Construction of Partial MDS and Sector-Disk Codes With Two Global Parity Symbols

Mario Blaum, *Fellow, IEEE*, James S. Plank, *Member, IEEE*, Moshe Schwartz, *Senior Member, IEEE*, and Eitan Yaakobi, *Member, IEEE*

***Abstract*— Partial MDS (PMDS) codes are erasure codes combining local (row) correction with global additional correction of entries, while sector-disk (SD) codes are erasure codes that address the mixed failure mode of current redundant arrays of independent disk (RAID) systems. It has been an open problem to construct general codes that have the PMDS and the SD properties, and previous work has relied on Monte-Carlo searches. In this paper, we present a general construction that addresses the case of any number of failed disks and in addition, two erased sectors. The construction requires a modest field size. This result generalizes previous constructions extending RAID 5 and RAID 6.**

***Index Terms*— Partial MDS codes, sector-disk codes, locally recoverable codes, maximally recoverable codes.**

I. Introduction

CONSIDER an $r \times n$ array whose entries are elements in a finite field GF(2^w) [13] (in general, we could consider a field GF(p^w), p a prime number, but for simplicity, we constrain ourselves to binary fields). The array may correspond to a stripe on a disk system, where elements co-located in the same column reside on the same disk, or the elements may correspond to disk or SSD blocks in a large storage system. Normally, these arrays are protected using the well known architectures known as Redundant Arrays of Independent Disks (RAID) [7].

Recent work has explored the loosening of the MDS property of RAID codes by defining erasure codes that combine global array protection with protection of subsets of the array (typically rows). Examples include Pyramid codes [10], LRC codes [11], [14], [16] and STAIR codes [12]. The rationale for these codes is to improve storage efficiency, encoding complexity and decoding complexity over RAID, while tolerating combinations of failures that are practical in storage systems. Please see [15] for further discussion of the practical nature of these codes.

In this paper, we concentrate on two erasure codes that also follow this trend, loosening the MDS property of RAID codes for improved performance and storage efficiency. These codes are called Partial MDS (PMDS) codes and Sector-Disk (SD) codes [2], [15]. Both follow the same methodology—m entire columns of elements are devoted to coding, and each row composes an $[n, n-m, m+1]$ MDS code. In the remaining $n-m$ columns of the array, s more elements are also devoted to coding. The erasure protection that they provide differentiates PMDS and SD codes. SD codes tolerate the erasure of any m columns of elements, plus any additional s elements in the array. PMDS codes tolerate a broader class of erasures — any m elements per row may be erased, plus any additional s elements.

As their name implies, SD codes address the combination of disk and sector failures that occurs in modern disk systems. Column failures occur when entire disks break, and sector failures can accumulate over time, typically unnoticed until an entire disk breaks, and the failed sector is required for recovery [1], [8]. PMDS codes are maximally recoverable for codes laid out in the manner described above [2]. Maximally recoverable codes have been applied to cloud storage systems where each element resides on a different storage node [11]. The rows of the array correspond to collections of storage nodes that can decode together with good performance, while the extra s elements allow the system to tolerate broader classes of failures.

We label the codes with $(m; s)$, and illustrate the difference between PMDS and SD codes in Figure 1. The figure depicts five failure scenarios in a 4×5 array, encoded with a $(1; 2)$ code, where erased elements are shaded in gray. The first four scenarios may be tolerated by both PMDS and SD codes. The first scenario is tolerated by both since each row corresponds to a $[5, 4, 2]$ MDS code. The second scenario is also tolerated by both PMDS and SD codes, because four erasures are co-located in the same column. The third and fourth scenarios are also tolerated by both PMDS and SD codes, since rows with only one erasure are corrected by a $[5, 4, 2]$ MDS code, and then we are left with three erasures in the same row, which are within the erasure-correcting capability of a $(1; 2)$ code. These two cases are important, as they are not tolerated by RAID-6, even though RAID-6 devotes two full columns to

Manuscript received May 21, 2015; revised October 16, 2015; accepted February 16, 2016. Date of publication March 1, 2016; date of current version April 19, 2016. This work was supported in part by the National Science Foundation under Grant CSR-1016636 and in part by the IBM Faculty Award. This paper was presented at the 2014 IEEE International Symposium on Information Theory.

M. Blaum is with the IBM Research Division, Almaden Research Center, San Jose, CA 95120 USA (e-mail: mmblaum@us.ibm.com).

J. S. Plank is with the Department of Electrical Engineering and Computer Science, The University of Tennessee, Knoxville, TN 37996 USA (e-mail: jplank@utk.edu).

M. Schwartz is with the Department of Electrical and Computer Engineering, Ben-Gurion University of the Negev, Beer Sheva 84105, Israel (e-mail: schwartz@ee.bgu.ac.il).

E. Yaakobi was with the Electrical Engineering Department, California Institute of Technology, Pasadena, CA 91125 USA. He is now with the Department of Computer Science, Technion–Israel Institute of Technology, Haifa 32000, Israel (e-mail: yaakobi@cs.technion.ac.il).

Communicated by A. G. Dimakis, Associate Editor for Coding Techniques.

Digital Object Identifier 10.1109/TIT.2016.2536720

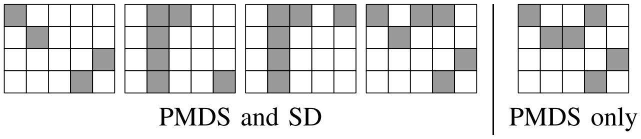

Fig. 1. Five failure scenarios on a 4 × 5 array of elements.

coding. The fifth scenario is tolerated by PMDS only, since once the rows with only one erasure are corrected, we are left with two rows with two erasures each, and none of them is in the same column.

The challenge is to construct PMDS and SD codes for general parameters. The case of $(m; 1)$ PMDS codes was solved in [2]. In this paper, we address the case of $(m; 2)$ PMDS and SD codes. We will also discuss possible methods for extending the results to general $(m; s)$ PMDS and SD codes.

As related work, let us mention [2], [9], that give constructions of $(1; s)$ PMDS codes (PMDS codes are called Maximally Recoverable codes in [9]). In [2], the construction is based on the field generated by $M_p(x) = 1 + x + \cdots + x^{p-1}$, where $p > mn$ is a prime number and 2 is primitive in $\mathrm{GF}(p)$ (which makes $M_p(x)$ irreducible). In [9], constructions of $(1; s)$ PMDS codes reducing the size of the field are presented. STAIR codes relax the failure-coverage of SD codes in order to allow for general constructions [12]. PMDS codes satisfy also the requirements of optimal Locally Repairable codes (LRC) [14], [17]. For example, consider a (1;2) PMDS code with $n > 3$, its minimum distance is 3, the same as an optimal LRC code with the same parameters. However, an optimal LRC code as described, for instance, in [17] cannot correct situations of two erasures in two different rows, as shown, for example, in the second and fifth arrays of Figure 1.

We begin with a formal definition of the two types of codes.

Definition 1: Let $\mathcal{C}$ be a linear $[rn, r(n-m)-s]$ code over a field such that when codewords are taken row-wise as $r \times n$ arrays, each row belongs in an $[n, n-m, m+1]$ MDS code. Then,

1) *$\mathcal{C}$ is an $(m; s)$ partial MDS (PMDS) code if,* for any *$(s_1, s_2, \ldots, s_t)$ such that each $s_j \geqslant 1$ and $\sum_{j=1}^{t} s_j = s$, and for any $i_1, i_2, \ldots, i_t$ such that*

$$0 \leqslant i_1 < i_2 < \cdots < i_t \leqslant r-1,$$

$\mathcal{C}$ can correct up to $s_j + m$ erasures in each row i_j, $1 \leqslant j \leqslant t$, of an array in $\mathcal{C}$.

2) *$\mathcal{C}$ is an $(m; s)$ sector-disk (SD) code if, for any $l_1, l_2, \ldots, l_m$ such that*

$$0 \leqslant l_1 < l_2 < \cdots < l_m \leqslant n-1,$$

for any $(s_1, s_2, \ldots, s_t)$ such that each $s_j \geqslant 1$ and $\sum_{j=1}^{t} s_j = s$, and for any $i_1, i_2, \ldots, i_t$ such that

$$0 \leqslant i_1 < i_2 < \cdots < i_t \leqslant r-1,$$

$\mathcal{C}$ can correct up to $s_j + m$ erasures in each row i_j, $1 \leqslant j \leqslant t$, of an array in $\mathcal{C}$ provided that locations $l_1, l_2, \ldots l_m$ in each of the rows i_j have been erased.

Constructions of $(1; 2)$ SD codes were given in [6] and of $(2; 2)$ codes in [4]. These constructions are also summarized in [15] and the construction of $(3; 2)$ SD codes was verified for all r, n in $\mathrm{GF}(2^8)$ and for $r, n \leqslant 24$ in $\mathrm{GF}(2^{16})$. Hence, our results extend those constructions. We finally note here that we can use an MDS code (like a RS code for example) over the entire array. This will work for the purpose of correcting the maximum number of erasures in the array, but it does not guarantee the first property of PMDS or SD codes, namely that each row belongs in an $[n, n-m, m+1]$ MDS code.

In Section II we give a general code construction for $r \times n$ arrays over a field of size at least rn, i.e., the total number of symbols. We prove that the construction gives $(m; 2)$ SD codes. We then show how to adapt the construction to obtain $(m; 2)$ PMDS codes over fields of size at least $r((m+1)(n-m-1)+1)$. In Section III we present codes for a more constrained model of erasures that we call disjoint-sector-disk codes. These codes cover all possible parameters m and s and require a much smaller field size. Lastly, in Section IV we give a summary of the results and some open questions.

II. Code Construction

Consider the field $\mathrm{GF}(2^w)$ and let α be an element in $\mathrm{GF}(2^w)$. The (multiplicative) order of α, denoted $\mathcal{O}(\alpha)$, is the minimum $\ell > 0$ such that $\alpha^\ell = 1$. If α is a primitive element [13], then $\mathcal{O}(\alpha) = 2^w - 1$. To each element $\alpha \in \mathrm{GF}(2^w)$, there is an associated (irreducible) minimal polynomial [13] that we denote $f_\alpha(x)$.

Let $\alpha \in \mathrm{GF}(2^w)$ and $rn \leqslant \mathcal{O}(\alpha)$. We want to construct an SD-code consisting of $r \times n$ arrays over $\mathrm{GF}(2^w)$, such that m of the columns correspond to parity (in RAID 5, $m = 1$, while in RAID 6, $m = 2$). In addition, two extra symbols also correspond to parity. When read row-wise, the codewords belong in an $[rn, r(n-m)-2]$ code over $\mathrm{GF}(2^w)$.

Construction A: For $\alpha \in \mathrm{GF}(2^w)$ with $rn \leqslant \mathcal{O}(\alpha)$, let $\mathcal{C}(r, n, m, 2; f_\alpha(x))$ be the $[rn, r(n-m)-2]$ code whose $(mr+2) \times rn$ parity-check matrix is given by

$$\mathcal{H} = \left(\begin{array}{c|c|c|c} H_0 & 0 & \ldots & 0 \\ \hline 0 & H_0 & \ldots & 0 \\ \hline \vdots & \vdots & \ddots & \vdots \\ \hline 0 & 0 & \ldots & H_0 \\ \hline H_1 & H_2 & \ldots & H_r \end{array}\right) \tag{1}$$

where

$$H_0 = \begin{pmatrix} 1 & 1 & 1 & \ldots & 1 \\ 1 & \alpha & \alpha^2 & \ldots & \alpha^{n-1} \\ 1 & \alpha^2 & \alpha^4 & \ldots & \alpha^{2(n-1)} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & \alpha^{m-1} & \alpha^{2(m-1)} & \ldots & \alpha^{(m-1)(n-1)} \end{pmatrix} \tag{2}$$

and, for $0 \leqslant j \leqslant r-1$,

$$H_{j+1} = \begin{pmatrix} 1 & \alpha^m & \alpha^{2m} & \ldots & \alpha^{m(n-1)} \\ \alpha^{-jn} & \alpha^{-jn-1} & \alpha^{-jn-2} & \ldots & \alpha^{-jn-(n-1)} \end{pmatrix}. \tag{3}$$

□

$$\mathcal{H}_1 = \left(\begin{array}{ccccc|ccccc|ccccc}
1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \hline
0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ \hline
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 \\ \hline
1 & \alpha & \alpha^2 & \alpha^3 & \alpha^4 & 1 & \alpha & \alpha^2 & \alpha^3 & \alpha^4 & 1 & \alpha & \alpha^2 & \alpha^3 & \alpha^4 \\
1 & \alpha^{14} & \alpha^{13} & \alpha^{12} & \alpha^{11} & \alpha^{10} & \alpha^9 & \alpha^8 & \alpha^7 & \alpha^6 & \alpha^5 & \alpha^4 & \alpha^3 & \alpha^2 & \alpha
\end{array}\right) \tag{4}$$

$$\mathcal{H}_2 = \left(\begin{array}{ccccc|ccccc|ccccc}
1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
1 & \alpha & \alpha^2 & \alpha^3 & \alpha^4 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \hline
0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 1 & \alpha & \alpha^2 & \alpha^3 & \alpha^4 & 0 & 0 & 0 & 0 & 0 \\ \hline
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & \alpha & \alpha^2 & \alpha^3 & \alpha^4 \\ \hline
1 & \alpha^2 & \alpha^4 & \alpha^6 & \alpha^8 & 1 & \alpha^2 & \alpha^4 & \alpha^6 & \alpha^8 & 1 & \alpha^2 & \alpha^4 & \alpha^6 & \alpha^8 \\
1 & \alpha^{14} & \alpha^{13} & \alpha^{12} & \alpha^{11} & \alpha^{10} & \alpha^9 & \alpha^8 & \alpha^7 & \alpha^6 & \alpha^5 & \alpha^4 & \alpha^3 & \alpha^2 & \alpha
\end{array}\right) \tag{5}$$

$$M(i_0, \ldots, i_m;\, j_0, \ldots, j_m;\, r;\, n;\, \ell) = \left(\begin{array}{cccc|cccc}
1 & 1 & \ldots & 1 & 0 & 0 & \ldots & 0 \\
\alpha^{i_0} & \alpha^{i_1} & \ldots & \alpha^{i_m} & 0 & 0 & \ldots & 0 \\
\alpha^{2i_0} & \alpha^{2i_1} & \ldots & \alpha^{2i_m} & 0 & 0 & \ldots & 0 \\
\vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\
\alpha^{(m-1)i_0} & \alpha^{(m-1)i_1} & \ldots & \alpha^{(m-1)i_m} & 0 & 0 & \ldots & 0 \\ \hline
0 & 0 & \ldots & 0 & 1 & 1 & \ldots & 1 \\
0 & 0 & \ldots & 0 & \alpha^{j_0} & \alpha^{j_1} & \ldots & \alpha^{j_m} \\
0 & 0 & \ldots & 0 & \alpha^{2j_0} & \alpha^{2j_1} & \ldots & \alpha^{2j_m} \\
\vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\
0 & 0 & \ldots & 0 & \alpha^{(m-1)j_0} & \alpha^{(m-1)j_1} & \ldots & \alpha^{(m-1)j_m} \\ \hline
\alpha^{mi_0} & \alpha^{mi_1} & \ldots & \alpha^{mi_m} & \alpha^{mj_0} & \alpha^{mj_1} & \ldots & \alpha^{mj_m} \\
\alpha^{-i_0} & \alpha^{-i_1} & \ldots & \alpha^{-i_m} & \alpha^{-n\ell-j_0} & \alpha^{-n\ell-j_1} & \ldots & \alpha^{-n\ell-j_m}
\end{array}\right) \tag{6}$$

We will show under which conditions the codes $\mathcal{C}(r, n, m, 2;\, f_\alpha(x))$ are SD codes. Unless stated otherwise, for simplicity, let us denote $\mathcal{C}(r, n, m, 2;\, f_\alpha(x))$ by $\mathcal{C}(r, n, m, 2)$. We start by giving some examples.

Example 2: Consider the finite field GF(16) *and let α be a primitive element, i.e., $\mathcal{O}(\alpha) = 15$. Then, the parity-check matrices $\mathcal{H}_1$ and $\mathcal{H}_2$, of $\mathcal{C}(3, 5, 1, 2)$ and $\mathcal{C}(3, 5, 2, 2)$, are given by* (4) *and* (5)*, as shown at the top of this page, respectively.* □

Let us point out that the construction of this type of codes is valid also over the ring of polynomials modulo $M_p(x) = 1 + x + \cdots + x^{p-1}$, p a prime number, as done with the Blaum-Roth (BR) codes [5]. In that case, $\mathcal{O}(\alpha) = p$, where $\alpha^{p-1} = 1 + \alpha + \cdots + \alpha^{p-2}$. The construction proceeds similarly, and we denote it $\mathcal{C}(r, n, m, 2;\, M_p(x))$. Utilizing the ring modulo $M_p(x)$ allows for XOR operations at the encoding and the decoding without look-up tables in a finite field, which is advantageous in erasure decoding [5]. It is well known that $M_p(x)$ is irreducible if and only if 2 is primitive in GF(p) [13].

Next we give a lemma that is key to proving the conditions under which codes $\mathcal{C}(r, n, m, 2)$ are PMDS or SD. Throughout the paper the notation $\oplus$ denotes the XOR operation.

Lemma 3: Let $\alpha \in$ GF(2^w)*, $rn \leqslant \mathcal{O}(\alpha)$, $1 \leqslant \ell \leqslant r - 1$, and, if $1 \leqslant m \leqslant n - 2$, let*

$$0 \leqslant i_0 < i_1 < i_2 < \cdots < i_m \leqslant n - 1$$

and

$$0 \leqslant j_0 < j_1 < j_2 < \cdots < j_m \leqslant n - 1.$$

Consider the $(2m + 2) \times (2m + 2)$ matrix $M(i_0, i_1, \ldots, i_m;\, j_0, j_1, \ldots, j_m;\, r;\, n;\, \ell)$ given by (6)*, as shown at the top of this page. Let*

$$\Delta = \det M(i_0, i_1, \ldots, i_m;\, j_0, j_1, \ldots, j_m;\, r;\, n;\, \ell).$$

Then,

$$\Delta = \left(\prod_{0 \leqslant u < v \leqslant m} \left(\alpha^{i_u} \oplus \alpha^{i_v}\right)\left(\alpha^{j_u} \oplus \alpha^{j_v}\right)\right) \cdot \left(\alpha^{-\sum_{u=0}^{m} i_u} \oplus \alpha^{-n\ell-\sum_{u=0}^{m} j_u}\right). \tag{7}$$

Proof: Since the field has characteristic 2, in the determinant expansions we don't have to worry about signs. Consider the $m \times (m + 1)$ matrices

$$M = \begin{pmatrix}
1 & 1 & \ldots & 1 \\
\alpha^{i_0} & \alpha^{i_1} & \ldots & \alpha^{i_m} \\
\alpha^{2i_0} & \alpha^{2i_1} & \ldots & \alpha^{2i_m} \\
\vdots & \vdots & \ddots & \vdots \\
\alpha^{(m-1)i_0} & \alpha^{(m-1)i_1} & \ldots & \alpha^{(m-1)i_m}
\end{pmatrix}$$

and

$$M' = \begin{pmatrix}
1 & 1 & \ldots & 1 \\
\alpha^{j_0} & \alpha^{j_1} & \ldots & \alpha^{j_m} \\
\alpha^{2j_0} & \alpha^{2j_1} & \ldots & \alpha^{2j_m} \\
\vdots & \vdots & \ddots & \vdots \\
\alpha^{(m-1)j_0} & \alpha^{(m-1)j_1} & \ldots & \alpha^{(m-1)j_m}
\end{pmatrix}.$$

For each $0 \leqslant u \leqslant m$, let M_u and M'_u denote the $m \times m$ Vandermonde matrices obtained from deleting column u from M and M' respectively. Also, for $0 \leqslant u, v \leqslant 2m+1$, $u \neq v$, let $X^{(u,v)}$ be the $(2m) \times (2m)$ matrix obtained from removing columns u and v and the last two rows from $M(i_0, i_1, \ldots, i_m; j_0, j_1, \ldots, j_m; r; n; \ell)$.

If $0 \leqslant u, v \leqslant m$, $u \neq v$,

$$X^{(u,v)} = \left(\begin{array}{c|c} P & \underline{0} \\ \hline \underline{0} & M' \end{array}\right),$$

where P denotes an $m \times (m-1)$ matrix and $\underline{0}$ are zero matrices. Notice that $X^{(u,v)}$ has rank smaller than $2m$, since the first m rows have rank smaller than m. Thus,

$$\det\left(X^{(u,v)}\right) = 0 \quad \text{for } 0 \leqslant u, v \leqslant m,\ u \neq v. \tag{8}$$

If $0 \leqslant u \leqslant m$ and $m+1 \leqslant v \leqslant 2m+1$,

$$X^{(u,v)} = \left(\begin{array}{c|c} M_u & \underline{0} \\ \hline \underline{0} & M'_{v-m-1} \end{array}\right).$$

By properties of determinants,

$$\det\left(X^{(u,v)}\right) = (\det(M_u))\left(\det(M'_{v-m-1})\right) \tag{9}$$

for $0 \leqslant u \leqslant m$, $m+1 \leqslant v \leqslant 2m+1$. Similarly,

$$\det\left(X^{(u,v)}\right) = \left(\det(M'_{u-m-1})\right)(\det(M_v)) \tag{10}$$

for $m+1 \leqslant u \leqslant 2m+1$, $0 \leqslant v \leqslant m$, and

$$\det\left(X^{(u,v)}\right) = 0, \tag{11}$$

for $m+1 \leqslant u, v \leqslant 2m+1$, $u \neq v$.

Expanding the determinant Δ from the bottom row of (6) and then from the next to bottom row, using (8), (9), (10), (11) and standard factorization, we obtain

$$\begin{aligned}
\Delta = &\left(\bigoplus_{u=0}^{m} \alpha^{-i_u} \bigoplus_{\substack{v=0 \\ v\neq u}}^{m} \alpha^{m i_v} \det\left(X^{(u,v)}\right)\right) \\
&\oplus\left(\bigoplus_{u=0}^{m} \alpha^{-i_u} \bigoplus_{v=m+1}^{2m+1} \alpha^{m j_{v-m-1}} \det\left(X^{(u,v)}\right)\right) \\
&\oplus\left(\bigoplus_{u=m+1}^{2m+1} \alpha^{-n\ell-j_{u-m-1}} \bigoplus_{v=0}^{m} \alpha^{m i_v} \det\left(X^{(u,v)}\right)\right) \\
&\oplus\left(\bigoplus_{u=m+1}^{2m+1} \alpha^{-n\ell-j_{u-m-1}} \bigoplus_{\substack{v=m+1 \\ v\neq u}}^{2m+1} \alpha^{m j_{v-m-1}} \det\left(X^{(u,v)}\right)\right) \\
= &\left(\bigoplus_{u=0}^{m} \alpha^{-i_u} \bigoplus_{v=0}^{m} \alpha^{m j_v} \det(M_u) \det(M'_v)\right) \\
&\oplus\left(\bigoplus_{u=0}^{m} \alpha^{-n\ell-j_u} \bigoplus_{v=0}^{m} \alpha^{m i_v} \det(M_v) \det(M'_u)\right) \\
= &\left(\bigoplus_{u=0}^{m} \alpha^{-i_u} \det(M_u)\right)\left(\bigoplus_{u=0}^{m} \alpha^{m j_u} \det(M'_u)\right) \\
&\oplus\left(\bigoplus_{u=0}^{m} \alpha^{-n\ell-j_u} \det(M'_u)\right)\left(\bigoplus_{u=0}^{m} \alpha^{m i_u} \det(M_u)\right).
\end{aligned} \tag{12}$$

Let

$$W_0 = \begin{pmatrix} 1 & 1 & \ldots & 1 \\ \alpha^{i_0} & \alpha^{i_1} & \ldots & \alpha^{i_m} \\ \alpha^{2i_0} & \alpha^{2i_1} & \ldots & \alpha^{2i_m} \\ \vdots & \vdots & \ddots & \vdots \\ \alpha^{(m-1)i_0} & \alpha^{(m-1)i_1} & \ldots & \alpha^{(m-1)i_m} \\ \alpha^{mi_0} & \alpha^{mi_1} & \ldots & \alpha^{mi_m} \end{pmatrix},$$

$$W_1 = \begin{pmatrix} 1 & 1 & \ldots & 1 \\ \alpha^{i_0} & \alpha^{i_1} & \ldots & \alpha^{i_m} \\ \alpha^{2i_0} & \alpha^{2i_1} & \ldots & \alpha^{2i_m} \\ \vdots & \vdots & \ddots & \vdots \\ \alpha^{(m-1)i_0} & \alpha^{(m-1)i_1} & \ldots & \alpha^{(m-1)i_m} \\ \alpha^{-i_0} & \alpha^{-i_1} & \ldots & \alpha^{-i_m} \end{pmatrix},$$

$$W'_0 = \begin{pmatrix} 1 & 1 & \ldots & 1 \\ \alpha^{j_0} & \alpha^{j_1} & \ldots & \alpha^{j_m} \\ \alpha^{2j_0} & \alpha^{2j_1} & \ldots & \alpha^{2j_m} \\ \vdots & \vdots & \ddots & \vdots \\ \alpha^{(m-1)j_0} & \alpha^{(m-1)j_1} & \ldots & \alpha^{(m-1)j_m} \\ \alpha^{mj_0} & \alpha^{mj_1} & \ldots & \alpha^{mj_m} \end{pmatrix},$$

$$W'_1 = \begin{pmatrix} 1 & 1 & \ldots & 1 \\ \alpha^{j_0} & \alpha^{j_1} & \ldots & \alpha^{j_m} \\ \alpha^{2j_0} & \alpha^{2j_1} & \ldots & \alpha^{2j_m} \\ \vdots & \vdots & \ddots & \vdots \\ \alpha^{(m-1)j_0} & \alpha^{(m-1)j_1} & \ldots & \alpha^{(m-1)j_m} \\ \alpha^{-n\ell-j_0} & \alpha^{-n\ell-j_1} & \ldots & \alpha^{-n\ell-j_m} \end{pmatrix}.$$

It is clear that W_0 and W'_0 are Vandermonde matrices. Observe also that W_1 and W'_1 become Vandermonde matrices when multiplying each column t, $0 \leqslant t \leqslant m$, by α^{i_t} and by α^{j_t} respectively (and in the case of W'_1 extracting $\alpha^{-n\ell}$ as a common factor from the last row). Then, by properties of determinants and of Vandermonde determinants,

$$\begin{aligned}
\det(W_0) &= \bigoplus_{u=0}^{m} \alpha^{m i_u} \det(M_u) \\
&= \prod_{0 \leqslant u < v \leqslant m} (\alpha^{i_u} \oplus \alpha^{i_v}), \\
\det(W_1) &= \bigoplus_{u=0}^{m} \alpha^{-i_u} \det(M_u) \\
&= \alpha^{-\sum_{u=0}^{m} i_u} \prod_{0 \leqslant u < v \leqslant m} (\alpha^{i_u} \oplus \alpha^{i_v}), \\
\det(W'_0) &= \bigoplus_{u=0}^{m} \alpha^{m j_u} \det(M'_u) \\
&= \prod_{0 \leqslant u < v \leqslant m} (\alpha^{j_u} \oplus \alpha^{j_v}), \\
\det(W'_1) &= \bigoplus_{u=0}^{m} \alpha^{-n\ell-j_u} \det(M'_u) \\
&= \alpha^{-n\ell-\sum_{u=0}^{m} j_u} \prod_{0 \leqslant u < v \leqslant m} (\alpha^{j_u} \oplus \alpha^{j_v}).
\end{aligned}$$

Thus, (12) becomes

$$\begin{aligned}\Delta &= \begin{pmatrix}\det(W_0) & \det(W_0') \\ \det(W_1) & \det(W_1')\end{pmatrix} \\ &= \left(\prod_{0\leqslant u<v\leqslant m}\left(\alpha^{i_u}\oplus\alpha^{i_v}\right)\left(\alpha^{j_u}\oplus\alpha^{j_v}\right)\right) \\ &\quad\cdot\det\begin{pmatrix}1 & 1 \\ \alpha^{-\sum_{u=0}^{m} i_u} & \alpha^{-n\ell-\sum_{u=0}^{m} j_u}\end{pmatrix}\end{aligned}$$

and (7) follows. ■

Lemma 3 is valid also over the ring of polynomials modulo $M_p(x)$, p prime, where $rn < p$. Let us illustrate it with an example for $m = 1$ and $m = 2$.

Example 4: Let $m = 1$, then

$$M(i_0, i_1; j_0, j_1; r; n; \ell) = \left(\begin{array}{cc|cc} 1 & 1 & 0 & 0 \\ \hline 0 & 0 & 1 & 1 \\ \hline \alpha^{i_0} & \alpha^{i_1} & \alpha^{j_0} & \alpha^{j_1} \\ \alpha^{-i_0} & \alpha^{-i_1} & \alpha^{-n\ell-j_0} & \alpha^{-n\ell-j_1} \end{array}\right)$$

and

$$\Delta = \left(\alpha^{i_0}\oplus\alpha^{i_1}\right)\left(\alpha^{j_0}\oplus\alpha^{j_1}\right)\left(\alpha^{-i_0-i_1}\oplus\alpha^{-n\ell-j_0-j_1}\right).$$

If $m = 2$, Lemma 3 gives

$$M(i_0, i_1, i_2; j_0, j_1, j_2; r; n; \ell) = \left(\begin{array}{ccc|ccc} 1 & 1 & 1 & 0 & 0 & 0 \\ \alpha^{i_0} & \alpha^{i_1} & \alpha^{i_2} & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & \alpha^{j_0} & \alpha^{j_1} & \alpha^{j_2} \\ \hline \alpha^{2i_0} & \alpha^{2i_1} & \alpha^{2i_2} & \alpha^{j_0} & \alpha^{j_1} & \alpha^{j_2} \\ \alpha^{-i_0} & \alpha^{-i_1} & \alpha^{-i_2} & \alpha^{-n\ell-j_0} & \alpha^{-n\ell-j_1} & \alpha^{-n\ell-j_2} \end{array}\right)$$

and

$$\begin{aligned}\Delta = &\left(\alpha^{i_0}\oplus\alpha^{i_1}\right)\left(\alpha^{i_0}\oplus\alpha^{i_2}\right)\left(\alpha^{i_1}\oplus\alpha^{i_2}\right)\left(\alpha^{j_0}\oplus\alpha^{j_1}\right) \\ &\left(\alpha^{j_0}\oplus\alpha^{j_2}\right)\left(\alpha^{j_1}\oplus\alpha^{j_2}\right)\left(\alpha^{-i_0-i_1-i_2}\oplus\alpha^{-n\ell-j_0-j_1-j_2}\right).\end{aligned}$$

□

We now state the main result for SD codes.

Theorem 5: The codes $\mathcal{C}(r, n, m, 2; f_\alpha(x))$ and $\mathcal{C}(r, n, m, 2; M_p(x))$ from Construction A are SD codes.

Proof: Assume that m columns have been erased, and in addition, we have two random erasures. Assume first that these two random erasures occurred in the same row ℓ of the stripe, where $0 \leqslant \ell \leqslant r-1$. The rows that are different from ℓ are corrected since each one of them has m erasures, which are handled by the horizontal code, that is, each horizontal code is given by the parity-check matrix H_0, which is the parity-check matrix of a RS code that can correct up to m erasures [13]. Thus, we have to solve a linear system with $m+2$ unknowns. Assume that the erasures in row ℓ occurred in locations $i_0, i_1, \ldots, i_m, i_{m+1}$, where $0 \leqslant i_0 < i_1 < \cdots < i_m < i_{m+1} \leqslant n-1$. According to the parity-check matrix of the code as given by (1), (2) and (3), there will be a unique solution if and only if the $(m+2)\times(m+2)$ matrix

$$\begin{pmatrix} 1 & 1 & \ldots & 1 & 1 \\ \alpha^{i_0} & \alpha^{i_1} & \ldots & \alpha^{i_m} & \alpha^{i_{m+1}} \\ \alpha^{2i_0} & \alpha^{2i_1} & \ldots & \alpha^{2i_m} & \alpha^{2i_{m+1}} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ \alpha^{mi_0} & \alpha^{mi_1} & \ldots & \alpha^{mi_m} & \alpha^{mi_{m+1}} \\ \alpha^{-n\ell-i_0} & \alpha^{-n\ell-i_1} & \ldots & \alpha^{-n\ell-i_m} & \alpha^{-n\ell-i_{m+1}} \end{pmatrix}$$

is invertible. By taking $\alpha^{-n\ell}$ in the last row as a common factor, and by multiplying each column j, $0 \leqslant j \leqslant m+1$, by α^{i_j}, this matrix is transformed into a Vandermonde matrix, which is always invertible in a field and also in the ring of polynomials modulo $M_p(x)$ [5].

Consider now the case in which the two random failures occur in different rows. Specifically, assume that columns $i_0, i_1, \ldots, i_{m-1}$ were erased, where $0 \leqslant i_0 < i_1 < \ldots < i_{m-1} \leqslant n-1$, and in addition, entries (ℓ, t) and (ℓ', t') were erased, where

$$t, t' \notin \{i_0, i_1, \ldots, i_{m-1}\} \quad\text{and}\quad 0 \leqslant \ell < \ell' \leqslant r-1.$$

Again, using the parity-check matrix of the code as given in (1), (2), and (3), there will be a unique solution if and only if the $(2m+2)\times(2m+2)$ matrix of (13), as shown at the bottom of this page, is invertible. Taking $\alpha^{-n\ell}$ as a common

$$\left(\begin{array}{cccc|cccc} 1 & \ldots & 1 & 1 & 0 & \ldots & 0 & 0 \\ \alpha^{i_0} & \ldots & \alpha^{i_{m-1}} & \alpha^{t} & 0 & \ldots & 0 & 0 \\ \alpha^{2i_0} & \ldots & \alpha^{2i_{m-1}} & \alpha^{2t} & 0 & \ldots & 0 & 0 \\ \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ \alpha^{(m-1)i_0} & \ldots & \alpha^{(m-1)i_{m-1}} & \alpha^{(m-1)t} & 0 & \ldots & 0 & 0 \\ \hline 0 & \ldots & 0 & 0 & 1 & \ldots & 1 & 1 \\ 0 & \ldots & 0 & 0 & \alpha^{i_0} & \ldots & \alpha^{i_{m-1}} & \alpha^{t'} \\ 0 & \ldots & 0 & 0 & \alpha^{2i_0} & \ldots & \alpha^{2i_{m-1}} & \alpha^{2t'} \\ \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & \ldots & 0 & 0 & \alpha^{(m-1)i_0} & \ldots & \alpha^{(m-1)i_{m-1}} & \alpha^{(m-1)t'} \\ \hline \alpha^{mi_0} & \ldots & \alpha^{mi_{m-1}} & \alpha^{mt} & \alpha^{mi_0} & \ldots & \alpha^{mi_{m-1}} & \alpha^{mt'} \\ \alpha^{-n\ell-i_0} & \ldots & \alpha^{-n\ell-i_{m-1}} & \alpha^{-n\ell-t} & \alpha^{-n\ell'-i_0} & \ldots & \alpha^{-n\ell'-i_{m-1}} & \alpha^{-n\ell'-t'} \end{array}\right) \tag{13}$$

factor in the last row, we obtain the matrix

$$M(i_0, i_1, i_2, \ldots, i_{m-1}, t; i_0, i_1, i_2, \ldots, i_{m-1}, t'; r; n; \ell' - \ell)$$

as defined by (6) in Lemma 3, whose determinant, by (7), is given by

$$\Delta = \left(\prod_{0 \leqslant u < v \leqslant m-1} \left(\alpha^{i_u} \oplus \alpha^{i_v} \right)^2 \right) \cdot \left(\prod_{0 \leqslant u \leqslant m-1} \left(\alpha^{i_u} \oplus \alpha^{t} \right) \left(\alpha^{i_u} \oplus \alpha^{t'} \right) \right) \cdot \alpha^{-\sum_{u=0}^{m-1} i_u} \left(\alpha^{-t} \oplus \alpha^{-n(\ell'-\ell)-t'} \right).$$

For simplicity, redefine $\ell \leftarrow \ell' - \ell$, hence, $1 \leqslant \ell \leqslant r-1$. Each binomial $(\alpha^{i_u} \oplus \alpha^{i_v})$, $(\alpha^{i_u} \oplus \alpha^{t})$, and $(\alpha^{i_u} \oplus \alpha^{t'})$, above is invertible, so it remains to be proven that $(\alpha^{-t} \oplus \alpha^{-n\ell-t'})$ is invertible. If it is not,

$$n\ell + t' - t \equiv 0 \pmod{\mathcal{O}(\alpha)}. \tag{14}$$

But

$$0 < n\ell + t' - t \leqslant n(r-1) + t' - t \leqslant n(r-1) + (n-1) = nr - 1 < \mathcal{O}(\alpha),$$

so, $n\ell + t' - t \not\equiv 0 \pmod{\mathcal{O}(\alpha)}$, contradicting (14). ∎

Next, let us prove a similar result for PMDS codes. In fact, the codes $\mathcal{C}(r, n, m, 2; f_\alpha(x))$ and $\mathcal{C}(r, n, m, 2; M_p(x))$ are not PMDS, but we will obtain PMDS codes with a modification that requires a larger field or ring. Let

$$N = (m+1)(n-m-1) + 1, \tag{15}$$

$\alpha \in \mathrm{GF}(2^w)$ and $rN \leqslant \mathcal{O}(\alpha)$. For example, if $m = 1$, $N = 2n - 3$. As in the case of SD codes, we construct a PMDS code consisting of $r \times n$ arrays over $\mathrm{GF}(2^w)$, such that m of the columns correspond to parity and in addition, two extra symbols also correspond to parity. When read row-wise, the codewords belong in an $[rn, r(n-m)-2]$ code over $\mathrm{GF}(2^w)$.

Construction B: Let $\alpha \in \mathrm{GF}(2^w)$ and $rN \leqslant \mathcal{O}(\alpha)$, where $N = (m+1)(n-m-1)+1$, and let $\mathcal{C}'(r, n, m, 2; f_\alpha(x))$ be the $[rn, r(n-m)-2]$ code whose $(mr+2) \times rn$ parity-check matrix is given by

$$\mathcal{H}' = \left(\begin{array}{c|c|c|c} H_0 & \underline{0} & \ldots & \underline{0} \\ \hline \underline{0} & H_0 & \ldots & \underline{0} \\ \hline \vdots & \vdots & \ddots & \vdots \\ \hline \underline{0} & \underline{0} & \ldots & H_0 \\ \hline H'_1 & H'_2 & \ldots & H'_r \end{array} \right) \tag{16}$$

where H_0 is given by (2) and, for $0 \leqslant j \leqslant r-1$,

$$H'_{j+1} = \begin{pmatrix} 1 & \alpha^{m} & \alpha^{2m} & \ldots & \alpha^{m(n-1)} \\ \alpha^{-jN} & \alpha^{-jN-1} & \alpha^{-jN-2} & \ldots & \alpha^{-jN-(n-1)} \end{pmatrix}. \tag{17}$$

□

As before, the construction is also valid over the ring of polynomials $M_p(x)$, p prime, in which case we denote the codes $\mathcal{C}'(r, n, m, 2; M_p(x))$. Let us give an example.

Example 6: Let $n = 5$, $m = 1$ and $r = 3$. According to (15), $N = 7$. Thus, we need $\mathcal{O}(\alpha) \geqslant rN = 21$. For instance we may consider the field GF(32) *and α primitive in* GF(32), *i.e., $\mathcal{O}(\alpha) = 31 > 21$ (we can also handle $r = 4$ in this example). Thus, the parity-check matrix of $\mathcal{C}'(3, 5, 1, 2; f_\alpha(x))$ is given by* (18), *as shown at the bottom of this page.* □

Theorem 7: The codes $\mathcal{C}'(r, n, m, 2; f_\alpha(x))$ and $\mathcal{C}'(r, n, m, 2; M_p(x))$ from Construction B are PMDS codes.

Proof: The case of $m+2$ erasures in the same row and at most m erasures in the remaining rows is handled as in Theorem 5.

Assume, without loss of generality, that row 0 has $m+1$ erasures in locations

$$0 \leqslant i_0 < i_1 < \ldots < i_m \leqslant n-1$$

and row ℓ, $1 \leqslant \ell \leqslant r-1$ has $m+1$ erasures in locations

$$0 \leqslant j_0 < j_1 < \ldots < j_m \leqslant n-1,$$

while the remaining rows have no erasures. We have to prove that the matrix

$$M(i_0, i_1, \ldots, i_m; j_0, j_1, \ldots, j_m; r; N; \ell)$$

as given by (6) is invertible, which will occur if and only if $\left(\alpha^{-\sum_{u=0}^{m} i_u} \oplus \alpha^{-N\ell - \sum_{u=0}^{m} j_u} \right)$ is invertible by Lemma 3. If it is not,

$$N\ell + \sum_{u=0}^{m} j_u - \sum_{u=0}^{m} i_u \equiv 0 \pmod{\mathcal{O}(\alpha)}. \tag{19}$$

Notice that

$$\frac{m(m+1)}{2} = \sum_{u=0}^{m} u \leqslant \sum_{u=0}^{m} i_u \tag{20}$$

and

$$\begin{aligned} \sum_{u=0}^{m} i_u &\leqslant \sum_{u=0}^{m} (n-m-1) + u \\ &= (m+1)(n-m-1) + \sum_{u=0}^{m} u \\ &= N - 1 + \frac{m(m+1)}{2}, \end{aligned} \tag{21}$$

$$\mathcal{H}' = \left(\begin{array}{ccccc|ccccc|ccccc} 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 \\ \hline 1 & \alpha & \alpha^2 & \alpha^3 & \alpha^4 & 1 & \alpha & \alpha^2 & \alpha^3 & \alpha^4 & 1 & \alpha & \alpha^2 & \alpha^3 & \alpha^4 \\ \hline 1 & \alpha^{30} & \alpha^{29} & \alpha^{28} & \alpha^{27} & \alpha^{24} & \alpha^{23} & \alpha^{22} & \alpha^{21} & \alpha^{20} & \alpha^{17} & \alpha^{16} & \alpha^{15} & \alpha^{14} & \alpha^{13} \end{array} \right) \tag{18}$$

by (15). The same is valid for $\sum_{u=0}^{m} j_u$. Combining bounds (20) and (21), we obtain

$$-(N-1) \leqslant \sum_{u=0}^{m} j_u - \sum_{u=0}^{m} i_u \leqslant N-1.$$

Therefore,

$$\begin{aligned} 1 = N-(N-1) &\leqslant N\ell + \sum_{u=0}^{m} j_u - \sum_{u=0}^{m} i_u \\ &\leqslant N(r-1) + N - 1 = Nr - 1 < \mathcal{O}(\alpha), \end{aligned}$$

so,

$$N\ell + \sum_{u=0}^{m} j_u - \sum_{u=0}^{m} i_u \not\equiv 0 \pmod{\mathcal{O}(\alpha)},$$

contradicting (19). ■

Let us point out that Lemma 3 and Theorems 5 and 7 not only prove that the codes $\mathcal{C}(r, n, m, 2;\, f_\alpha(x))$ and $\mathcal{C}'(r, n, m, 2;\, f_\alpha(x))$ are SD and PMDS respectively, but also provide for efficient encoding and decoding algorithms. In effect, solving the linear systems corresponding to erasures, for instance, using Cramer's rule, involves inverting either Vandermonde determinants or determinants of matrices

$$M(i_0, i_1, \ldots, i_{m-1}, t;\, j_0, j_1, \ldots, j_{m-1}, t';\, r;\, n;\, \ell' - \ell),$$

as given by (6) in Lemma 3. Both types of determinants involve products of binomials, which are easily inverted both in GF(q) and in the ring of polynomials modulo $M_p(x)$ [5].

III. Disjoint-Sector-Disk Codes

In this section we study a narrower case of SD codes which we call *disjoint-sector-disk (DSD) codes*. These are SD codes whose extra sector erasures reside within disjoint disks, i.e., within distinct columns of the arrays.

Definition 8: Let $\mathcal{C}$ be a linear $[rn, r(n-m)-s]$ code over a field such that when codewords are taken row-wise as $r \times n$ arrays, each row belongs in an $[n, n-m, m+1]$ MDS code. Then $\mathcal{C}$ is an $(m; s)$ disjoint-sector-disk (DSD) code if, for any $l_1, l_2, \ldots, l_m$ such that

$$0 \leqslant l_1 < l_2 < \cdots < l_m \leqslant n-1,$$

for any $(s_1, s_2, \ldots, s_t)$ such that each $s_j \geqslant 1$ and $\sum_{j=1}^{t} s_j = s$, and for any $i_1, i_2, \ldots, i_t$ such that

$$0 \leqslant i_1 < i_2 < \cdots < i_t \leqslant r-1,$$

$\mathcal{C}$ can correct up to $s_j + m$ erasures in each row i_j, $1 \leqslant j \leqslant t$, of an array in $\mathcal{C}$ provided that locations $l_1, l_2, \ldots l_m$ in each of the rows i_j have been erased, and provided the rest of the s sector erasures occur in s distinct columns.

We will construct general $(m; s)$ DSD codes for all m and s, and with a small field size, much smaller than those required by Construction A for SD codes.

The general strategy we employ is to replace the underlying Vandermonde construction, with one based on Cauchy matrices. We again assume $F = \mathrm{GF}(2^w)$ for ease of presentation only (the results carry over to general fields as well). Let $x, y \in F^n$,

$$x = x_1, x_2, \ldots, x_n \quad y = y_1, y_2, \ldots, y_n,$$

be two sequences of elements from the field, where

$$\{x_1, \ldots, x_n\} \cap \{y_1, \ldots, y_n\} = \emptyset.$$

The Cauchy matrix $\mathscr{C} = (\mathscr{C}_{i,j})$ is defined as

$$\mathscr{C}_{i,j} = \frac{1}{x_i + y_j}.$$

It is well-known that

$$\det(\mathscr{C}) = \frac{\prod_{i<j}(x_i + x_j)(y_i + y_j)}{\prod_{i,j}(x_i + y_j)}.$$

To make the notation easier, we denote by $\mathscr{C}(x, y)$ the Cauchy matrix defined by the sequences x and y, which are not necessarily of the same length, i.e., $\mathscr{C}$ is not necessarily square.

Construction C: Fix $F = \mathrm{GF}(2^w)$, and let $\overline{x} \in F^m$, $y \in F^n$, $\underline{x} \in F^s$, be three sequences of elements of F, where in total, there are $m + n + s$ distinct elements appearing in $\overline{x}$, y, and $\underline{x}$ together.

Define $\mathcal{C}''(r, n, m, s)$ to be the $[rn, r(n-m)-s]$ code whose $(mr + s) \times rn$ parity-check matrix is given by

$$\mathcal{H}'' = \left(\begin{array}{c|c|c|c} T & \underline{0} & \ldots & \underline{0} \\ \hline \underline{0} & T & \ldots & \underline{0} \\ \hline \vdots & \vdots & \ddots & \vdots \\ \hline \underline{0} & \underline{0} & \ldots & T \\ \hline B & B & \ldots & B \end{array}\right) \tag{22}$$

where

$$T = \mathscr{C}(\overline{x}, y) \quad \text{and} \quad B = \mathscr{C}(\underline{x}, y), \tag{23}$$

i.e., T is an $m \times n$ Cauchy matrix, and B is an $s \times n$ Cauchy matrix. □

Theorem 9: The code $\mathcal{C}''(r, n, m, s)$ from Construction C is a DSD code.

Proof: Consider a transmitted $r \times n$ array, with each of its rows containing m erasures, except for t rows, with indices $i_1, i_2, \ldots, i_t$, that contain $m + s_k$ erasures, $k = 1, 2, \ldots, t$, where $s_k \geqslant 1$, and $s_1 + s_2 + \cdots + s_t = s$. The rows having only m erasures may be corrected using the Cauchy-code for the row. We are therefore left only with rows i_k with $m + s_k$ erasures each, $1 \leqslant k \leqslant t$.

For additional notation, for each $k = 1, 2, \ldots, t$, let us denote the column locations of the $m + s_k$ erasures in row i_k as $j_1^{(k)}, j_2^{(k)}, \ldots, j_{m+s_k}^{(k)}$. In our setting, m entire columns are erased, and so we assume

$$j_\ell^{(1)} = j_\ell^{(2)} = \cdots = j_\ell^{(t)}, \tag{24}$$

for all $\ell = 1, 2, \ldots, m$. The remaining s erased sectors are all in distinct columns, i.e.,

$$\left| \bigcup_{k=1}^{t} \left\{ j_{m+1}^{(k)}, \ldots, j_{m+s_k}^{(k)} \right\} \right| = s.$$

We project the sequence y that defines the Cauchy matrices onto to the appropriate indices, i.e.,

$$y^{(k)} = y_{j_1^{(k)}}, y_{j_2^{(k)}}, \ldots, y_{j_{m+s_k}^{(k)}}.$$

As we did in the previous constructions, we take the columns of the parity-check matrix $\mathcal{H}''$ that correspond to the erasures, and obtain the matrix

$$M = \left(\begin{array}{c|c|c|c} T^{(1)} & \underline{0} & \dots & \underline{0} \\ \hline \underline{0} & T^{(2)} & \dots & \underline{0} \\ \hline \vdots & \vdots & \ddots & \vdots \\ \hline \underline{0} & \underline{0} & \dots & T^{(t)} \\ \hline B^{(1)} & B^{(2)} & \dots & B^{(t)} \end{array}\right),$$

where

$$T^{(k)} = \mathscr{C}(\overline{x}, y^{(k)}),$$

is an $m \times (m + s_k)$ Cauchy matrix, and

$$B^{(k)} = \mathscr{C}(\underline{x}, y^{(k)}),$$

is an $s \times (m + s_k)$ Cauchy matrix.

The erasures are correctable if and only if

$$\det(M) \neq 0.$$

We show this by showing an exact expression for $\det(M)$. For ease of presentation, we refer to the columns of M containing $T^{(k)}$ (and $B^{(k)}$) as the kth *column-block* of H. Similarly, the rows of M containing $T^{(k)}$ are referred to as the kth *row-block* of M. Finally, the bottom s rows of M are called the *bottom row-block* of H.

To calculate $\det(M)$ we perform the following procedure:

1) For $\ell = 1, 2, \dots, m$ do:
   a) For each column-block $k = 1, 2, \dots, t$, add column ℓ in the block to columns $\ell + 1, \ell + 2, \dots, m + s_k$ in the block.
   b) Collect $1/(\overline{x}_i + y_\ell^{(k)})$, $i = \ell, \ell + 1, \dots, m$, from the ith row of each of the top t row-blocks.
   c) Collect $1/(\underline{x}_i + y_\ell^{(k)})$, $i = 1, 2, \dots, s$, from the ith row of the bottom row-block.
   d) For each column-block k, collect $(y_\ell^{(k)} + y_j^{(k)})$, $j = \ell + 1, \ell + 2, \dots, m + s_k$, from the jth column of the kth column block.
   e) For each row-block $k = 1, 2, \dots, t$, add row ℓ in the block to rows $\ell + 1, \ell + 2, \dots, m$ in the block, as well as to rows $1, 2, \dots, s$ of the bottom block.
   f) For each column-block k, collect $1/(\overline{x}_\ell + y_j^{(k)})$, $j = \ell + 1, \ell + 2, \dots, m + s_k$, from the jth column of the kth column block.
   g) Collect $(\overline{x}_\ell + \overline{x}_i)$, $i = \ell, \ell + 1, \dots, m$ from the ith row of each of the top t row-blocks.
   h) Collect $(\overline{x}_\ell + \underline{x}_i)$, $i = 1, 2, \dots, s$, from the ith row of the bottom row-block.

At the end of the procedure, we move the last s_k columns of each of the column blocks to form a new rightmost column block with $s_1 + s_2 + \dots, s_t = s$ columns. We call the resulting matrix M'. Following the procedure carefully, we can verify that

$$M' = \left(\begin{array}{c|c} I_{tm} & \underline{0} \\ \hline \underline{0} & N \end{array}\right),$$

where I_{tm} is the $tm \times tm$ identity matrix, and N is an $s \times s$ Cauchy matrix, $N = \mathscr{C}(\underline{x}, \underline{y})$, with

$$\underline{y} = (y_{m+1}^{(1)}, \dots, y_{m+s_1}^{(1)}, \dots, y_{m+1}^{(t)}, \dots, y_{m+s_t}^{(t)}).$$

We also conveniently denote

$$\overline{y} = (y_1^{(k)}, y_2^{(k)}, \dots, y_m^{(k)}),$$

where the choice of k does not matter according to (24). Finally, we denote

$$\mathbf{y} = (\overline{y} \mid \underline{y}), \quad \text{and } \mathbf{x} = (\overline{x} \mid \underline{x}),$$

where $(\cdot \mid \cdot)$ denotes concatenation of vectors.

With this notation, we get

$$\begin{aligned} \det M &= \det(\mathscr{C}(\mathbf{x}, \mathbf{y})) \cdot \det(\mathscr{C}(\overline{x}, \overline{y}))^{t-1} \\ &= \frac{\prod_{i<j} (\mathbf{x}_i + \mathbf{x}_j)(\mathbf{y}_i + \mathbf{y}_j)}{\prod_{i,j} (\mathbf{x}_i + \mathbf{y}_j)} \\ &\quad \cdot \left(\frac{\prod_{i<j} (\overline{x}_i + \overline{x}_j)(\overline{y}_i + \overline{y}_j)}{\prod_{i,j} (\overline{x}_i + \overline{y}_j)} \right)^{t-1} . \end{aligned}$$

Since the elements of **x** and **y** are together all distinct, we have $\det(M) \neq 0$. ■

IV. Conclusions

We described constructions of SD and PMDS codes where the number s of additional sectors equals two. The minimal field size required by the construction for SD codes is only the total number of sectors in the array, and in the case of PMDS codes, at most of linear order on the total number of sectors.

We also presented a general construction for DSD codes using Cauchy matrices. These codes are more limited in their erasure-correction capabilities than SD codes, but the construction spans all possible parameters and requires a much smaller field size, which is linear in the number of disks and erased sectors.

The problem of constructing PMDS and SD codes for $s > 2$ is still open. An option for addressing this problem is by using existing constructions, like the ones given in [3] and in [17], to obtain suboptimal codes for a fixed value of s. For example, the construction in [3] is based upon generalized concatenated (GC) codes to correct erasure patterns which are more restricted than the ones PMDS and SD codes correct. However, the advantage of these codes is that they exist for all parameters while their field size is much smaller than the one required by PMDS and SD codes. It can be shown that in order to correct s extra erasures using these codes, the redundancy is given by $rm + D(s)$, where

$$D(s) = \sum_{h=1}^{s} \left\lfloor \frac{s}{h} \right\rfloor .$$

In particular,

$$D(s) \leqslant \sum_{h=1}^{s} \frac{s}{h} \leqslant s + \int_1^s \frac{s}{h} dh = s \ln(s) + s,$$

$$D(s) \geqslant \sum_{h=1}^{s} \left(\frac{s}{h} - 1 \right) \geqslant -s + \int_1^{s+1} \frac{s}{h} dh = s \ln(s+1) - s.$$

Assume next that we want to use the codes in [17] to correct s extra erasures, and for simplicity, the minimum distance of the code d satisfies $d < n$. Then, we can easily see that the redundancy would be $rm+2s-1$. For $s=3$, $D(3)=(2)(3)-1=5$, and both constructions have the same extra redundancy. For $s=4$ and $s=5$, the construction based on GC codes has one more extra redundancy than the construction in [17]. The difference between the two constructions gets larger as s increases. However, the construction in [3] requires a finite field of size at least n, while the construction in [17] requires a finite field of size at least mn. This also emphasizes the fact that minimizing the size of the finite field in SD and PMDS codes is another open problem in general constructions.

Acknowledgement

The authors thank three anonymous reviewers as well as the Associate Editor Prof. Alexandros Dimakis for their valuable comments and suggestions, which have contributed for the clarity of the paper and its presentation.

References

[1] L. N. Bairavasundaram, G. R. Goodson, B. Schroeder, A. C. Arpaci-Dusseau, and R. H. Arpaci-Dusseau, "An analysis of data corruption in the storage stack," in *Proc. 6th Usenix Conf. File Storage Technol. (FAST)*, San Jose, CA, USA, Feb. 2008, Art. no. 8.

[2] M. Blaum, J. L. Hafner, and S. Hetzler, "Partial-MDS codes and their application to RAID type of architectures," *IEEE Trans. Inf. Theory*, vol. 59, no. 7, pp. 4510–4519, Jul. 2013.

[3] M. Blaum and S. Hetzler. (Jul. 2014). "Generalized concatenated types of codes for erasure correction." [Online]. Available: http://arxiv.org/pdf/1406.6270.pdf.

[4] M. Blaum and J. S. Plank. (May 2013). "Construction of two SD codes." [Online]. Available: http://arxiv.org/abs/1305.1221.

[5] M. Blaum and R. M. Roth, "New array codes for multiple phased burst correction," *IEEE Trans. Inf. Theory*, vol. 39, no. 1, pp. 66–77, Jan. 1993.

[6] M. Blaum. (Apr. 2013). "Construction of PMDS and SD codes extending RAID 5." [Online]. Available: http://arxiv.org/abs/1305.0032.

[7] P. M. Chen, E. K. Lee, G. A. Gibson, R. H. Katz, and D. A. Patterson, "RAID: High-performance, reliable secondary storage," *ACM Comput. Surv.*, vol. 26, no. 2, pp. 145–185, Jun. 1994.

[8] J. G. Elerath and M. Pecht, "A highly accurate method for assessing reliability of redundant arrays of inexpensive disks (RAID)," *IEEE Trans. Comput.*, vol. 58, no. 3, pp. 289–299, Mar. 2009.

[9] P. Gopalan, C. Huang, B. Jenkins, and S. Yekhanin, "Explicit maximally recoverable codes with locality," *IEEE Trans. Inf. Theory*, vol. 60, no. 9, pp. 5245–5256, Sep. 2014.

[10] C. Huang, M. Chen, and J. Li, "Pyramid codes: Flexible schemes to trade space for access efficiency in reliable data storage systems," *ACM Trans. Storage*, vol. 9, no. 1, Mar. 2013, Art. no. 3.

[11] C. Huang, H. Simitci, Y. Xu, A. Ogus, B. Calder, P. Gopalan, J. Li, and S. Yekhanin, "Erasure coding in windows azure storage," in *Proc. USENIX Annu. Tech. Conf.*, 2012, pp. 15–26.

[12] M. Li and P. P. C. Lee, "STAIR codes: A general family of erasure codes for tolerating device and sector failures in practical storage systems," in *Proc. 12th USENIX Conf. File Storage Technol. (FAST)*, Santa Clara, CA, USA, Feb. 2014, pp. 147–162.

[13] F. J. MacWilliams and N. J. A. Sloane, *The Theory of Error-Correcting Codes*. Amsterdam, The Netherlands: North Holland, 1978.

[14] D. S. Papailiopoulos and A. G. Dimakis, "Locally repairable codes," in *Proc. IEEE Int. Symp. Inf. Theory (ISIT)*, Cambridge, MA, USA, Jul. 2012, pp. 2771–2775.

[15] J. S. Plank and M. Blaum, "Sector-disk (SD) erasure codes for mixed failure modes in RAID systems," *ACM Trans. Storage*, vol. 10, no. 1, Jan. 2014, Art. no. 4.

[16] M. Sathiamoorthy, M. Asteris, D. Papailiopoulos, A. G. Dimakis, R. Vadali, S. Chen, and D. Borthakur, "XORing elephants: Novel erasure codes for big data," in *Proc. 39th Int. Conf. Very Large Data Bases*, Aug. 2013, pp. 325–336.

[17] I. Tamo and A. Barg, "A family of optimal locally recoverable codes," *IEEE Trans. Inf. Theory*, vol. 60, no. 8, pp. 4661–4676, Aug. 2014.

Mario Blaum (S'84–M'85–SM'92–F'00) was born in Buenos Aires, Argentina, in 1951. He received the degree of Licenciado from the University of Buenos Aires in 1977, the M. Sc. degree from the Israel Institute of Technology in 1981 and the Ph. D. degree from the California Institute of Technology in 1984, all these degrees in Mathematics. From January to June, 1985, he was a Research Fellow at the Department of Electrical Engineering at Caltech. In 1985, he joined the IBM Almaden Research Center. In 2003, his division (data storage) was transferred to Hitachi Global Storage Technologies, where he continued until 2009. In 2009 he rejoined the IBM Almaden Research Center, where he is at present. Since 2001, he is an Academic Adviser at the Universidad Complutense of Madrid, Spain. From 2009 to 2012 Dr. Blaum served as Associate Editor for IEEE TRANSACTIONS ON INFORMATION THEORY. Dr. Blaum's research interests include Storage Technology, comprising all aspects of coding and synchronization. He has authored and co-authored numerous articles in the scientific literature. He also holds more than 60 US Patents. He was named an IEEE Fellow in 2000 "for Contributions to the Theory and Practice of Unidirectional and Array Codes."

James S. Plank (M'94) is a Professor in the EECS department at the University of Tennessee. He received his BS from Yale in 1988, and his PhD from Princeton in 1993. He has been at the University of Tennessee ever since. Professor Plank's research has spanned many areas of fault-tolerance, including checkpointing systems, wide-area storage systems, and erasure coding for storage systems. He is currently researching Neuromorphic computing systems. Professor Plank has been an associate editor of IEEE TRANSACTIONS ON PARALLEL AND DISTRIBUTED SYSTEMS, and has chaired conferences in storage and applications. He has left a legacy of publicly available software that includes the Unix graphing package Jgraph, the checkpointing library Libckpt, and the erasure-coding library Jerasure.

Moshe Schwartz (M'03–SM'10) is an associate professor at the Department of Electrical and Computer Engineering, Ben-Gurion University of the Negev, Israel. His research interests include algebraic coding, combinatorial structures, and digital sequences.

Prof. Schwartz received the B.A. (summa cum laude), M.Sc., and Ph.D. degrees from the Technion - Israel Institute of Technology, Haifa, Israel, in 1997, 1998, and 2004 respectively, all from the Computer Science Department. He was a Fulbright post-doctoral researcher in the Department of Electrical and Computer Engineering, University of California San Diego, and a post-doctoral researcher in the Department of Electrical Engineering, California Institute of Technology. While on sabbatical 2012-2014, he was a visiting scientist at the Massachusetts Institute of Technology (MIT).

Prof. Schwartz received the 2009 IEEE Communications Society Best Paper Award in Signal Processing and Coding for Data Storage, and the 2010 IEEE Communications Society Best Student Paper Award in Signal Processing and Coding for Data Storage.

Eitan Yaakobi (S'07–M'12) is an Assistant Professor at the Computer Science Department at the Technion–Israel Institute of Technology. He received the B.A. degrees in computer science and mathematics, and the M.Sc. degree in computer science from the Technion–Israel Institute of Technology, Haifa, Israel, in 2005 and 2007, respectively, and the Ph.D. degree in electrical engineering from the University of California, San Diego, in 2011. Between 2011-2013, he was a postdoctoral researcher in the department of Electrical Engineering at the California Institute of Technology. His research interests include information and coding theory with applications to non-volatile memories, associative memories, data storage and retrieval, and voting theory. He received the Marconi Society Young Scholar in 2009 and the Intel Ph.D. Fellowship in 2010-2011.