Title

Minimum Storage Regenerating Codes for All Parameters

Permalink

https://escholarship.org/uc/item/7hk293h1

Journal

IEEE Transactions on Information Theory, 63(10)

ISSN

0018-9448

Authors

Goparaju, Sreechakra
Fazeli, Arman
Vardy, Alexander

Publication Date

2017-10-01

DOI

10.1109/tit.2017.2690662

Peer reviewed

Minimum Storage Regenerating Codes For All Parameters

Sreechakra Goparaju **Arman Fazeli** **Alexander Vardy**
University of California San Diego, La Jolla, CA 92093, USA
Email: {sgoparaju, afazelic, avardy}@ucsd.edu

Abstract—**Regenerating codes for distributed storage have attracted much research interest in the past decade. Such codes trade the bandwidth needed to repair a failed node with the overall amount of data stored in the network. Minimum storage regenerating (MSR) codes are an important class of *optimal* regenerating codes that minimize (first) the amount of data stored per node and (then) the repair bandwidth. Specifically, an $[n,k,d]$-(α) MSR code $\mathbb{C}$ over $\mathbb{F}_q$ stores a file $\mathcal{F}$ consisting of αk symbols over $\mathbb{F}_q$ among n nodes, each storing α symbols, in such a way that:**

- **the file $\mathcal{F}$ can be recovered by downloading the content of any k of the n nodes; and**
- **the content of any failed node can be reconstructed by accessing any d of the remaining $n-1$ nodes and downloading $\alpha/(d-k+1)$ symbols from each of these nodes.**

In practice, the file $\mathcal{F}$ is typically available in uncoded form on some k of the n nodes, known as *systematic nodes*, and the defining node-repair condition above can be relaxed to requiring the optimal repair bandwidth for systematic nodes *only*. Such codes are called *systematic–repair MSR codes*.

Unfortunately, finite–α constructions of $[n,k,d]$ MSR codes are known only for certain special cases: either low rate, namely $k/n \leqslant 0.5$, or high repair connectivity, namely $d = n-1$. Our main result in this paper is a finite–α construction of systematic-repair $[n,k,d]$ MSR codes for all possible values of parameters n, k, d. We also introduce a generalized construction for $[n,k]$ MSR codes to achieve the optimal repair bandwidth for all values of d simultaneously.

Index Terms—**distributed storage systems, regenerating codes, interference alignment**

I. Introduction

Distributed storage systems form the backbone for modern cloud computing, large–scale data servers, and peer–to–peer systems. The data in these systems is stored in a redundant fashion — typically via replication (for instance, Hadoop [2] and Google file systems [3] adopt a triple replication policy) — to safeguard data against not–so–infrequently occurring disk failures. An alternative approach to storing data on these systems, which highly reduces the redundancy involved in replication, is to use maximum distance separable (MDS) codes such as Reed–Solomon codes. Though MDS codes are the most space–efficient for a targeted worst–case number of simultaneous node failures, they, unlike repetition codes, incur a high repair bandwidth[1] when the system undergoes the repair of a single node failure. A new class of erasure codes, called regenerating codes, was recently defined by Dimakis et al. [5] over a set of n nodes, which simultaneously optimizes storage efficiency, worst–case resilience and repair bandwidth for single node failures. These codes follow a trade–off curve which is intuitively evidenced by the contrast between repetition codes and MDS codes: the repair bandwidth decreases as the storage redundancy per node increases.

Formally, a file $\mathcal{F}$ of size M, is said to be stored on a DSS consisting of n nodes, each with a storage capacity of α, using an $[n,k,d]$-(α) (or, in short, $[n,k,d]$) regenerating code, if it satisfies two properties:

(a) *data recovery*: the file $\mathcal{F}$ can be recovered using the contents of any k of the n nodes (this property will also be referred to as the *MDS property*); and

(b) *repair property*: the contents of any node can be recovered using the contents of a *helper set* of any d other *helper* nodes, where each node transmits β number of symbols to the replacement node.

An *optimal* $[n,k,d]$ regenerating code achieves the optimal value of total repair bandwidth $\gamma = d\beta$ (minimum repair bandwidth) for a given storage capacity α and M. This is given implicitly by the following trade–off:

$$M \quad = \quad \sum_{i=0}^{k-1} \min\{\alpha, (d-i)\beta\}\,. \qquad (1)$$

Most of the regenerating codes research (e.g. [6]–[15]) is focussed on the extremal points of this trade–off: MBR and MSR codes. *Minimum bandwidth regenerating* (MBR) codes achieve the optimal α when the repair bandwidth equals that of a repetition code. This paper concerns *minimum storage regenerating* (MSR) codes, often dubbed as *optimal bandwidth MDS codes*, because they are optimal regenerating codes that are *also* MDS codes[2]. For these codes, $\alpha = M/k$, and the optimal repair bandwidth is given by:

$$\beta \quad = \quad \frac{\alpha}{d-k+1}. \qquad (2)$$

The main result in this paper was first presented as an invited talk at the 53rd Allerton Conference, and later appears in part in the proceedings of the IEEE International Symposium on Information Theory (ISIT) 2016 [1]. This work was supported in part by the National Science Foundation under Grant CCF-1405119.

[1] A recent work [4] revisits this for the case of Reed–Solomon codes.

[2] To be precise, these are vector MDS codes, i.e., MDS codes over $\mathbb{F}_q^\alpha$.

It is easy to see that the total repair bandwidth $d\beta$ is optimized when the number of helper nodes $d = n-1$. However, it is not always practical to connect to *all* the remaining nodes to aid the repair of a failed node. We therefore consider the following question: *Are there constructions of $[n,k,d]$ MSR codes, for $d < n-1$?*

A. Previous Work

This question had not been wholly unanswered before this paper. The first MSR code constructions appeared in [7], [16], which roughly correspond to the family of parameters $\{n,k,d\}$ with rate $k/n \leqslant 1/2$. The asymptotic existence of MSR codes for all triples $\{n,k,d\}$ was eventually shown in [11] using interference alignment techniques developed for a wireless interference channel; these codes achieve optimality as a regenerating code (as well as approach the MSR point) *only* when $\alpha \to \infty$, i.e., $\beta/\alpha \to 1/(d-k+1)$, as $M \to \infty$.

MSR codes, being MDS vector codes, can be expressed as a set of k systematic vectors and $n-k$ parity vectors (the corresponding nodes are referred to as systematic and parity nodes, respectively). For the high–rate ($k/n \geqslant 1/2$) regime, code constructions were discovered independently in [12]–[14], [17] for the specific case of $d = n-1$. Of these, the constructions in [12], [13], [17] focus on the relaxation of restricting optimal repair to *systematic nodes* in the system; we call the corresponding codes *systematic–repair MSR codes*. Practical systems usually store information in a systematic format. Parity nodes may fail, but as in the above works, we do *not* require optimal bandwidth repair for such nodes (maybe they are less urgent or critical). Clearly, any node can be repaired by reconstructing the whole file, so this covers the node repairability (even if suboptimally). Wang et al. construct an MSR code for $d = n-1$ in [14] that achieves the optimal repair bandwidth also for parity nodes, albeit at the cost of some other metrics such as the number of symbols read from a node (*access-optimality*) and the complexity of updating parities when systematic data changes (*update-optimality*). Agarwal et al. in [18], and Raviv et al. in [19] have also presented alternative constructions of access-optimal MSR codes for $d = n-1$ while achieving the corresponding optimal sub-packetization.

B. Contribution & Outline

We present the first[3] high–rate finite–α constructions for systematic–repair MSR codes for $d < n-1$. We start by describing in Section II the representative code construction that contains the ideas behind those in [12], [13], [17]. Leveraging on this, we present our construction in Section III, but restrict to the case when the helper nodes contain the remaining $k-1$ systematic nodes. This restriction is removed in Section IV, thus rounding out the code construction. In Section V, we further generalize the construction to systematic–repair $[n,k]$ MSR codes that universally achieve the optimal repair bandwidth for any given d helper nodes ($k \leqslant d \leqslant n-1$). We conclude with some remarks in Section VI.

C. Subsequent Work

Since this work was first presented, a flurry of recent results have essentially completely solved the problem of constructing $[n,k,d]$ MSR codes. Rawat et al. [21] construct MSR codes which optimally repair all nodes. However, the flavor of their construction, which is not systematic in nature, differs from ours. Most recently, Ye and Barg [22], [23] show that $[n,k,d]$ MSR codes can be explicitly constructed[4] over a small finite field and with a near optimal sub-packetization α. Birenjith et al. [24] also construct explicit $[n,k,d = n-1]$ MSR codes with these properties. An interesting and related direction has been covered in [25] and [26] where MDS codes have been constructed that have a significantly reduced sub-packetization α at the expense of achieving only a near-optimal (and *not* the optimal) repair bandwidth.

II. Primer: Code Construction for $d = n-1$

Let $n = k+r$ denote the number of nodes in the distributed storage system, where each node has the capacity to store a vector of size α over $\mathbb{F}_q$. Throughout this paper, we discuss systematic constructions and assume that the first k nodes are systematic nodes and store raw information, while the remaining r nodes correspond to the parities. We use the notation $\mathbf{x}_i$, $i \in [k]$, for the raw systematic vectors stored in the systematic nodes. The parity nodes are defined by

$$\mathbf{x}_{k+i} = \sum_{j=1}^{k} A_{ij}\mathbf{x}_j, \qquad i \in [r], \tag{3}$$

where A_{ij}'s are $\alpha \times \alpha$ encoding matrices. The generator matrix of the code is then given by

$$G = \begin{bmatrix} I & & 0 \\ & \ddots & \\ 0 & & I \\ A_{1,1} & \cdots & A_{1,k} \\ \vdots & \ddots & \vdots \\ A_{r,1} & \cdots & A_{r,k} \end{bmatrix}. \tag{4}$$

In this section, we consider MSR codes where $d = n-1$. In other words, when a single node failure occurs, all the remaining nodes aid in its repair. We also restrict our attention to codes that consider failures only of the systematic nodes, and discuss in this section, a construction that underlies the ideas in [13], [17] and [12]. Note

[3] This work was first presented as an invited talk at the 53rd Annual Allerton Conference on Communication, Control, and Computing. A simultaneous result was presented at the same venue by Tamo and En Gad [20].

[4] Explicitness implies that the generator or the parity check matrix of the code can be explicitly specified, not merely proved to exist.

that the construction is optimal neither in terms of subpacketization for access-optimal MSR codes nor in terms of field size. However, its simplicity will inform our generalization for the general parameter triple $\{n,k,d\}$ in Section III.

A commonly adopted strategy in constructing an MSR code is to first guarantee the optimal repair bandwidth property for a single failure (in this case, for a single systematic node failure), and then transform the construction to ensure the MDS property. This is illustrated in Example 1 below.

Example 1. Assume $(n,k,d) = (4,2,3)$ and $\alpha = 4$. Let the first two nodes $\mathbf{x}_1$ and $\mathbf{x}_2$ be the systematic nodes, and let the parity nodes $\mathbf{x}_3$ and $\mathbf{x}_4$ be defined as

$$\mathbf{x}_3 = \underbrace{\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}}_{I} \mathbf{x}_1 + \underbrace{\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}}_{I} \mathbf{x}_2,$$

$$\mathbf{x}_4 = \underbrace{\begin{pmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix}}_{P_1} \mathbf{x}_1 + \underbrace{\begin{pmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix}}_{P_2} \mathbf{x}_2.$$

Fig. 1 (a) depicts the component-wise storage in each node. It can be observed that a single failure in either $\mathbf{x}_1$ or $\mathbf{x}_2$ can be reconstructed by downloading $\alpha/2 = 2$ elements from each of the remaining $d = 3$ nodes. However, the data is not recoverable if both $\mathbf{x}_1$ and $\mathbf{x}_2$ fail and hence, the code is not MDS. To overcome this problem, we associate a coefficient λ with P_2 such that $\begin{pmatrix} I & I \\ P_1 & \lambda P_2 \end{pmatrix}$, which is the sub-matrix of the generator matrix G corresponding to the nodes $\mathbf{x}_3$ and $\mathbf{x}_4$, is non-singular. Note that,

$$\begin{vmatrix} I & I \\ P_1 & \lambda P_2 \end{vmatrix} = \det(\lambda P_2 - P_1) = \begin{vmatrix} 0 & \lambda & -1 & 0 \\ \lambda & 0 & 0 & -1 \\ -1 & 0 & 0 & \lambda \\ 0 & -1 & \lambda & 0 \end{vmatrix} = (\lambda^2 - 1)^2,$$

which is non-zero[5] if the field size $q = 5$ and $\lambda = 2$. Fig. 1 (b) shows the component-wise storage for the resulting MSR code.

Construction 1 generalizes the construction given in Example 1 for an $[n,k,n-1]$ MSR code. Note that any MSR code construction must specify both the generator matrix of the code as well as the optimal bandwidth repair strategy that is implemented on the code.

C_1	C_2	C_3	C_4
$x_{1,1}$	$x_{2,1}$	$x_{1,1}+x_{2,1}$	$x_{1,3}+x_{2,2}$
$x_{1,2}$	$x_{2,2}$	$x_{1,2}+x_{2,2}$	$x_{1,4}+x_{2,1}$
$x_{1,3}$	$x_{2,3}$	$x_{1,3}+x_{2,3}$	$x_{1,1}+x_{2,4}$
$x_{1,4}$	$x_{2,4}$	$x_{1,4}+x_{2,4}$	$x_{1,2}+x_{2,3}$

(a)

C_1	C_2	C_3	C_4
$x_{1,1}$	$x_{2,1}$	$x_{1,1}+x_{2,1}$	$x_{1,3}+2x_{2,2}$
$x_{1,2}$	$x_{2,2}$	$x_{1,2}+x_{2,2}$	$x_{1,4}+2x_{2,1}$
$x_{1,3}$	$x_{2,3}$	$x_{1,3}+x_{2,3}$	$x_{1,1}+2x_{2,4}$
$x_{1,4}$	$x_{2,4}$	$x_{1,4}+x_{2,4}$	$x_{1,2}+2x_{2,3}$

(b)

Fig. 1. (a) Component wise storage in a $[4,2,3]$ binary array code with optimal repair bandwidth for a single systematic node failure, described by $(\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_1 + \mathbf{x}_2, P_1\mathbf{x}_1 + P_2\mathbf{x}_2)$; (b) A $(4,2,3)$ MSR code in $\mathbb{F}_5$ described by $(\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_1 + \mathbf{x}_2, P_1\mathbf{x}_1 + 2P_2\mathbf{x}_2)$. In both cases, gray cells are accessed to rebuild C_1.

Construction 1. Let $\alpha = r^k$ and label the α elements $[0 : r^k - 1]$ by r-ary vectors in $\mathbb{Z}_r^k$. Define permutation f_j^ℓ on $[0 : r^k - 1]$ as follows:

$$\begin{array}{rccc} f_j^\ell : & \mathbb{Z}_r^k & \rightarrow & \mathbb{Z}_r^k \\ & v & \mapsto & v + \ell e_j, \end{array}$$

for $j \in [k]$ and $\ell \in [0 : r-1] := \{0, 1, \ldots, r-1\}$, where $\{e_1, e_2, \ldots, e_k\}$ is the standard vector basis for $\mathbb{Z}_r^k$. The mapping f_j^ℓ is bijective, and therefore, corresponds to a permutation on $[0 : r^k - 1]$. Let $P_{\ell,j}$ be the $\alpha \times \alpha$ matrix corresponding to the permutation f_j^ℓ, that is, $P_{\ell,j}\,\mathbf{x} = \mathbf{y}$, where $\mathbf{x}$, $\mathbf{y} \in \mathbb{F}_q^\alpha$, and $\mathbf{x}(v) = \mathbf{y}(f_j^\ell(v))$. Here, $\mathbf{x}(v)$ corresponds to the v^{th} coordinate of vector $\mathbf{x}$, where v is the k-ary representation of a number in $[0 : \alpha - 1]$. In other words, $P_{\ell,j}$ scrambles the elements of a vector according to the permutation f_j^ℓ. (Notice that $P_{0,j} = I_\alpha$.)

1) *MSR Code*: The generator matrix of the code is given by (4), where $A_{i,j} = \lambda_{i,j} P_{i-1,j}$, $i \in [r]$ and $j \in [k]$. The non-zero coefficients $\lambda_{i,j} \in \mathbb{F}_q$ will be defined in Section II-B to ensure the MDS property.
2) *Repair Strategy*: Let $Y_j = \{v \in [0, r^k - 1] : v \cdot e_j = 0\}$ denote a subset of $[0 : r^k - 1]$. Y_j can be interpreted as those elements in $[0 : \alpha - 1]$ whose label representation in $\mathbb{Z}_r^k$ have a 0 in their j^{th} coordinate. If systematic node j fails, it is repaired by accessing the elements corresponding to Y_j from each of the remaining nodes, i.e., by accessing $\mathbf{x}_i(v)$, where $v \in Y_j$ and $j \neq i \in [n]$.

Construction 1 is obtained by first constructing an $[n,k]$ array code[6] (Section II-A) which guarantees the optimal bandwidth repair for a single systematic node failure. The array code is then transformed (Section II-B) to an MDS array code (and thereby, a systematic–repair MSR code) by transforming the encoding matrices of the parity nodes, while retaining the repair property.

[5] In general for $\alpha \times \alpha$ matrices A, B, C, and D, if D is nonsingular, then the calculation of its Schur complement yields in $\det\left(\begin{bmatrix} A & B \\ C & \lambda D \end{bmatrix}\right) = \det(D)\det(\lambda A - BD^{-1}C)$, which is a polynomial of degree at most α in λ. Therefore, if the field size is large enough, *i.e.* $q > \alpha$, one can always find a value for λ so that the 2×2 block matrix becomes non-singular as well. The same approach can be used to prove Lemma 5.

[6] By an $[n,k]$ array code, we mean a set of k systematic vectors, and $n-k$ parity vectors defined according to (3), which may or may not satisfy any properties.

A. Repair Property: Interference Alignment

The optimal repair bandwidth property of an $[n,k,n-1]$ MSR code can be viewed as a *signal interference* problem: the objective is to retrieve the desired signal — the contents of the failed systematic node, say, $\mathbf{x}_i$ — which, in the repair data downloaded from the remaining nodes, is interfered by partial contents of the remaining systematic nodes, $\mathbf{x}_j$, where $i \neq j \in [n]$. The solution, turns out to be an *interference alignment* strategy, where the repair data associated with the interfering systematic data is aligned, so as to minimize the interference. This is crystallized in the following lemma[7].

Lemma 1. *Let $\mathbf{x}_i$, $i \in [k]$, be the failed systematic node. For an $[n,k,n-1]$ MSR code, the set of $d = n-1$ helper nodes is given by $\mathcal{D} = \{\mathbf{x}_j \mid j \in [n]\backslash\{i\}\}$. To recover the contents of the failed systematic node with the optimal repair bandwidth, it is necessary and sufficient to find $n-1$ (repair) matrices denoted by $\{S^i_j \in \mathbb{F}_q^{\alpha/r\times\alpha} \mid j \in [n]\backslash\{i\}\}$, where $r = n-k$, such that, for $j \in [k], j \neq i$, the following two conditions are satisfied:*
(a) signal recovery:

$$\mathsf{rank}\left(\left(\begin{array}{c} S^i_{k+1}A_{1,i} \\ S^i_{k+2}A_{2,i} \\ \vdots \\ S^i_{k+r}A_{r,i} \end{array}\right)\right) \quad = \quad \alpha, \tag{5}$$

(b) interference alignment:

$$\mathsf{rank}\left(\left(\begin{array}{c} S^i_j \\ S^i_{k+1}A_{1,j} \\ \vdots \\ S^i_{k+r}A_{r,j} \end{array}\right)\right) \quad = \quad \frac{\alpha}{r}. \tag{6}$$

Stated otherwise, to optimally repair $\mathbf{x}_i$, it is necessary and sufficient to find $n-1$ (repair) subspaces of dimension α/r, denoted[8] by $\{S^i_j \mid j \in [n]\backslash\{i\}\}$, where $r = n-k$, such that, for $j \in [k], j \neq i$, the following two conditions are satisfied:
(a) signal recovery:

$$S^i_{k+1}A_{1,i} \oplus \cdots \oplus S^i_{k+r}A_{r,i} \quad \backsimeq \quad \mathbb{F}_q^{\alpha}, \tag{7}$$

(b) interference alignment:

$$S^i_j \quad \backsimeq \quad S^i_{k+s}A_{s,j},\ \forall\, s \in [r], \tag{8}$$

where $\backsimeq$ denotes equality of subspaces, SA is the subspace obtained by operating the subspace S by the matrix A, and $\oplus$ denotes the subspace sum.

For completeness, we provide a proof for Lemma 1 in Appendix A. Lemma 2 generalizes Lemma 1 when the number of helper nodes $d < n-1$. This will be used later in Section III.

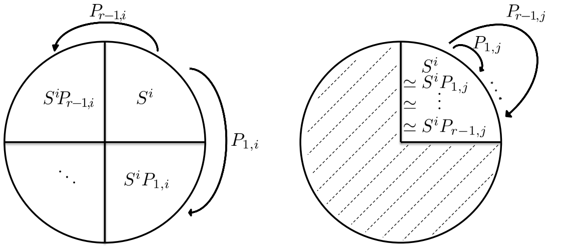

Fig. 2. Visualization of Lemma 1(a) (left), and Lemma 1(b) (right) to justify repair optimality in Construction 1.

Lemma 2. (Corollary of Lemma 1.) *In general, for an $[n,k,d]$ MSR code, if the set of $d = (k-1)+t < n-1$ helper nodes is given by $\mathcal{D} = \{\mathbf{x}_j \mid j \in \mathcal{J} = [k]\backslash\{i\} \cup \{b_1, \cdots, b_t\}\}$ (where $b_i \in \{k+1, \ldots, n\}$ denote the t parity nodes in the helper set), it is necessary and sufficient to find d (repair) subspaces of dimension α/t denoted by $\{S^i_j \mid j \in \mathcal{J}\}$, such that, for $j \in [k], j \neq i$, the following two conditions are satisfied:*

$$S^i_{b_1}A_{b_1-k,i} \oplus \cdots \oplus S^i_{b_t}A_{b_t-k,i} \quad \backsimeq \quad \mathbb{F}_q^{\alpha}, \tag{9}$$

$$S^i_j \quad \backsimeq \quad S^i_{b_s}A_{b_s-k,j},\ \forall\, s \in [t]. \tag{10}$$

The optimal repair property of Construction 1 can now be justified.

Lemma 3. *The repair strategy in Construction 1 is optimal with respect to repair bandwidth.*

Proof: Define $S^i_j \triangleq S^i \triangleq Y_i, j \neq i$. Notice that the rank of subspace S^i is $r^{k-1} = \alpha/r$. Per definition, the permutation $P_{\ell,i}$ maps Y_i to $Y_i + \ell e_i = \{v \in [0, r^k - 1] : v \cdot e_i = \ell\}$. This implies that for any distinct $\ell, \ell' \in [0 : r-1]$, the intersection $S^iP_{\ell,i} \cap S^iP_{\ell',i}$ contains only the all-zero vector. Thus the subspaces: $S^i, S^iP_{1,i}, \ldots, S^iP_{r-1,i}$, span the space $\mathbb{F}_q^{\alpha}$ $(\alpha = r^k)$ and the signal recovery condition(s) in Lemma 1 are satisfied; see Fig. 2. Furthermore, applying a permutation $P_{\ell,j}$ corresponding to a different coordinate $j \neq i$ maps Y_i to itself. This validates the interference alignment condition(s) in Lemma 1. Finally, note that the two conditions continue to be satisfied when replacing the permutations $P_{i-1,j}$ with any scaled versions $A_{i,j} = \lambda_{i,j}P_{i-1,j}$, because the scaling of the basis vectors does not change the relevant subspaces and thereby does not affect the conditions in Lemma 1. ■

B. MDS Property

This second step relies on the following two lemmas, the proofs of which are left to the reader.

Lemma 4. *Let B denote the parity part of the generator matrix for an $[n,k]$ array code denoted by $\mathbb{C}$, where*

$$B = \left[\begin{array}{ccc} B_{1,1} & \cdots & B_{1,k} \\ \vdots & \ddots & \vdots \\ B_{r,1} & \cdots & B_{r,k} \end{array}\right].$$

[7] This result is known and has been used in several papers on MSR codes, but we state and prove it for completeness.

[8] Whenever this lemma is referenced, we use the subspace and matrix notation interchangeably as some proofs or expressions are clearer in one of the formats. We accordingly overload the notation S^i_j to refer to both the matrix and the subspace spanned by the row vectors of the matrix.

Given that $B_{i,j}$ is non-singular for all i, j, $\mathbb{C}$ is an MDS array code if and only if any square sub-block-matrix B' of B is also non-singular, where

$$B' = \begin{bmatrix} B_{i_1,j_1} & \cdots & B_{i_1,j_t} \\ \vdots & \ddots & \vdots \\ B_{i_t,j_1} & \cdots & B_{i_t,j_t} \end{bmatrix},$$

for some $\{i_1, \cdots, i_t\} \subset [r], \{j_1, \cdots, j_t\} \subset [k]$.

Lemma 5. *Let B denote the $r\alpha \times k\alpha$ matrix associated with the parity part of the generator matrix for an $[n, k]$ array code, as defined in Lemma 4. Given that $B_{i,j}$ is non-singular for all $i \in [r]$, $j \in [k]$, and the field size q is large enough, there exist coefficients $\lambda_{i,j} \in \mathbb{F}_q$, such that all square sub-block-matrices of A are non-singular, where*

$$A = \begin{bmatrix} \lambda_{1,1}B_{1,1} & \cdots & \lambda_{1,k}B_{1,k} \\ \vdots & \ddots & \vdots \\ \lambda_{r,1}B_{r,1} & \cdots & \lambda_{r,k}B_{r,k} \end{bmatrix}.$$

In other words, any parity generator matrix B for an $[n, k]$ array code with non-singular encoding matrices can be transformed into a parity generator matrix A for an $[n, k]$ MDS array code by multiplying the encoding matrices with appropriate scalar coefficients.

Proof Sketch: To obtain a valid set of $\lambda_{i,j}$'s, one may first sort the pairs (i, j) with respect to $i + j$ increasingly, and then recursively choose a value for each $\lambda_{i,j}$ such that all sub-block-matrices with $\lambda_{i,j}A_{i,j}$ on their bottom right corner become non-singular. It suffices to have the field size q greater than the number of such sub-block-matrices at any step multiplied by α;

$$|\mathbb{F}| > q_{\text{MDS}} = \alpha \max_t \left\{ \binom{n-k-1}{t} \times \binom{k-1}{t} \middle| t \in [k] \right\}. \tag{11}$$

■

Remark: Notice that this construction is slightly different than that described in [17], where the authors choose different permutation matrices with a potentially smaller value of m in $\alpha = r^m$. This is achieved by choosing a different set of vectors than $\{e_i, i \in [k]\}$ when defining the permutations in Construction 1. However, we forego the analysis of reducing m for the sake of simplicity and readability, and take the current construction as a starting point for the generalized construction below.

III. Code Construction for Restricted Helper Set

We now move to the construction of $[n, k, d]$ systematic–repair MSR codes for any n, k, and d, where $k + 1 \leqslant d \leqslant n - 1$. In this section, we start with the restricted case when the helper set $\mathcal{D}$ includes all remaining $k - 1$ systematic nodes. Let us begin with an example.

Example 2. Let us look at the case when $[n, k, d] = [k + 3, k, k + 1]$ for $k \in \mathbb{N}$. Given a failure at the systematic node i, we are interested in repairing it by downloading $\frac{\alpha}{d-k+1} = \frac{\alpha}{2}$ symbols from each node in the helper set $\mathcal{D}_i$. Let us assume that $\mathcal{D}_i$ includes all of the remaining $k - 1$ systematic nodes. Hence, there are $\binom{3}{2} = 3$ different ways to choose $\mathcal{D}_i$ depending on which two parity nodes are included in it. Let us use an indicator $a \in [3]$ to differentiate between these scenarios, and denote the helper set for each scenario by $\mathcal{D}_{i,a}$.

Construction. Let $\alpha = 2^{3k}$ and label the α elements $[0 : 2^{3k} - 1]$ by binary vectors in $\mathbb{Z}_2^{3k}$. Define permutation f_j^ℓ on $[0 : 2^{3k} - 1]$ as follows:

$$\begin{array}{rccc} f_j^\ell : & \mathbb{Z}_2^{3k} & \to & \mathbb{Z}_2^{3k} \\ & v & \mapsto & v + \ell e_j, \end{array}$$

for $j \in [3k]$ and $\ell \in \{0, 1\}$, where $\{e_1, e_2, \cdots, e_{3k}\}$ is the standard vector basis for $\mathbb{Z}_2^{3k}$. The mapping f_j^ℓ is again bijective and therefore corresponds to a permutation on $[0 : 2^{3k} - 1]$. As before, let $P_{\ell,j}$ be the $\alpha \times \alpha$ matrix corresponding to the permutation f_j^ℓ, that is, $P_{\ell,j}\mathbf{x} = \mathbf{y}$, where $\mathbf{x}, \mathbf{y} \in \mathbb{F}_q^\alpha$, and $\mathbf{x}(v) = \mathbf{y}(f_j^\ell(v))$. As before, $\mathbf{x}(v)$ is the v^{th} coordinate of $\mathbf{x} \in \mathbb{F}_q^\alpha$, where v is the k-ary vector representation of the coordinate, and $P_{0,j} = I_\alpha$.)

1) *MSR Code*: The generator matrix of the code is given by

$$G = \begin{bmatrix} I & & 0 \\ & \ddots & \\ 0 & & I \\ A_{1,1} & \cdots & A_{1,k} \\ A_{2,1} & \cdots & A_{2,k} \\ A_{3,1} & \cdots & A_{3,k} \end{bmatrix},$$

where

$$\begin{aligned} A_{1,j} &= \lambda_{1,j} \; P_{0,3j-2} \; P_{0,3j-1}, \\ A_{2,j} &= \lambda_{2,j} \; P_{1,3j-2} \; P_{0,3j}, \\ A_{3,j} &= \lambda_{3,j} \; P_{1,3j-1} \; P_{1,3j}, \end{aligned} \tag{12}$$

for $j \in [k]$. The non-zero coefficients $\lambda_{i,j} \in \mathbb{F}_q$ are again selected according to the discussion in Section II-B to establish the MDS property.

2) *Repair Strategy via $\mathcal{D}_{i,1} = \{\mathbf{x}_j | j \in [k+3], j \neq i, k+3\}$*: Let $Y_{i,1} = \{v \in [0 : 2^{3k} - 1] : v \cdot e_{3i-2} = 0\}$ denote a subset of $[0 : 2^{3k} - 1]$. $Y_{i,1}$ can be interpreted as those elements in $[0 : 2^{3k} - 1]$ whose label representation in $\mathbb{Z}_2^{3k}$ has a 0 in its $(3i-2)^{\text{th}}$ coordinate. If systematic node i fails, it can be repaired by accessing the elements corresponding to $Y_{i,1}$ from each of the helper nodes, i.e., by accessing $\mathbf{x}_j(v)$, where $v \in Y_{i,1}$ and $j \in \{1, 2, \cdots, i-1, i+1, \cdots, k, k+1, k+2\}$.

3) *Repair Strategy via $\mathcal{D}_{i,2} = \{\mathbf{x}_j | j \in [k+3], j \neq i, k+2\}$*: Similarly, let $Y_{i,2} = \{v \in [0 : 2^{3k} - 1] : v \cdot e_{3i-1} = 0\}$. If systematic node i fails, it can be repaired by accessing $\mathbf{x}_j(v)$, where $v \in Y_{i,2}$ and $j \in \{1, 2, \cdots, i-1, i+1, \cdots, k, k+1, k+3\}$.

4) *Repair Strategy via $\mathcal{D}_{i,3} = \{\mathbf{x}_j | j \in [k+3], j \neq i, k+1\}$*: Finally, let $Y_{i,3} = \{v \in [0 : 2^{3k} - 1] : v \cdot e_{3i} = 0\}$ denote the location of the elements that are accessed if systematic node i fails, i.e., node i can

be repaired by accessing $\mathbf{x}_j(v)$, where $v \in Y_{i,3}$ and $j \in \{1,2,\cdots,i-1,i+1,\cdots,k,k+2,k+3\}$.

Justification of the repair strategy: Let $\mathbf{x}_i$, $i \in [k]$, be the failed systematic node. Define $Q_{u,v} = A_{u,v}\lambda_{u,v}^{-1}$, $u \in [3], v \in [k]$; this is a product of multiple permutation matrices, and hence can be viewed as a permutation matrix itself. In order to justify the repair strategy, it suffices to define the proper subspaces S_j^i that fulfill the two interference alignment conditions in Lemma 2. Let $U_{i,a}$ be the complimentary subset of $Y_{i,a}$ in $\mathbb{Z}_2^{3k}$, i.e.,

$$\begin{aligned} U_{i,1} &= \{v \in [0 : 2^{3k}-1] : v \cdot e_{3i-2} = 1\},\\ U_{i,2} &= \{v \in [0 : 2^{3k}-1] : v \cdot e_{3i-1} = 1\},\\ U_{i,3} &= \{v \in [0 : 2^{3k}-1] : v \cdot e_{3i} = 1\}. \end{aligned}$$

Given the code construction in (12), we can verify that

$$\begin{array}{lll} Y_{i,1}Q_{1,i} = Y_{i,1}, & Y_{i,2}Q_{1,i} = Y_{i,2}, & Y_{i,3}Q_{1,i} = Y_{i,3},\\ U_{i,1}Q_{1,i} = U_{i,1}, & U_{i,2}Q_{1,i} = U_{i,2}, & U_{i,3}Q_{1,i} = U_{i,3},\\ Y_{i,1}Q_{2,i} = U_{i,1}, & Y_{i,2}Q_{2,i} = Y_{i,2}, & Y_{i,3}Q_{2,i} = Y_{i,3},\\ U_{i,1}Q_{2,i} = Y_{i,1}, & U_{i,2}Q_{2,i} = U_{i,2}, & U_{i,3}Q_{2,i} = U_{i,3},\\ Y_{i,1}Q_{3,i} = Y_{i,1}, & Y_{i,2}Q_{3,i} = U_{i,2}, & Y_{i,3}Q_{3,i} = U_{i,3},\\ U_{i,1}Q_{3,i} = U_{i,1}, & U_{i,2}Q_{3,i} = Y_{i,2}, & U_{i,3}Q_{3,i} = Y_{i,3},\\ \text{and,} & & \end{array}$$

$$Y_{i',u}Q_{i,j} = Y_{i',u}\ ,\ \ U_{i',u}Q_{i,j} = U_{i',u}\ , \qquad \text{for } i' \neq i. \quad (13)$$

Now we define subspaces $S_{j,a}^i \triangleq S_a^i \triangleq Y_{i,a}, j \neq i, a \in [3]$. Let us for simplicity assume $a = 1$. The other scenarios can be justified similarly. Based on (13), we observe that the permutation $Q_{2,i}$ maps the basis $Y_{i,1}$ to its complementary subset $U_{i,1}$ and vice versa, while $Q_{1,i}$ preserves both of them. Hence,

$$\mathsf{rank}\left(\begin{pmatrix} S_1^iQ_{1,i}\\ S_1^iQ_{2,i}\end{pmatrix}\right) = \mathsf{rank}\left(\begin{pmatrix} Y_{i,1}\\ U_{i,1}\end{pmatrix}\right) = \alpha.$$

Furthermore, $Y_{i,1}$ remains unchanged under any other permutation $Q_{t,j}, j \neq i$, and hence

$$\mathsf{rank}\left(\begin{pmatrix} S_1^i\\ S_1^iQ_{1,i'}\\ S_1^iQ_{2,i'}\end{pmatrix}\right) = \mathsf{rank}\left(\begin{pmatrix} Y_{i,1}\\ Y_{i,1}\\ Y_{i,1}\end{pmatrix}\right) = \frac{\alpha}{2}.$$

■

The key element in the construction is to satisfy the two requirements in Lemma 2 for any systematic failure and any such helper set $\mathcal{D}$. Let $\rho = d-k+1$ denote the number of parity nodes in the helper set of size d. There are $\binom{r}{\rho}$ different ways to choose ρ parity nodes during the repair. Let us label these cases with numbers $a \in [\binom{r}{\rho}]$, and set $\mathcal{R}_a$ to be the subset of parity nodes corresponding to case a.

Assume that $\mathcal{R}_a = \{\mathbf{x}_{k+d_1^{(a)}}, \mathbf{x}_{k+d_2^{(a)}}, \cdots, \mathbf{x}_{k+d_\rho^{(a)}}\}$, where $\{d_1^{(a)}, \cdots, d_\rho^{(a)}\} \subset [r]$. Finally, define r-ary vectors ω_a for $a \in [\binom{r}{\rho}]$ as

$$\omega_a(i) = \begin{cases} t-1 & \text{if } \exists t : i = d_t^{(a)},\\ 0 & \text{otherwise.}\end{cases}$$

$$\begin{array}{cc} & \begin{array}{ccc} \omega_1 & \omega_2 & \omega_3\\ \downarrow & \downarrow & \downarrow \end{array}\\ \Omega = & \begin{bmatrix} 0 & 0 & 0\\ 1 & 0 & 0\\ 0 & 1 & 1 \end{bmatrix} \Rightarrow \begin{array}{llll} Q_{1,j} = & P_{0,3j-2} & P_{0,3j-1} & I_\alpha\\ Q_{2,j} = & P_{1,3j-2} & I_\alpha & P_{0,3j}\\ Q_{3,j} = & I_\alpha & P_{1,3j-1} & P_{1,3j} \end{array} \end{array}$$

Fig. 3. Relation between ω_a and (12).

Construction 2. Let $\alpha = \rho^{k\binom{r}{\rho}}$ and label the α elements $[0 : \alpha-1]$ by ρ-ary vectors in $\mathbb{Z}_\rho^{k\binom{r}{\rho}}$. Define permutation f_j^ℓ on $[0 : \alpha-1]$ as follows:

$$\begin{array}{cccc} f_j^\ell : & \mathbb{Z}_\rho^{k\binom{r}{\rho}} & \rightarrow & \mathbb{Z}_\rho^{k\binom{r}{\rho}}\\ & v & \mapsto & v+\ell e_j, \end{array}$$

for $j \in [k\binom{r}{\rho}]$ and $\ell \in [0 : \rho-1]$, where $\{e_1, \cdots, e_{k\binom{r}{\rho}}\}$ is the standard vector basis of $\mathbb{Z}_\rho^{k\binom{r}{\rho}}$. Let $P_{\ell,j}$ be the $\alpha \times \alpha$ matrix corresponding to the permutation f_j^ℓ.

1) *MSR Code*: The generator matrix of the $[n,k,d]$ code is given by (4), where

$$A_{i,j} = \lambda_{i,j}\prod_{a\in[\binom{r}{\rho}]} P_{w_a(i),a+(j-1)\binom{r}{\rho}}, \quad \text{for } j \in [k], i \in [r].$$

The non-zero coefficients $\lambda_{i,j} \in \mathbb{F}_q$ are defined according to Section II-B to ensure the MDS property; and later will be modified again in Section IV.

2) *Repair Strategy*: Let $\mathcal{R}_a$ correspond to the parity subset of the helper set $\mathcal{D}$. Define $Y_{j,a} \subset [0 : \alpha-1]$ as $\{x \in [0,\alpha-1] : x \cdot e_{a+(j-1)\binom{r}{\rho}} = 0\}$. If systematic node j fails, it is repaired by accessing the elements corresponding to $Y_{j,a}$ from helper nodes, i.e., by accessing $\mathbf{x}_i(v)$, where $i \in \mathcal{D}$, and $v \in Y_{j,a}$.

Lemma 6. *The repair strategy in* Construction 2 *is optimal with respect to repair bandwidth.*

Proof: Let us first explain the role of ω_a by revisiting Example 2 via Fig. 3. Note that the matrix $P_{0,x}$ for any x is also equal to the identity matrix I_α. Here we assumed that

$$\begin{aligned} a = 1 \rightarrow\ \ \mathcal{R}_1 &= \{\mathbf{x}_{k+1}, \mathbf{x}_{k+2}\} \ \rightarrow\ \omega_1 = (0,1,0)^t,\\ a = 2 \rightarrow\ \ \mathcal{R}_2 &= \{\mathbf{x}_{k+1}, \mathbf{x}_{k+3}\} \ \rightarrow\ \omega_2 = (0,0,1)^t,\\ a = 3 \rightarrow\ \ \mathcal{R}_3 &= \{\mathbf{x}_{k+2}, \mathbf{x}_{k+3}\} \ \rightarrow\ \omega_3 = (0,0,1)^t. \end{aligned}$$

In general, the matrix $\Omega = \{\omega_1|\omega_2|\cdots|\omega_{\binom{r}{\rho}}\}$ is designed in a way that for any choice of $a \in [\binom{r}{\rho}]$ we can always find a column in Ω, denoted by ω_a, such that its intersection with ρ rows associated with scenario a, forms $\{0,1,\cdots,\rho-1\}$.

Now assume that systematic node i fails and we are to perform an optimal repair given that the parity nodes in the helper set are in $\mathcal{R}_a = \{\mathbf{x}_{k+d_1^{(a)}}, \mathbf{x}_{k+d_2^{(a)}}, \cdots, \mathbf{x}_{k+d_\rho^{(a)}}\}$.

Define $Q_{u,v} = \lambda_{u,v}^{-1} A_{u,v}$; this is a product of multiple permutation matrices, and hence can be viewed as a permutation matrix itself. It again suffices to define the proper subspaces S_j^i that fulfill the two interference alignment conditions in Lemma 2. Let us select our subspaces as $S_{j,a}^i \triangleq S_a^i \triangleq Y_{i,a} = \{x | x \cdot e_{a+(i-1)\binom{r}{\rho}} = 0\}$. We have

$$\begin{aligned} Y_{i,a} Q_{d_1^{(a)},i} = & \qquad \{x | x \cdot e_{a+(i-1)\binom{r}{\rho}} = 0\}, \\ Y_{i,a} Q_{d_2^{(a)},i} = & \qquad \{x | x \cdot e_{a+(i-1)\binom{r}{\rho}} = 1\}, \\ \vdots & \\ Y_{i,a} Q_{d_\rho^{(a)},i} = & \qquad \{x | x \cdot e_{a+(i-1)\binom{r}{\rho}} = \rho - 1\}, \end{aligned}$$

and hence,

$$\mathsf{rank} \left(\begin{pmatrix} S_a^i Q_{d_1^{(a)},i} \\ S_a^i Q_{d_2^{(a)},i} \\ \vdots \\ S_a^i Q_{d_\rho^{(a)},i} \end{pmatrix} \right) = \rho \times \mathsf{rank}\left(Y_{i,a}\right) = \rho \frac{\alpha}{\rho} = \alpha.$$

The second condition in Lemma 2 is also automatically satisfied since

$$Y_{i,a} \simeq Y_{i,a} Q_{1,i'} \simeq Y_{i,a} Q_{2,i'} \simeq \cdots \simeq Y_{i,a} Q_{\binom{r}{\rho},i'} \quad \text{for } i' \neq i.$$

■

Remark: Note that the sub-packetization α can be optimized further, but it is not the goal in this paper. Although Construction 2 suggests a fairly large value, i.e. $\alpha = \rho^{k\binom{r}{\rho}}$, it is clear that we do not need $\binom{r}{\rho}$ columns in Ω to cover all the $\binom{r}{\rho}$ helper set selection scenarios. Indeed, α in Example 2 can be reduced to 2^{2k}, where $\Omega_{\text{new}} = \{\omega_1 | \omega_2\}$. We leave the optimization strategies of this kind to future work. We also refer the reader to [22]–[24], where MSR constructions with near-optimal sub-packetization parameter, e.g. $\alpha = r^{\lfloor n/r \rfloor}$, are introduced. A different approach to solve this problem is presented in [27], which is a method to reduce the sub-packetization of the existing constructions dramatically in exchange for a slight increase in repair bandwidth.

IV. Code Construction for any Helper Set

In this section, we show that Construction 2 in fact holds, even when an arbitrary set of d helper nodes is allowed to be chosen from the $(n-1)$ surviving nodes. This generality merely imposes some additional constraints on the selection of the scaling coefficients $\lambda_{i,j}$ of the encoding matrices $A_{i,j} = \lambda_{i,j} Q_{i,j}$, where $Q_{i,j}$ is the (product) permutation matrix corresponding to $A_{i,j}$, as defined in Construction 2. We now arrive at the main theorem.

Theorem 7. Construction 2 *gives an* $[n,k,d]$ *systematic–repair MSR code for any set of d helper nodes, for a large enough field size for the scaling coefficients* $\lambda_{i,j}$ *for the encoding matrices* $A_{i,j}$.

Proof: *Part 1:* First, we illustrate the proof by fixing $d = k+1$, and taking an example set of helper nodes for an example failure of node $\mathbf{x}_1$ (or node 1). Let us denote the (indices of the) helper set by $\mathcal{D}$, and let $\mathcal{D} = \{h, h+1, \ldots, k, k+1, \ldots, k+h\}$, that is, there are h parity nodes and $d - h = k + 1 - h$ systematic nodes in the helper set. Let $S_i^j(\mathcal{D})\mathbf{x}_i$ denote the repair information that node i sends to help in the repair of node j when $\mathcal{D}$ is the set of helper nodes. (Wherever clear, we ignore the $\mathcal{D}$ in the notation and simply write S_i^j.) When node 1 fails, the information we therefore have at its replacement node can be written as:

$$\begin{array}{c} \begin{array}{cccccccc} \mathbf{x}_1 & \mathbf{x}_2 & \cdots & \mathbf{x}_h & \mathbf{x}_{h+1} & \cdots & \mathbf{x}_{k-1} & \mathbf{x}_k \end{array} \\ \begin{pmatrix} & & & S_h^1 & & & & \\ & & & & S_{h+1}^1 & & & \\ & & & & & \ddots & & \\ & & & & & & S_{k-1}^1 & \\ & & & & & & & S_k^1 \\ S_{k+1}^1 A_{11} & S_{k+1}^1 A_{12} & & \cdots & & \cdots & & S_{k+1}^1 A_{1k} \\ \vdots & \vdots & & \vdots & & \vdots & & \vdots \\ S_{k+h}^1 A_{h1} & S_{k+h}^1 A_{h2} & & \cdots & & \cdots & & S_{k+h}^1 A_{hk} \end{pmatrix} \begin{pmatrix} \mathbf{x}_1 \\ \mathbf{x}_2 \\ \vdots \\ \mathbf{x}_h \\ \mathbf{x}_{h+1} \\ \vdots \\ \mathbf{x}_{k-1} \\ \mathbf{x}_k \end{pmatrix} \end{array} \tag{14}$$

Suppose all S_j^1's in (14) be replaced by a repair subspace S^1 (corresponding to Lemma 1(b)) that we would have used if $\mathcal{D} = \{2, 3, \ldots, k, k+1, k+2\}$. Specifically, suppose $S^1 A_{1,1}$ and S^1 complete the space $\mathbb{F}_q^\alpha$. Since S^1 and $S^1 A_{i,j}$ denote the same subspace, for $j \neq 1$, the components of $\mathbf{x}_i, i \in \{h, h+1, \ldots, k\}$ can be easily subtracted from the information coming from the parity nodes , using that coming from the systematic nodes h to k. Thus, in order to recover $\mathbf{x}_1$, we can concentrate on the following information at the replacement node:

$$\begin{array}{c} \begin{array}{ccccc} \mathbf{x}_1 & \mathbf{x}_2 & \cdots & \mathbf{x}_{h-2} & \mathbf{x}_{h-1} \end{array} \\ \begin{pmatrix} S^1 A_{11} & S^1 A_{12} & \cdots & S^1 A_{1,h-2} & S^1 A_{1,h-1} \\ S^1 A_{21} & S^1 A_{22} & \cdots & S^1 A_{2,h-2} & S^1 A_{2,h-1} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ S^1 A_{h1} & S^1 A_{h2} & \cdots & S^1 A_{h,h-2} & S^1 A_{h,h-1} \end{pmatrix} \begin{pmatrix} \mathbf{x}_1 \\ \mathbf{x}_2 \\ \vdots \\ \mathbf{x}_{h-2} \\ \mathbf{x}_{h-1} \end{pmatrix} \end{array} \tag{15}$$

Let $S^1 A_{i,j} = \lambda_{i,j} S^1 Q_{i,j} = \lambda_{i,j} \widetilde{Q}_{i,j} S^1$, where $\widetilde{Q}_{i,j}$ is an $\alpha/2 \times \alpha/2$ matrix, and $(i,j) \neq (1,1)$. It must be noted that not only is S^1 dependent on the choice of $\mathcal{D}$, but so in turn is $\widetilde{Q}_{i,j}$. Let us also denote $S^1 \mathbf{x}_i$ by $\widetilde{\mathbf{x}}_i$. Then, (15) can be rewritten as:

$$\begin{pmatrix} \lambda_{1,1} S^1 Q_{1,1} & \lambda_{1,2} \widetilde{Q}_{1,2} & \cdots & \lambda_{1,h-1} \widetilde{Q}_{1,h-1} \\ \lambda_{2,1} \widetilde{Q}_{2,1} S^1 & \lambda_{2,2} \widetilde{Q}_{2,2} & \cdots & \lambda_{2,h-1} \widetilde{Q}_{2,h-1} \\ \vdots & \vdots & \ddots & \vdots \\ \lambda_{h,1} \widetilde{Q}_{h,1} S^1 & \lambda_{h,2} \widetilde{Q}_{h,2} & \cdots & \lambda_{h,h-1} \widetilde{Q}_{h,h-1} \end{pmatrix} \begin{pmatrix} \mathbf{x}_1 \\ \widetilde{\mathbf{x}}_2 \\ \vdots \\ \widetilde{\mathbf{x}}_{h-1} \end{pmatrix} \tag{16}$$

The matrix in (16) — call it M — is a square matrix of dimensions $h\alpha/2 \times h\alpha/2$. A sufficient condition to recover $\mathbf{x}_1$ is that M is invertible. Notice that the determinant of M, $\det(M)$, is a polynomial in the following variables: $\lambda_{i,j}, i \in [h], j \in [h-1]$. Hence, $\det(M)$ is a nonzero polynomial of degree $h\alpha/2$ in the given variables. From Schwartz–Zippel–DeMillo–Lipton lemma [28], if the finite field $\mathbb{F}_q$ over which the determinant is defined has cardinality $|\mathbb{F}_q| = q > h\alpha/2$, there exist $\lambda_{i,j}$'s for which the determinant $\det(M)$ above is nonzero.

Part 2: Notice that M above is defined for a particular example scenario. In general, let the number of helper nodes be d, the failed systematic node be $f \in [k]$, the set of helper nodes by $\mathcal{D} \subseteq [n] \backslash \{f\}$, the set of systematic helper nodes be $\mathcal{D}_s \subseteq [k] \backslash \{f\}$, and the set of parity helper nodes be $\mathcal{D}_p \subseteq [k+1 : k+r]$. Let the number of parity helper nodes be denoted by h, where h ranges from $d-k+1$ to r. Let us represent by $\mathcal{H}_p$ the set of parity helper nodes but indexed within $[r]$, where i corresponds to node $k+i$ of the system, that is, $\mathcal{H}_p = \{i \,|\, k+i \in \mathcal{D}_p\} \subseteq [r]$.

The matrix M in (16), in general, can be seen to be a square matrix of dimensions $h\alpha/(d-k+1) \times h\alpha/(d-k+1)$. In particular, M is a function of f, $\mathcal{D}_s$, and $\mathcal{H}_p$, and the determinant polynomial has degree which is a function of $|\mathcal{D}_p| = h$ and d. For each f, $\mathcal{D}_s$ and $\mathcal{H}_p$, we obtain a sufficiency condition that the corresponding M is invertible. Therefore, the product of the corresponding determinant polynomials is a nonzero polynomial of degree

$$\begin{aligned} q_{\text{ANY}} &= k\left(\sum_{h=d-k+1}^{r} \binom{r}{h}\binom{k-1}{d-h}\frac{h\alpha}{d-k+1}\right) \\ &= \left(\sum_{h=d-k+1}^{r} h\binom{r}{h}\binom{k-1}{d-h}\right)\frac{k\alpha}{d-k+1}; \quad (17) \end{aligned}$$

consequently, there exist $\lambda_{i,j}$'s in $\mathbb{F}$ such that any systematic node is repairable with optimal repair bandwidth using any arbitrary set of d helper nodes, as long as the field size $|\mathbb{F}| > q_{\text{ANY}}$.

Part 3: Finally, using Lemma 4, Lemma 5, and Lemma 6, we obtain an $[n,k,d]$ systematic–repair MSR code for any set of d helper nodes, when the field size $q > q_{\text{ANY}} + q_{\text{MDS}}$. ■

Remark: As shown in [17], the sufficiency conditions on the field size such as equation (11) are not always necessary. Indeed, the authors in [17] show that for small values of r there exist smart choices of coefficients $\lambda_{i,j}$ such that a field size of 3 or 4 would suffice. However for arbitary values of r, their construction also follows a combinatorial Nullstellensatz argument to prove the existence of such codes. Note that for the case of $d = n-1$, there is only one selection of helper sets, and hence condition (17) becomes unnecessary.

Table I provides a comparison of these sufficiency conditions for a select set of code parameters. The calculation of q_{MDS} and q_{ANY} assumes a large value for n. The field size is only given based on sufficiency conditions in equations (11) and (17), which is based on an existence argument. However as shown in [17], the explicit and optimized constructions of MSR codes may require much smaller field sizes. The last row in the table corresponds to the all-d construction in Example 3.

$[n,k,d]$	α	q_{MDS}/α	q_{ANY}/α	$\vert\mathbb{F}\vert/\alpha$
$[n, n-2, n-1]$	2^{n-2}	$n-1$	$n-2$	$\mathcal{O}(n)$
$[n, n-3, n-1]$	3^{n-3}	$2\binom{n-4}{2}$	$n-3$	$\mathcal{O}(n^2)$
$[n, n-3, n-2]$	$2^{3(n-3)}$	$2\binom{n-4}{2}$	$3\binom{n-2}{2}$	$\mathcal{O}(n^2)$
$[n, n-4, n-1]$	4^{n-4}	$3\binom{n-5}{3}$	$n-4$	$\mathcal{O}(n^3)$
$[n, n-4, n-2]$	$3^{6(n-4)}$	$3\binom{n-5}{3}$	$\frac{4(n-2)(n-4)}{3}$	$\mathcal{O}(n^3)$
$[n, n-4, n-3]$	$2^{4(n-4)}$	$3\binom{n-5}{3}$	$\frac{(n-4)(2n^2-9n+52)}{2}$	$\mathcal{O}(n^3)$
$[n, n-3]$ all-d	$6^{4(n-3)}$	$2\binom{n-4}{2}$	$\frac{(n-3)(3n-4)}{2}$	$\mathcal{O}(n^2)$

TABLE I
SUB-PACKETIZATION AND FIELD SIZE OF THE PROPOSED MSR CODES FOR A SELECT SET OF PARAMETERS.

V. CONSTRUCTION OF $[n,k]$ MSR CODES

This section is devoted to generalizing the construction of $[n,k,d]$ MSR codes from previous sections to $[n,k]$ MSR codes, where a fixed code construction achieves the optimal repair bandwidth for any given d helper nodes ($k \leqslant d \leqslant n-1$).

Definition. *Let Ω be an $r \times \theta$ matrix, where its elements are chosen from $\mathbb{Z}_\tau$ and τ is the least common multiple* $\mathsf{lcm}\{1, 2, \cdots, r\}$. *We call Ω an (r,θ)-*Product Matrix *if and only if for any subset of rows in Ω such as $\{i_1, \cdots, i_\rho\} \subset [r]$ (for $\rho \leqslant r$), there exists a column $c \in [\theta]$ which intersects with these rows in the set of all the elements modulo $\mathbb{Z}_\rho$.*

We previously mentioned that the matrix Ω in Fig. 3 satisfies the property that for any selection of two rows, there exists a column such that it intersects with those rows in $\{0,1\}$, the set of all elements modulo $\mathbb{Z}_2$. Before proceeding with an example construction of a Product Matrix, recall the definition of $\omega_a(i)$, which was the key element in Construction 2. Consider an $r \times \binom{r}{\rho}$ matrix Ω where

$$\Omega_{i,a} = \omega_a(i) \quad \text{for} \quad i \in [r] \text{ and } a \in \left[\binom{r}{\rho}\right].$$

It is easy to verify that for any selection of ρ rows from Ω, one can find a column whose intersection with these rows form all elements in $\{0, 1, \cdots, \rho - 1\}$.

Example 3. A simple way to construct a Product Matrix is to concatenate the Ω matrices from Construction 2 for different values of $\rho \leqslant r$ and arrive at an $r \times \theta$ matrix where $\theta = \binom{r}{r} + \binom{r}{r-1} + \cdots + \binom{r}{2}$. The $(3,4)$-Product Matrix defined as

$$\Omega = \begin{bmatrix} 0 & 0 & 0 & 1 \\ 1 & 0 & 1 & 0 \\ 2 & 1 & 0 & 0 \end{bmatrix},$$

where the elements are chosen from $\mathbb{Z}_6$, serves as such an example.

Construction 3. Let Ω be an (r,θ)-Product Matrix. Define $\alpha = \tau^{k\theta}$, where $\tau = \mathsf{lcm}\{1, 2, \cdots, r\}$, and label the

α elements $[0 : \alpha - 1]$ by τ-ary vectors in $\mathbb{Z}_\tau^{k\theta}$. Define permutation f_j^ℓ on $[0 : \alpha - 1]$ as follows:

$$\begin{array}{rccc} f_j^\ell : & \mathbb{Z}_\tau^{k\theta} & \to & \mathbb{Z}_\tau^{k\theta} \\ & v & \mapsto & v + \ell e_j, \end{array}$$

for $j \in [k\theta]$ and $\ell \in [0 : \tau - 1]$, where $\{e_1, \cdots, e_{k\theta}\}$ is the standard vector basis of $\mathbb{Z}_\rho^{k\theta}$. Let P_j^ℓ be the $\alpha \times \alpha$ matrix corresponding to the permutation f_j^ℓ.

1) *MSR Code*: The generator matrix of the $[n, k]$ code is given by (4), where

$$A_{i,j} = \lambda_{i,j} \prod_{a \in [\theta]} P_{a+(j-1)\theta}^{\Omega_{i,a}}, \quad \text{for } j \in [k], i \in [r].$$

The non-zero coefficients $\lambda_{i,j} \in \mathbb{F}_q$ are defined according to the discussion in Section II-B and Section IV to ensure the MDS property and to allow arbitrary selections of helper nodes.

2) *Repair Strategy*: Assume that the helper set $\mathcal{D}$ consists of the remaining $k - 1$ systematic nodes (an assumption that can be dropped by a clever selection of $\lambda_{i,j} \in \mathbb{F}_q$ according to Section IV) and a subset of parity nodes denoted by $\mathcal{R}$. Furthermore, assume $a \in [\theta]$ corresponds to a column in Ω whose intersection with the rows associated with these parities forms all elements mod $\mathbb{Z}_\rho$, where $\rho = |\mathcal{R}|$. Define $Y_{j,a} \subset [0 : \alpha - 1]$ as $\{x \in [0, \alpha - 1] : x \cdot e_{a+(j-1)\theta} = 0 \ (\text{mod } \rho)\}$. If systematic node j fails, it is repaired by downloading the elements corresponding to $Y_{j,a}$ from helper nodes in $\mathcal{D}$, i.e., by accessing $\mathbf{x}_i(v)$, where $i \in \mathcal{D}$, and $v \in Y_{j,a}$.

Lemma 8. *The repair strategy in* Construction 3 *is optimal with respect to repair bandwidth.*

Proof: The proof in general follows steps similar to those for that of Theorem 6 and is omitted to avoid repetition. We refer the reader to the previous sections for notation, definitions and the overall flow of the proof. However, it is helpful to mention that the matrix Ω is designed in a way that for any repair scenario a, we can always find a column in Ω, denoted by Ω_a, such that its intersection with ρ rows associated with parity nodes in scenario a is given by $\{0, 1, \cdots, \rho - 1\}$.

Now assume that systematic node i fails and we are to perform an optimal repair given parity nodes in $\mathcal{R}_a = \{\mathbf{x}_{k+d_1^{(a)}}, \mathbf{x}_{k+d_2^{(a)}}, \cdots, \mathbf{x}_{k+d_\rho^{(a)}}\}$. It is now clear that if we select our subspaces as $S_{j,a}^i \triangleq S_a^i \triangleq Y_{i,a} = \{x | x \cdot e_{a+(i-1)\theta} = 0 \ (\text{mod } \rho)\}$, then $\dim(S_a^i) = \frac{\alpha}{\rho}$, and

$$\begin{aligned} Y_{i,a} Q_{d_1^{(a)}, i} &= \quad \{x | x \cdot e_{a+(i-1)\theta} = 0 \ (\text{mod } \rho)\}, \\ Y_{i,a} Q_{d_2^{(a)}, i} &= \quad \{x | x \cdot e_{a+(i-1)\theta} = 1 \ (\text{mod } \rho)\}, \\ &\ \ \vdots \\ Y_{i,a} Q_{d_\rho^{(a)}, i} &= \quad \{x | x \cdot e_{a+(i-1)\theta} = \rho - 1 \ (\text{mod } \rho)\}, \end{aligned}$$

and therefore,

$$\mathsf{rank}\left(\begin{pmatrix} S_a^i Q_{d_1^{(a)}, i} \\ S_a^i Q_{d_2^{(a)}, i} \\ \vdots \\ S_a^i Q_{d_\rho^{(a)}, i} \end{pmatrix}\right) = \rho \times \mathsf{rank}\left(Y_{i,a}\right) = \rho \frac{\alpha}{\rho} = \alpha.$$

■

Notice that arriving at a construction which guarantees the optimal repair bandwidth for any selection of helper nodes comes at the cost of a huge increase in the sub-packetization parameter α.

Remark: This increase in sub-packetization can be viewed also as a trade-off between α and the flexibility of choosing the helper set size. For instance, assume that only a few choices of d such as $\{d_1, d_2, \cdots, d_\delta\} \subset [r]$ are desired. A naive construction of the product matrix as in Example 3 would result in an $r \times \theta$ matrix, where

$$\theta = \binom{r}{d_1 - k + 1} + \binom{r}{d_2 - k + 1} + \cdots + \binom{r}{d_\delta - k + 1},$$

and its elements are chosen from $\mathbb{Z}_\tau$, with

$$\tau = \text{l.c.m.}\{d_1 - k + 1, d_2 - k + 1, \cdots, d_\delta - k + 1\}.$$

The sub-packetization of this MSR code — call it an $[n, k, \{d_1, d_2, \cdots, d_\delta\}]$ MSR code — from Construction 3 is then given by $\alpha = \tau^{k\theta}$, which establishes a trade-off between α and the flexibility $\{d_1, d_2, \cdots, d_\delta\} \subset [r]$ of the helper set size.

For practical purposes one may seek to reduce α. One way to do so is to minimize the number of columns in a Product Matrix for a fixed number of rows. For example, it is easy to verify that the matrix Ω defined as

$$\Omega = \begin{bmatrix} 0 & 0 \\ 1 & 0 \\ 2 & 1 \end{bmatrix},$$

is also a Product Matrix with $r = 3$, and would be preferred over the $(3, 4)$-Product Matrix introduced in Example 3. Although the trivial construction of Product Matrices guarantees $\theta = \mathcal{O}(2^r)$, we believe that the optimal value for θ is much smaller. We finish this section by stating this as an interesting combinatorial problem, which will be addressed in future work.

Open Problem. Given a positive integer $r \in \mathbb{Z}$, what is the smallest value for θ such that there exists an (r, θ)-Product Matrix, i.e., an $r \times \theta$ matrix Ω with elements in $\mathbb{Z}_\tau$ ($\tau = \mathsf{lcm}\{1, 2, \cdots, r\}$), where for any selection of ρ ($1 \leqslant \rho \leqslant r$) rows from Ω, one can always find a column in Ω which intersects with these ρ rows in all the elements in $\mathbb{Z}_\rho$?

VI. Conclusion

In this paper we first presented a new construction for systematic–repair $[n, k, d]$ MSR codes for all possible values of parameters n, k, and d. We discussed how a proper selection of coefficients in $\mathbb{F}_q$ guarantees the MDS property of the code, which can also be utilized to drop the restriction of choosing all the remaining systematic nodes when selecting the helper nodes.

Next, we introduced a more generalized construction, where a single $[n,k]$ code simultaneously satisfies the optimal repair for all $d \in \{k+1, \cdots, n-1\}$. Both generalizations come at the cost of increasing α. A lower bound on α is proved in [29] when $d = n-1$. Whether similar bounds exist for general $\{n,k,d\}$ or not is left for future work.

References

[1] A. Fazeli, S. Goparaju, and A. Vardy, "Minimum Storage Regenerating Codes for all Parameters," in *Proceedings of IEEE International Symposium on Information Theory (ISIT)*, Jul. 2016, pp. 76–80.

[2] D. Borthakur, "The Hadoop Distributed File System: Architecture and Design," in *hadoop.apache.org*, 2007. [Online]. Available: http://hadoop.apache.org/docs/r0.18.0/hdfs_design.pdf

[3] S. Ghemawat, H. Gobioff, and S.-T. Leung, "The Google File System," in *Proceedings of the 19th ACM Symposium on Operating Systems Principles*, 2003, pp. 20–43.

[4] V. Guruswami and M. Wootters, "Repairing Reed-Solomon Codes," in *arxiv.org*, September 2015. [Online]. Available: http://arxiv.org/abs/1509.04764

[5] A. G. Dimakis, P. B. Godfrey, Y. Wu, M. Wainwright, and K. Ramchandran, "Network Coding for Distributed Storage Systems," in *IEEE Transactions on Information Theory*, vol. 56, no. 9, September 2010, pp. 4539–4551.

[6] N. Shah, K. Rashmi, P. V. Kumar, and K. Ramchandran, "Distributed Storage Codes with Repair-by-Transfer and Nonachievability of Interior Points on the Storage-Bandwidth Tradeoff," in *IEEE Transactions on Information Theory*, vol. 58, no. 3, March 2012, pp. 1837–1852.

[7] 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," vol. 57, Aug. 2011, pp. 5227–5239.

[8] Y. Wu and A. G. Dimakis, "Reducing Repair Traffic for Erasure Coding-Based Storage via Interference Alignment," in *Proceedings of the IEEE International Symposium on Information Theory (ISIT)*, June-July 2009, pp. 2276–2280.

[9] D. Cullina, A. G. Dimakis, and T. Ho, "Searching for Minimum Storage Regenerating Codes," in *arxiv.org*, October 2009. [Online]. Available: http://arxiv.org/abs/0910.2245

[10] Y. Wu, "A Construction of Systematic MDS Codes with Minimum Repair Bandwidth," in *IEEE Transactions on Information Theory*, June 2011, pp. 3738–3741.

[11] V. Cadambe, S. Jafar, H. Maleki, K. Ramchandran, and C. Suh, "Asymptotic Interference Alignment for Optimal Repair of MDS codes in Distributed Storage," vol. 59, May 2013, pp. 2974–2987.

[12] V. R. Cadambe, C. Huang, J. Li, and S. Mehrotra, "Polynomial Length MDS Codes with Optimal Repair in Distributed Storage," in *Proceedings of the 45th Asilomar Conference on Signals, Systems and Computers*, Nov. 2011, pp. 1850–1854.

[13] D. S. Papailiopoulos, A. G. Dimakis, and V. R. Cadambe, "Repair Optimal Erasure Codes through Hadamard Designs," in *Proceedings of the 49th Annual Allerton Conference on Communication, Control, and Computing (Allerton)*, Sep. 2011, pp. 1382–1389.

[14] Z. Wang, I. Tamo, and J. Bruck, "On Codes for Optimal Rebuilding Access," in *Proceedings of the 49th Annual Allerton Conference on Communication, Control, and Computing (Allerton)*, September 2011, pp. 1374–1381.

[15] ——, "Long MDS Codes for Optimal Repair Bandwidth," in *Proceedings of IEEE International Symposium on Information Theory (ISIT)*, Jul. 2012, pp. 1182–1186.

[16] C. Suh and K. Ramchandran, "Exact-Repair MDS Codes for Distributed Storage using Interference Alignment," in *Proceedings of IEEE International Symposium on Information Theory (ISIT)*, June 2010, pp. 161–165.

[17] I. Tamo, Z. Wang, and J. Bruck, "Zigzag Codes: MDS Array Codes With Optimal Rebuilding," in *IEEE Transactions on Information Theory*, vol. 59, March 2013, pp. 1597–1616.

[18] G. Agarwal, B. Sasidharan, and P. V. Kumar, "An Alternate Construction of an Access-Optimal Regenerating Code with Optimal Sub-Packetization Level," in *National Conference on Communications (NCC)*, 2015, pp. 1–6.

[19] N. Raviv, N. Silberstein, and T. Etzion, "Constructions of High-Rate MSR Codes Over Small Fields," in *arxiv.org*, 2015.

[20] I. Tamo and E. En Gad, "$[n,k]$ Minimum-Storage Regenerating Codes for all d's in the Range $\{k, k+1, \ldots, n-1\}$ Simultaneously," *53rd Allerton Conference on Communications, Control, and Computing (invited)*, 2015.

[21] A. S. Rawat, O. O. Koyluoglu, and S. Vishwanath, "Progress on High–rate MSR Codes: Enabling Arbitrary Number of Helper Nodes," in *arxiv.org*, January 2016. [Online]. Available: http://arxiv.org/abs/1601.06362

[22] M. Ye and A. Barg, "Explicit Constructions of High-Rate MDS Array Codes with Optimal Repair Bandwidth," in *arxiv.org*, April 2016. [Online]. Available: http://arxiv.org/abs/1604.00454

[23] ——, "Explicit Constructions of Optimal-Access MDS Codes with Nearly Optimal Sub-Packetization," in *arxiv.org*, May 2016. [Online]. Available: http://arxiv.org/abs/1605.08630

[24] 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," in *arxiv.org*, August 2016. [Online]. Available: https://arxiv.org/abs/1607.07335

[25] I. Tamo and K. Efremenko, "New Results on MSR Codes," *Information Theory and Applications Workshop (ITA)*, February 2016.

[26] V. Guruswami and A. S. Rawat, "New MDS Codes with Small Sub-Packetization and Near-Optimal Repair Bandwidth," in *arxiv.org*, August 2016. [Online]. Available: http://arxiv.org/abs/1608.00191

[27] K. Kralevska, D. Gligoroski, and H. Overby, "General Sub-Packetized Access-Optimal Regenerating Codes," in *IEEE Communications Letters*, vol. 20, Jul. 2016, pp. 1281–1284.

[28] J. T. Schwartz, "Fast Probabilistic Algorithms for Verification of Polynomial Identities," in *Journal of ACM (JACM)*, Oct. 1980, pp. 701–717.

[29] S. Goparaju, I. Tamo, and R. Calderbank, "An Improved Sub-Packetization Bound for Minimum Storage Regenerating Codes," in *IEEE Transactions on Information Theory*, vol. 60, no. 5, 2014, pp. 2770–2779.

Appendix A
Interference Alignment

Proof of Lemma 1: We prove the result for the failure of systematic node $i = 1$. The argument generalizes for the failure of other systematic nodes. Let us assume that node $\mathbf{x}_1$ fails, and let each of the remaining $d = n-1$ nodes send $\beta = \alpha/r$ symbols to recover $\mathbf{x}_1$. In other words, node $\mathbf{x}_j$ (where $j \in [n]$, $j \neq i$) sends $S_j^1 \mathbf{x}_j$ for some $\alpha/r \times \alpha$ matrix S_j^1. We therefore need to recover $\mathbf{x}_1$ from the following functions of $\mathbf{x}_i, i \in [k]$:

$$\begin{array}{c}
\begin{array}{ccccc} \mathbf{x}_1 & \mathbf{x}_2 & \mathbf{x}_3 & \cdots & \mathbf{x}_k \end{array} \\
\begin{pmatrix}
 & S_2^1 & & & \\
 & & S_3^1 & & \\
 & & & \ddots & \\
 & & & & S_k^1 \\
S_{k+1}^1 A_{1,1} & S_{k+1}^1 A_{1,2} & \cdots & \cdots & S_{k+1}^1 A_{1,k} \\
S_{k+2}^1 A_{2,1} & S_{k+2}^1 A_{2,2} & \cdots & \cdots & S_{k+2}^1 A_{2,k} \\
\vdots & \vdots & \vdots & \vdots & \vdots \\
S_{k+r}^1 A_{r,1} & S_{k+r}^1 A_{r,2} & \cdots & \cdots & S_{k+r}^1 A_{r,k}
\end{pmatrix}
\end{array}
\begin{pmatrix} \mathbf{x}_1 \\ \mathbf{x}_2 \\ \mathbf{x}_3 \\ \vdots \\ \mathbf{x}_k \end{pmatrix}. \quad (18)$$

Necessity: Suppose the systematic vectors $\mathbf{x}_2$ through $\mathbf{x}_k$ be the zero vectors. Then, (18) simplifies to:

$$\begin{pmatrix} S_{k+1}^1 A_{1,1} \\ S_{k+2}^1 A_{2,1} \\ \vdots \\ S_{k+r}^1 A_{r,1} \end{pmatrix} \mathbf{x}_1,$$

where the matrix is an $\alpha \times \alpha$ square matrix. Since $\mathbf{x}_1$ is recoverable, it is necessary that the matrix be non-singular, thus proving the signal recovery conditions (5) and (7). Note that this also implies that all encoding matrices $A_{i,j}, i \in [r], j \in [k]$, are non-singular. s

Suppose now, without loss of generality, that the interference alignment condition (6) is not satisfied for $j = 2$. Again, without loss of generality, let $S_2^1 \not\simeq S_{k+1}^1 A_{1,2}$. This implies that

$$\operatorname{rank}\left(\begin{pmatrix} S_2^1 \\ S_{k+1}^1 A_{1,2} \end{pmatrix} \right) \quad = \quad \frac{\alpha}{r} + \epsilon, \qquad (19)$$

for some $\epsilon > 0$. Since $\mathbf{x}_1$ is recoverable, from (18), we have access to the following information at the replacement node:

$$\begin{array}{c} \begin{array}{ccccc} \mathbf{x}_1 & \mathbf{x}_2 & \mathbf{x}_3 & \cdots & \mathbf{x}_k \end{array} \\ \begin{pmatrix} I_\alpha & & & & \\ & S_2^1 & & & \\ & & S_3^1 & & \\ & & & \ddots & \\ & & & & S_k^1 \\ 0 & S_{k+1}^1 A_{1,2} & \cdots & \cdots & S_{k+1}^1 A_{1,k} \end{pmatrix} \end{array} \begin{pmatrix} \mathbf{x}_1 \\ \mathbf{x}_2 \\ \mathbf{x}_3 \\ \vdots \\ \mathbf{x}_k \end{pmatrix}. \qquad (20)$$

From (19), the rank of the matrix in (20) is at least

$$\alpha + \frac{\alpha}{r} + \epsilon + (k-2)\frac{\alpha}{r} \quad > \quad (n-1)\frac{\alpha}{r},$$

the total number of symbols available at the replacement node. In other words, we are able to recover more number of linearly independent symbols that are functions of the systematic data vectors $\mathbf{x}_1$ through $\mathbf{x}_k$, than the number of repair symbols available at the replacement node — a contradiction! Thus, conditions (6) and (8) must be true.

Sufficiency: Suppose that we have the required repair matrices S_j^1 that satisfy the signal recovery and interference alignment conditions (5) and (6). Using (6), we can eliminate the contribution of systematic vectors $\mathbf{x}_2$ through $\mathbf{x}_k$ in the information transmitted by the parity nodes (that is, the last r rows in (18)). For instance, $S_2^1 \simeq S_{k+1}^1 A_{1,2}$ implies that $S_{k+1}^1 A_{1,2} = B S_2^1$, for some $\alpha/r \times \alpha/r$ matrix B, and therefore the contribution of $S_{k+1}^1 A_{1,2} \mathbf{x}_2$ can be removed from the repair information transmitted by the parity node $k+1$ using the repair information $S_2^1 \mathbf{x}_2$ (or equivalently, $B S_2^1 \mathbf{x}_2$) transmitted by systematic node 2. Using (5), it is then easy to recover $\mathbf{x}_1$. ■

Sreechakra Goparaju is a postdoctoral scholar in the California Institute for Telecommunications and Information Technology (Calit2) at the University of California, San Diego. He received his Ph.D. in Electrical Engineering at Princeton University, where he joined as a Gordon Wu Fellow from 2008. He received a B.Tech. and an M.Tech. (dual degree) in Electronics and Electrical Communication Engineering from Indian Institute of Technology, Kharagpur. His research interests are information theory, coding on storage, and combinatorics.

Arman Fazeli (S'14) was born in Tehran, Iran, in 1989. He received the B.S. degree in electrical engineering from Sharif University of Technology, Tehran, Iran, in 2012. Since then, he has been working towards his Ph.D. degree in the in the Department of Electrical and Computer Engineering at the University of California, San Diego, where he is supervised by Prof. Alexander Vardy. His current research interests include information and coding theory, with particular emphasis on coding for distributed storage systems. Arman received silver and bronze medals at the International Mathematical Olympiad (IMO) in 2006 and 2007, when he was in the Iran national mathematics team.

Alexander Vardy (S88–M91–SM94–F99) was born in Moscow, U.S.S.R., in 1963. He earned his B.Sc. (summa cum laude) from the Technion, Israel, in 1985, and Ph.D. from the Tel-Aviv University, Israel, in 1991. During 1985–1990 he was with the Israeli Air Force, where he worked on electronic counter measures systems and algorithms. During the years 1992 and 1993 he was a Visiting Scientist at the IBM Almaden Research Center, in San Jose, CA. From 1993 to 1998, he was with the University of Illinois at Urbana-Champaign, first as an Assistant Professor then as an Associate Professor. Since 1998, he has been with the University of California San Diego (UCSD), where he is the Jack Keil Wolf Endowed Chair Professor in the Department of Electrical and Computer Engineering and the Department of Computer Science. While on sabbatical from UCSD, he has held long-term visiting appointments with CNRS, France, the EPFL, Switzerland, the Technion, Israel, and Nanyang Technological University, Singapore.

His research interests include error-correcting codes, algebraic and iterative decoding algorithms, lattices and sphere packings, coding for digital media, cryptography and computational complexity theory, as well as fun math problems.

He received an IBM Invention Achievement Award in 1993, and NSF Research Initiation and CAREER awards in 1994 and 1995. In 1996, he was appointed Fellow in the Center for Advanced Study at the University of Illinois, and received the Xerox Award for faculty research. In the same year, he became a Fellow of the David and Lucile Packard Foundation. He received the IEEE Information Theory Society Paper Award (jointly with Ralf Koetter) for the year 2004. In 2005, he received the Fulbright Senior Scholar Fellowship, and the Best Paper Award at the IEEE Symposium on Foundations of Computer Science (FOCS). During 1995–1998, he was an Associate Editor for Coding Theory and during 1998–2001, he was the Editor-in-Chief of the IEEE TRANSACTIONS ON INFORMATION THEORY. From 2003 to 2009, he was an Editor for the *SIAM Journal on Discrete Mathematics*. He is currently serving on the Executive Editorial Board for the IEEE TRANSACTIONS ON INFORMATION THEORY. He has been a member of the Board of Governors of the IEEE Information Theory Society during 1998–2006, and again during 2011–2017.