apply when m is composite [19]. The best upper bound for general m shows that $|\mathcal{F}| \leq 2^{n/2}$ [28].

We start by bounding $k(m,n)$ in the prime case.

A. The prime case

We present two bounds for the prime case. The first is based on the linear algebra method [3] and is tight when p is a constant.

Theorem 22: For any positive integer n and any prime p, we have

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

Note that equation (10) shows that for constant p and growing n, the above bound is asymptotically tight.

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 $\mathbb{F}_p^n \backslash \{0^n\}$ under the equivalence relation $\lambda \mathbf{v} \equiv \mathbf{v}$ for $\lambda \in \mathbb{F}_p^*$. Projective hyperplanes are specified by vectors $\mathbf{u} \in \mathbb{F}_p^n \backslash \{0^n\}$ 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 $\mathbb{Z}_p^n$.

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 [1, 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 $|X| = x$,

$$|N(X)| \geq u - u^{\frac{n}{n-1}}/x. \tag{15}$$

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

$$k(p,n) \leq 4p^{n/2} + 2. \tag{16}$$

Equation (9) shows that $k(p,n) = \Omega\left(p^{(n-3)/2}\right)$, so the above upper bound is nearly tight when n is a constant and p grows to infinity. Note that for this setting of parameters, the linear-algebra bound gives $k(p,n) \leq O(p^{n-1})$, so the bound above gives a significant improvement.

B. The prime power case

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

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

C. The composite case

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

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

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

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

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

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

We now proceed to the case of general n.

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

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

The upper bound of lemma 27 (that applies only when m is prime) is substantially stronger than the bounds of theorems 30 and 32. We feel that latter bounds can be improved. Specifically, we propose the following

Conjecture 33: Let m and n be arbitrary positive integers; then $k(m,n) \leq O\left(m^{n/2}\right)$.

We discuss the implications of the conjecture for the lower bounds for MV codes in remark 36.

VII. 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 $2^{\Omega(\sqrt{\log k})}$. All proofs are deferred to the full version of the paper [11].

Theorem 34: 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

$$k \leq \frac{N}{2^{0.4\sqrt{\log N}}} \;\Rightarrow\; N \geq k2^{0.4\sqrt{\log k}}.$$

One can generalize theorem 34 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 35: 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 $2^{\Omega(\sqrt{\log k_\ell})}$.

A. 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 (12) 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

$$n = O\left(\sqrt{\log k}/\log\log k\right). \tag{17}$$

Given k we choose n to be the largest integer satisfying (17). 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.

Remark 36: It is not hard to verify that if conjecture 33 holds; then any MV code must have length $N = \Omega(k^2)$. This would imply that RM LDCs improve on MV codes once $r \geq \log^2 k$, (by an argument similar to the one above). Since MV codes improve on RM codes for $r \leq \log k/(\log\log k)^{O(1)}$, this would give a clearer picture of the comparison between the two families of codes.

VIII. 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 some of them.

- It is very interesting to see if one can get MV codes that improve upon RM codes for values of $r \geq \log k/(\log\log k)^{O(1)}$. This calls for constructions of bounded matching families in $\mathbb{Z}_m^n$, where m is comparable to (or larger than) n.
- It is very interesting to prove (or disprove) conjecture 33. A simple case where it is open is when n is a constant and m is a product of two nearly equal primes.
- Our results show that MV codes share many properties of RM codes. We would like to know if MV codes are (or can be made) locally correctable [4], [10]. Note that to date, RM LDCs constitute the only known class of locally correctable codes.

References

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

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

[3] Laszlo Babai and Peter Frankl. *Linear algebra methods in combinatorics*. 1998.

[4] Omer Barkol, Yuval Ishai, and Enav Weinreb. On locally decodable codes, self-correctable codes, and t-private PIR. In *International Workshop on Randomization and Computation (RANDOM)*, pages 311–325, 2007.

[5] 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. Springer, Berlin, Heidelberg, 1990.

[6] 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.

[7] 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.

[8] Avraham Ben-Aroya, Klim Efremenko, and Amnon Ta-Shma. Local list decoding with a constant number of queries. In *Electronic Colloquium on Computational Complexity (ECCC)*, TR010-047, 2010.

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

[10] Zeev Dvir. On matrix rigidity and locally self-correctable codes. In *26th IEEE Computational Complexity Conference (CCC)*, pages 102–113, 2010.

[11] Zeev Dvir, Parikshit Gopalan, and Sergey Yekhanin. Matching vector codes. In *Electronic Colloquium on Computational Complexity (ECCC)*, TR010-012, 2010.

[12] Zeev Dvir and Amir Shpilka. Locally decodable codes with 2 queries and polynomial identity testing for depth 3 circuits. *SIAM Journal on Computing*, 36(5):1404–1434, 2006.

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

[14] David Forney. Generalized minimum distance decoding. *IEEE Transactions on Information Theory*, 12:125–131, 1966.

[15] Anna Gal and Andrew Mills. Three query locally decodable codes with higher correctness require exponential length. Manuscript, submitted, 2009.

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

[17] 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.

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

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

[20] Toshiya Itoh and Yasuhiro Suzuki. New constructions for query-efficient locally decodable codes of subexponential length. *IEICE Transactions on Information and Systems*, E93-D:263–270, 2010.

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

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

[23] 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.

[24] 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. Springer, Berlin, Heidelberg, 1990.

[25] F. J. MacWilliams and N. J. A. Sloane. *The Theory of Error Correcting Codes*. North Holland, Amsterdam, New York, 1977.

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

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

[28] Jiri Sgall. Bounds on pairs of families with restricted intersections. *Combinatorica*, 19:555–566, 1999.

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

[30] Madhu Sudan, Luca Trevisan, and Salil Vadhan. Pseudorandom generators without the XOR lemma. In *39th ACM Symposium on Theory of Computing (STOC)*, pages 537–546, 1999.

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

[32] 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. Springer, Berlin, Heidelberg, 2005.

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

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

[35] 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.

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

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