Explicit constructions of MDS array codes and RS codes with optimal repair bandwidth

Min Ye Alexander Barg

***Abstract*—Given any r and n, we present an explicit construction of high-rate maximum distance separable (MDS) array codes that can optimally repair any d failed nodes from any h helper nodes for all $h, 1 \leqslant h \leqslant r$ and $d, k \leqslant d \leqslant n-h$ simultaneously. These codes can be constructed over any base field F as long as $|F| \geqslant sn$, where $s = \text{lcm}(1, 2, \ldots, r)$. The encoding, decoding, repair of failed nodes, and update procedures of these codes all have low complexity. Our results present a significant improvement over earlier results which can only construct explicit codes for the case of at most 3 parity nodes, and these existing constructions can only optimally repair a single node failure by accessing all the surviving nodes.**

In the second part of the paper we give an explicit construction of Reed-Solomon codes with asymptotically optimal repair bandwidth.

I. Introduction

An (n, k, l) MDS array code has k information nodes[1] and $r = n - k$ redundancy nodes in each codeword with the property that any k out of n nodes can recover the codeword. Each node is a column vector in F^l, where F is some finite field. While originally MDS array codes were studied for error correction in memories [1], recent applications in distributed storage brought forth the problem of *efficient regeneration* of a failed node [2].

Recovering failed (erased) node(s) from the information available at the other nodes is called the *repair process*.

Definition I.1. *For $h \leqslant r$ and $k \leqslant d < n$, define $N(h, d)$ as the smallest integer such that the contents of any h nodes can be recovered by accessing any d surviving nodes and downloading the total of at most $N(h, d)$ symbols of F from these nodes*[2]. *The parameter $N(h, d)$ is called the* (h, d)-repair bandwidth *of the code $\mathcal{C}$.*

By a result of [2], [7],

$$N(h, d) \geqslant d\, l \frac{h}{d+h-k}. \tag{1}$$

We say that an (n, k, l) MDS array code $\mathcal{C}$ has the (h, d)-*optimal repair* property if this lower bound is achieved, and omit the reference to h if $h = 1$. If $d = n-1$ we also omit the reference to d. For $k \leqslant (n+1)/2$ (the low rate regime), MDS array codes with the optimal repair property were constructed in [3]–[6]. For arbitrary code rate, [7] proved that the bound (1) is asymptotically achievable when $l \to \infty$. For finite l and $k > (n+1)/2$ (the high-rate regime) [8]–[12] showed that for F large enough there exist MDS array codes that can optimally repair any systematic node using all the surviving nodes, and [13] showed the same for all rather than only systematic nodes. At the same time, explicit MDS array code constructions for optimal all-node or even only systematic node repair in the high rate regime are known only if $r \leqslant 3$ [10]–[14].

In Section III, we present an explicit MDS array code construction with the optimal repair property for any number of parity nodes and any code length using a field F of size $|F| \geqslant rn$. The encoding, decoding, and repair of a single failed node involve only simple operations with $r \times r$ matrices over F, and thus have low complexity[3]. An additional property of the proposed codes is optimal update, i.e., the need to change only the minimal number of coordinates in parity nodes if one coordinate in systematic node is updated. In our construction we rely on a (non-systematic) parity-check representation of the codes as opposed to the systematic generator form used in most earlier works. This representation does not distinguish between systematic nodes and parity nodes, and leads naturally to the optimal repair of all nodes. Moreover, the parity-check form combined with the block Vandermonde structure [12] and the idea of using r-ary expansions [8], [11] makes the explicit construction for larger number of parity nodes possible.

In Section IV, we give an explicit construction of (n, k, l) MDS array codes with d-optimal repair property for any positive integers n, k, d, l such that $k \leqslant d < n, l = (d+1-k)^n$ using a field F of size $|F| \geqslant (d+1-k)n$.

In Section V, we construct MDS codes with d-optimal repair property for several values of d simultaneously. Moreover, we show that (n, k, r^n) MDS array codes constructed in Section III will automatically have d-optimal repair property for all d such that $(d+1-k)|(n-k)$. In Section VI we further extend our construction to obtain (n, k, l) MDS array codes with (h, d)-optimal repair property for all $h \leqslant r$ and $k \leqslant d \leqslant n-h$ simultaneously, where $l = s^n, s = \text{lcm}(1, 2, \ldots, r)$. These codes can be constructed over any base field F as long as $|F| \geqslant sn$, and they also have the optimal update property. Moreover, the encoding, decoding, and repair procedures only require operations with matrices of size not greater than $n \times n$.

Recently, [16] studied the repair bandwidth of Reed-Solomon (RS) codes and introduced an efficient linear repair scheme for RS codes. In Section VII we use this linear repair scheme and the r-ary expansion idea to construct an explicit family of RS codes with asymptotically optimal repair bandwidth: we show that the ratio of the actual repair bandwidth of the codes and the optimal value approaches 1 when the code length goes to infinity.

The authors are with Dept. of ECE and ISR, University of Maryland, College Park, MD 20742. Emails: yeemmi@gmail.com and abarg@umd.edu.
Research supported by NSF grants CCF1217245 and CCF1422955.

[1] following the recent literature, we refer to codeword coordinates as nodes.

[2] these symbols can be some functions of the contents of the nodes.

[3] An expanded version of Sections II-VI of this paper is available online as arXiv:1604.00454 [15].

II. General code construction

Let $\mathcal{C} \in F^{ln}$ be an (n, k, l) array code with nodes $C_i \in F^l, i = 1, \dots, n$, where each C_i is a column vector. Throughout this paper we consider codes defined in the following parity-check form:

$$\mathcal{C} = \{(C_1, C_2, \dots, C_n) : \sum_{i=1}^{n} A_{t,i} C_i = 0, \, t = 1, \dots, r\}, \quad (2)$$

where $A_{t,i}, t = 1, \dots, r, i = 1, \dots, n$ are $l \times l$ matrices over F.

Given positive integers r and n, define an $(n, k = n - r, l)$ array code $\mathcal{C}$ by setting in (2)

$$A_{t,i} = A_i^{t-1}, t \in [r], i \in [n], \quad (3)$$

where $A_1, A_2, \dots, A_n$ are some $l \times l$ matrices. (We use the convention $A^0 = I$.) The specific code families in Section III-VI are obtained by choosing different forms of the matrices $A_1, A_2, \dots, A_n$.

III. Construction of MDS array codes with optimal repair property

A. Code construction

Construction 1. *Let F be a finite field of size $|F| \geqslant rn$, and let $l = r^n$. Let $\{\lambda_{i,j}\}_{i \in [n], j=0,1,\dots,r-1}$ be rn distinct elements in F. Consider the code family given by* (2)-(3)*, where we take*

$$A_i = \sum_{a=0}^{l-1} \lambda_{i,a_i} e_a e_a^T, \; i = 1, \dots, n.$$

Here $\{e_a : a = 0, 1, \dots, l-1\}$ is the standard basis of F^l over F, and a_i is the i-th digit from the right in the representation of a in the r-ary form, $a = (a_n, a_{n-1}, \dots, a_1)$.

Since the $A_i, i = 1, \dots, n$ are diagonal matrices, we can write out the parity-check equations (2) coordinatewise. Let $c_{i,a}$ denote the a-th coordinate of the column vector C_i for all $a = 0, \dots, l-1$, i.e., $C_i = (c_{i,0}, c_{i,1}, \dots, c_{i,l-1})^T$. We have

$$\sum_{i=1}^{n} \lambda_{i,a_i}^t c_{i,a} = 0 \quad (4)$$

for all $t = 0, \dots, r-1$ and $a = 0, \dots, l-1$.

Theorem III.1. *Codes given by Construction 1 attain optimal repair bandwidth for repairing any single failed node.*

Proof: For $u = 0, 1, \dots, r-1$, let $a(i, u) := (a_n, \dots, a_{i+1}, u, a_{i-1}, \dots, a_1)$. We will show that for any $i \in [n]$ and $a = 0, 1, \dots, l-1$, the coordinates $\{c_{i,a(i,0)}, c_{i,a(i,1)}, \dots, c_{i,a(i,r-1)}\}$ in C_i are functions of the following set of $n-1$ elements of F:

$$\mu_{j,i}^{(a)} := \sum_{u=0}^{r-1} c_{j,a(i,u)}, \quad j \in [n] \backslash \{i\}. \quad (5)$$

In other words, each surviving node only needs to transmit one scalar in F to recover r coordinates in the failed node, so the optimal repair bandwidth is achieved. Replacing a with $a(i, u)$ in (4), we obtain

$$\lambda_{i,u}^t c_{i,a(i,u)} + \sum_{j \neq i} \lambda_{j,a_j}^t c_{j,a(i,u)} = 0. \quad (6)$$

Summing (6) over $u = 0, 1, \dots, r-1$ and then writing the result in matrix form, we get

$$\begin{bmatrix} 1 & 1 & \dots & 1 \\ \lambda_{i,0} & \lambda_{i,1} & \dots & \lambda_{i,r-1} \\ \vdots & \vdots & \vdots & \vdots \\ \lambda_{i,0}^{r-1} & \lambda_{i,1}^{r-1} & \dots & \lambda_{i,r-1}^{r-1} \end{bmatrix} \begin{bmatrix} c_{i,a(i,0)} \\ c_{i,a(i,1)} \\ \vdots \\ c_{i,a(i,r-1)} \end{bmatrix} = - \begin{bmatrix} \sum_{j \neq i} \mu_{j,i}^{(a)} \\ \sum_{j \neq i} \lambda_{j,a_j} \mu_{j,i}^{(a)} \\ \vdots \\ \sum_{j \neq i} \lambda_{j,a_j}^{r-1} \mu_{j,i}^{(a)} \end{bmatrix}. \quad (7)$$

By construction $\lambda_{i,0}, \dots, \lambda_{i,r-1}$ are distinct, so we can solve this system for $\{c_{i,a(i,0)}, c_{i,a(i,1)}, \dots, c_{i,a(i,r-1)}\}$ given the set of elements in (5). ■

The repair procedure of a single node has low complexity: indeed, according to (7), it can be accomplished by operations with $r \times r$ matrices (rather than much larger $l \times l$ matrices).

Theorem III.2. *The code $\mathcal{C}$ given by Construction 1 is MDS.*

Proof: We write out the parity-check equations (2) coordinatewise. For all $a = 0, 1, \dots, l-1$, we have

$$\begin{bmatrix} 1 & 1 & \dots & 1 \\ \lambda_{1,a_1} & \lambda_{2,a_2} & \dots & \lambda_{n,a_n} \\ \vdots & \vdots & \vdots & \vdots \\ \lambda_{1,a_1}^{r-1} & \lambda_{2,a_2}^{r-1} & \dots & \lambda_{n,a_n}^{r-1} \end{bmatrix} \begin{bmatrix} c_{1,a} \\ c_{2,a} \\ \vdots \\ c_{n,a} \end{bmatrix} = 0 \quad (8)$$

Clearly every r columns of the parity-check matrix in (8) have rank r, so any k out of n elements in the set $\{c_{1,a}, c_{2,a}, \dots, c_{n,a}\}$ can recover the whole set. Since this holds for all $a = 0, 1, \dots, l-1$, we conclude that any k nodes of a codeword in $\mathcal{C}$ can recover the whole codeword. ■

B. Complexity of encoding, decoding, and updates

The code given by Construction 1 can be efficiently transformed into systematic form. Without loss of generality we assume that the first k nodes are systematic (information) nodes. By (8), for all $a = 0, 1, \dots, l-1$, we have

$$\begin{bmatrix} 1 & 1 & \dots & 1 \\ \lambda_{k+1,a_{k+1}} & \lambda_{k+2,a_{k+2}} & \dots & \lambda_{k+r,a_{k+r}} \\ \vdots & \vdots & \vdots & \vdots \\ \lambda_{k+1,a_{k+1}}^{r-1} & \lambda_{k+2,a_{k+2}}^{r-1} & \dots & \lambda_{k+r,a_{k+r}}^{r-1} \end{bmatrix} \begin{bmatrix} c_{k+1,a} \\ c_{k+2,a} \\ \vdots \\ c_{k+r,a} \end{bmatrix} = - \begin{bmatrix} 1 & 1 & \dots & 1 \\ \lambda_{1,a_1} & \lambda_{2,a_2} & \dots & \lambda_{k,a_k} \\ \vdots & \vdots & \vdots & \vdots \\ \lambda_{1,a_1}^{r-1} & \lambda_{2,a_2}^{r-1} & \dots & \lambda_{k,a_k}^{r-1} \end{bmatrix} \begin{bmatrix} c_{1,a} \\ c_{2,a} \\ \vdots \\ c_{k,a} \end{bmatrix}. \quad (9)$$

Consequently, in the encoding process we do not need to invert an $rl \times rl$ matrix, instead, we only need to invert $r \times r$ matrices l times, gaining a factor of l^2 in complexity. Similarly, in the decoding process, if some r nodes are erased, then in order to recover them, we only need to invert $r \times r$ matrices l times.

Another useful parameter of codes is *update complexity* [1]. On account of the MDS property, in order to update the value of a stored element $c_{i,a}$ in an information node, one needs to update at least one coordinate in every parity node [17]. From

(9) it is easy to see that for any $i \in [k]$ and $a = 0, \dots, l-1$, to update $c_{i,a}$, we only need to update $c_{k+1,a}, \dots, c_{k+r,a}$. Thus Construction 1 gives an optimal update code.

IV. Explicit MDS array codes with d-optimal repair property

The general construction in (2)-(3) can also be used to construct an $(n, k = n-r, l)$ MDS array code $\mathcal{C}$ with d-optimal repair property, $k \leqslant d \leqslant n-1$.

Construction 2. *Let F be a finite field of size $|F| \geqslant sn$, where $s = d+1-k$. Let $\{\lambda_{i,j}\}_{i\in[n],j=0,1,\dots,s-1}$ be sn distinct elements in F. Consider the code family given by (2)-(3), where $l = s^n$ and*

$$A_i = \sum_{a=0}^{l-1} \lambda_{i,a_i} e_a e_a^T, \ i = 1, \dots, n.$$

Here $\{e_a : a = 0, 1, \dots, l-1\}$ is the standard basis of F^l over F and a_i is the i-th digit from the right in the representation of a in the s-ary form, $a = (a_n, a_{n-1}, \dots, a_1)$.

Define $a(i,u)$ and $c_{i,a}$ in the same way as in Sect. III.

Theorem IV.1. *The code $\mathcal{C}$ given by Construction 2 is an MDS code.*

Proof: Same as the proof of Theorem III.2. ■

By the same arguments as in the previous section, $\mathcal{C}$ also has low-complexity encoding, decoding, and the optimal update property.

Let us show that the code $\mathcal{C}$ has d-optimal repair property. Recall the definition of Generalized Reed-Solomon codes.

Definition IV.2. *A* Generalized Reed-Solomon code $\mathrm{GRS}(n, k, \Omega, v) \subseteq F^n$ *of dimension k over F with evaluation points $\Omega = \{\omega_1, \omega_2, \dots, \omega_n\} \subseteq F$ is the set of vectors*

$$\{(v_1 f(\omega_1), \dots, v_n f(\omega_n)) \in F^n : f \in F[x], \deg f \leqslant k-1\}$$

where $v = (v_1, \dots, v_n) \in (F^*)^n$ are some nonzero coefficients. If $v = (1, \dots, 1)$, then the GRS code is called a Reed-Solomon code.

Theorem IV.3. *The code $\mathcal{C}$ given by Construction 2 has d-optimal repair property.*

Proof: Without loss of generality, we consider the case of repairing C_1. Let

$$\mu_{j,1}^{(a)} := \sum_{u=0}^{s-1} c_{j,a(1,u)}, \quad j \in \{2, 3, \dots, n\}. \tag{10}$$

Using arguments similar to those that lead to (7), we obtain

$$\begin{aligned}
&\begin{bmatrix} 1 & 1 & \dots & 1 \\ \lambda_{1,0} & \lambda_{1,1} & \dots & \lambda_{1,s-1} \\ \lambda_{1,0}^2 & \lambda_{1,1}^2 & \dots & \lambda_{1,s-1}^2 \\ \vdots & \vdots & \vdots & \vdots \\ \lambda_{1,0}^{r-1} & \lambda_{1,1}^{r-1} & \dots & \lambda_{1,s-1}^{r-1} \end{bmatrix} \begin{bmatrix} c_{1,a(1,0)} \\ c_{1,a(1,1)} \\ \vdots \\ c_{1,a(1,s-1)} \end{bmatrix} \\
= -&\begin{bmatrix} 1 & 1 & \dots & 1 \\ \lambda_{2,a_2} & \lambda_{3,a_3} & \dots & \lambda_{n,a_n} \\ \lambda_{2,a_2}^2 & \lambda_{3,a_3}^2 & \dots & \lambda_{n,a_n}^2 \\ \vdots & \vdots & \vdots & \vdots \\ \lambda_{2,a_2}^{r-1} & \lambda_{3,a_3}^{r-1} & \dots & \lambda_{n,a_n}^{r-1} \end{bmatrix} \begin{bmatrix} \mu_{2,1}^{(a)} \\ \mu_{3,1}^{(a)} \\ \vdots \\ \mu_{n,1}^{(a)} \end{bmatrix}.
\end{aligned} \tag{11}$$

Define polynomials $p_0(x) = \prod_{u=0}^{s-1}(x - \lambda_{1,u})$, and $p_i(x) = x^i p_0(x)$ for $i = 0, 1, \dots, r-s-1$. We have proved the case of $d = n-1$ in the previous section, so here we only consider the case when $d < n-1$, and so $r - s - 1 \geqslant 0$. Since the degree of $p_i(x)$ is less than r for all $i = 0, 1, \dots, r-s-1$, we can write

$$p_i(x) = \sum_{j=0}^{r-1} p_{i,j} x^j.$$

Define the $(r-s) \times r$ matrix

$$P = \begin{bmatrix} p_{0,0} & p_{0,1} & \cdots & p_{0,r-1} \\ p_{1,0} & p_{1,1} & \cdots & p_{1,r-1} \\ \vdots & \vdots & \vdots & \vdots \\ p_{r-s-1,0} & p_{r-s-1,1} & \cdots & p_{r-s-1,r-1} \end{bmatrix}.$$

Since

$$\begin{aligned}
&P \begin{bmatrix} 1 & 1 & \dots & 1 \\ \lambda_{1,0} & \lambda_{1,1} & \dots & \lambda_{1,s-1} \\ \lambda_{1,0}^2 & \lambda_{1,1}^2 & \dots & \lambda_{1,s-1}^2 \\ \vdots & \vdots & \vdots & \vdots \\ \lambda_{1,0}^{r-1} & \lambda_{1,1}^{r-1} & \dots & \lambda_{1,s-1}^{r-1} \end{bmatrix} \\
&= \begin{bmatrix} p_0(\lambda_{1,0}) & p_0(\lambda_{1,1}) & \cdots & p_0(\lambda_{1,s-1}) \\ p_1(\lambda_{1,0}) & p_1(\lambda_{1,1}) & \cdots & p_1(\lambda_{1,s-1}) \\ \vdots & \vdots & \vdots & \vdots \\ p_{r-s-1}(\lambda_{1,0}) & p_{r-s-1}(\lambda_{1,1}) & \cdots & p_{r-s-1}(\lambda_{1,s-1}) \end{bmatrix} \\
&= 0,
\end{aligned}$$

together with (11), we have

$$P \begin{bmatrix} 1 & 1 & \dots & 1 \\ \lambda_{2,a_2} & \lambda_{3,a_3} & \dots & \lambda_{n,a_n} \\ \lambda_{2,a_2}^2 & \lambda_{3,a_3}^2 & \dots & \lambda_{n,a_n}^2 \\ \vdots & \vdots & \vdots & \vdots \\ \lambda_{2,a_2}^{r-1} & \lambda_{3,a_3}^{r-1} & \dots & \lambda_{n,a_n}^{r-1} \end{bmatrix} \begin{bmatrix} \mu_{2,1}^{(a)} \\ \mu_{3,1}^{(a)} \\ \vdots \\ \mu_{n,1}^{(a)} \end{bmatrix} = 0. \tag{12}$$

By (13) and the fact that $p_0(\lambda_{2,a_2}), p_0(\lambda_{3,a_3}), \dots, p_0(\lambda_{n,a_n})$ are all nonzero, $(\mu_{2,1}^{(a)}, \mu_{3,1}^{(a)}, \dots, \mu_{n,1}^{(a)})$ forms a Generalized Reed-Solomon code of length $n-1$ and dimension d. Thus any d out of $n-1$ elements in $\{\mu_{2,1}^{(a)}, \mu_{3,1}^{(a)}, \dots, \mu_{n,1}^{(a)}\}$ suffice to recover the whole set. Moreover, (11) implies that $\{c_{1,a(1,0)}, c_{1,a(1,1)}, \dots, c_{1,a(1,s-1)}\}$ can be determined by $\{\mu_{2,1}^{(a)}, \mu_{3,1}^{(a)}, \dots, \mu_{n,1}^{(a)}\}$. Consequently, we can recover C_1 by accessing any d surviving nodes and downloading the total of dl/s symbols of F from these nodes. This completes the proof. ■

V. MDS array codes with d-optimal repair property for several values of d simultaneously

In the previous two sections, we constructed MDS array codes with d-optimal repair property for a single value of d. In this section we give a simple extension of the previous constructions to make the code have d-optimal repair property for several values of d simultaneously. Let $n, k, m, d_1, d_2, \dots, d_m$ be any positive integers such that $k \leqslant d_1, \dots, d_m < n$. We will show that by replacing s in Construction 2 with the value

$$s = \mathrm{lcm}(d_1 + 1 - k, d_2 + 1 - k, \dots, d_m + 1 - k)$$

$$
\begin{aligned}
P\begin{bmatrix} 1 & 1 & \cdots & 1 \\ \lambda_{2,a_2} & \lambda_{3,a_3} & \cdots & \lambda_{n,a_n} \\ \lambda_{2,a_2}^2 & \lambda_{3,a_3}^2 & \cdots & \lambda_{n,a_n}^2 \\ \vdots & \vdots & \vdots & \vdots \\ \lambda_{2,a_2}^{r-1} & \lambda_{3,a_3}^{r-1} & \cdots & \lambda_{n,a_n}^{r-1} \end{bmatrix} &= \begin{bmatrix} p_0(\lambda_{2,a_2}) & p_0(\lambda_{3,a_3}) & \cdots & p_0(\lambda_{n,a_n}) \\ p_1(\lambda_{2,a_2}) & p_1(\lambda_{3,a_3}) & \cdots & p_1(\lambda_{n,a_n}) \\ \vdots & \vdots & \vdots & \vdots \\ p_{r-s-1}(\lambda_{2,a_2}) & p_{r-s-1}(\lambda_{3,a_3}) & \cdots & p_{r-s-1}(\lambda_{n,a_n}) \end{bmatrix} \\
&= \begin{bmatrix} p_0(\lambda_{2,a_2}) & p_0(\lambda_{3,a_3}) & \cdots & p_0(\lambda_{n,a_n}) \\ p_0(\lambda_{2,a_2})\lambda_{2,a_2} & p_0(\lambda_{3,a_3})\lambda_{3,a_3} & \cdots & p_0(\lambda_{n,a_n})\lambda_{n,a_n} \\ \vdots & \vdots & \vdots & \vdots \\ p_0(\lambda_{2,a_2})\lambda_{2,a_2}^{r-s-1} & p_0(\lambda_{3,a_3})\lambda_{3,a_3}^{r-s-1} & \cdots & p_0(\lambda_{n,a_n})\lambda_{n,a_n}^{r-s-1} \end{bmatrix}.
\end{aligned}
\tag{13}
$$

we obtain an $(n, k, l = s^n)$ MDS array code $\mathcal{C}$ with d_i-optimal repair property for all $i = 1, \dots, m$ simultaneously.

By Theorem IV.1, $\mathcal{C}$ is an MDS array code. In the next theorem we establish results about the repair properties of the code $\mathcal{C}$.

Theorem V.1. *The code $\mathcal{C}$ has d_i-optimal repair property for any $i \in [m]$.*

The proof can be found in [15].

Corollary V.2. *The $(n, k, (n-k)^n)$ MDS array code given by Construction 1 has d-optimal repair property if $(d+1-k)|(n-k)$.*

Example V.3. *A $(k+4, k, 4^{k+4})$ MDS array code given by Construction 1 will automatically have $(k+1)$-optimal repair property. A $(k+6, k, 6^{k+6})$ MDS array code given by Construction 1 has both $(k+1)$-optimal repair property and $(k+2)$-optimal repair property.*

VI. Explicit MDS array codes with (h, d)-optimal repair property for all $h \leqslant r$ and $k \leqslant d \leqslant n-h$ simultaneously

Given integers n and r, we construct a family of $(n, k = n-r, l)$ MDS array codes with (h, d)-optimal repair property for all $h \leqslant r$ and $k \leqslant d \leqslant n-h$ simultaneously. (The proofs of this section can be found in [15].)

Construction 3. *Let F be a finite field of size $|F| \geqslant sn$, where $s = lcm(1, 2, \dots, r)$. Let $\{\lambda_{i,j}\}_{i \in [n], j=0,1,\dots,s-1}$ be sn distinct elements in F. Let $l = s^n$. Consider the code family given by (2)-(3), where the matrices A_i are given by*

$$A_i = \sum_{a=0}^{l-1} \lambda_{i,a_i} e_a e_a^T,\ i = 1, \dots, n. \tag{14}$$

Here $\{e_a : a = 0, 1, \dots, l-1\}$ is the standard basis of F^l over F and a_i is defined in Construction 2.

VII. A family of Reed-Solomon Codes with asymptotically optimal repair bandwidth

In this section we take a different perspective of the repair problem: rather than constructing codes with optimal repair bandwidth, we study the repair bandwidth of a classical code family, the RS codes. Such a study was recently undertaken in [16], and we couple its linear repair scheme with the r-ary expansion idea of [8], [11] to construct a family of RS codes with asymptotically optimal repair bandwidth.

Given any n and k, we will specify a symbol field E, which is a degree l finite field extension over some finite field F, and a set of evaluation points Λ, and view the $\mathrm{RS}(n, k, \Lambda)$ codes as (n, k, l) array codes over F. We will show that they have repair bandwidth bounded above by $\frac{l(n+1)}{n-k}$ over the base field F. Since the optimal repair bandwidth for an (n, k, l) MDS array code is $\frac{l(n-1)}{n-k}$, we conclude that when $n \to \infty$, the ratio between the actual and the optimal repair bandwidth approaches 1 (the corresponding quantity of the construction in [16] is about 1.5).

A. The linear repair scheme of [16]

Suppose the symbol field of the code $\mathcal{C} = \mathrm{RS}(n, k, \Lambda)$ is E and we want to repair it over the base field $F \subseteq E$. More precisely, if a single codeword symbol is erased, we will recover this symbol by download sub-symbols of the base field F from the surviving nodes. Let $\mathrm{tr}(\beta) = \mathrm{tr}_{E/F}(\beta) := \beta + \beta^q + \beta^{q^2} + \dots + \beta^{q^{l-1}}$ be the trace function. In order to make the repair scheme F-linear, [16] uses F-linear transforms $L_\gamma : E \to F$ given by the *trace functionals* $L_\gamma(\beta) = \mathrm{tr}(\gamma\beta)$.

Let $\{\zeta_1, \dots, \zeta_l\}$ be a basis for E over F, and let $\{\mu_1, \dots, \mu_l\}$ be its dual (trace-orthogonal) basis, then for all $\beta \in E$

$$\beta = \sum_{i=1}^{l} (\mathrm{tr}(\zeta_i \beta)\mu_i).$$

Therefore, we can make the following observation: *If $\{\zeta_1, \dots, \zeta_l\}$ is a basis for E over F, then $\{tr(\zeta_i\beta)\}_{i=1}^l$ uniquely determines β.*

Let $\mathcal{C}^\perp$ be the dual code of $\mathcal{C} = \mathrm{RS}(n, k, \Lambda)$. Suppose that the codeword symbol c_i in a codeword $c = (c_1, \dots, c_n) \in \mathcal{C}$ is erased. We can find l codewords $\{c_j^\perp = (c_{j,1}^\perp, \dots, c_{j,n}^\perp)\}_{j=1}^l$ in $\mathcal{C}^\perp$ such that $\{c_{1,i}^\perp, \dots, c_{l,i}^\perp\}$ is a basis of E over F. By the observation above, knowing the values of $\{\mathrm{tr}(c_{j,i}^\perp c_i)\}_{j=1}^l$ suffices to recover the erased symbol c_i. Since the trace is an F-linear transformation, we have

$$\mathrm{tr}(c_{j,i}^\perp c_i) = -\sum_{t \neq i} \mathrm{tr}(c_{j,t}^\perp c_t) \text{ for all } j \in [l].$$

Thus knowing the values of $\{\{\mathrm{tr}(c_{j,t}^\perp c_t)\}_{j \in [l]}\}_{t \in [n], t \neq i}$ suffices to recover c_i. Let B_t be a maximal linearly independent subset of the set $\{c_{j,t}^\perp\}_{j \in [l]}$ over F. Again due to the F-linearity of the trace function, $\{\mathrm{tr}(c_{j,t}^\perp c_t)\}_{j \in [l]}$ can be calculated from $\{\mathrm{tr}(\beta c_t)\}_{\beta \in B_t}$. Consequently, c_i can be recovered from $\{\{\mathrm{tr}(\beta c_t)\}_{\beta \in B_t}\}_{t \in [n], t \neq i}$. The total number of sub-symbols in F we need to download from the surviving nodes to recover c_i is $\sum_{t\in[n],t\neq i}\dim_F(\{c_{j,t}^{\perp}\}_{j\in[l]})$.

We conclude that to efficiently recover c_i, we need to find l codewords in $\mathcal{C}^{\perp}$ that minimize the quantity $\sum_{t\in[n],t\neq i}\dim_F(\{c_{j,t}^{\perp}\}_{j\in[l]})$ under the condition that $\{c_{1,i}^{\perp},\dots,c_{l,i}^{\perp}\}$ is a basis for E over F.

As already remarked, $\mathcal{C}^{\perp}=\mathrm{GRS}(n,n-k,\Lambda,v)$ for some nonzero coefficients $v=(v_1,\dots,v_n)\in E^n$. Choosing a codeword from $\mathcal{C}^{\perp}=\mathrm{GRS}(n,n-k,\Lambda,v)$ is equivalent to choosing a polynomial with degree less than $n-k$. Suppose $\Lambda=\{\alpha_1,\dots,\alpha_n\}$. Since $v_1,\dots,v_n$ are nonzero constants, our task of efficiently repairing c_i is reduced to finding l polynomials $\{f_j\}_{j\in[l]}$ of degree less than $n-k$ such that the quantity

$$\sum_{t\in[n],t\neq i}\dim_F(\{f_j(\alpha_t)\}_{j\in[l]}) \tag{15}$$

is minimized under the condition that $\{f_1(\alpha_i),\dots,f_l(\alpha_i)\}$ is a basis for E over F.

B. The choice of symbol field and evaluation points

In this section we show how to find a symbol field E and a set of evaluation points Λ such that the corresponding RS code has nearly optimal repair bandwidth.

Suppose that n and k are arbitrary fixed numbers. Let F be a finite field and let $h(x)\in F[x]$ be a degree l irreducible polynomial over F, where $l=r^n, r=n-k$. Let β be a root of $h(x)$ and set the symbol field to be $E=F(\beta)$, i.e., the field generated by β over F. Clearly $\{1,\beta,\beta^2,\dots,\beta^{l-1}\}$ is a basis for E over F. Choose the set of evaluation points to be $\Lambda=\{\beta^{r^0},\beta^{r^1},\dots,\beta^{r^{n-1}}\}$.

Theorem VII.1. *The repair bandwidth of the code RS(n,k,Λ) over F is less than $l\frac{n+1}{n-k}$.*

Proof. We need to show that for every $i\in[n]$, we can find polynomials $f_{i,j}$ with $\deg(f_{i,j})<r, j=1,\dots,l$ such that $f_{i,1}(\beta^{r^{i-1}}),\dots,f_{i,l}(\beta^{r^{i-1}})$ form a basis for E over F and

$$\sum_{0\leqslant t<n,t\neq i-1}\dim_F(\{f_{i,j}(\beta^{r^t})\}_{j\in[l]})<\frac{l(n+1)}{n-k}.$$

For $a=0,1,\dots,l-1$, write its r-ary expansion as $a=(a_n,a_{n-1},\dots,a_1)$, where a_i is the i-th digit from the right. Define the set of l polynomials $\{f_{i,j}\}_{j\in[l]}=\{\beta^a x^s: a_i=0, s=0,1,\dots,r-1\}$.

It is easy to verify that

$$\{f_{i,j}(\beta^{r^{i-1}}):j\in[l]\}=\{1,\beta,\beta^2,\dots,\beta^{l-1}\}$$

(as sets), so the elements $\{f_{i,j}(\beta^{r^{i-1}})\}_{j\in[l]}$ form a basis for E over F. When $t<i-1$, we have

$$\{f_{i,j}(\beta^{r^t})\}_{j\in[l]}=\{\beta^a:a_i=0\}\bigcup$$
$$\Big(\bigcup_{u=0}^{r-2}\{\beta^a:a_i=1,a_{i-1}=\cdots=a_{t+2}=0,a_{t+1}=u\}\Big).$$

Thus $\dim_F(\{f_{i,j}(\beta^{r^t})\}_{j\in[l]})\leqslant\frac{l}{r}+(r-1)\frac{l}{r^{i-t}}$ if $t<i-1$. When $t>i-1$, we have

$$\{f_{i,j}(\beta^{r^t})\}_{j\in[l]}=\{\beta^a:a_i=0\}\bigcup$$
$$\Big(\bigcup_{u=0}^{r-2}\{\beta^{l+a}:a_n=\cdots=a_{t+2}=0,a_{t+1}=u,a_i=0\}\Big).$$

Thus $\dim_F(\{f_{i,j}(\beta^{r^t})\}_{j\in[l]})\leqslant\frac{l}{r}+(r-1)\frac{l}{r^{n-t+1}}$ for $t>i-1$. An upper bound on the sum of the dimensions is given by:

$$\begin{aligned}&\sum_{0\leqslant t<n,t\neq i-1}\dim_F(\{f_{i,j}(\beta^{r^t})\}_{j\in[l]})\\&\leqslant(n-1)\frac{l}{r}+(r-1)\sum_{t=0}^{i-2}\frac{l}{r^{i-t}}+(r-1)\sum_{t=i}^{n-1}\frac{l}{r^{n-t+1}}\\&=l\Big(\frac{n-1}{r}+\frac{r^{i-1}-1}{r^i}+\frac{r^{n-i}-1}{r^{n-i+1}}\Big)\\&<l\frac{n+1}{n-k}.\end{aligned}$$

The proof is complete. □

References

[1] M. Blaum, P. G. Farell, and H. van Tilborg, "Array codes," in *Handbook of Coding Theory*, V. Pless and W. C. Huffman, Eds. Elsevier Science, 1998, vol. II, ch. 22, pp. 1855–1909.

[2] A. Dimakis, P. Godfrey, Y. Wu, M. Wainwright, and K. Ramchandran, "Network coding for distributed storage systems," *IEEE Trans. on Information Theory*, vol. 56, no. 9, pp. 4539–4551, 2010.

[3] K. Rashmi, N. Shah, and P. Kumar, "Optimal exact-regenerating codes for distributed storage at the MSR and MBR points via a product-matrix construction," *IEEE Trans. on Information Theory*, vol. 57, no. 8, pp. 5227–5239, 2011.

[4] N. Shah, K. Rashmi, P. Kumar, and K. Ramchandran, "Interference alignment in regenerating codes for distributed storage: Necessity and code constructions," *IEEE Trans. on Information Theory*, vol. 58, no. 4, pp. 2134–2158, 2012.

[5] C. Suh and K. Ramchandran, "Exact-repair MDS code construction using interference alignment," *IEEE Trans. on Information Theory*, vol. 57, no. 3, pp. 1425–1442, 2011.

[6] Y. Wu and A. Dimakis, "Reducing repair traffic for erasure coding-based storage via interference alignment," in *Proc. 2009 IEEE Int. Sympos. Inform. Theory*, 2009, pp. 2276–2280.

[7] V. Cadambe, S. Jafar, H. Maleki, K. Ramchandran, and C. Suh, "Asymptotic interference alignment for optimal repair of MDS codes in distributed storage," *IEEE Trans. on Information Theory*, vol. 59, no. 5, pp. 2974–2987, 2013.

[8] V. Cadambe, C. Huang, and J. Li, "Permutation code: Optimal exact-repair of a single failed node in MDS code based distributed storage systems," in *Proc. 2011 IEEE Int. Sympos. Inform. Theory*, 2011, pp. 1225–1229.

[9] V. Cadambe, C. Huang, J. Li, and S. Mehrotra, "Polynomial length MDS codes with optimal repair in distributed storage," in *Proc. 45th Asilomar Conference on Signals, Systems and Computers*, 2011, pp. 1850–1854.

[10] D. Papailiopoulos, A. Dimakis, and V. Cadambe, "Repair optimal erasure codes through hadamard designs," *IEEE Trans. on Information Theory*, vol. 59, no. 5, pp. 3021–3037, 2013.

[11] I. Tamo, Z. Wang, and J. Bruck, "Zigzag codes: MDS array codes with optimal rebuilding," *IEEE Trans. on Information Theory*, vol. 59, no. 3, pp. 1597–1616, 2013.

[12] Z. Wang, I. Tamo, and J. Bruck, "Explicit MDS codes for optimal repair bandwidth," 2014, arXiv:1411.6328.

[13] ——, "On codes for optimal rebuilding access," in *Proceedings of the 49th Annual Allerton Conference on Communication, Control and Computing*, 2011, pp. 1374–1381.

[14] N. Raviv, N. Silberstein, and T. Etzion, "Constructions of high-rate MSR codes over small fields," 2015, arXiv:1505.00919.

[15] M. Ye and A. Barg, "Explicit constructions of high-rate MDS array codes with optimal repair bandwidth," 2016, arXiv:1604.00454.

[16] V. Guruswami and M. Wootters, "Repairing Reed-Solomon codes," 2015, arXiv:1509.04764.

[17] I. Tamo, Z. Wang, and J. Bruck, "Access versus bandwidth in codes for storage," *IEEE Trans. on Information Theory*, vol. 60, no. 4, pp. 2028–2037, 2014.