

Efficient Repair of $(k + 2, k)$ Degraded Read Friendly MDS Array Codes With Sub-packetization 2

Jie Li, *Senior Member, IEEE*, Xiaohu Tang, *Fellow, IEEE*

Abstract

In this paper, we present two constructions of degraded read friendly (DRF) MDS array codes with two parity nodes and a sub-packetization level of 2 over small finite fields, applicable for any arbitrary code length. The first construction achieves the smallest repair bandwidth among all existing constructions with the same parameters, and is asymptotically optimal with respect to the lower bound on the average repair bandwidth characterized by Zhang *et al.* The second construction supports two repair mechanisms, depending on whether computation within the helper nodes is permitted or not during the node repair process, thereby optimizing either the repair bandwidth or the rebuilding access.

I. INTRODUCTION

Distributed storage systems have widespread applications in practice, such as Windows Azure Storage [1] and Hadoop-based systems deployed in Facebook and Yahoo [2]. In these applications, data are stored across multiple unreliable storage nodes, making node failures common rather than a rare exception. To ensure reliability, erasure codes have been widely deployed in previous distributed storage systems, e.g., HDFS RAID [3]. As an important erasure code, MDS codes provide the optimal tradeoff between fault tolerance and storage overhead. However, MDS codes are inefficient in the face of node repair. For instance, in a distributed storage system based on an (n, k) MDS code, repairing a failed node requires downloading the entire content from any k surviving nodes. This results in an excessive *repair bandwidth*, which is defined as the amount of data downloaded to repair a failed node.

To enable efficient node repair, one can employ MDS array codes, where a codeword is represented by an $N \times n$ array for $N \geq 1$. In this context, N is referred to as the *sub-packetization level*. According to the MDS property and the cut-set bound demonstrated in [4], it has been shown that the repair bandwidth of (n, k) MDS array codes with sub-packetization level N is lower bounded by

$$\gamma \geq \gamma_{\text{optimal}} = \begin{cases} k, & \text{if } N = 1, \\ \frac{d}{d-k+1}N, & \text{if } N > 1. \end{cases} \quad (1)$$

Here, d such that $k \leq d < n$ denotes the number of helper nodes contacted during the repair process. MDS array codes ($N > 1$) that achieve this lower bound are said to have the *optimal repair bandwidth* and are also referred to as MSR codes in [4].

Besides the repair bandwidth, *rebuilding access* is also a critical metric for evaluate the performance of an MDS array code, which is defined as the amount of data accessed from the helper nodes during a repair process. Notably, the rebuilding access of an MDS array code is equal to or larger than its repair bandwidth. There are MSR codes where the rebuilding access also meets the lower bound of the repair bandwidth γ_{optimal} , e.g., the ones in [5]–[9].

Over the past decade, MSR codes have garnered significant attention [5]–[24]. However, MSR codes have the drawback of a large sub-packetization level N , e.g., $N \geq r^{\frac{n}{r+1}}$ when $d = n - 1$, where $r = n - k$. This constraint affects the selection of system parameters and complicates metadata management, e.g., n and r should be kept small to avoid a large sub-packetization level, thereby hindering implementation in practical systems [25].

In this paper, we consider MDS array codes that achieve both small repair bandwidth/rebuilding access and the smallest sub-packetization level. To reduce repair bandwidth/rebuilding access, the smallest sub-packetization level is 2 by (1). Moreover, the highest code-rate is $\frac{k}{k+2}$. These aspects are the focus of this work, and the lower bound in (1) becomes $\gamma \geq n - 1$ (which has been shown to be unachievable in subsequent research). An advantage of MDS array codes with a sub-packetization level of 2 is that discontinuous disk I/O can be effectively avoided during the repair process, codes with such a focus were also studied in [26] and [27] recently.

In the literature, there are a few constructions of $(n = k + 2, k)$ MDS array codes with an efficient repair strategy and a sub-packetization level of 2. The MSR code construction in [10] includes an MSR code with two parity nodes and a sub-packetization level of 2, but the code length is limited to $n = 5$, and the required finite field size should be $q > 10$. In [16],

J. Li was with the Department of Mathematics and Systems Analysis, Aalto University, FI-00076 Aalto, Finland, (e-mails: jieli873@gmail.com).

X. Tang is with the Information Coding and Transmission Key Lab of Sichuan Province, CSNMT Int. Coop. Res. Centre (MoST), Southwest Jiaotong University, Chengdu, 610031, China (e-mail: xhutang@swjtu.edu.cn).

an example of $(n = 6, k = 4)$ MDS array code with a sub-packetization level of 2 was presented, however, the two parity nodes do not endow an optimal repair bandwidth, resulting in an average repair bandwidth higher than that in [10]. In [28], an $(n = 5, k = 3)$ MSR code with a sub-packetization level of 2 was proposed, which can build over a finite field with size $q \geq 4$, i.e., much smaller than the one in [10] while the other properties are the same. This construction was later generalized to $n = 6$ in [29]. For convenience, we refer to the above MSR codes in [10], [16], [28], [29] as the product-matrix MSR code, the long MDS code, the GKW code, and the GKWX code, respectively.

In practice, during the interval between failure and recovery, users may request for temporarily unavailable data, commonly referred to as degraded reads [30], which should be optimized for efficiency. The DRF property is quite essential in practical systems, as it ensures the system's availability, enhancing users' experience by providing faster access to data, etc.

In [31], a lower bound for the average rebuilding access of $(n = k + 2, k)$ DRF MDS array codes with a sub-packetization level of 2 was characterized for general n , and a construction matching this lower bound was proposed, which will be referred to as the WHLBZZW code in this paper. The results in [31] and recent works [25], [32]–[34] demonstrate that MDS array codes with small sub-packetization levels and large code lengths can be constructed by sacrificing the optimality of the repair bandwidth and rebuilding access w.r.t. the lower bound in (1). However, lower bounds of the average repair bandwidth of $(n = k + 2, k)$ DRF MDS array codes with a sub-packetization level of 2 are still open.

Recall that rebuilding access is also a critical metric in distributed storage. Unlike MSR codes, the lower bound of rebuilding access is not the same as that of the repair bandwidth for general MDS array codes. In the recent work [35], Zhang *et al.* showed that the lower bound of the repair bandwidth is smaller than that of the rebuilding access for $(n = k + 2, k)$ MDS array codes of sub-packetization level 2 without the DRF property.

Since both repair bandwidth and rebuilding access are important metrics, therefore, in this paper, we focus on optimizing the two metrics separately. Specifically, we propose two new $(n = k + 2, k)$ DRF MDS array codes with a sub-packetization level of 2. The first code aims to optimize the repair bandwidth instead of rebuilding access, its average repair bandwidth is significantly smaller than the lower bound of the average rebuilding access of $(n = k + 2, k)$ MDS array codes with a sub-packetization level of 2 derived in [31]. The second MDS array code supports two repair mechanisms, one optimizing rebuilding access while the other optimizing repair bandwidth. One can choose the repair mechanism based on the computing capability of the helper nodes during the repair process. As in some distributed storage systems, such as Ceph [36] and Hadoop [37], [38], storage nodes are capable of performing computational tasks. Ideally, helper nodes should avoid computation during node repairs when they are already engaged in these tasks.

To summarize, this paper contains the following contributions.

- We propose an $(n = 4m, k = 4m - 2)$ MDS array code with a sub-packetization level of 2 over the finite field $\mathbf{F}_q$ with $q = 2^{2t} \geq \frac{3n}{4} + 1$. The normalized repair bandwidth (i.e., the ratio of the repair bandwidth to the file size) of the new MDS array code is $\frac{5n-8}{8n-16}$, which is much smaller than that of existing $(n = k + 2, k)$ MDS array codes with sub-packetization level 2, and is asymptotically optimal w.r.t. the lower bound derived in [35].
- We propose an $(n = 3m, k = 3m - 2)$ MDS array code with a sub-packetization level of 2 over the finite field $\mathbf{F}_q$ with $q \geq \frac{2n}{3}$ if $2 \mid q$ and $q \geq n$ otherwise. This MDS array code supports two kinds of repair mechanisms, the first one leads to an average normalized repair bandwidth of $\frac{2n-3}{3n-6}$, smaller than that in [31]. The second one leads to an average normalized rebuilding access of $\frac{13n-18}{18n-36}$, which is very close to the lower bound in [31].

The rest of this paper is organized as follows. Section II introduces some preliminaries on MDS array codes. Section III proposes the new construction of $(n = k + 2, k)$ MDS array codes with sub-packetization level 2 and small repair bandwidth. Section IV presents another construction of $(n = k + 2, k)$ MDS array codes with multiple repair strategies. An extensive comparison is carried out in Section V. Finally, Section VI concludes this paper.

II. PRELIMINARIES

In this section, we introduce some preliminaries on high-rate MDS codes. Denote by q a prime power and $\mathbf{F}_q$ the finite field with q elements. For any two integers a and b with $b > a$, denote by $[a : b]$ the set $\{a, a + 1, \dots, b - 1\}$.

A. $(n = k + 2, k)$ MDS codes

Let $\mathbf{f}_0, \mathbf{f}_1, \dots, \mathbf{f}_{n-1}$ represent the data stored across a distributed storage system consisting of n nodes, based on an $(n = k + 2, k)$ MDS array code with a sub-packetization level of 2. Here $\mathbf{f}_i = \begin{pmatrix} f_{i,0} \\ f_{i,1} \end{pmatrix}$ is a column vector of length 2 over $\mathbf{F}_q$. In this paper, similar to that in [31], we consider $(n = k + 2, k)$ DRF MDS codes that permit a definition in the following parity-check form:

$$\underbrace{\begin{pmatrix} I & I & \cdots & I \\ A_0 & A_1 & \cdots & A_{n-1} \end{pmatrix}}_A \begin{pmatrix} \mathbf{f}_0 \\ \mathbf{f}_1 \\ \vdots \\ \mathbf{f}_{n-1} \end{pmatrix} = \mathbf{0}_4, \quad (2)$$

where A_i ($i \in [0 : n)$) is a 2×2 matrix, $\mathbf{0}_4$ denotes the zero column vector of length 4, and will be abbreviated as $\mathbf{0}$ in the sequel if its length is clear. The $4 \times 2n$ matrix A in (2) is referred to as the *parity-check matrix* of the $(n = k + 2, k)$ MDS code with a sub-packetization level of 2.

Remark 1. The parity-check matrix of a DRF MDS code can also take other forms, such as replacing some identity matrices in the first block row of the parity-check matrix A with permutation matrices. In this paper, we define the $(k + 2, k)$ DRF MDS code with sub-packetization level 2 as the one that has a parity-check form in (2).

An advantage of restricting the parity-check matrix of the form in (2) rather than a general one as

$$\begin{pmatrix} B_0 & B_1 & \cdots & B_{n-1} \\ A_0 & A_1 & \cdots & A_{n-1} \end{pmatrix} \quad (3)$$

is the improved performance for degraded read of temporarily unavailable data. For example, when node i is failed, the degraded read to $f_{i,0}$ can be performed efficiently by accessing just $f_{j,0}$, $0 \leq j \neq i < n$ and computing $-\sum_{j=0, j \neq i}^{n-1} f_{j,0}$. In contrast, with a parity-check matrix in the general form, degraded read to $f_{i,0}$ becomes less efficient, as it may require accessing more symbols and performing multiplications of field elements.

An $(n = k + 2, k)$ MDS array code defined by (2) possesses the MDS property that the source file can be reconstructed by connecting to any k out of the n nodes. That is, any 2×2 sub-block matrix of A is nonsingular, i.e., $A_i - A_j$ is nonsingular for $i, j \in [0 : n)$ with $i \neq j$ [8].

To end this subsection, we present a lemma that will be used to verify the MDS property of the new array code in the next section.

Lemma 1. Let $q = 2^{2t}$ for some positive integer t , and let w be a primitive element of the finite field $\mathbf{F}_q$. Then we have

$$w^{2i} + w^i + 1 \neq 0 \text{ for } 0 \leq i < \frac{q-1}{3}. \quad (4)$$

Proof. For $0 \leq i < \frac{q-1}{3}$, it follows that $0 \leq 3i < q-1$, thus, $w^{3i} \neq 1$. Since $(w^i + 1)(w^{2i} + w^i + 1) = w^{3i} + 1 \neq 0$, then $w^{2i} + w^i + 1 \neq 0$. $\square$

B. Repair Process

Consider the repair of a failed node i ($i \in [0, n)$) of an $(n = k + 2, k)$ MDS array code. Let R_i be a 2×4 matrix, multiply which with (2) from both sides we obtain

$$R_i \begin{pmatrix} I & I & \cdots & I \\ A_0 & A_1 & \cdots & A_{n-1} \end{pmatrix} \begin{pmatrix} \mathbf{f}_0 \\ \mathbf{f}_1 \\ \vdots \\ \mathbf{f}_{n-1} \end{pmatrix} = \mathbf{0}. \quad (5)$$

Node i is then regenerated by solving the above equations. Rewrite (5) as

$$\underbrace{R_i \begin{pmatrix} I \\ A_i \end{pmatrix} \mathbf{f}_i}_{\text{useful data}} + \sum_{j=0, j \neq i}^{n-1} \underbrace{R_i \begin{pmatrix} I \\ A_j \end{pmatrix} \mathbf{f}_j}_{\text{interference by } \mathbf{f}_j} = \mathbf{0}. \quad (6)$$

Clearly, regenerating node i requires the coefficient matrix of the useful data in (6) to be of full rank, i.e.,

$$\text{rank} \left(R_i \begin{pmatrix} I \\ A_i \end{pmatrix} \right) = 2, \quad i \in [0, n), \quad (7)$$

and the interference can be cancelled, which is done by downloading a sufficient amount of data from the surviving nodes. The amount of data that needs to be downloaded (i.e., the repair bandwidth of node i) to cancel the interference in (6) is given by

$$\gamma_i = \sum_{j=0, j \neq i}^{n-1} \text{rank} \left(R_i \begin{pmatrix} I \\ A_j \end{pmatrix} \right). \quad (8)$$

Meanwhile, the rebuilding access of node i , i.e., the amount of data that needs to be accessed to cancel the interference in (6) is

$$\Gamma_i = \sum_{j=0, j \neq i}^{n-1} N_c \left(R_i \begin{pmatrix} I \\ A_j \end{pmatrix} \right), \quad (9)$$

where $N_c(A)$ denotes the number of nonzero columns of the matrix A .

For example, suppose that

$$R_i \begin{pmatrix} I \\ A_j \end{pmatrix} = \begin{pmatrix} 1 & 1 \\ x & x \end{pmatrix}, \quad (10)$$

for some x and $\mathbf{f}_j = \begin{pmatrix} f_{j,0} \\ f_{j,1} \end{pmatrix}$. To cancel the interference

$$R_i \begin{pmatrix} I \\ A_j \end{pmatrix} \mathbf{f}_j,$$

one needs to access two symbols $f_{j,0}$ and $f_{j,1}$ from node j , but only needs to download one symbol $f_{j,0} + f_{j,1}$, which is consistent with

$$N_c \left(\begin{pmatrix} 1 & 1 \\ x & x \end{pmatrix} \right) = 2 \text{ and } \text{rank} \left(\begin{pmatrix} 1 & 1 \\ x & x \end{pmatrix} \right) = 1.$$

C. Bounds on the Repair Bandwidth and Rebuilding Access of $(k+2, k)$ MDS Array Codes

To facilitate performance comparisons, we define the normalized repair bandwidth of node i of an (n, k) array code with sub-packetization N as the ratio of the repair bandwidth to the file size, i.e.,

$$\gamma_i^{Nor} = \frac{\gamma_i}{kN}.$$

We further define the *average normalized repair bandwidth* as

$$\gamma_{Ave}^{Nor} = \frac{\sum_{i=0}^{n-1} \gamma_i^{Nor}}{n} = \frac{\sum_{i=0}^{n-1} \gamma_i}{nkN}.$$

Similarly, the *average normalized rebuilding access* is defined as

$$\Gamma_{Ave}^{Nor} = \frac{\sum_{i=0}^{n-1} \Gamma_i}{nkN}.$$

In [31], a lower bound for the average normalized rebuilding access of $(n = k+2, k)$ DRF MDS array codes with a sub-packetization level of 2 is derived, showing that

$$\gamma_{Ave}^{Nor} \geq \frac{\min\{\Delta_3, \Delta_4\}}{2nk}, \quad (11)$$

where

$$\Delta_3 = \min_{l_1+l_2+l_3=n, l_1 \leq l_2} 2n(n-1) - n^2 + l_1^2 + l_2^2 + l_3^2 + l_1(l_3-1)$$

and

$$\Delta_4 = \min_{l_1+l_2+l_3+l_4=n, l_1 \leq l_2, l_3 \leq l_4} l_1(l_2+1) - l_2(n-l_2) - l_3(l_4+1) - l_4(n-l_4).$$

Furthermore, numerical experiments indicate that $\Gamma_{Ave}^{Nor} \geq \frac{\Delta_3}{2nk} = \frac{\min\{\Delta_3, \Delta_4\}}{2nk}$ for $n \in [4 : 51]$ and

$$\Gamma_{Ave}^{Nor} > 0.72. \quad (12)$$

In [35], the lower bounds on average normalized repair bandwidth and rebuilding access of $(n = k+2, k)$ MDS array codes with a sub-packetization level of 2 are derived for the non-DRF case, which shows

$$\gamma_{Ave}^{Nor} \geq \frac{5}{8}, \quad (13)$$

and

$$\Gamma_{Ave}^{Nor} \geq \frac{4k+1}{6k}. \quad (14)$$

By (12) and (14), we note that by sacrificing the DRF property, the rebuilding access of $(k+2, k)$ MDS array codes with a sub-packetization level of 2 can be smaller than that of the DRF case.

III. A NEW CONSTRUCTION OF $(k+2, k)$ MDS ARRAY CODES WITH ASYMPTOTICALLY OPTIMAL REPAIR BANDWIDTH

Construction 1. Let w be a primitive element of the finite field $\mathbf{F}_q$ and $a, b \in \mathbf{F}_q \setminus \{0, 1\}$. We construct an $(n = 4m, k = 4m - 2)$ array code with sub-packetization level 2 over $\mathbf{F}_q$, where the parity-check matrix is defined by A in (2) and

$$A_{4i+j} = \begin{cases} w^i \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}, & \text{if } j = 0, \\ w^i \begin{pmatrix} a & 0 \\ 0 & b \end{pmatrix}, & \text{if } j = 1, \\ w^i \begin{pmatrix} b & 0 \\ a & a \end{pmatrix}, & \text{if } j = 2, \\ w^i \begin{pmatrix} b & b \\ 0 & a \end{pmatrix}, & \text{if } j = 3, \end{cases}$$

for $i \in [0 : m)$. The repair matrices are defined by

$$R_{4i+j} = \begin{cases} \begin{pmatrix} 1 & a & & \\ & & 1 & 1 \end{pmatrix}, & \text{if } j = 0, \\ \begin{pmatrix} a & 1 & & \\ & & 1 & b \end{pmatrix}, & \text{if } j = 1, \\ \begin{pmatrix} 0 & 1 & & \\ & & 0 & 1 \end{pmatrix}, & \text{if } j = 2, \\ \begin{pmatrix} 1 & 0 & & \\ & & 1 & 0 \end{pmatrix}, & \text{if } j = 3, \end{cases}$$

for $i \in [0 : m)$.

Theorem 1. The average normalized repair bandwidth and average normalized rebuilding access of the array code in Construction 1 are

$$\gamma_{Ave}^{Nor} = \frac{5n - 8}{8n - 16} \text{ and } \Gamma_{Ave}^{Nor} = \frac{13n - 16}{16n - 32}$$

if the array code is constructed over the finite field $\mathbf{F}_q$ with $q = 2^{2t}$ for some positive integer t , and $a = w^{\frac{q-1}{3}}$, $b = w^{\frac{2(q-1)}{3}}$. This implies that the average normalized repair bandwidth is asymptotically optimal w.r.t. the lower bound in (13).

Proof. Consider the repair of node $4i + j$, where $i \in [0 : m)$ and $j \in [0 : 4)$.

i) When $j = 0$, for $i, i' \in [0 : m)$, we have

$$R_{4i} \begin{pmatrix} I \\ A_{4i'} \end{pmatrix} = \begin{pmatrix} 1 & a & & \\ & & 1 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 \\ 0 & 1 \\ w^{i'} & 0 \\ 0 & w^{i'} \end{pmatrix} = \begin{pmatrix} 1 & a \\ w^{i'} & w^{i'} \end{pmatrix},$$

which is of full rank if and only if

$$a \neq 1. \tag{15}$$

$$R_{4i} \begin{pmatrix} I \\ A_{4i'+1} \end{pmatrix} = \begin{pmatrix} 1 & a & & \\ & & 1 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 \\ 0 & 1 \\ w^{i'}a & 0 \\ 0 & w^{i'}b \end{pmatrix} = \begin{pmatrix} 1 & a \\ w^{i'}a & w^{i'}b \end{pmatrix},$$

which has rank 1 if and only if

$$b = a^2. \tag{16}$$

$$R_{4i} \begin{pmatrix} I \\ A_{4i'+2} \end{pmatrix} = \begin{pmatrix} 1 & a & & \\ & & 1 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 \\ 0 & 1 \\ w^{i'}b & 0 \\ w^{i'}a & w^{i'}a \end{pmatrix} = \begin{pmatrix} 1 & a \\ w^{i'}(a+b) & w^{i'}a \end{pmatrix},$$

which has rank 1 if and only if

$$a + b = 1 \quad (17)$$

$$R_{4i} \begin{pmatrix} I \\ A_{4i'+3} \end{pmatrix} = \begin{pmatrix} 1 & a & & \\ & & 1 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 \\ 0 & 1 \\ w^{i'}b & w^{i'}a \\ 0 & w^{i'}b \end{pmatrix} = \begin{pmatrix} 1 & a \\ w^{i'}b & w^{i'}(a+b) \end{pmatrix},$$

which has rank 1 if and only if

$$ab = a + b. \quad (18)$$

Therefore, if (15)–(18) are satisfied, the repair bandwidth and rebuilding access of node $4i$ are given by

$$\begin{aligned} \gamma_{4i} &= \sum_{j=0, j \neq 4i}^{n-1} \text{rank} \left(R_{4i} \begin{pmatrix} I \\ A_j \end{pmatrix} \right) \\ &= \sum_{i'=0, i' \neq i}^{m-1} \text{rank} \left(R_{4i} \begin{pmatrix} I \\ A_{4i'} \end{pmatrix} \right) + \sum_{i'=0}^{m-1} \sum_{j'=1}^3 \text{rank} \left(R_{4i} \begin{pmatrix} I \\ A_{4i'+j'} \end{pmatrix} \right) \\ &= 2(m-1) + 3m \\ &= 5m - 2 \end{aligned}$$

and

$$\Gamma_{4i} = \sum_{j=0, j \neq 4i}^{n-1} N_c \left(R_{4i} \begin{pmatrix} I \\ A_j \end{pmatrix} \right) = 8m - 2.$$

ii) When $j = 1$, for $i, i' \in [0 : m)$, we have

$$R_{4i+1} \begin{pmatrix} I \\ A_{4i'} \end{pmatrix} = \begin{pmatrix} a & 1 & & \\ & & 1 & b \end{pmatrix} \begin{pmatrix} 1 & 0 \\ 0 & 1 \\ w^{i'} & 0 \\ 0 & w^{i'} \end{pmatrix} = \begin{pmatrix} a & 1 \\ w^{i'} & w^{i'}b \end{pmatrix},$$

which has rank 1 if and only if

$$ab = 1. \quad (19)$$

$$R_{4i+1} \begin{pmatrix} I \\ A_{4i'+1} \end{pmatrix} = \begin{pmatrix} a & 1 & & \\ & & 1 & b \end{pmatrix} \begin{pmatrix} 1 & 0 \\ 0 & 1 \\ w^{i'}a & 0 \\ 0 & w^{i'}b \end{pmatrix} = \begin{pmatrix} a & 1 \\ w^{i'}a & w^{i'}b^2 \end{pmatrix},$$

which is of full rank if and only if

$$b^2 \neq 1. \quad (20)$$

$$R_{4i+1} \begin{pmatrix} I \\ A_{4i'+2} \end{pmatrix} = \begin{pmatrix} a & 1 & & \\ & & 1 & b \end{pmatrix} \begin{pmatrix} 1 & 0 \\ 0 & 1 \\ w^{i'}b & 0 \\ w^{i'}a & w^{i'}a \end{pmatrix} = \begin{pmatrix} a & 1 \\ w^{i'}(b+ab) & w^{i'}ab \end{pmatrix},$$

which has rank 1 if and only if

$$a^2 = a + 1. \quad (21)$$

$$R_{4i+1} \begin{pmatrix} I \\ A_{4i'+3} \end{pmatrix} = \begin{pmatrix} a & 1 & & \\ & & 1 & b \end{pmatrix} \begin{pmatrix} 1 & 0 \\ 0 & 1 \\ w^{i'}b & w^{i'}b \\ 0 & w^{i'}a \end{pmatrix} = \begin{pmatrix} a & 1 \\ w^{i'}b & w^{i'}(b+ab) \end{pmatrix},$$

which has rank 1 if and only if

$$a^2 + a = 1 \quad (22)$$

holds.

Similarly, if (19)–(22) are satisfied, the repair bandwidth and rebuilding access of node $4i + 1$ are respectively

$$\gamma_{4i+1} = 5m - 2$$

and

$$\Gamma_{4i+1} = 8m - 2.$$

iii) When $j = 2, 3$, for $i, i' \in [0 : m)$, it is easy to obtain

$$N_c \left(R_{4i+j} \begin{pmatrix} I \\ A_{4i'+j'} \end{pmatrix} \right) = \text{rank} \left(R_{4i+j} \begin{pmatrix} I \\ A_{4i'+j'} \end{pmatrix} \right) = \begin{cases} 1, & \text{if } j' \neq j, \\ 2, & \text{if } j' = j. \end{cases}$$

Thus, the repair bandwidth and rebuilding access of node $4i + j$ are

$$\gamma_{4i+j} = \Gamma_{4i+j} = 5m - 2, j = 2, 3.$$

Note that if (15)–(22) are satisfied, then $2 \mid q$, $a \neq 1$, $a^3 = 1$, $b = a^2$, and $a^2 + a = 1$. These conditions can be satisfied if a with $a \neq 1$ is a cubic root of unity in $\mathbf{F}_{2^{2t}}$ for some positive integer t . Specifically, (15)–(22) hold if $a = w^{\frac{q-1}{3}}$ and $b = w^{\frac{2(q-1)}{3}}$ and $q = 2^{2t}$.

With the above repair strategy, a direct calculation shows that $\gamma_{Ave}^{Nor} = \frac{5n-8}{8n-16}$ and $\Gamma_{Ave}^{Nor} = \frac{13n-16}{16n-32}$, where $\gamma_{Ave}^{Nor} = \frac{5k+2}{8k}$ is asymptotically optimal w.r.t. the lower bound in (13). $\square$

Theorem 2. *The array code in Construction 1 is MDS if*

- i) $a, b \notin \{w^t | t \in [0 : m) \cup [q - m : q - 1)\}$;
- ii) $a \neq w^t b$ for $t \in [-m + 1 : m)$;
- iii) $w^{2i} + w^{2i'} - w^i w^{i'} \neq 0$ for $i, i' \in [0 : m)$.

Proof. It suffices to analyze the nonsingularity of $A_{4i+j} - A_{4i'+j'}$ for $i, i' \in [0 : m)$ and $j, j' \in [0 : 4)$ with $(i, j) \neq (i', j')$, which can be proceeded according to the following four cases.

- i) When $j = j'$ and $i \neq i'$,

$$\det(A_{4i+j} - A_{4i'+j'}) = (1 - w^{i'-i}) \det(A_{4i+j}) \neq 0.$$

- ii) When $j = 0, j' = 1, 2, 3$,

$$\det(A_{4i+j} - A_{4i'+j'}) = (w^i - w^{i'} a)(w^i - w^{i'} b),$$

which is nonzero if $a, b \neq w^{i-i'}$, i.e., $a, b \notin \{w^t | t \in [0 : m) \cup [q - m : q - 1)\}$.

- iii) When $j = 1, j' = 2, 3$,

$$\det(A_{4i+j} - A_{4i'+j'}) = (w^i a - w^{i'} b)(w^i b - w^{i'} a),$$

which is nonzero if $a \neq w^{i'-i} b, w^{i-i'} b$, i.e., $a \neq w^t b$ for $t \in [-m + 1 : m)$.

- iv) When $j = 2, j' = 3$,

$$\det(A_{4i+j} - A_{4i'+j'}) = \det \begin{pmatrix} w^i b - w^{i'} b & -w^{i'} b \\ w^i a & w^i a - w^{i'} a \end{pmatrix} = ab \det \begin{pmatrix} w^i - w^{i'} & -w^{i'} \\ w^i & w^i - w^{i'} \end{pmatrix} = ab(w^{2i} + w^{2i'} - w^i w^{i'}),$$

which is nonzero if $a, b \neq 0$ and $w^{2i} + w^{2i'} - w^i w^{i'} \neq 0$ for $i, i' \in [0 : m)$. $\square$

Theorem 3. *The $(n = 4m, k = 4m - 2)$ array code in Construction 1 is an MDS array code over $\mathbf{F}_q$ with an average normalized repair bandwidth of $\gamma_{Ave}^{Nor} = \frac{5n-8}{8n-16}$ and an average normalized rebuilding access of $\Gamma_{Ave}^{Nor} = \frac{13n-16}{16n-32}$ if $q \geq 3m + 1$ and $q = 2^{2t}$ for some positive integer t .*

Proof. By Theorems 1 and 2, it suffices to prove that i)–iii) of Theorem 2 hold under $a = w^{\frac{q-1}{3}}$ and $b = w^{\frac{2(q-1)}{3}}$.

If $q \geq 3m + 1$, then we have,

$$m \leq \frac{q-1}{3} < q - m, m \leq \frac{2(q-1)}{3} < q - m,$$

which implies that Theorem 2-i) is satisfied.

Clearly, Theorem 2-ii) holds since $a = w^{\frac{q-1}{3}}$ and $b = w^{\frac{2(q-1)}{3}}$.

Now we verify Theorem 2-iii). For $0 \leq i \leq i' < m$,

$$w^{2i} + w^{2i'} - w^i w^{i'} = w^{2i}(1 + w^{2(i'-i)} + w^{i'-i}),$$

which is nonzero by Lemma 1 since $0 \leq i' - i < m \leq \frac{q-1}{3}$. The statement holds similarly for $0 \leq i' < i < m$. Thus, Theorem 2-iii) is satisfied. $\square$

IV. A CONSTRUCTION OF MDS ARRAY CODES WITH TWO REPAIR STRATEGIES

In this section, we present another construction of $(n = k + 2, k)$ MDS array codes that incorporates two repair strategies.

Construction 2. Let w be a primitive element of the finite field $\mathbf{F}_q$, we construct an $(n = 3m, k = 3m - 2)$ array code with a sub-packetization level of 2 over $\mathbf{F}_q$. The parity-check matrix is defined by A in (2) and

$$A_{3i+j} = \begin{cases} \begin{pmatrix} \lambda_i & -1 \\ 0 & \lambda_i + 1 \end{pmatrix}, & \text{if } j = 0, \\ \begin{pmatrix} \lambda_i & 0 \\ 1 & \lambda_i + 1 \end{pmatrix}, & \text{if } j = 1, \\ \begin{pmatrix} \lambda_i + 1 & 0 \\ 0 & \lambda_i \end{pmatrix}, & \text{if } j = 2, \end{cases} \quad (23)$$

for $i \in [0 : m)$.

Theorem 4. The $(n = 3m, k = 3m - 2)$ array code in Construction 2 is MDS if the following conditions are met:

- i) $\lambda_i \neq \lambda_{i'}$ for $0 \leq i < i' < m$;
- ii) $\lambda_i - \lambda_{i'} \neq \pm 1$ for $i, i' \in [0 : m)$.

Furthermore, these requirements can be satisfied over a finite field $\mathbf{F}_q$ with

$$q \geq \begin{cases} \frac{2n}{3}, & \text{if } 2 \mid q, \\ n, & \text{otherwise.} \end{cases}$$

Proof. It suffices to analyze the nonsingularity of $A_{3i+j} - A_{3i'+j'}$ for $i, i' \in [0 : m)$ and $j, j' \in [0 : 3)$ with $(i, j) \neq (i', j')$, which can be proceeded according to the following four cases.

Case 1: When $j = j'$ and $i \neq i'$, if $j = 0$,

$$\det(A_{3i+j} - A_{3i'+j'}) = \det \begin{pmatrix} \lambda_i - \lambda_{i'} & 0 \\ 0 & \lambda_i - \lambda_{i'} \end{pmatrix} \neq 0$$

if and only if $\lambda_i \neq \lambda_{i'}$. The same argument applies for $j = 1, 2$.

Case 2: When $j = 0, j' = 1$,

$$\det(A_{3i+j} - A_{3i'+j'}) = \det \begin{pmatrix} \lambda_i - \lambda_{i'} & -1 \\ -1 & \lambda_i - \lambda_{i'} \end{pmatrix} = (\lambda_i - \lambda_{i'})^2 - 1 \neq 0,$$

which is nonzero if $\lambda_i - \lambda_{i'} \neq \pm 1$.

Case 3: When $j = 0, j' = 2$,

$$\det(A_{3i+j} - A_{3i'+j'}) = \det \begin{pmatrix} \lambda_i - \lambda_{i'} - 1 & -1 \\ 0 & \lambda_i - \lambda_{i'} + 1 \end{pmatrix} = (\lambda_i - \lambda_{i'})^2 - 1,$$

which is nonzero if $\lambda_i - \lambda_{i'} \neq \pm 1$.

Case 4: When $j = 1, j' = 2$,

$$\det(A_{3i+j} - A_{3i'+j'}) = \det \begin{pmatrix} \lambda_i - \lambda_{i'} - 1 & 0 \\ 1 & \lambda_i - \lambda_{i'} + 1 \end{pmatrix} = (\lambda_i - \lambda_{i'})^2 - 1,$$

which is nonzero if $\lambda_i - \lambda_{i'} \neq \pm 1$.

Now, let us determine the required finite field size. If the underlying finite field has characteristic two, then a finite field with size $q \geq \frac{2n}{3} = 2m$ is sufficient. Since we can partition the finite field into two disjoint sets S and T such that for any $a \in S$ we have $a + 1 \in T$. We can then select pairwise distinct $\lambda_0, \lambda_1, \dots, \lambda_{m-1}$ from the set S to fulfill conditions i) and ii).

If the underlying finite field has an odd characteristic, a field size $q \geq n = 3m$ is sufficient. Initially, we can choose λ_0 to be an arbitrary element in $\mathbf{F}_q$. For $1 \leq i < m$, we can select λ_i to be any element in

$$\mathbf{F}_q \setminus \{\lambda_j, \lambda_j - 1, \lambda_j + 1 \mid 0 \leq j < i\}.$$

With these assignments, i) and ii) are fulfilled. $\square$

For the array code in Construction 2, we have two repair strategies for nodes $3i + 2$ for $i \in [0 : m)$. One strategy aims to minimize the repair bandwidth, while the other focuses on minimizing the rebuilding access. We first present the strategy that minimizes the repair bandwidth of nodes $3i + 2$ for $i \in [0 : m)$ and demonstrate the average normalized repair bandwidth and average normalized rebuilding access of the array code in Construction 2 under this strategy.

Theorem 5. *The nodes of Construction 2 can be repaired with an average normalized repair bandwidth of $\gamma_{Ave}^{Nor} = \frac{2n-3}{3n-6}$ and an average normalized rebuilding access of $\Gamma_{Ave}^{Nor} = \frac{7n-9}{9n-18}$.*

Proof. Let the repair matrices be

$$R_{im+j} = \begin{cases} \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix}, & \text{if } j = 0, \\ \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}, & \text{if } j = 1, \\ \begin{pmatrix} 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \end{pmatrix}, & \text{if } j = 2, \end{cases}$$

for $i \in [0 : m)$.

i) When $j = 0$, for $i, i' \in [0 : m)$, we have

$$R_{3i} \begin{pmatrix} I \\ A_{3i'} \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} \begin{pmatrix} 1 & 0 \\ 0 & 1 \\ \lambda_{i'} & -1 \\ 0 & \lambda_{i'} + 1 \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ \lambda_{i'} & -1 \end{pmatrix},$$

which is of full rank.

$$R_{3i} \begin{pmatrix} I \\ A_{3i'+1} \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} \begin{pmatrix} 1 & 0 \\ 0 & 1 \\ \lambda_{i'} & 0 \\ 1 & \lambda_{i'} + 1 \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ \lambda_{i'} & 0 \end{pmatrix},$$

which has rank 1.

$$R_{3i} \begin{pmatrix} I \\ A_{3i'+2} \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} \begin{pmatrix} 1 & 0 \\ 0 & 1 \\ \lambda_{i'} + 1 & 0 \\ 0 & \lambda_{i'} \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ \lambda_{i'} + 1 & 0 \end{pmatrix},$$

which has rank 1.

Thus, the repair bandwidth of node $3i$ is

$$\begin{aligned} \gamma_{3i} &= \sum_{j=0, j \neq 3i}^{n-1} \text{rank} \left(R_{3i} \begin{pmatrix} I \\ A_j \end{pmatrix} \right) \\ &= \sum_{i'=0, i' \neq i}^{m-1} \text{rank} \left(R_{3i} \begin{pmatrix} I \\ A_{3i'} \end{pmatrix} \right) + \sum_{i'=0}^{m-1} \sum_{j'=1}^2 \text{rank} \left(R_{3i} \begin{pmatrix} I \\ A_{3i'+j'} \end{pmatrix} \right) \\ &= 2(m-1) + 2m \\ &= 4m - 2, \end{aligned}$$

while the rebuilding access is

$$\begin{aligned} \Gamma_{3i} &= \sum_{j=0, j \neq 3i}^{n-1} N_c \left(R_{3i} \begin{pmatrix} I \\ A_j \end{pmatrix} \right) \\ &= \sum_{i'=0, i' \neq i}^{m-1} N_c \left(R_{3i} \begin{pmatrix} I \\ A_{3i'} \end{pmatrix} \right) + \sum_{i'=0}^{m-1} \sum_{j'=1}^2 N_c \left(R_{3i} \begin{pmatrix} I \\ A_{3i'+j'} \end{pmatrix} \right) \\ &= 2(m-1) + 2m \\ &= 4m - 2, \end{aligned}$$

ii) When $j = 1$, similar to the previous case, we have that the repair bandwidth and rebuilding access of node $3i + 1$ are

$$\gamma_{3i+1} = \Gamma_{3i+1} = 4m - 2.$$

iii) When $j = 2$, for $i, i' \in [0 : m)$, we have

$$R_{3i+2} \begin{pmatrix} I \\ A_{3i'+2} \end{pmatrix} = \begin{pmatrix} 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 \\ 0 & 1 \\ \lambda_{i'} + 1 & 0 \\ 0 & \lambda_{i'} \end{pmatrix} = \begin{pmatrix} 1 & 1 \\ \lambda_{i'} + 1 & \lambda_{i'} \end{pmatrix},$$

which is of full rank.

$$R_{3i+2} \begin{pmatrix} I \\ A_{3i'} \end{pmatrix} = \begin{pmatrix} 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 \\ 0 & 1 \\ \lambda_{i'} & -1 \\ 0 & \lambda_{i'} + 1 \end{pmatrix} = \begin{pmatrix} 1 & 1 \\ \lambda_{i'} & \lambda_{i'} \end{pmatrix},$$

which has rank 1.

$$R_{3i+2} \begin{pmatrix} I \\ A_{3i'+1} \end{pmatrix} = \begin{pmatrix} 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 \\ 0 & 1 \\ \lambda_{i'} & 0 \\ 1 & \lambda_{i'} + 1 \end{pmatrix} = \begin{pmatrix} 1 & 1 \\ \lambda_{i'} + 1 & \lambda_{i'} + 1 \end{pmatrix},$$

which has rank 1.

That is, the repair bandwidth of node $3i + 2$ is

$$\begin{aligned} \gamma_{3i+2} &= \sum_{j=0, j \neq 3i+2}^{n-1} \text{rank} \left(R_{3i+2} \begin{pmatrix} I \\ A_j \end{pmatrix} \right) \\ &= \sum_{i'=0, i' \neq i}^{m-1} \text{rank} \left(R_{3i+2} \begin{pmatrix} I \\ A_{3i'+2} \end{pmatrix} \right) + \sum_{i'=0}^{m-1} \sum_{j'=0}^1 \text{rank} \left(R_{3i+2} \begin{pmatrix} I \\ A_{3i'+j'} \end{pmatrix} \right) \\ &= 2(m-1) + 2m \\ &= 4m - 2. \end{aligned}$$

Similarly, the rebuilding access of node $3i + 2$ is $\Gamma_{3i+2} = 6m - 2$.

With the above repair strategy, a direct calculation shows that $\gamma_{Ave}^{Nor} = \frac{2n-3}{3n-6}$ and $\Gamma_{Ave}^{Nor} = \frac{7n-9}{9n-18}$. $\square$

In the following, we propose an alternative repair strategy for nodes $3i + 2$ where $i \in [0 : m)$, which aims to minimize the rebuilding access. We also present the average normalized repair bandwidth and average normalized rebuilding access of the array code in Construction 2 under this repair strategy.

Theorem 6. *The nodes of Construction 2 can be repaired with the average normalized repair bandwidth and average normalized rebuilding access being*

$$\gamma_{Ave}^{Nor} = \Gamma_{Ave}^{Nor} = \frac{13n - 18}{18n - 36}.$$

Proof. We modify the repair matrix of node $3i + 2$ to

$$R'_{im+2} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 \end{pmatrix}$$

for $i \in [0 : m)$, while keeping the repair matrices of the other nodes the same as those used in the first repair strategy in the proof of Theorem 5.

Now for $i, i' \in [0 : m)$, we have

$$R'_{3i+2} \begin{pmatrix} I \\ A_{3i'+2} \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 \end{pmatrix} \begin{pmatrix} 1 & 0 \\ 0 & 1 \\ \lambda_{i'} + 1 & 0 \\ 0 & \lambda_{i'} \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ \lambda_{i'} + 1 & 1 \end{pmatrix},$$

which is of full rank.

$$R'_{3i+2} \begin{pmatrix} I \\ A_{3i'} \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 \end{pmatrix} \begin{pmatrix} 1 & 0 \\ 0 & 1 \\ \lambda_{i'} & -1 \\ 0 & \lambda_{i'} + 1 \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ \lambda_{i'} & 0 \end{pmatrix},$$

which has rank 1.

$$R'_{3i+2} \begin{pmatrix} I \\ A_{3i'+1} \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 \end{pmatrix} \begin{pmatrix} 1 & 0 \\ 0 & 1 \\ \lambda_{i'} & 0 \\ 1 & \lambda_{i'} + 1 \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ \lambda_{i'} & 1 \end{pmatrix},$$

which has a rank of 2.

With this repair strategy, the repair bandwidth of node $3i + 2$ is

$$\begin{aligned} \gamma'_{3i+2} &= \sum_{j=0, j \neq 3i+2}^{n-1} \text{rank} \left(R'_{3i+2} \begin{pmatrix} I \\ A_j \end{pmatrix} \right) \\ &= \sum_{i'=0, i' \neq i}^{m-1} \text{rank} \left(R'_{3i+2} \begin{pmatrix} I \\ A_{3i'+2} \end{pmatrix} \right) + \sum_{i'=0}^{m-1} \sum_{j'=0}^1 \text{rank} \left(R'_{3i+2} \begin{pmatrix} I \\ A_{3i'+j'} \end{pmatrix} \right) \\ &= 2(m-1) + 3m \\ &= 5m - 2. \end{aligned}$$

Similarly, the rebuilding access of node $3i + 2$ is $\Gamma'_{3i+2} = 5m - 2$.

With the new repair strategy for nodes $3i+2$ where $i \in [0 : m)$, a direct calculation shows that $\gamma'_{Ave} = \Gamma'_{Ave} = \frac{13n-18}{18n-36}$. $\square$

Remark 2. In Construction 2, we assumed $n = 3m$ and there are three types of matrices in the building blocks of the parity-check matrix, i.e., upper triangular matrices, lower triangular matrices, and diagonal matrices, with each type appearing an equal number of times. While this construction is systematic, it does not always yield the smallest average rebuilding access. An extreme optimization of the average rebuilding access may dynamically adjust the number of occurrences of each type of building block. For example, we can replace the building blocks of the parity-check matrix in (23) with the following

$$A_i = \begin{cases} \begin{pmatrix} \lambda_i & -1 \\ 0 & \lambda_i + 1 \end{pmatrix}, & \text{if } 0 \leq i < l_1, \\ \begin{pmatrix} \lambda_{i-l_1} & 0 \\ 1 & \lambda_{i-l_1} + 1 \end{pmatrix}, & \text{if } l_1 \leq i < l_1 + l_2, \\ \begin{pmatrix} \lambda_{i-l_1-l_2} + 1 & 0 \\ 0 & \lambda_{i-l_1-l_2} \end{pmatrix}, & \text{if } l_1 + l_2 \leq i < n. \end{cases}$$

Remark 3. If we replace the building blocks of the parity-check matrix in (23) according to Remark 2, and let $l_3 = n - l_1 - l_2$, then a direct calculation shows that

$$\Gamma_{Ave}^{Nor} = \frac{n^2 - 2n + l_1^2 + l_2^2 + l_3^2 + l_2 l_3}{2nk}, \quad (24)$$

and we can choose the values of l_1 and l_2 such that $l_3 \leq l_1$ and Γ_{Ave}^{Nor} in (24) is minimized. In addition, the repair of only the last l_3 nodes requires access to one more symbol than the construction in [31] that achieves the lower bound in (11).

In fact, $\lim_{n \rightarrow +\infty} \frac{13n-18}{18n-36} \in [0.7222, 0.7223)$, indicating that the average normalized rebuilding access of the MDS code in Construction 2 is very close to the lower bounds in (11) and (12).

V. COMPARISONS

In this section, we compare some key parameters among the proposed MDS array codes and several existing notable MDS codes with two parity nodes and a sub-packetization level of 2.

From Table I, we observe that the proposed MDS codes have the following advantages:

- The product-matrix MSR code in [10], the long MDS code in [16], the GKW code in [28], and the GKWX code in [29] with two parities and a sub-packetization level of 2 only support a code length of 5 or 6. In contrast, the new MDS array codes in Constructions 1 and 2 support arbitrary code lengths.

TABLE I
A COMPARISON OF SOME KEY PARAMETERS AMONG THE MDS ARRAY CODES PROPOSED IN THIS PAPER AND SOME EXISTING ONES WITH TWO PARITY
NODES AND A SUB-PACKETIZATION LEVEL OF 2

	Code length n	γ_{Avg}^{Nor}	Γ_{Avg}^{Nor}	Required field size q	References
The product-matrix MSR code	5	$2/3$	$4/3$	≥ 10	[10]
The long MDS code	6	$3/4$	$23/24$	≥ 4	[16]
The GKW code	5	$2/3$	$7/6$	≥ 4	[28]
The GKWX code	6	$5/8$	$47/48$	≥ 4	[29]
The WHLBZZW code	Arbitrary	> 0.72	> 0.72	Sufficient large	[31]
Non-DRF code 1	Arbitrary	$\frac{k + \lfloor n/4 \rfloor + \frac{1}{n}(n \bmod 4)\lceil n/4 \rceil}{2k}$	Not discussed	$> n + 2$	[35]
Non-DRF Code 2	Arbitrary	Not discussed	$\frac{k + \lfloor n/3 \rfloor + \frac{1}{n}(n \bmod 3)\lceil n/3 \rceil}{2k}$	$> n$	[35]
MDS codes in Construction 1	Arbitrary	$\frac{5n-8}{8n-16}$	$\frac{13n-16}{16n-32}$ (Asymptotically optimal)	$> \frac{3n}{4}$	Thm. 3
MDS codes in Construction 2	Arbitrary	$\begin{cases} \frac{2n-3}{3n-6}, & \text{1st repair strategy} \\ \frac{13n-18}{18n-36}, & \text{2nd repair strategy} \end{cases}$	$\begin{cases} \frac{7n-9}{9n-18}, & \text{1st repair strategy} \\ \frac{13n-18}{18n-36}, & \text{2nd repair strategy} \end{cases}$	$\geq \begin{cases} \frac{2n}{3}, & \text{if } 2 \mid q \\ n, & \text{otherwise} \end{cases}$	Thms. 4, 5 Thms. 4, 6

- Compared to the WHLBZZW code in [31], the new MDS code in Construction 1 has a smaller repair bandwidth and is explicit over a small finite field.
- The new MDS array code in Construction 1 is asymptotically optimal w.r.t. the lower bound on the average repair bandwidth in (13).
- Compared to the codes in [35], both new MDS array codes in Constructions 1 and 2 exhibit the DRF property and require a smaller finite field size.
- The new MDS code in Construction 2 supports two repair mechanisms. The first mechanism achieves a smaller repair bandwidth than that in [31], while the second mechanism results in rebuilding access that is very close to the lower bounds in (11) and (12).

VI. CONCLUSION

In this paper, we proposed two constructions of MDS array codes with two parity nodes and a sub-packetization level of 2 for arbitrary code lengths. The required finite field sizes, which are smaller than the code lengths, were determined for both constructions. The first construction offers the smallest repair bandwidth among all existing constructions with the same parameters, and is asymptotically optimal w.r.t. the lower bound derived in [35]. The second one supports two repair mechanisms, with one focusing on optimizing the repair bandwidth while the other emphasizing rebuilding access. While extending these constructions to include more parities is feasible, determining the required finite field sizes presents challenges that will be addressed in future research.

REFERENCES

- [1] B. Calder, J. Wang, A. Ogus, N. Nilakantan, A. Skjolsvold, S. McKelvie, Y. Xu, S. Srivastav, J. Wu, H. Simitci, *et al.*, “Windows azure storage: a highly available cloud storage service with strong consistency,” in *Proceedings of the Twenty-Third ACM Symposium on Operating Systems Principles*, pp. 143–157, 2011.
- [2] D. Borthakur, “The hadoop distributed file system: Architecture and design,” *Hadoop Project Website*, vol. 11, no. 2007, p. 21, 2007.
- [3] D. Borthakur, R. Schmit, R. Vadali, S. Chen, and P. Kling, “HDFS RAID,” <http://www.slideshare.net/ydn/hdfs-raid-facebook>, Nov. 2012.
- [4] A. G. Dimakis, P. B. Godfrey, Y. Wu, M. J. Wainwright, and K. Ramchandran, “Network coding for distributed storage systems,” *IEEE transactions on information theory*, vol. 56, no. 9, pp. 4539–4551, 2010.
- [5] J. Li, X. Tang, and C. Tian, “A generic transformation to enable optimal repair in MDS codes for distributed storage systems,” *IEEE Transactions on Information Theory*, vol. 64, no. 9, pp. 6257–6267, 2018.
- [6] J. Li, X. Tang, and C. Tian, “A generic transformation for optimal repair bandwidth and rebuilding access in MDS codes,” in *2017 IEEE International Symposium on Information Theory (ISIT)*, pp. 1623–1627, IEEE, 2017.
- [7] M. Ye and A. Barg, “Explicit constructions of optimal-access MDS codes with nearly optimal sub-packetization,” *IEEE Transactions on Information Theory*, vol. 63, no. 10, pp. 6307–6317, 2017.
- [8] M. Ye and A. Barg, “Explicit constructions of high-rate MDS array codes with optimal repair bandwidth,” *IEEE Transactions on Information Theory*, vol. 63, no. 4, pp. 2001–2014, 2017.
- [9] I. Tamo, Z. Wang, and J. Bruck, “Zigzag codes: MDS array codes with optimal rebuilding,” *IEEE Transactions on Information Theory*, vol. 59, no. 3, pp. 1597–1616, 2012.
- [10] K. V. Rashmi, N. B. Shah, and P. V. Kumar, “Optimal exact-regenerating codes for distributed storage at the MSR and MBR points via a product-matrix construction,” *IEEE Transactions on Information Theory*, vol. 57, no. 8, pp. 5227–5239, 2011.
- [11] D. S. Papailiopoulos, A. G. Dimakis, and V. R. Cadambe, “Repair optimal erasure codes through hadamard designs,” *IEEE Transactions on Information Theory*, vol. 59, no. 5, pp. 3021–3037, 2013.
- [12] C. Tian, “Characterizing the rate region of the (4, 3, 3) exact-repair regenerating codes,” *IEEE Journal on Selected Areas in Communications*, vol. 32, no. 5, pp. 967–975, 2014.
- [13] Y. S. Han, H.-T. Pai, R. Zheng, and P. K. Varshney, “Update-efficient error-correcting product-matrix codes,” *IEEE Transactions on Communications*, vol. 63, no. 6, pp. 1925–1938, 2015.
- [14] J. Li, X. Tang, and U. Parampalli, “A framework of constructions of minimal storage regenerating codes with the optimal access/update property,” *IEEE Transactions on Information theory*, vol. 61, no. 4, pp. 1920–1932, 2015.

- [15] B. Sasidharan, M. Vajha, and P. V. Kumar, "An explicit, coupled-layer construction of a high-rate MSR code with low sub-packetization level, small field size and all-node repair," *arXiv preprint arXiv:1607.07335*, 2016.
- [16] Z. Wang, I. Tamo, and J. Bruck, "Explicit minimum storage regenerating codes," *IEEE Transactions on Information Theory*, vol. 62, no. 8, pp. 4466–4480, 2016.
- [17] J. Li and X. Tang, "Optimal exact repair strategy for the parity nodes of the $(k+2, k)$ zigzag code," *IEEE Transactions on Information Theory*, vol. 62, no. 9, pp. 4848–4856, 2016.
- [18] S. Goparaju, A. Fazeli, and A. Vardy, "Minimum storage regenerating codes for all parameters," *IEEE Transactions on Information Theory*, vol. 63, no. 10, pp. 6318–6328, 2017.
- [19] S. Balaji, M. N. Krishnan, M. Vajha, V. Ramkumar, B. Sasidharan, and P. V. Kumar, "Erasure coding for distributed storage: An overview," *Science China Information Sciences*, vol. 61, no. 10, pp. 1–45, 2018.
- [20] Z. Chen and A. Barg, "Explicit constructions of MSR codes for clustered distributed storage: The rack-aware storage model," *IEEE Transactions on Information Theory*, vol. 66, no. 2, pp. 886–899, 2019.
- [21] H. Hou and P. P. Lee, "Binary MDS array codes with optimal repair," *IEEE Transactions on Information Theory*, vol. 66, no. 3, pp. 1405–1422, 2019.
- [22] M. Elyasi and S. Mohajer, "Cascade codes for distributed storage systems," *IEEE Transactions on Information Theory*, vol. 66, no. 12, pp. 7490–7527, 2020.
- [23] Y. Liu, J. Li, and X. Tang, "A generic transformation to generate MDS array codes with δ -optimal access property," *IEEE Transactions on Communications*, vol. 70, no. 2, pp. 759–768, 2022.
- [24] G. Li, N. Wang, S. Hu, and M. Ye, "Msr codes with linear field size and smallest sub-packetization for any number of helper nodes," *IEEE Transactions on Information Theory*, vol. 70, no. 11, pp. 7790 – 7806, 2024.
- [25] A. S. Rawat, I. Tamo, V. Guruswami, and K. Efremenko, "MDS code constructions with small sub-packetization and near-optimal repair bandwidth," *IEEE Transactions on Information Theory*, vol. 64, no. 10, pp. 6506–6525, 2018.
- [26] B. Ma, Y. Hu, D. Feng, R. Wu, and K. Zhang, "Repair I/O optimization for clay codes via gray-code based sub-chunk reorganization in ceph," 2024.
- [27] Z. Liu and Z. Zhang, "A family of access-friendly MDS array codes," in *2024 IEEE International Symposium on Information Theory (ISIT)*, pp. 2353–2358, IEEE, 2024.
- [28] S. Guan, H. Kan, and X. Wang, "A construction of $(5, 3)$ MDS codes with optimal repair capability for distributed storage systems," in *2017 9th International Conference on Wireless Communications and Signal Processing (WCSP)*, pp. 1–6, IEEE, 2017.
- [29] S. Guan, H. Kan, J. Wen, and S. Xia, "A new construction of exact-repair MSR codes using linearly dependent vectors," *IEEE Communications Letters*, vol. 21, no. 8, pp. 1691–1694, 2017.
- [30] O. Khan, R. C. Burns, J. S. Plank, W. Pierce, and C. Huang, "Rethinking erasure codes for cloud file systems: minimizing I/O for recovery and degraded reads," in *FAST*, p. 20, 2012.
- [31] T.-Y. Wu, Y. S. Han, Z. Li, B. Bai, G. Zhang, X. Zhang, and X. Wu, "Achievable lower bound on the optimal access bandwidth of $(k+2, k, 2)$ -MDS array code with degraded read friendly," in *2021 IEEE Information Theory Workshop (ITW)*, pp. 1–5, IEEE, 2021.
- [32] J. Li, Y. Liu, and X. Tang, "A systematic construction of MDS codes with small sub-packetization level and near-optimal repair bandwidth," *IEEE Transactions on Information Theory*, vol. 67, no. 4, pp. 2162–2180, 2021.
- [33] J. Li, Y. Liu, X. Tang, Y. S. Han, B. Bai, and G. Zhang, "MDS array codes with (near) optimal repair bandwidth for all admissible repair degrees," *IEEE Transactions on Communications*, vol. 71, no. 10, pp. 5633–5646, 2023.
- [34] J. Li, Y. Liu, X. Tang, Y. S. Han, B. Bai, and G. Zhang, "MDS array codes with small sub-packetization levels and small repair degrees," *arXiv preprint arXiv: 2308.14527*, 2023.
- [35] Z. Zhang, G. Li, and S. Hu, "Optimal repair of $(k+2, k, 2)$ MDS array codes," *arXiv preprint arXiv:2509.21036*, 2025.
- [36] "Guide on Red Hat Ceph Storage architecture." https://docs.redhat.com/en/documentation/red_hat_ceph_storage/2/html-single/architecture_guide/index, 2025-10-04.
- [37] G. Porter, "Decoupling storage and computation in hadoop with superdatanodes," *ACM SIGOPS Operating Systems Review*, vol. 44, no. 2, pp. 41–46, 2010.
- [38] "Hadoop cluster." <https://www.databricks.com/glossary/hadoop-cluster>, 2025-10-04.