Centralized Multi-Node Repair Regenerating Codes

Marwen Zorgui, *Student Member, IEEE*, Zhiying Wang, *Member, IEEE*

Abstract—In a distributed storage system, recovering from multiple failures is a critical and frequent task that is crucial for maintaining the system's reliability and fault-tolerance. In this work, we focus on the problem of repairing multiple failures in a centralized way, which can be desirable in many data storage configurations, and we show that a significant repair traffic reduction is possible. First, the fundamental tradeoff between the repair bandwidth and the storage size for functional repair is established. Using a graph-theoretic formulation, the optimal tradeoff is identified as the solution to an integer optimization problem, for which a closed-form expression is derived. Expressions of the extreme points, namely the minimum storage multi-node repair (MSMR) and minimum bandwidth multi-node repair (MBMR) points, are obtained. Second, we describe a general framework for converting single erasure minimum storage regenerating codes to MSMR codes. The repair strategy for e failures is similar to that for a single failure, however certain extra requirements need to be satisfied by the repairing functions for a single failure. For illustration, the framework is applied to product-matrix codes and interference alignment codes. Furthermore, we prove that the functional MBMR point is not achievable for linear exact-repair codes. We also show that exact-repair minimum bandwidth cooperative repair (MBCR) codes achieve an interior point, that lies near the MBMR point, when $k \equiv 1 \mod e$, k being the minimum number of nodes needed to reconstruct the entire data. Finally, for $k > 2e, e \mid k$ and $e \mid d$, where d is the number of helper nodes during repair, we show that the functional repair tradeoff is not achievable under exact repair, except for maybe a small portion near the MSMR point, which parallels the results for single-erasure repair by Shah et al.

Index Terms—Regenerating codes, distributed storage, multi-node centralized repair, minimum storage, minimum bandwidth.

I. INTRODUCTION

Ensuring data reliability is of paramount importance in modern storage systems. Reliability is typically achieved through the introduction of redundancy. Traditionally, simple replication of data has been adopted in many systems. For instance, Google file systems opted for a triple replication policy [3]. However, for the same redundancy factor, replication systems fall short on providing the highest level of reliability. On the other hand, erasure codes can be optimal in terms of the redundancy-reliability tradeoff. In erasure codes, a file of size $\mathcal{M}$ is divided into k fragments, each of size $\frac{\mathcal{M}}{k}$. The k fragments are then encoded into n fragments using an (n, k) maximum distance separable (MDS) code and then stored at n different nodes. Using such a scheme, the data is guaranteed to be recovered from any $n - k$ node erasures,

providing the highest level of worst-case data reliability for the given redundancy. However, traditional erasure codes suffer from high repair bandwidth. In the case of a single node erasure, they require downloading the entire data of size $\mathcal{M}$ to repair a single node storing a fragment of size $\frac{\mathcal{M}}{k}$. This expansion factor made erasure codes impractical in some applications using distributed storage systems. In the last decade, the repair problem has gained increasing interest and motivated the research for a new class of erasure codes with better repair capabilities. The seminal work in [4] proposed regenerating codes that optimally solve the repair bandwidth problem. Interestingly, the authors in [4] proved that one can significantly reduce the amount of bandwidth required for repair and the bandwidth decreases as each node stores more information. Formally, suppose any k out of n nodes are sufficient to recover the entire file of size $\mathcal{M}$. Assuming that d nodes, termed helpers, participate in the repair process, denoting the storage capacity of each node by α and the amount of information downloaded from each helper by β, then, an optimal $(\mathcal{M}, n, k, d, \alpha, \beta)$ regenerating code satisfies

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

Equation (1) describes the fundamental tradeoff between the storage capacity α and the bandwidth β. Two extreme points can be obtained from the tradeoff. Minimum storage regenerating (MSR) codes correspond to the best storage efficiency with $\alpha = \frac{\mathcal{M}}{k}$, while minimum bandwidth regenerating (MBR) codes achieve the lowest possible bandwidth at the expense of extra storage per node.

If we recover the exact same information as the failed node, we call it *exact repair*, otherwise we call it *functional repair*. Using network coding [5], [6], it is possible to construct functional regenerating codes satisfying (1) [4]. Following the seminal work in [4], there has been a flurry of interest in designing exact-repair regenerating codes that achieve the optimal tradeoff, focusing mainly on the extreme MSR and MBR points, e.g., [7]–[16]. For interior points that are between the MBR and MSR points in the tradeoff of (1), [17] showed that most points are not achievable for exact repair. Moreover, there has been a growing literature focused on understanding the fundamental limits of exact-repair regenerating codes. Other outer bounds for exact repair include [18]–[20] for general parameters, and [21] for linear codes. The aforementioned references, as most of the studies on regenerating codes in the literature, focus on the single erasure repair problem. However, in many practical scenarios, such as in large scale storage systems, multiple failures are more frequent than a single failure. Moreover, many systems (e.g., [22]) apply a lazy repair strategy, which seeks to limit the repair cost of erasure

codes. Instead of immediately repairing every single failure, a a lazy repair strategy waits until e erasures occur, $e \leq n - k$, then, the repair is done by downloading the equivalent of the total information in the system to regenerate the erased nodes. However, a natural question of interest is, whether one can reduce the amount of download in such scenarios.

In this work, we consider centralized repair. Indeed, there are situations in which, due to architectural constraints, it is more desirable to regenerate the lost nodes at a central server before dispatching the regenerated content to the replacement nodes [22]. For instance, one can think of a rack-based node placement architecture [23] in which failures frequently occur to nodes corresponding to a particular rack. In this scenario, a centralized repair of the entire rack is favorable as opposed to repairing the rack on a per-node basis. Furthermore, [23] showed that a centralized repair framework can have interesting applications in communication-efficient secret sharing. Finally, centralized repair can be used in a broadcast network, where the repair information is transmitted to all replacement nodes (e.g. [24]).

Our centralized repair framework requires the content of any k out of n nodes in the system to be sufficient to reconstruct the entire data. Upon the failure of e nodes in the system, the repair is carried out by contacting any d helpers out of the $n-e$ available nodes, $d \leq n-e$, and downloading β amount of information from each of the d helpers. Our first objective is to characterize the functional repair tradeoff between the storage per node α and the repair bandwidth β under the centralized multiple failure repair framework. We also seek to investigate the achievability of the functional tradeoff under exact repair.

A. Related work

Cooperative regenerating codes (also known as coordinated regenerating codes) have been studied to address the repair of multiple erasures [25], [26] in a distributed manner. In this framework, each replacement node downloads information from d helpers in the first stage. Then, the replacement nodes exchange information between themselves before regenerating the lost nodes. Cooperative regenerating codes that achieve the extreme points on the cooperative tradeoff have been developed; namely, minimum storage cooperative regenerating (MSCR) codes [26]–[28] and minimum bandwidth cooperative regenerating (MBCR) codes [29]. In [30], the authors proved that the interference alignment MSR construction of [8], originally designed for repairing any single node failure, can recover from multiple failures in a cooperative way. Specifically, it is shown that any set of systematic nodes, set of parity-check nodes, or pair of nodes can be repaired cooperatively with optimal bandwidth.

The number of nodes involved in the repair of a single node, known as locality, is another important measure of node repair efficiency [31]. Various bounds and code constructions have been proposed in the literature [31], [32]. Recent works have investigated the problem of multiple node repair under locality constraints [33], [34].

The problem of centralized repair has been considered in [14], in which the authors restricted themselves to MDS codes,

corresponding to the point of minimum storage per node. The authors in [14] showed the existence of MDS codes with optimal repair bandwidth in the asymptotic regime where the storage per node (as well as the entire information) tends to infinity. In [35], the authors proved that Zigzag codes, which are MDS codes designed initially for repairing optimally single erasures [15], can also be used to optimally repair multiple erasures in a centralized manner. In [23], the authors independently proved that multiple failures can be repaired in Zigzag codes with optimal bandwidth. Moreover, [23] defines the minimum bandwidth multi-node repair codes as codes satisfying the property of having the downloaded information $d\beta$ matching the entropy of e nodes[1]. Based on that, the authors derived a lower bound on β for systems having a certain entropy accumulation property and then showed achievability of the minimum bandwidth codes using MBCR codes. However, the optimal storage size per node α is not known under these conditions. In [36], the authors presented an explicit MDS code construction that provides optimal repair for all $e \leq n - k$ and $k \leq d \leq n - e$ simultaneously. It is worth pointing out that the previous constructions are for high-rate codes, with large subpacketization α. The authors in [24] studied the problem of broadcast repair for wireless distributed storage which is equivalent to the model we study in this paper. In [37], the authors presented an approach that enables single erasure MSR codes to recover from multiple failures simultaneously with near-optimal bandwidth. Based on simulations, [37] showed that their approach can provide efficient recovery of most of the failure patterns, but not all of them. The repair problem of Reed Solomon codes has been recently investigated in [38] for single erasure and in [39]–[42] for multiple erasures.

B. Contributions of the paper

The main contributions of this paper are the characterization of functional tradeoff, and the examination of its achievability under exact repair for the extreme points and the interior points. They are summarized as follows.

- We first establish the explicit functional tradeoff between the repair bandwidth and the storage size for functional repair (Theorems 1, 2, 3). We obtain the tradeoff using information flow graphs. From the functional tradeoff, we characterize the minimum storage multi-node repair (MSMR) point, and the minimum bandwidth multi-node repair (MBMR) point.
- When the number of erasures e satisfies $e \geq k$, k being the minimum number of nodes needed to reconstruct the entire data, the tradeoff reduces to a single point, for which we provide an explicit code construction.
- We formalize a construction for exact-repair MSMR codes. Given an instance of an exact linear MSR code, we present a framework to construct an instance of an exact linear MSMR code. We note here that [27] and [37] used a similar approach for MSCR codes and their numerical

---

[1]The definition of minimum bandwidth multi-node repair codes in our paper is simply the minimum bandwidth point on the functional tradeoff, which is different from [23] for $e \nmid k$.

results, respectively. Based on this framework, we study the product-matrix (PM) MSR codes [43] and the interference alignment (IA) MSR construction in [8]. We prove the existence of PM and IA MSMR codes for any number of failures e, $e \leq n-k$ (Theorems 4, 5, 9). Moreover, for the IA code, we prove that the code can always efficiently recover from any set of $e \leq n - k$ node failures as long as the failed nodes are either all systematic nodes or all parity nodes (Theorem 6); for failures including both systematic and parity nodes, we derive explicit design conditions under which exact recovery is ensured, for some particular system parameters (Theorems 7, 8). We note here that unlike previous constructions, our codes are applicable when the code rate is low and they use a small subpacketization size of $\alpha = k - 1$ or k.

- We prove that, to our surprise, functional MBMR point is not achievable for linear exact repair codes for $1 < e < k$ (Theorems 10, 11), while linear codes achieve such point for single erasure [43].
- We show that exact-repair MBCR codes achieve an interior point, that lies near the MBMR point, when $k \equiv 1 \mod e$ (Theorem 12).
- We show that the functional repair tradeoff is not achievable under exact repair for interior points between MBMR and MSMR points, except for maybe a small portion near the MSMR point, for k, d being multiples of e and $k > 2e$ (Theorems 13, 14), which parallels the results for single erasure repair [17]. The achievability of the functional tradeoff under exact repair is summarized in Table I.
- Finally, we study the adaptive repair problem of multiple erasures in MBR codes and present an MBR construction with optimal repair, simultaneously for varying numbers of helpers and varying numbers of erasures (Theorem 15).

C. Organization of the paper

The remainder of the paper is organized as follows. In Section II, we first describe the system model before analyzing the fundamental functional repair tradeoff between the storage size and the repair bandwidth. Section III describes our code construction for the case $e \geq k$, as well as the MSMR codes framework and its application to the product-matrix and the interference alignment codes. We prove the non-achievability of MBMR point under linear exact repair in Section IV. The non-achievability of the interior points under exact repair is investigated in Section V. The adaptive repair of multiple erasures for an MBR code is presented in Section VI and Section VII draws conclusions.

Notation. $[n]$ denotes the set of elements $\{1, \ldots, n\}$. $\lceil \cdot \rceil$ and $\lfloor \cdot \rfloor$ represent the ceiling and the floor functions. For two sets $\mathcal{A}, \mathcal{B}$, $\mathcal{A} \backslash \mathcal{B}$ denotes the set of elements that are in $\mathcal{A}$ but not in $\mathcal{B}$. $|\mathcal{A}|$ denotes the size of $\mathcal{A}$. The symbol $\mathbb{1}_{\{E\}}$ denotes the indicator function of an event E, which is 1 if E is true, and 0 otherwise. The notations $e \mid k$ and $e \nmid k$ are used to denote whether k is a multiple of e, or not, respectively. The superscript t is used to denote the transpose of a matrix. For a matrix A, $|A|$ denotes its determinant and $A_{i,j}$ refers to its

entry at position (i, j). I_n denotes the identity matrix of size n and $\text{diag}\{\lambda_1, \ldots, \lambda_n\}$ denotes the $(n \times n)$ diagonal matrix with the corresponding elements. Vectors are denoted with lower-case bold letters. $\mathbf{u} = [u_1, \ldots, u_m]$ denotes a vector of length m. Note that the notation $[k]$ may refer to a vector of size 1, or the set $\{1, \ldots, k\}$, however the meaning is clear from the context. $\mathbf{e}_i$ denotes the i-th standard basis vector whose dimension is clear from the context.

II. FUNCTIONAL STORAGE-BANDWIDTH TRADEOFF

A. System model

The centralized mutli-node repair problem is characterized by parameters $(\mathcal{M}, n, k, d, e, \alpha, \beta)$. We consider a distributed storage system with n nodes storing $\mathcal{M}$ amount of information. The data elements are distributed across the n storage nodes such that each node can store up to α amount of information. Every node corresponds to a codeword symbol. The system should satisfy the following two properties:

- Reconstruction property: a data collector (DC) connecting to any $k \leq n$ nodes should be able to reconstruct the entire data.
- Regeneration property: upon failure of e nodes, a central node is assumed to contact d helpers, $k \leq d \leq n-e$, and download β amount of information from each of them. New replacement nodes join the system and the content of each is determined by the central node. β is called the repair bandwidth. The total bandwidth is denoted $\gamma = d\beta$.

We consider functional repair and exact repair. In the former case, the replacement nodes are not required to be exact copies of the failed nodes, but the repaired code should again satisfy the above two properties. Our objective is to characterize the tradeoff between the storage per node α and the repair bandwidth β under the centralized multiple failure repair framework. On the optimal functional tradeoff, the minimum bandwidth mutli-node repair point is called *MBMR*, and it has the minimum possible β, while the minimum storage mutli-node repair point is called *MSMR* and has the minimum possible α. When considering exact repair, the minimum storage and minimum bandwidth points may be different from the above functional extreme points. While it has been shown for single erasure that the extreme points match for functional and exact repair, we will show later that MBMR is not achievable under exact repair.

In the paper, we will use the notation $k = \eta e + r$, such that $\eta = \lfloor \frac{k}{e} \rfloor$ and $0 \leq r \leq e - 1$. We now study the fundamental tradeoff between the storage size α and the repair bandwidth β for e erasures under functional repair. We use the technique of evaluating the minimum cut of a multicast information flow graph similar to the single erasure codes [4] and the cooperative regenerating codes [26].

B. Information flow graphs

The performance of a storage system can be characterized by the concept of information flow graphs (IFGs). Our constructed IFG depicts the amount of information transferred, processed and stored during repair. We design our IFG with the

	MSMR point	MBMR point	Interior points
$e = 1$	✓ [8], [15], [43]	✓ [43]	✗, except maybe for a small portion near the MSMR point [17].
$1 < e < k$	✓ [14], [36], [Sections III-B, III-C, III-D]	✗ (for linear codes) [Section IV]	• if $k \equiv 1 \mod e$: an interior point near the MBMR point is achievable [Section IV-D]. • if $e \mid k, e \mid d, k > 2e$: ✗, except maybe for a small portion near the MSMR point [Section V].
$e \geq k$	✓ Section III-A	✓ Section III-A	✓ Section III-A

TABLE I: Summary of achievability results of functional repair tradeoff under exact repair for an $(n, k, d, e, \alpha, \beta)$ distributed storage system. MSMR and MBMR points are defined to be the minimum storage point and the minimum bandwidth point on the functional tradeoff, respectively. Here $e \mid k, e \mid d$ means that k, d are multiples of e. The symbol ✓ denotes achievability while ✗ denotes non-achievability, both of which are under exact repair.

following different kinds of nodes (see Figure 1). It contains a single source node s that represents the source of the data object. Each storage node $x^i, i \in [n]$, of the IFG is represented by two distinct nodes: an input storage node x^i_{in} and an output storage node x^i_{out}. Each output node x^i_{out} is connected to its input node x^i_{in} with an edge of capacity α, reflecting the storage constraint of each individual node. The information flow graph is formed with n initial storage nodes, connected to the source node with edges of capacity ∞. The IFG evolves with time whereupon failure of e nodes, e new nodes simultaneously join the system. Each of the replacement nodes $x^j, j \geq n$, is similarly represented by an input node x^j_{in} and an output node x^j_{out}, linked with an edge of capacity α. To model the centralized repair nature of the system, we add a virtual node $x^i_{virt}, i \geq 1$, that links the d helpers to the new storage nodes. The virtual node x^i_{virt} is connected to the d helpers through d incoming edges each of capacity β. The same node x^i_{virt} is also connected to the input nodes x^j_{in} of the replacement nodes, with edges of capacity ∞. We define a *repair group* to be any set of e nodes that have been repaired simultaneously. In an IFG, a repair group is then associated with the virtual node that performs the repair operation.

Each IFG represents one particular history of the failure patterns. The ensemble of IFGs is denoted by $\mathcal{G}(n, k, d, e, \alpha, \beta)$. For convenience, we drop the parameters whenever it is clear from the context. Given an IFG $G \in \mathcal{G}$, there are $\binom{n}{k}$ different data collectors connecting to k output storage nodes in G with edges of capacity ∞. The set of all data collectors (DCs) nodes in a graph G is denoted by DC(G). For an IFG $G \in \mathcal{G}$ and a data collector $t \in$ DC(G), we partition the nodes of G into two disjoint sets, $U, \bar{U}$, $s \in U, t \in \bar{U}$, and a cut is the sum of the capacities for all edges from nodes in U to nodes in $\bar{U}$. The minimum cut (min-cut) value separating the source node s and the data collector t is denoted by mincut$_G(s, t)$.

C. Network coding analysis

The key idea behind representing the repair problem by an IFG lies in the observation that the repair problem can be cast as a multicast network coding problem [4]. Celebrated results from network coding [5], [6] are then invoked to establish the fundamental limits of the repair problem.

According to the max-flow bound of network coding [5], for a data collector to be able to reconstruct the data, the min-cut separating the source to the data collector should be larger than or equal to the data object size $\mathcal{M}$. Considering all possible data collectors and all possible failure patterns, and assuming that the number of failures/repairs is bounded, the following condition is necessary and sufficient for the existence of centralized multi-node repair codes [4, Proposition 1]

$$\min_{G \in \mathcal{G}} \min_{t \in \text{DC}(G)} \text{mincut}_G(s, t) \geq \mathcal{M}. \quad (2)$$

Analyzing the minimum cut of all IFGs results in the following theorem.

Theorem 1. *For fixed system parameters $(\mathcal{M}, n, k, d, e, \alpha, \beta)$, assuming that the number of failures/repairs is bounded, regenerating codes satisfying the centralized multi-node repair condition exist if and only if*

$$\mathcal{M} \leq \min_{\mathbf{u} \in \mathcal{P}} f(\mathbf{u}), \quad (3)$$

where

$$f(\mathbf{u}) = \sum_{i=1}^{g} \min(u_i \alpha, (d - \sum_{j=1}^{i-1} u_j)\beta), \quad (4)$$

$$\mathcal{P} = \{\mathbf{u} = [u_1, \ldots, u_g] : 1 \leq u_i \leq e, g \in \mathbb{N}, \sum_{i=1}^{g} u_i = k\}. \quad (5)$$

Note that g in (5) corresponds to the support of $\mathbf{u}$, and it satisfies $\lceil \frac{k}{e} \rceil \leq g \leq k$. We call the vector $\mathbf{u} \in \mathcal{P}$ a recovery *scenario*.

Proof: Consider a data collector that connects to a subset of k nodes $\{x^j_{out} : j \in I\}$, where I is the set of k contacted nodes. Then, the reconstruction process can be described by a scenario $\mathbf{u} \in \mathcal{P}$ as follows. The size of the support of $\mathbf{u}$ corresponds to the number of repair groups of size e taking part in the reconstruction process, while u_i corresponds to the number of nodes contacted from repair group i.

We first show that the min-cut minimized over all possible IFGs and over all data collectors is lower bounded by $\min_{\mathbf{u} \in \mathcal{P}} f(\mathbf{u})$. As all incoming edges of DC have infinite capacity, we only examine cuts $(U, \bar{U})$ with $s \in U$ and
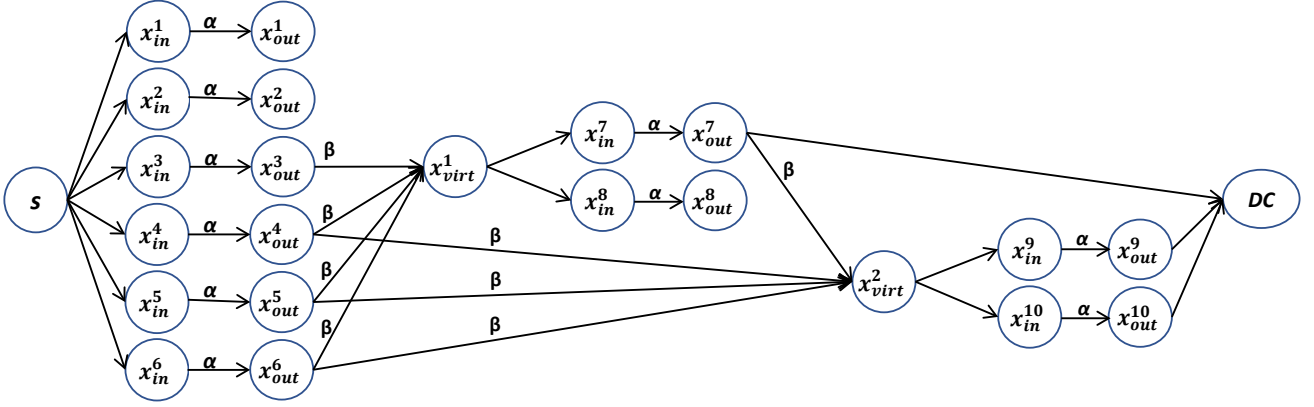
Fig. 1: Example of an information flow graph: $k = 3, d = 4, n = 6, e = 2$. The unlabeled edges have capacity ∞. Nodes 1 and 2 are repaired in the first stage and nodes 3 and 4 are repaired in the second stage. A data collector connecting to any 3 nodes should be able to recover the entire information.

$\{x_{out}^i : i \in I\} \subseteq \bar{U}$. Every directed acyclic graph has a topological sorting, which is an ordering "$<$" of its vertices such that the existence of an edge $x \to y$ implies $x < y$. We recall that nodes within the same repair group are repaired simultaneously, hence it is possible that all input (or output) nodes in a repair group are adjacent in the ordering. We thus order the g repair groups connected to DC according to the sorting. Since nodes are sorted, nodes in the i-th repair group do not have incoming edges from nodes in the j-th repair group, with $j > i, i, j \in [g]$. Considering the i-th repair group, we consider two cases:

- $x_{virt}^i \in U$: as x_{virt}^i is connected with edges of capacity ∞ to x_{in}^j, for j in repair group i, we only consider the case $x_{in}^j \in U, x_{out}^j \in \bar{U}$, for all j in repair group i such that x_{out}^j is connected to the DC. The contribution to the cut is $u_i \alpha$.
- $x_{virt}^i \in \bar{U}$: since the i-th repair group is the topologically i-th repair group, at most $\sum_{j=1}^{i-1} u_j$ edges come from output nodes in $\bar{U}$ and are not part of the cut. The contribution to the cut is at least $(d - \sum_{j=1}^{i-1} u_j)\beta$.

Thus, the contribution of the i-the repair group is at least $\min(u_i \alpha, (d - \sum_{j=1}^{i-1} u_j)\beta)$. Finally, summing all contributions from different repair groups and considering the worst-case for $\mathbf{u} \in \mathcal{P}$ implies that

$$\min_{G \in \mathcal{G}} \min_{t \in \mathrm{DC}(G)} \mathrm{mincut}_G(s,t) \geq \min_{\mathbf{u} \in \mathcal{P}} (\sum_{i=1}^{g} \min(u_i \alpha, (d - \sum_{j=1}^{i-1} u_j)\beta)),$$

with $\mathcal{P}$ defined as in (5).

Now, considering a scenario $\mathbf{u} \in \mathcal{P}$, we show that there exists an information flow graph where the the min-cut is equal to $f(\mathbf{u})$. In this graph, there are initially nodes labeled from 1 to n, and we consider g repair groups (i.e., eg newcomers). Nodes in the i-th repair group are labeled from $n + ie + 1$ to $n + ie$. The i-th repair group connects to nodes $n - d - (\sum_{j=1}^{i-1} u_i) + 1, \ldots, n$, and the first u_j nodes

from repair group j, for $j \leq i - 1$. Figure 1 illustrates the graph for $n = 6, k = 3, d = 4, e = 2, \mathbf{u} = [1, 2]$. Consider a DC that connects to the first u_i nodes from the i-th repair group, for $i \in [g]$. According to the first part of the proof, the min-cut $(U, \bar{U})$ should be as follows. For each $i \in [g]$, if $u_i \alpha \leq (d - \sum_{j=1}^{i-1} u_j)\beta)$, then we include the u_i output nodes of the contacted nodes from the i-th repair group in U and their corresponding output nodes in $\bar{U}$; otherwise, we include x_{virt}^i in $\bar{U}$. Then, this cut $(U, \bar{U})$ achieves $f(\mathbf{u})$. Hence, the min-cut minimized over all graphs and over all data collectors should be equal to $\min_{\mathbf{u} \in \mathcal{P}} f(\mathbf{u})$. The theorem follows according to the necessary and sufficient condition in (2). ∎

Our characterization of Theorem 1 relies on the boundedness assumption of the total number of failures/repairs. A future direction is to investigate the correctness of Theorem 1 for arbitrary number of failures/repairs, similar to [26], [44].

D. Solving the minimum cut problem

In this section, we derive the structure of the optimal scenario $\mathbf{u}$ in (3) for any set of parameters (α, β). For instance, we show that for $me < k \leq (m+1)e$, the number of optimal repair groups g^* (the support of $\mathbf{u}$) is equal to $m + 1$. The result is formalized in the following theorem. Recall that we denote $\eta = \lfloor k/e \rfloor, r = k - \eta e$.

Theorem 2. *For fixed system parameters $(\mathcal{M}, n, k, d, e, \alpha, \beta)$, functional regenerating codes satisfying the centralized multinode repair condition exist if and only if*

$$\mathcal{M} \leq f(\mathbf{u}^*) = \sum_{i=1}^{\lceil \frac{k}{e} \rceil} \min(u_i^* \alpha, (d - \sum_{j=1}^{i-1} u_j^*)\beta),$$

where

$$\mathbf{u}^* = \begin{cases} [k], & \text{if } k \leq e, \\ [\underbrace{e,\ldots,e}_{\eta \text{ times}}], & \text{else if } k = \eta e, \\ [r,\underbrace{e,\ldots,e}_{\eta \text{ times}}], & \text{else if } k = \eta e + r \text{ and } \alpha \leq \frac{d+\eta r - \eta e}{r}\beta, \\ [\underbrace{e,\ldots,e}_{\eta \text{ times}},r], & \text{otherwise,} \end{cases}$$

$$(6)$$

where $0 < r < e$.

Note that $[k]$ in (6) means a vector with a single entry k. We note that [23], [24] have independently developed Theorem 1 or an equivalent of Theorem 1, without entirely characterizing the optimal solution. The authors of [45] independently proved via a different approach Theorem 2, except for the last case in (6).

We denote by $[\mathbf{v}, \mathbf{u}, \mathbf{w}]$ the vector that is the concatenation of the vectors $\mathbf{v}, \mathbf{u}, \mathbf{w}$. The next lemma shows that the minimum cut can be obtained by optimizing any subsequence of $\mathbf{u}$ first. The proof follows directly from the definition of $f()$ in (4) and is omitted.

Lemma 1. *Consider vectors* $\mathbf{v}, \mathbf{w}, \mathbf{u}, \mathbf{u}'$ *such that* $\sum_i u_i = \sum_i u_i'$. *If*

$$f(\mathbf{u}) \geq f(\mathbf{u}'),$$

then,

$$f([\mathbf{v}, \mathbf{u}, \mathbf{w}]) \geq f([\mathbf{v}, \mathbf{u}', \mathbf{w}]).$$

Lemma 2. *Let* α, β *be non-negative reals,* u_1, u_2, d, e, s, l *be non-negative integers such that* $u_1 + u_2 = s \leq e$, *then the following inequality holds*

$$f([u_1, \underbrace{e,\ldots,e}_{l \text{ times}}, u_2]) \geq \min(f([s, \underbrace{e,\ldots,e}_{l \text{ times}}]), f([\underbrace{e,\ldots,e}_{l \text{ times}}, s])),$$

where $f(\mathbf{u})$ *is defined as in (4).*

Proof: To prove the result, we cast it as an optimization problem:

$$\begin{aligned} \underset{\mathbf{u}=[u_1,u_2]}{\text{minimize}} \quad & \min(u_1\alpha, d\beta) + \sum_{i=0}^{l-1}\min(e\alpha, (d-ie-u_1)\beta) \\ & + \min(u_2\alpha, (d-(l+1)e-u_1)\beta) \\ \text{subject to} \quad & 0 \leq u_1 \leq s, \\ & 0 \leq u_2 \leq e, \\ & u_1 + u_2 = s. \end{aligned} \quad (7)$$

Substituting u_2 by $s-u_1$ in (7), using the identity $\min(x,y) = \frac{x+y-|x-y|}{2}$ and after eliminating constant terms, (7) becomes equivalent to

$$\begin{aligned} \underset{u_1}{\text{minimize}} \; & -u_1 l\beta - |u_1\alpha - d\beta| \\ & -\sum_{i=0}^{l-1}|e\alpha - d\beta + ie\beta + u_1\beta| - |s\alpha - u_1(\alpha-\beta) - (d-le)\beta| \\ \text{subject to} \quad & 0 \leq u_1 \leq s. \end{aligned} \quad (8)$$

The objective function in (8), as a function of u_1, is concave over the interval $[0, s]$. The concavity is due to the convexity of $x \to |x|$. Therefore, the minimum is achieved at one of the extreme values. Equivalently, $u_1^* = s$ or $u_1^* = 0$. ∎

Lemma 2 addresses the case $u_1 + u_2 \leq e$. Generalizing it to the case where $e \leq u_1 + u_2 \leq 2e$ follows the same approach.

Lemma 3. *Let* α, β *be non-negative reals,* u_1, u_2, d, e, s, l *be non-negative integers such that* $u_1 + u_2 = e + s$ *and* $0 \leq u_1, u_2, s \leq e$. *Then, the following inequality holds*

$$f([u_1, \underbrace{e,\ldots,e}_{l \text{ times}}, u_2]) \geq \min(f([s, \underbrace{e,\ldots,e}_{l+1 \text{ times}}]), f([\underbrace{e,\ldots,e}_{l+1 \text{ times}}, s])),$$

where $f(\mathbf{u})$ *is defined as in (4).*

Proof: First, we notice that $u_1 = e + s - u_2 \geq s$ as $u_2 \leq e$. Then, the proof follows along similar lines as that of Lemma 2 by replacing the constraint in (8) by $s \leq u_1 \leq e$. ∎

In proving the result of Theorem 2, we first characterize the optimal solution in the case of $k \leq e$. Insight and intuition gained from this case are used to motivate and derive the general optimal solution. We state the following lemma, which represents a key step towards proving our result.

1) Case $k \leq e$: In this scenario, the data collector connecting to k nodes from the same repair group yields the worst-case scenario from an information flow perspective. Given a particular repair scenario characterized by a vector $\mathbf{u}$, for any two adjacent repair groups (i.e., two adjacent entries in $\mathbf{u}$) with u_1 and u_2 nodes respectively, we have $u_1 + u_2 \leq e$. One can combine these two groups into a single repair group to achieve a lower cut value. Indeed, from the cut expression in (3), the contribution of the initial set $[u_1, u_2]$ to the cut is $\min(u_1\alpha, l\beta) + \min(u_2\alpha, (l-u_1)\beta)$, for some non-negative integer l. After combining the groups into a single repair group, the contribution of the newly formed repair group is $\min((u_1 + u_2)\alpha, l\beta)$, which is lower than the initial contribution by virtue of Lemma 2, thus achieving a lower cut. This means that starting from an IFG, we construct a new IFG that has one less repair group and lower min-cut value. This process can be repeated until we end up with a single repair group consisting of $k \leq e$ nodes, which corresponds to the minimum cut over all graphs in this case.

Therefore, the tradeoff in (3) is simply characterized by $\mathcal{M} \leq \min(k\alpha, d\beta)$. Moreover, $\alpha_{\text{MSMR}} = \alpha_{\text{MBMR}} = \frac{\mathcal{M}}{k}$ and $\beta_{\text{MSMR}} = \beta_{\text{MBMR}} = \frac{\mathcal{M}}{d}$. Equivalently, the functional storage bandwidth tradeoff reduces to a single point given by $(\alpha_{\text{MSMR}}, \beta_{\text{MSMR}}) = (\alpha_{\text{MBMR}}, \beta_{\text{MBMR}}) = (\frac{\mathcal{M}}{k}, \frac{\mathcal{M}}{d})$.

2) Case $e < k$: Motivated by the previous case, the intuition is that, according to Lemmas 1, 2, and 3, given a scenario $\mathbf{u}$, one should form a new scenario which exhibits as many groups of size e as possible. Subsequently, one constructs a scenario $\mathbf{u}$ such that all its entries, except maybe one entry, are equal to e.

For a fixed β, we denote the cut corresponding to $\mathbf{u} = [\underbrace{e,\ldots,e}_{j \text{ times}}, r, \underbrace{e,\ldots,e}_{\eta-j \text{ times}}]$, as a function of α, by $C_j(\alpha), j = 0, \ldots, \eta$. As will be shown later in the proof of Theorem 2, a careful analysis of the behavior of the $\eta + 1$ different scenarios $C_j(\alpha), 0 \leq j \leq \eta$, is needed to

determine the overall optimal scenario. We state the result in the following lemma, whose proof is relegated to Appendix A.

Lemma 4. *Assume $e \nmid k$. There exists a real number $\alpha_c(\eta) \in [\frac{d}{e}\beta, \frac{d}{r}\beta]$ such that, for any $0 \leq j \leq \eta$,*

$$C_j(\alpha) \begin{cases} \geq C_0(\alpha), & \text{if } \alpha \leq \alpha_c(\eta), \\ \geq C_\eta(\alpha), & \text{if } \alpha \geq \alpha_c(\eta), \end{cases} \tag{9}$$

with

$$\alpha_c(\eta) = \frac{d + \eta r - \eta e}{r}\beta. \tag{10}$$

Proof of Theorem 2: Now that we have the necessary machinery, we proceed as follows: given any scenario $\mathbf{u}$, we keep combining and/or changing repair groups by means of successive applications of Lemma 2 and Lemma 3 on subsequences of $\mathbf{u}$ until we can no longer reduce the minimum cut. By Lemma 1 we reduced the overall minimum cut. The algorithm terminates because at each step, either the number of repair groups in $\mathbf{u}$ is reduced by one, or the number of repair groups of full size e is increased by one. As the number of repair groups is lower bounded by $\eta + 1$, and as the number of repair groups of full size e is upper bounded by η, the algorithm must terminate after a finite number of steps. It can be seen then that the above reduction procedure has a finite number of outcomes, given by

- $\mathbf{u} = [\underbrace{e, \ldots, e}_{\eta \text{ times}}]$ if $k = \eta e$,
- $\mathbf{u} = [\underbrace{e, \ldots, e}_{j \text{ times}}, r, \underbrace{e, \ldots, e}_{\eta - j \text{ times}}]$ when $k = \eta e + r$, with $0 < r < e$ and $j \in \{0, \ldots, \eta\}$.

Therefore, if $e \mid k$, then the optimal scenario corresponds to considering exactly η repair groups. On the other hand, if $e \nmid k$, then, it is optimal to consider exactly $\eta + 1$ repair groups. However, the optimal position of the repair group with r nodes needs to be determined. Then, using Lemma 4, the result in Theorem 2 follows. ∎

Example 1. *Let $\mathbf{u} = [1, 3, 2, 3, 2]$ with $e = 3$. Then, one can start by reducing the first three repair groups $[1, 3, 2]$. This leads to $\mathbf{u} = [3, 3, 3, 2]$. Another approach would be to consider the last three repair groups $[2, 3, 2]$. Reducing this vector leads to either $\mathbf{u} = [1, 3, 3, 3, 1]$ or $\mathbf{u} = [1, 3, 1, 3, 3]$. Reducing further $\mathbf{u} = [1, 3, 3, 3, 1]$ leads to $\mathbf{u} = [2, 3, 3, 3]$ or $\mathbf{u} = [3, 3, 3, 2]$. Reducing $\mathbf{u} = [1, 3, 1, 3, 3]$ leads to $\mathbf{u} = [3, 2, 3, 3]$ or $\mathbf{u} = [2, 3, 3, 3]$. It remains to compare the cuts given by $\mathbf{u} = [3, 3, 3, 2]$, $\mathbf{u} = [3, 3, 2, 3]$, $\mathbf{u} = [3, 2, 3, 3]$ and $\mathbf{u} = [2, 3, 3, 3]$. Following Theorem 2, either $\mathbf{u} = [2, 3, 3, 3]$ or $\mathbf{u} = [3, 3, 3, 2]$ gives the lowest min-cut.*

E. Explicit expression of the tradeoff

Having characterized the optimal scenario generating the minimum cut in the last section, we are now ready to state the admissible storage-repair bandwidth region for the centralized multi-node repair problem, the proof of which is in Appendix B.
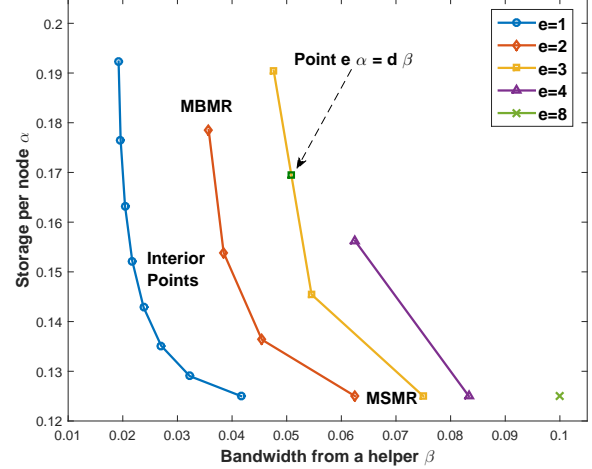

Fig. 2: Multi-node functional repair tradeoff: $k = 8, d = 10, \mathcal{M} = 1, e \in \{1, 2, 3, 4, 8\}$. When $e \nmid k$, the point with $e\alpha = d\beta$ is not the MBMR point.

Theorem 3. *For an $(\mathcal{M}, n, k, d, e, \alpha, \beta)$ storage system with total bandwidth $\gamma = d\beta$, there exists a threshold function $\alpha^*(\mathcal{M}, n, k, d, e, \gamma)$ such that for any $\alpha \geq \alpha^*(\mathcal{M}, n, k, d, e, \gamma)$, regenerating codes exist. For any $\alpha < \alpha^*(\mathcal{M}, n, k, d, e, \gamma)$, it is impossible to construct codes achieving the target parameters. The threshold function $\alpha^*(\mathcal{M}, n, k, d, e, \gamma)$ is defined as follows:*

- *if $k \leq e$, then: $\alpha^* = \frac{\mathcal{M}}{k}$, $\gamma \in [\mathcal{M}, +\infty)$,*
- *if $k = \eta e, \eta \geq 2$, then:*

$$\alpha^* = \begin{cases} \frac{\mathcal{M}}{k}, & \gamma \in [f_0(\eta-1), +\infty), \\ \frac{\mathcal{M} - \gamma g_0(i)}{ie}, & \gamma \in [f_0(i-1), f_0(i)], i = \eta - 1, \ldots 1, \end{cases}$$

- *if $k = \eta e + r$ with $\eta \geq 1, 1 \leq r \leq e - 1$, then:*

$$\alpha^* = \begin{cases} \frac{\mathcal{M}}{k}, & \gamma \in [f_r(\eta-1), +\infty), \\ \frac{\mathcal{M} - \gamma g_r(i)}{r + ie}, & \gamma \in [f_r(i-1), f_r(i)], i = \eta - 1, \ldots 1, \\ \frac{\mathcal{M} - \gamma g_r(0)}{r}, & \gamma \in [\frac{d\mathcal{M}}{(\eta+1)d - e\binom{\eta+1}{2}}, f_r(0)], \end{cases} \tag{11}$$

where

$$f_r(i) = \frac{2ed\mathcal{M}}{-k^2 - r^2 + e(k - r) + 2kd - e^2(i^2 + i) - 2ier}, \tag{12}$$

$$g_r(i) = \frac{(\eta - i)(-2r + e + 2d - \eta e - ei)}{2d}.$$

The functional repair tradeoff is illustrated in Figure 2 for multiple values of $e \in \{1, 2, 3, 4, 8\}$ and $k = 8, d = 10, \mathcal{M} = 1$.

Remark 1. *In the case of $e|k, e|d$, the following equality holds for all points on the tradeoff*

$$\mathcal{M} = \sum_{i=0}^{\eta-1} \min(e\alpha, (d - ie)\beta) \iff \frac{\mathcal{M}}{e} = \sum_{i=0}^{\eta-1} \min(\alpha, (\frac{d}{e} - i)\beta).$$

Therefore, the tradeoff between α and β is the same as the

single erasure tradeoff of a system with reduced parameters given by $\frac{\mathcal{M}}{e}$, $\frac{k}{e} = \eta$ *and* $\frac{d}{e}$. *The expression of the tradeoff in this case can be recovered from [4] with the appropriate parameters.*

We next present the expressions of the two extreme points on the optimal tradeoff. We focus on the case $e < k$, as otherwise the optimal tradeoff reduces to a single point.

MSMR. The MSMR point is the same irrespective of the relation between k and e, and it is given by

$$\alpha_{\text{MSMR}} = \frac{\mathcal{M}}{k}, \gamma_{\text{MSMR}} = \frac{\mathcal{M}}{k}\frac{ed}{d-k+e}. \tag{13}$$

MBMR. Interestingly, the MBMR point depends on whether e divides k or not.

- If $k = \eta e$, we obtain

$$\gamma_{\text{MBMR}} = \frac{2ed\mathcal{M}}{-k^2 + ek + 2kd} = \frac{d\mathcal{M}}{d\eta - e\binom{\eta}{2}}, \tag{14}$$

$$\alpha_{\text{MBMR}} = \frac{\gamma_{\text{MBMR}}}{e}. \tag{15}$$

The amount of information downloaded for repair is equal to the amount of information stored at the e replacement nodes. This property of the MBMR point is similar to the minimum bandwidth point in the single erasure case [4] and also the minimum bandwidth cooperative repair point [26].

- If $k = \eta e + r$, we obtain

$$\gamma_{\text{MBMR}} = \frac{2ed\mathcal{M}}{(k-r+e)(2d-k+r)} = \frac{d\mathcal{M}}{d(\eta+1) - e\binom{\eta+1}{2}}, \tag{16}$$

$$\alpha_{\text{MBMR}} = \gamma_{\text{MBMR}}\frac{d + \eta r - e\eta}{rd}. \tag{17}$$

This situation is novel for multiple erasures as the e nodes need to store more than the overall downloaded information. This is an extra cost in order to achieve the low value of the repair bandwidth. Figure 2 illustrates this situation with $e = 3, k = 8$. However, later we will see that for both $e|k$ and $e \nmid k$, the total bandwidth at MBMR is equal to the entropy of the failed nodes (see Lemma 6 and Lemma 11):

$$H(W_E) = d\beta = \gamma,$$

where $E \subset [n]$ is any subset of nodes of size e and W_E is the information stored across the nodes in E.

Remark 2. *From the statement of Theorem 2, we note that if we only consider points between the MSMR and the MBMR points, then the scenario* $\mathbf{u} = [r, e, \ldots, e]$ *always generates the lowest cut. In fact, the scenario* $\mathbf{u} = [e, \ldots, e, r]$ *corresponds to points beyond the MBMR point, namely, points with* $\alpha \geq \alpha_{MBMR}, \beta = \beta_{MBMR}$.

Centralized strategy v.s. separate strategy. We compare the centralized repair scheme repairing e nodes to a separate strategy repairing each of the e nodes separately using single erasure regenerating codes. We fix k, α and $\mathcal{M}$.

Case I: both strategies use d helpers. The separate strategy requires a total bandwidth given by $ed\beta_1$, while the centralized repair requires $d\beta_e$, where the subscript indicates the number

of erasures repaired at a time. For simplicity, we assume that $e \mid k$. The case $e \nmid k$ can be treated in a similar way. For points on the multi-node repair tradeoff, we have

$$\mathcal{M} = \sum_{j=0}^{\eta-1} \min(e\alpha, (d - je)\beta_e).$$

Consider a point with the same $\alpha, k, d, \mathcal{M}$ on the single erasure tradeoff, we write

$$\mathcal{M} = \sum_{j=0}^{\eta-1} \min(e\alpha, (d - je)\beta_e) = \sum_{j=0}^{k-1} \min(\alpha, (d-j)\beta_1)$$

$$= \sum_{j=0}^{\eta-1}\sum_{i=0}^{e-1} \min(\alpha, (d - i - je)\beta_1)$$

$$\leq \sum_{j=0}^{\eta-1} e \min(\alpha, (d - je)\beta_1) = \sum_{j=0}^{\eta-1} \min(e\alpha, (d-je)e\beta_1).$$

It follows that $\beta_e \leq e\beta_1$ with equality if and only if $e = 1$. Therefore, for any storage capacity α, multi-node repair requires strictly less bandwidth than a separate strategy for the same number of helpers d.

Case II: multi-node repair uses $d - e + 1$ helpers, and separate repair uses d helpers. In this case, the original number of available nodes that can serve as helpers is assumed to be d, and $e \geq 1$ erasures occur within the available nodes. Then a separate strategy may require a smaller bandwidth for some values of α, as illustrated by Figure 3. However, as d is sufficiently large, we observe numerically that multi-node repair with $d - e + 1$ helpers performs better than a separate strategy for all values of α. Moreover, for the MSMR point, the separate repair bandwidth is $ed\beta_{1,\text{MSR}} = e\frac{\mathcal{M}}{k}\frac{d}{d-k+1}$, and centralized repair bandwidth is $(d-e+1)\beta_{e,\text{MSMR}} = \frac{\mathcal{M}}{k}\frac{e(d-e+1)}{d-k+1}$. It follows that a centralized repair is always better that a separate repair strategy, specifically, for $e > 2$,

$$\frac{(d - e + 1)\beta_{e,\text{MSMR}}}{ed\beta_{1,\text{MSR}}} = \frac{d - (e-1)}{d} < 1.$$

III. Exact-repair MSMR codes constructions

In the remainder of the paper, we study exact repair. In this section, we first analyze the case $e \geq k$ and then construct MSMR codes when $e < k$. In later sections, we study the feasibility of MBMR codes and the interior points under exact repair for $e < k$.

A. Construction when $k \leq e$

In the case of $k \leq e$, the optimal tradeoff reduces to a single point, so our MSMR construction in this section is also an MBMR code. The optimal parameters satisfy $\alpha = \frac{\mathcal{M}}{k}, \beta = \frac{\mathcal{M}}{d}$ and $\gamma = \mathcal{M}$. We note that the overall repair bandwidth $d\beta$ and the reconstruction bandwidth $k\alpha$ are the same. Therefore, one can achieve α and γ by dividing the data into k symbols and encoding them using an (n, k) MDS code (for example, a Reed-Solomon code). The repair can be done by downloading the full content of any k out of d helpers while not using $d-k$ helpers. Such repair is asymmetric in nature. We describe one alternative approach for achieving the repair with equal contribution from d helpers.
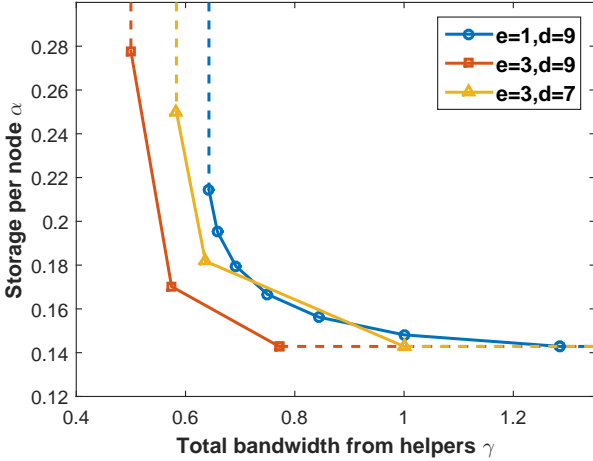
Fig. 3: Centralized multi-node repair vs separate repair strategy: $k = 7, \mathcal{M} = 1$. Separate repair strategy uses $d = 9$ to repair 3 nodes successively while multi-node repair is plotted for $d = 7$ and $d = 9$.

1) Divide the original file into kd symbols (that is $\mathcal{M} = kd$) and encode them using an (nd, kd) MDS code.
2) Store the encoded symbols at n nodes, such that each node stores $\alpha = d$ encoded symbols.
3) For reconstruction, from any k nodes, we obtain kd different symbols. By virtue of the MDS property, we can reconstruct the data.
4) For repair, each helper node transmits any $\beta = \frac{\mathcal{M}}{d} = k$ symbols. The replacement nodes receive dk different coded symbols, which are sufficient to reconstruct the whole data and thus regenerate the missing symbols.

Remark 3. *The above procedure works for a specific predetermined d. However, it can be generalized to support any value of d satisfying $k \leq d \leq n - e$. For instance, let $\delta = lcm(k, k + 1, k + 2, \ldots, n - e)$ (lcm denotes the least common multiple). Assume $\mathcal{M} = k\delta$. The file of size $\mathcal{M}$ is then encoded using an $(n\delta, k\delta)$ MDS code. Each node stores $\alpha = \frac{\mathcal{M}}{k} = \frac{k\delta}{k} = \delta$ coded symbols. To repair with d helpers, for any $k \leq d \leq n-e$, each node transmits any $\beta = \frac{\mathcal{M}}{d} = \frac{k\delta}{d}$ coded symbols for his node. Similarly, it can be seen that reconstruction is always feasible.*

B. Minimum storage codes framework

In the following subsections, we discuss an explicit MSMR code construction method using existing MSR codes designed for single failures for $k > e$. We first describe the general framework, and then present two specific codes. We denote the code parameters by $(n, k, d, e, \alpha, \gamma)$.

The framework described in this section has been developed in [37] for numerical simulations. We present it here in a formal and analytical way. Consider an instance of an exact linear (n, k, d, α, β) MSR code, where $\beta = \frac{\alpha}{d-k+1}$. Consider e nodes, indexed by $f_1, \ldots, f_e$, and other distinct $d - e + 1$ nodes, indexed by $h_1, \ldots, h_{d-e+1}$, such that $d-e+1 \geq k$. Let $\mathcal{H} = \{f_1, \ldots, f_e, h_1, \ldots, h_{d-e+1}\}$ and define $\mathcal{H}_{f_j} = \mathcal{H} \backslash \{f_j\}$.

Consider the single-node repair algorithm corresponding to failed node f_j and helper nodes $\mathcal{H}_{f_j}$. We denote by $\mathbf{s}_{h,f_j}^{\mathcal{H}}$ the information sent by node h to repair node f_j, for helpers $h \in \mathcal{H}_{f_j}$. We drop the superscript $\mathcal{H}$ when it is clear from the context. The size of $\mathbf{s}_{h,f_j}$ is β symbols.

Now we construct an $(n, k, d-e+1, e, \alpha, e\beta)$ MSMR code. Upon failure of the e nodes $f_1, \ldots, f_e$, the centralized node carrying the repair connects to the set of $d - e + 1$ helpers $h_1, \ldots, h_{d-e+1}$. Each helper node h_i transmits $e\beta$ symbols given by $\cup_{j=1}^{e}\{\mathbf{s}_{h_i,f_j}\}$. One can check that the parameters of an MSMR code in (13) are satisfied with equality.

The approach consists in using the underlying MSR repair procedure for each of the e failed nodes. Note that $\mathbf{s}_{h_i,f_j}$ can be obtained from the $d - e + 1$ helpers, for $i \in [d - e + 1]$. To this end, the MSR repair procedure requires $\mathbf{s}_{f_i,f_j}^{\mathcal{H}}$ for all $\{(i,j) : i, j \in [e], i \neq j\}$, which we treat as unknowns. Let $E_{i,j}(\cdot)$ denote the encoding function used to encode the information sent from node h_i to node f_j. Also, let $D_i(\cdot)$ denote the decoding function used by the MSR code to repair node f_i given information from d helpers. Then, we write

$$\mathbf{s}_{f_i,f_j}^{\mathcal{H}} = E_{i,j}(\mathbf{w}_{f_i})$$
$$= E_{i,j}(D_i(\mathbf{s}_{h,f_i}^{\mathcal{H}}, h \in \mathcal{H}_{f_i})), \quad (18)$$

where $\mathbf{w}_j$ denotes the content of node j, and $i, j \in [e], i \neq j$. Equation (18) generates $e(e-1)\beta$ linear equations in $e(e-1)\beta$ unknowns. Let $\mathbf{s}$ be a vector containing the unknowns $\mathbf{s}_{f_i,f_j}$. Then, we seek to form a system of linear equations as

$$A\mathbf{s} = \mathbf{b}, \quad (19)$$

where A is a known $(e(e - 1)\beta \times e(e - 1)\beta)$ matrix and b is a known $(e(e-1)\beta \times 1)$ vector. If A is non-singular, one can thus recover $\mathbf{s}$. Then, the centralized node can recover the failed node $\mathbf{w}_{f_i}$ as $\mathbf{w}_{f_i} = D_i(\mathbf{s}_{h,f_i}, h \in \mathcal{H}_{f_i})$. We adopt the above framework throughout the section.

Remark 4. *While the described framework applies to codes with arbitrary rates, we focus in the sequel on low-rate codes. High-rate MSMR constructions have been presented in [36]. However, in the low-rate regime, our constructions perform better. For instance, for a target MSMR code with rate $\frac{1}{2}$, the construction in [36] yields a storage size $\alpha = k^{2k-1}$, while applying the above approach to IA codes [8] or to PM codes [43] results in a smaller storage size $\alpha = k$ and $\alpha = k - 1$, respectively.*

C. Product-matrix codes

In this subsection, we construct MSMR codes for any e erasures based on product-matrix (PM) codes [43]. The PM framework allows the design of MBR codes for any value of d and the design of MSR codes for $d \geq 2k - 2$. Moreover, the PM construction offers simple encoding and decoding and ensures optimal repair of all nodes. Product-matrix MSR codes are a family of scalar MSR codes, i.e., $\beta = 1$. We first focus on the case $d = 2k-2$. Under this setup, $\alpha = d-k+1 = k-1$, $\mathcal{M} = k(k - 1)$. The codeword is represented by an $(n \times \alpha)$

code matrix C such that its i-th row corresponds to the α symbols stored by the i-th node. The code matrix is given by

$$C = \Psi M, \Psi = \begin{bmatrix} \Phi & \Lambda\Phi \end{bmatrix}, M = \begin{bmatrix} S_1 \\ S_2 \end{bmatrix},$$

where Ψ is an $(n \times d)$ encoding matrix and M is a $(d \times \alpha)$ message matrix. S_1 and S_2 are $(\alpha \times \alpha)$ symmetric matrices constructed such that the $\binom{\alpha+1}{2}$ entries in the upper-triangular part of each of the two matrices are filled up by $\binom{\alpha+1}{2}$ distinct file symbols. Φ is an $(n \times \alpha)$ matrix and Λ is an $(n \times n)$ diagonal matrix. The elements of Ψ should satisfy:

1) any d rows of Ψ are linearly independent;
2) any α rows of Φ are linearly independent;
3) the n diagonal elements of Λ are distinct.

The above conditions may be met by choosing Ψ to be a Vandermonde matrix, in which case its ith row is given by $\psi_i^t = \begin{bmatrix} 1 & \lambda_i & \cdots & \lambda_i^{d-1} \end{bmatrix}$. It follows that $\Lambda = \text{diag}\{\lambda_1^\alpha, \ldots, \lambda_n^\alpha\}$ for some coefficients $\lambda_1, \lambda_2, \ldots, \lambda_n$. In the following, we assume that Ψ is a Vandermonde matrix.

Repair of a single erasure in PM codes. The single erasure repair algorithm [43] is reviewed below. Let $\mathbf{w}_i^t$ denote the content stored at a failed node. Let ϕ_i^t be the ith row of Φ. Then, $\mathbf{w}_i^t = \psi_i^t M = \begin{bmatrix} \phi_i^t & \lambda_i^\alpha \phi_i^t \end{bmatrix} M = \phi_i^t S_1 + \lambda_i^\alpha \phi_i^t S_2$. Let $\mathcal{H}_i = \{h_1, \ldots, h_d\}$ denote the set of d helpers. Each helper h transmits $s_{h,i} = \mathbf{w}_h^t \phi_i = \psi_h^t M \phi_i$ to the replacement node, who obtains $\Psi_{\mathcal{H}_i} M \phi_i$, where $\Psi_{\mathcal{H}_i}^t = \begin{bmatrix} \psi_{h_1} & \cdots & \psi_{h_d} \end{bmatrix}$.

Note that $\Psi_{\mathcal{H}_i}$ is invertible by construction. Thus, using the symmetry of S_1 and S_2, we obtain $(M\phi_i)^t = \begin{bmatrix} \phi_i^t S_1 & \phi_i^t S_2 \end{bmatrix}$. We can then reconstruct $\mathbf{w}_i^t = \phi_i^t S_1 + \lambda_i^\alpha \phi_i^t S_2$.

Repair of multiple erasures in PM codes. Given the symmetry of PM codes, we can assume w.l.o.g that nodes in $\mathcal{E} = \{1, \ldots, e\}$ have failed. Define $\mathcal{E}_i = \mathcal{E}\setminus\{i\}$. Let $\mathcal{H} = \{1, \ldots, d+1\}$. The centralized node connects to helper node $h \in \{e+1, \ldots, d+1\}$, and obtains $\{\psi_h^t M\phi_j, j \in \mathcal{E}\}$.

Let $\mathbf{s} = [s_{1,2}, s_{2,1}, \ldots, s_{1,e}, s_{e,1}, \ldots, s_{e-1,e}, s_{e,e-1}]^t$. Our goal is to express explicitly A and $\mathbf{b}$ as in (19).

Consider the repair of node $i \in \mathcal{E}$ by the set of helpers in $\mathcal{H}_i = \mathcal{H}\setminus\{i\}$. From the single-node repair, we write

$$\mathbf{w}_i = \begin{bmatrix} I_\alpha & \lambda_i^\alpha I_\alpha \end{bmatrix} \Psi_{\mathcal{H}_i}^{-1} \mathbf{s}_{\mathcal{H}_i}, \text{ such that}$$
$$\Psi_{\mathcal{H}_i}^t = \begin{bmatrix} \psi_1 & \cdots & \psi_{i-1} & \psi_{i+1} & \cdots & \psi_{d+1} \end{bmatrix},$$
$$\mathbf{s}_{\mathcal{H}_i}^t = \begin{bmatrix} s_{1,i} & \cdots & s_{i-1,i} & s_{i+1,i} & \cdots & s_{d+1,i} \end{bmatrix}.$$

It follows that

$$\begin{aligned} s_{i,j} &= \phi_j^t \mathbf{w}_i \\ &= \begin{bmatrix} \phi_j^t & \lambda_i^\alpha \phi_j^t \end{bmatrix} \Psi_{\mathcal{H}_i}^{-1} (\sum_{l \in \mathcal{H}_i} s_{l,i} \mathbf{e}_{l,i}) \\ &= \sum_{l \in \mathcal{E}_i} (\begin{bmatrix} \phi_j^t & \lambda_i^\alpha \phi_j^t \end{bmatrix} \Psi_{\mathcal{H}_i}^{-1} \mathbf{e}_{l,i}) s_{l,i} \\ &\quad + \sum_{l=e+1}^{d+1} (\begin{bmatrix} \phi_j^t & \lambda_i^\alpha \phi_j^t \end{bmatrix} \Psi_{\mathcal{H}_i}^{-1} \mathbf{e}_{l,i}) s_{l,i}, \end{aligned} \quad (20)$$

Here, for $l \in [d+1]\setminus\{i\}$, we use the column standard basis $\mathbf{e}_l$ and define

$$\mathbf{e}_{l,i} \triangleq \begin{cases} \mathbf{e}_l, & l < i, \\ \mathbf{e}_{l-1}, & l > i. \end{cases}$$

Note that the second term in (20) is known from the helpers. Moreover, to compute (20), one may use the inverse of Vandermonde's matrix formula [46]. Let $h \in \{1, \ldots, d\}$, we have

$$(\Psi_{\mathcal{H}_i}^{-1} \mathbf{e}_{l,i})_h = \frac{\gamma_h(l,i)}{\prod_{m \in \mathcal{H}_i\setminus\{l\}}(\lambda_l - \lambda_m)} = \frac{\gamma_h(l,i)}{\sum_{j=1}^d \gamma_j(l,i)\lambda_l^{j-1}}, \quad (21)$$

where the subscript h in $(\cdot)_h$ means the h-th entry, and

$$\gamma_h(l,i) = (-1)^{d-h} \sum_{m_1 < \ldots < m_{d-h} \in \mathcal{H}_i\setminus\{l\}} \lambda_{m_1} \ldots \lambda_{m_{d-h}}. \quad (22)$$

We obtain

$$\begin{bmatrix} \phi_j^t & \lambda_i^\alpha \phi_j^t \end{bmatrix} \Psi_{\mathcal{H}_i}^{-1} \mathbf{e}_{l,i} = \frac{\sum_{h=1}^\alpha (\gamma_h(l,i) + \lambda_i^\alpha \gamma_{h+\alpha}(l,i))\lambda_j^{h-1}}{\sum_{h=1}^d \gamma_h(l,i)\lambda_l^{h-1}}. \quad (23)$$

Therefore, one can construct A and $\mathbf{b}$ in (19) as follows:

- The entries of $\mathbf{b}$ are indexed with (i,j), corresponding to $s_{i,j}$. The entry of $\mathbf{b}$ at index (i,j) is given by $\sum_{l=e+1}^{d+1} (\begin{bmatrix} \phi_j^t & \lambda_i^\alpha \phi_j^t \end{bmatrix} \Psi_{\mathcal{H}_i}^{-1} \mathbf{e}_{l,i}) s_{l,i}$.
- Index the $e(e-1)$ rows (and columns respectively) of A with (i,j). A has zero in all entries except: For every row in A indexed by (i,j):
  - the entry at column indexed by (i,j) is -1.
  - for $l \in \mathcal{E}_i$, the entry at column indexed by (l,i) is given by $\begin{bmatrix} \phi_j^t & \lambda_i^\alpha \phi_j^t \end{bmatrix} \Psi_{\mathcal{H}_i}^{-1} \mathbf{e}_{l,i}$ as in (23).

For clear presentation, we first prove the existence of product-matrix MSMR codes for 2 erasures, and then prove the result for general e.

Theorem 4. *There exists $(n, k, 2k-3, 2, k-1, 2)$ product-matrix MSMR codes, defined over a large enough finite field, such that any two erasures can be optimally repaired.*

Proof: In this case, the matrix A of (19) is given by

$$A = \begin{bmatrix} -1 & \begin{bmatrix} \phi_2^t & \lambda_1^\alpha \phi_2^t \end{bmatrix} \Psi_{\mathcal{H}_1}^{-1} \mathbf{e}_{2,1} \\ \begin{bmatrix} \phi_1^t & \lambda_2^\alpha \phi_1^t \end{bmatrix} \Psi_{\mathcal{H}_2}^{-1} \mathbf{e}_{1,2} & -1 \end{bmatrix}.$$

From (21), noting that $\mathcal{H}_1\setminus\{2\} = \mathcal{H}_2\setminus\{1\}$, we obtain the determinant of A to be

$$\begin{aligned} |A| &= 1 - \begin{bmatrix} \phi_2^t & \lambda_1^\alpha \phi_2^t \end{bmatrix} \Psi_{\mathcal{H}_1}^{-1} \mathbf{e}_{2,1} \begin{bmatrix} \phi_1^t & \lambda_2^\alpha \phi_1^t \end{bmatrix} \Psi_{\mathcal{H}_2}^{-1} \mathbf{e}_{1,2} \\ &= 1 - \frac{(\sum_{h=1}^\alpha \lambda_2^{h-1}(\gamma_h(1,2) + \lambda_1^\alpha \gamma_{h+\alpha}(1,2)))}{\sum_{h=1}^d \lambda_2^{h-1}\gamma_h(1,2) \sum_{h=1}^d \lambda_1^{h-1}\gamma_h(1,2)} \\ &\quad \times (\sum_{h=1}^\alpha \lambda_1^{h-1}(\gamma_h(1,2) + \lambda_2^\alpha \gamma_{h+\alpha}(1,2))) \\ &\triangleq 1 - \frac{N(\lambda_1, \ldots, \lambda_{d+1})}{D(\lambda_1, \ldots, \lambda_{d+1})}. \end{aligned}$$

$|A|$ can be viewed as a rational function of $(\lambda_1, \ldots, \lambda_{d+1})$, as N and D are polynomials in $(\lambda_1, \ldots, \lambda_{d+1})$. We want to show that the following polynomial is not identically zero:

$$P(\lambda_1, \ldots, \lambda_{d+1}) \triangleq D(\lambda_1, \ldots, \lambda_{d+1})|A|$$

$$= D(\lambda_1, \ldots, \lambda_{d+1}) - N(\lambda_1, \ldots, \lambda_{d+1}).$$

Let $y_\alpha = (-1)^\alpha \lambda_3 \cdots \lambda_{\alpha+2}$, $y_{\alpha-1} = (-1)^{\alpha-1} \lambda_3 \cdots \lambda_{\alpha+1}$. Then, it can be seen that P contains the term

$$y_\alpha y_{\alpha-1} (\lambda_1^{d-1} + \lambda_2^{d-1} - \lambda_1^\alpha \lambda_2^{\alpha-1} - \lambda_2^\alpha \lambda_1^{\alpha-1}),$$

which is not zero. Hence, $P(\lambda_1, \ldots, \lambda_{d+1})$ is a non-zero polynomial. The PM construction, when based on a Vandermonde matrix, requires $\lambda_i^\alpha \neq \lambda_j^\alpha$ [43], or equivalently, $g(\lambda_1, \lambda_2) \triangleq \lambda_i^\alpha - \lambda_j^\alpha \neq 0$. Let $Q(\lambda_1, \ldots, \lambda_n)$ denote the polynomial obtained by varying the set of helpers and failure patterns, taking the product of all corresponding polynomials P, and also multiplied by all g for all pairs of two nodes. Then, Q is not identically zero. By Combinatorial Nullstellensatz [47], we can find assignments of the variables $\{\lambda_1, \ldots, \lambda_n\}$ over a large enough finite field, such that the polynomial is not zero. Equivalently, we can guarantee the successful optimal repair of any two erasures among the n storage nodes. ∎

Theorem 5. *There exists $(n, k, 2k-e-1, e, k-1, e)$ product-matrix MSMR codes, defined over a large enough finite field, such that any e erasures can be optimally repaired.*

Proof: We only consider $e > 2$, as the case of $e = 2$ is covered by Theorem 4. Entries in each column indexed by $s_{i,j}$ in A is either -1 or some other $(e-1)$ non-zero entries whose denominator is the same and given by $\prod_{m \in \mathcal{H} \setminus \{i\}} (\lambda_i - \lambda_m)$. We multiply this common denominator to all entries in the column $s_{i,j}$, for all pairs $i \neq j$. When λ_i's are chosen to be distinct, this does not change the singularity of A. Denote this transformed matrix by B. Using (23), the entry of B in row (i, j) and column (l, m) is a polynomial in $\lambda_1, \ldots, \lambda_{d+1}$:

$$B_{(i,j),(l,m)} =$$
$$\begin{cases} -\sum_{h=1}^d \gamma_h(i, j) \lambda_i^{h-1}, & l = i, m = j, \\ \sum_{h=1}^\alpha \left(\gamma_h(l, i) \lambda_j^{h-1} + \gamma_{h+\alpha}(l, i) \lambda_i^\alpha \lambda_j^{h-1} \right), & m = i, \\ 0, & \text{otherwise.} \end{cases}$$

Notice that $e + \alpha - 1 \leq k - 1 + \alpha - 1 = d - 1$. Let $y = (-1)^{\alpha-1} \lambda_{e+1} \cdots \lambda_{e+\alpha-1}$, which is a term in $\gamma_{\alpha+1}(i, j)$ for all (i, j) by (22). We observe that there is a single term $\pm y \lambda_i^\alpha$ in the polynomial $B_{(i,j),(l,m)}$ for the non-zero entries of B.

Recall that the Leibniz formula for determinant of a $(m \times m)$ matrix B is given by

$$|B| = \sum_\sigma \text{sgn}(\sigma) \prod_i b_{\sigma(i),i},$$

where σ is a permutation from the permutation group S_m, sgn is the sign function of permutations, and $b_{i,j}$ is the entry (i, j) of B.

Claim 1. The term $T = \prod_{i=1}^e (y \lambda_i^\alpha)^{e-1}$ in $|B|$ has a non-zero coefficient.

Claim 1 implies that $|B|$ is not a zero polynomial. Then, proceeding as in the proof in Theorem 4, by Combinatorial Nullstellensatz [47], we can find assignments of the variables $\{\lambda_1, \ldots, \lambda_n\}$ over a large enough finite field, such that the code guarantees optimal repair of any set of e erasures.

Next, we prove Claim 1. Note that the term T can be created if and only if we take the single term $\pm y \lambda_i^\alpha$ in the non-zero

entries of B (depending on the permutation σ). Therefore, it is easy to see that the coefficient of term T in $|B|$ is the determinant of the following $(e(e-1) \times e(e-1))$ matrix C

$$C_{(i,j),(l,m)} = \begin{cases} -1, & l = i, m = j, \\ 1, & m = i, \\ 0, & \text{otherwise.} \end{cases}$$

One can verify that C is diagonalizable, and the eigenvalues satisfy:

- Eigenvalue $e - 2$ has multiplicity 1, with the corresponding (right) eigenvector $(1, 1, \ldots, 1)^t$.
- Eigenvalue -2 has multiplicity $e-1$, with the corresponding eigenspace $\{(x_{1,2}, \ldots, x_{e,e-1})^t : x_{1,j} = x_{1,2}, \forall j \in [e] \setminus \{1\}, x_{i,j} = x_{i,1}, \forall i \in [e] \setminus \{1\}, \forall j \in [e] \setminus \{i\}, x_{1,2} + \sum_{i=2}^e x_{i,1} = 0\}$ of dimension $e - 1$.
- Eigenvalue -1 has multiplicity $e(e - 2)$, with the corresponding eigenspace $\{(x_{1,2}, \ldots, x_{e,e-1})^t : \sum_{1 \leq i \leq e, i \neq j} x_{i,j} = 0, \forall j \in [e]\}$ of dimension $e(e - 2)$.

To ensure that $|C| \neq 0$, a sufficient condition is to require the finite field to have a characteristic such that the eigenvalues $\{e - 2, -2, -1\}$ are non-zero and $|C| \neq 0$. Therefore, Claim 1 is proved and the theorem statement follows. ∎

Remark 5. *There exists product-matrix MSMR codes, defined over a large enough finite field, that simultaneously repair any $e \in [n - k]$ erasures with optimal bandwidth. Indeed, let $\widetilde{Q} = \prod_{e=2}^{n-k} Q_e$, where Q_e is the polynomial corresponding to the code constraints for e erasures such that matrix A in (19) is invertible. Recall that the reconstruction process for PM codes requires that $\alpha_i^\alpha - \alpha_j^\alpha \neq 0$ for $\alpha_i \neq \alpha_j$. Let $g(\lambda_1, \ldots, \lambda_n) = \prod_{1 \leq i < j \leq n} (\lambda_j^\alpha - \lambda_i^\alpha)$. Let $Q(\lambda_1, \ldots, \lambda_n) = g(\lambda_1, \ldots, \lambda_n) \widetilde{Q}(\lambda_1, \ldots, \lambda_n)$. By Theorem 4 and Theorem 5, Q is not zero and the result follows by Combinatorial Nullstellensatz.*

We note that the sufficient condition that matrix C is invertible in the proof of Theorem 5 is not necessary for the existence of PM codes. Indeed, as it will be shown in Example 2, we can construct PM codes with optimal multi-node repair property over finite fields of characteristic 2.

Example 2. *Consider the product-matrix code with $n = 11, k = 6, d = 10, \alpha = 5$. The code is defined over $\mathbb{F}_{2^6}$ with $\Lambda = \text{diag}\{\lambda_1^\alpha, \ldots, \lambda_{11}^\alpha\}$ and $\lambda_i = g^{i-1}$ with g being the generator of the multiplicative group of $\mathbb{F}_{2^6}$. Recall that with the above choice of λ_i, any field of size at least $n\alpha = 55$ is sufficient to meet the PM code requirements [43]. We first consider repair of $e = 2$ erasures. One can check that out of the $\binom{11}{2} = 55$ possible 2 failure patterns, 2 patterns are not recoverable according to (19): $\mathcal{E} \in \{\{1, 2\}, \{10, 11\}\}$. Considering the same code structure, for $e = 3$ erasures, one observes that out of the $\binom{11}{3} = 165$ possible 3 failure patterns, 5 patterns are not recoverable: $\mathcal{E} \in \{\{1, 2, 11\}, \{2, 3, 7\}, \{2, 4, 8\}, \{3, 4, 7\}, \{5, 9, 10\}\}$. It is worth noting that a lazy repair strategy can be beneficial in the following way: if nodes 10 and 11 fail, i.e., $\mathcal{E} = \{10, 11\}$, then, one can optimally repair any 3 erasures $\mathcal{E} \in \{\{i, 10, 11\}, i \neq 10, i \neq 11\}$. Finally, as suggested by Theorem 4 and Theo-*

rem 5, we find that increasing the underlying field size to $\mathbb{F}_{2^8}$ suffices to ensure optimal repair of all two and three erasure patterns in this scenario.

Remark 6. *Following the code shortening procedure described in [43], we construct an $(n, k, d - e + 1, e, k - 1, e)$ product-matrix MSMR code $\mathcal{C}$ with optimal repair for any $e \in [n - k]$ erasures such that $2k - 2 \leq d \leq n - 1$. First, as described in Remark 5, we consider an $(n + (d - 2k + 2), k + (d - 2k + 2), d + (d - 2k + 2) - e + 1, e, d - k + 1, \beta = 1)$ product-matrix MSMR code $\mathcal{C}'$ in systematic form with varying $e \in [n - k]$. Note that the code $\mathcal{C}'$ exists because the parameters satisfy Theorem 5. The first $(d - 2k + 2)$ systematic nodes of $\mathcal{C}'$ are set to zeros. Then, the target code $\mathcal{C}$ is formed by deleting the first $(d - 2k - 2)$ rows in each code matrix of $\mathcal{C}'$. It can be seen that the repair procedure for e erasures in $\mathcal{C}$ can be done by invoking that of the original code $\mathcal{C}'$, which leads to the result.*

D. Interference alignment codes

In this subsection, we give explicit code coefficient conditions for optimal MSMR codes from IA codes [8] for $e = 2, 3, 4$ erasures, and for any $e \leq k$ erasures from only the systematic (or only the parity) nodes. Moreover, we show the existence of MSMR codes for any $e \leq k$ erasures.

The scalar MSR IA code construction is based on interference alignment techniques. The code is systematic and defined over a finite field $\mathbb{F}_q$ with optimal repair bandwidth for the case $\frac{k}{n} \leq \frac{1}{2}$ and $d \geq 2k - 1$. We focus on the case $n = 2k, d = 2k - 1, \beta = 1$. In this scenario, the storage size is $\alpha = d - k + 1 = k$.

Notation. For an invertible matrix B, we define its inverse transpose to be $B' \triangleq (B^{-1})^t$. The columns of B' constitute the dual basis of the column vectors of B. Recall that $B_{i,j}$ denotes the (i, j)-th element of matrix B. We use the following symbols to denote the transmission of information during repair operations.

- $s_{i,j}$: from systematic node i to parity node j.
- $r_{i,j}$: from systematic node i to systematic node j.
- $\bar{s}_{i,j}$: from parity node i to systematic node j.
- $\bar{r}_{i,j}$: from parity node i to parity node j.

The IA code is constructed as below. Consider k linearly independent vectors $\{\mathbf{v}_1, \ldots, \mathbf{v}_k\}$, $\mathbf{v}_i \in \mathbb{F}_q^k, i \in [k]$. Let

$$V = [\mathbf{v}_1, \ldots, \mathbf{v}_k], U = \kappa^{-1} V' P, \quad (24)$$

where every submatrix of the $(k \times k)$ matrix P is invertible and κ is an arbitrary non-zero constant in $\mathbb{F}_q$ satisfying $\kappa^2 - 1 \neq 0$. Let $\mathbf{w}_l, l \in [k]$ denote the content of systematic node l and $\bar{\mathbf{w}}_i$ the content of parity node i , $i \in [k]$. Let $\mathbf{u}_i, \mathbf{v}_i, \mathbf{u}'_i, \mathbf{v}'_i$ be the i-th column of U, V, U', V', respectively. Then, by the construction in [8],

$$\bar{\mathbf{w}}_i^t = \sum_{j=1}^k \mathbf{w}_j^t G_j^{(i)}, G_j^{(i)} = \mathbf{u}_i \mathbf{v}_j^t + P_{j,i} I,$$

such that the matrix $G_j^{(i)}$ indicates the encoding submatrix for parity node i, associated with information unit j, and I is the identity matrix of size $(k \times k)$.

Repair of a systematic node. Assume that systematic node l fails. The general repair procedure is described in [8]. In this section, we explicitly develop the exact expression of $\mathbf{w}_l$ as it is needed later in repairing multiple erasures. Each systematic node $j \in [k] \backslash \{l\}$ transmits $r_{j,l} = \mathbf{w}_j^t \mathbf{v}'_l$. Each parity node $i \in [k]$ transmits $\bar{s}_{i,l} = \bar{\mathbf{w}}_i^t \mathbf{v}'_l$. Noting that $G_j^{(i)} \mathbf{v}'_l = \mathbb{1}_{\{j=l\}} \mathbf{u}_i + P_{j,i} \mathbf{v}'_l$, it follows that

$$\bar{s}_{i,l} = \mathbf{w}_l^t (\mathbf{u}_i + P_{l,i} \mathbf{v}'_l) + \sum_{j \in [k] \backslash \{l\}} P_{j,i} r_{j,l}.$$

Canceling the interference from systematic nodes, and arranging the contributions of parity nodes in matrix form, we write

$$\begin{bmatrix} \bar{s}_{1,l} - \sum\limits_{j \in [k] \backslash \{l\}} P_{j,1} r_{j,l} \\ \vdots \\ \bar{s}_{k,l} - \sum\limits_{j \in [k] \backslash \{l\}} P_{j,k} r_{j,l} \end{bmatrix} = \begin{bmatrix} \mathbf{u}_1^t + P_{l,1} \mathbf{v}_l'^t \\ \vdots \\ \mathbf{u}_k^t + P_{l,k} \mathbf{v}_l'^t \end{bmatrix} \mathbf{w}_l$$

$$= \frac{1}{\kappa} P^t (I + \kappa \mathbf{e}_l \mathbf{e}_l^t) V^{-1},$$

where the last equality is obtained by substituting U by its expression in (24). Using the Sherman-Morrison formula, for an invertible square matrix A of size $(k \times k)$ and vectors $\mathbf{u}, \mathbf{v}$ of length k,

$$(A + \mathbf{u} \mathbf{v}^t)^{-1} = A^{-1} - \frac{A^{-1} \mathbf{u} \mathbf{v}^t A^{-1}}{1 + \mathbf{v}^t A^{-1} \mathbf{u}},$$

we obtain that $(\frac{1}{\kappa} P^t (I + \kappa \mathbf{e}_l \mathbf{e}_l^t) V^{-1})^{-1} = U' - \frac{\kappa^2}{1+\kappa} V \mathbf{e}_l \mathbf{e}_l^t P'$, it follows that

$$\mathbf{w}_l = \left(U' - \frac{\kappa^2}{1 + \kappa} V \mathbf{e}_l \mathbf{e}_l^t P' \right) \begin{bmatrix} \bar{s}_{1,l} - \sum\limits_{j \in [k] \backslash \{l\}} P_{j,1} r_{j,l} \\ \vdots \\ \bar{s}_{k,l} - \sum\limits_{j \in [k] \backslash \{l\}} P_{j,k} r_{j,l} \end{bmatrix}. \quad (25)$$

Repair of a parity node. The repair of a parity node is optimally achieved through the duality property of IA codes resulting in a structure that is also conducive to interference alignment. Indeed, inverting the roles of parity and systematic nodes, it follows from [8] that

$$\mathbf{w}_i = \sum_{j=1}^k \bar{\mathbf{w}}_j^t G_j'^{(i)}; G_j'^{(i)} = \frac{1}{1 - \kappa^2} (\mathbf{v}_i' \mathbf{u}_j'^t - \kappa^2 P_{i,j}' I).$$

Assume parity node l fails, then systematic node i transmits $s_{i,l} = \mathbf{w}_i^t \mathbf{u}_l$ and parity node j sends $\bar{r}_{j,l} = \bar{\mathbf{w}}_j^t \mathbf{u}_l$. Note that $G_j'^{(i)} \mathbf{u}_l = \frac{1}{1-\kappa^2} (-\kappa^2 u_j + \mathbb{1}_{\{j=l\}} \mathbf{v}_i')$. It follows that

$$s_{i,l} = \frac{1}{1 - \kappa^2} \bar{\mathbf{w}}_l^t (\mathbf{v}_i' - \kappa^2 P_{i,l}' \mathbf{u}_l) + \sum_{j \in [k] \backslash \{l\}} \frac{-\kappa^2}{1 - \kappa^2} P_{i,j}' \bar{r}_{j,l}.$$

Combining information from different helpers, we obtain after

simplification

$$\begin{bmatrix} s_{1,l} + \frac{\kappa^2}{1-\kappa^2}\sum_{j\in[k]\setminus\{l\}} P'_{1,j}\bar{r}_{j,l} \\ \vdots \\ s_{k,l} + \frac{\kappa^2}{1-\kappa^2}\sum_{j\in[k]\setminus\{l\}} P'_{k,j}\bar{r}_{j,l} \end{bmatrix} = \frac{1}{1-\kappa^2}\begin{bmatrix} \mathbf{v}_1'^t - \kappa^2 P'_{1,l}\mathbf{u}_l^t \\ \vdots \\ \mathbf{v}_k'^t - \kappa^2 P'_{k,l}\mathbf{u}_l^t \end{bmatrix}\bar{\mathbf{w}}_l$$

$$= \frac{\kappa}{1-\kappa^2}P'(I - \kappa \mathbf{e}_l\mathbf{e}_l^t)U^t\bar{\mathbf{w}}_l,$$

where the last equality is obtained by replacing $V^{-1} = \kappa P'U^t$. Inverting the system of equations and using the Sherman-Morrison formula, we obtain

$$\bar{\mathbf{w}}_l = ((1-\kappa^2)V+(1+\kappa)U'\mathbf{e}_l\mathbf{e}_l^t P^t)\begin{bmatrix} s_{1,l} + \frac{\kappa^2}{1-\kappa^2}\sum_{j\in[k]\setminus\{l\}} P'_{1,j}\bar{r}_{j,l} \\ \vdots \\ s_{k,l} + \frac{\kappa^2}{1-\kappa^2}\sum_{j\in[k]\setminus\{l\}} P'_{k,j}\bar{r}_{j,l} \end{bmatrix}.$$

$$(26)$$

Repair of multiple erasures. The goal is to construct the system of linear equations as in (19). We need to derive the equations relating the information transferred across the failed systematic and parity nodes according to (18). Consider a systematic node $l \in [k]$ and a parity node $m \in [k]$, from (25), we write

$$s_{l,m} = \mathbf{u}_m^t\mathbf{w}_l$$

$$= (\mathbf{u}_m^t U' - \frac{\kappa^2}{1+\kappa}\mathbf{u}_m^t V\mathbf{e}_l\mathbf{e}_l^t P')\begin{bmatrix} \bar{s}_{1,l} - \sum_{j\in[k]\setminus\{l\}} P_{j,1}r_{j,l} \\ \vdots \\ \bar{s}_{k,l} - \sum_{j\in[k]\setminus\{l\}} P_{j,k}r_{j,l} \end{bmatrix}$$

$$= (\mathbf{e}_m^t - \frac{\kappa}{1+\kappa}P_{l,m}\mathbf{e}_l^t P')\begin{bmatrix} \bar{s}_{1,l} - \sum_{j\in[k]\setminus\{l\}} P_{j,1}r_{j,l} \\ \vdots \\ \bar{s}_{k,l} - \sum_{j\in[k]\setminus\{l\}} P_{j,k}r_{j,l} \end{bmatrix} \quad (27)$$

$$= (\mathbf{e}_m^t - \frac{\kappa}{1+\kappa}P_{l,m}\mathbf{e}_l^t P')(\sum_{j\in[k]} \bar{s}_{j,l}\mathbf{e}_j - \sum_{j\in[k]\setminus\{l\}} r_{j,l}P^t\mathbf{e}_j)$$

$$= (1 - \frac{\kappa}{1+\kappa}P_{l,m}P'_{l,m})\bar{s}_{m,l}$$

$$- \sum_{j\in[k]\setminus\{m\}} (\frac{\kappa}{1+\kappa}P_{l,m}P'_{l,j})\bar{s}_{j,l} - \sum_{j\in[k]\setminus\{l\}} P_{j,m}r_{j,l}.$$

$$(28)$$

Here (27) is obtained by noting that $U^tV = \frac{1}{\kappa}P^t$, and (28) follows using $P'P^t = I$. Similarly, consider two systematic nodes $l_1, l_2 \in [k], l_1 \neq l_2$, starting from (25) and noting that $V'^tU' = \kappa P'$, we obtain after simplification

$$r_{l_1,l_2} = \mathbf{v}_{l_2}'^t\mathbf{w}_{l_1} = \sum_{j\in[k]} (\kappa P'_{l_2,j})\bar{s}_{j,l_1} - \kappa r_{l_2,l_1}. \quad (29)$$

Proceeding in a similar way, for a systematic node $l \in [k]$ and a parity node $m \in [k]$, starting from (26), we obtain

$$\bar{s}_{m,l} = \mathbf{v}_l'^t\bar{\mathbf{w}}_m = (1 - \kappa^2 + \kappa(1+\kappa)P'_{l,m}P_{l,m})s_{l,m}$$

$$+ \sum_{j\in[k]\setminus\{l\}} (\kappa(1+\kappa)P'_{l,m}P_{j,m})s_{j,m} + \sum_{j\in[k]} (\kappa^2 P'_{l,j})\bar{r}_{j,m}.$$

$$(30)$$

Finally, consider two parity nodes $m_1, m_2 \in [k], m_1 \neq m_2$, starting from (26), we obtain

$$\bar{r}_{m_1,m_2} = \mathbf{u}_{m_2}^t\bar{\mathbf{w}}_{m_1} = \sum_{j\in[k]} (\frac{1-\kappa^2}{\kappa}P_{j,m_2})s_{j,m_1} + \kappa\bar{r}_{m_2,m_1}.$$

$$(31)$$

The details of deriving (29), (30) and (31) can be found in Appendix C. Equations (28), (29), (30) and (31) can thus be used to derive A and $\mathbf{b}$ as defined in (19).

In the following theorem, we show that the IA code already provides optimal repair for systematic (respectively parity) failures, without the need to modify the coding matrices.

Theorem 6. *In the interference alignment MSR code [8], it is possible to optimally repair any set of $e \leq k$ systematic (respectively parity) failures.*

Proof: Assume w.l.o.g that nodes $\{1, \ldots, e\}$ have failed. Let $\mathbf{s} = \begin{bmatrix} r_{1,2}, r_{2,1}, \ldots, r_{e-1,e}, r_{e,e-1} \end{bmatrix}^t$. Then, from (29), it follows that A is a block-diagonal matrix given by

$$A = \begin{bmatrix} 1 & \kappa & & & \\ \kappa & 1 & & & \\ & & \ddots & & \\ & & & 1 & \kappa \\ & & & \kappa & 1 \end{bmatrix}.$$

It follows that $|A| = (1-\kappa^2)^{\frac{e(e-1)}{2}} \neq 0$ as $\kappa^2 \neq 1$ by design. The same procedure applies to any set of e failures among parity nodes using equation (31). ∎

Theorem 7. *The interference alignment MSR code achieves optimal simultaneous repair of one systematic node l and one parity node m if $P_{l,m}(P^{-1})_{m,l} \neq 1$.*

Proof: Assume that systematic node l and parity node m failed. Let $\mathbf{s} = [s_{l,m}, \bar{s}_{m,l}]^t$. From (28), we obtain

$$s_{l,m} = (1 - \frac{\kappa}{1+\kappa}P_{l,m}P'_{l,m})\bar{s}_{m,l} + c_1,$$

where c_1 is a known quantity independent of $\mathbf{s}$. Similarly, from (30), we obtain

$$\bar{s}_{m,l} = (1 - \kappa^2 + \kappa(1+\kappa)P_{l,m}P'_{l,m})s_{l,m} + c_2,$$

where c_2 is a known quantity independent of $\mathbf{s}$. It follows that A, as defined in (19), is given by

$$A = \begin{bmatrix} -1 & 1 - \frac{\kappa}{1+\kappa}P_{l,m}P'_{l,m} \\ 1 - \kappa^2 + \kappa(1+\kappa)P_{l,m}P'_{l,m} & -1 \end{bmatrix}.$$

After simplification, we have $|A| \neq 0 \iff \kappa^2(P_{l,m}P'_{l,m} - 1)^2 \neq 0 \iff P_{l,m}(P^{-1})_{m,l} \neq 1$, as $\kappa \neq 0$. ∎

Combining Theorems 6 and 7 we know that $(2k, k, 2k - 2, 2, k, 2)$ MSMR codes for 2 erasures can be constructed through IA codes. We point out that Theorems 6 and 7 have been derived in [30] for cooperative repair, using a different technique. Recall that MSCR codes are in particular MSMR

codes [23]. However, their technique cannot be extended to more than two node failures including systematic and parity nodes [30].

Theorem 8. *The interference alignment MSR code achieves optimal simultaneous repair of:*
- *two systematic failures l_1, l_2 and one parity failure m if $1 - P_{l_1,m}(P^{-1})_{m,l_1} - P_{l_2,m}(P^{-1})_{m,l_2} \neq 0$,*
- *one systematic failure l and two parity failures m_1, m_2 if $1 - P_{l,m_1}(P^{-1})_{m_1,l} - P_{l,m_2}(P^{-1})_{m_2,l} \neq 0$,*
- *three systematic failures l_1, l_2, l_3 and one parity failure m if*

$$1 - P_{l_1,m}(P^{-1})_{m,l} - P_{l_2,m}(P^{-1})_{m,l_2} - P_{l_3,m}(P^{-1})_{m,l_3} \neq 0,$$

- *one systematic failure l and three parity failures m_1, m_2, m_3 if*

$$1 - P_{l,m_1}(P^{-1})_{m_1,l} - P_{l,m_2}(P^{-1})_{m_2,l} - P_{l,m_3}(P^{-1})_{m_3,l} \neq 0,$$

- *two systematic failures l_1, l_2 and two parity failures m_1, m_2 if*

$$
\begin{aligned}
&1 - P_{l_1,m_1}(P^{-1})_{m_1,l_1} - P_{l_1,m_2}(P^{-1})_{m_2,l_1} - P_{l_2,m_1}(P^{-1})_{m_1,l_2} \\
&- P_{l_2,m_2}(P^{-1})_{m_2,l_2} + P_{l_1,m_1}(P^{-1})_{m_1,l_1}P_{l_2,m_2}(P^{-1})_{m_2,l_2} \\
&+ P_{l_1,m_2}(P^{-1})_{m_2,l_1}P_{l_2,m_1}(P^{-1})_{m_1,l_2} \\
&- P_{l_1,m_1}(P^{-1})_{m_1,l_2}P_{l_2,m_2}(P^{-1})_{m_2,l_1} \\
&- P_{l_1,m_2}(P^{-1})_{m_2,l_2}P_{l_2,m_1}(P^{-1})_{m_1,l_1} \neq 0.
\end{aligned}
$$

Proof: The proof follows along similar lines as Theorem 7 by constructing A using (28), (29), (30) and (31). The explicit expression of $|A|$ can then be obtained for example by using the Symbolic Math Toolbox of MATLAB, from which the above conditions can be readily obtained (the MATLAB source code can be found in [48]). ∎

Combining Theorems 6 and 8 we know that $(2k, k, 2k - e, e, k, e)$ MSMR codes for $e = 3, 4$ erasures can be constructed through IA codes.

Remark 7. *Deriving an exact condition under which the recovery of multiple failures for large e is not straightforward. However, we suspect that the general formula is given by the following expression*

$$|A| = k^{2sp}(1-k^2)^{\binom{s}{2}+\binom{p}{2}}\Big(1-$$
$$\sum_{\substack{L \subset S, J \subset P, \\ |L|=|J|, \\ |J| \leq \min(s,p)}} \sum_{\sigma \in \Pi_{L,J}} \sum_{\sigma' \in \Pi_{L,J}} sgn(\sigma)sgn(\sigma')\prod_{i \in L}P_{i,\sigma(i)}\prod_{j \in J}P'_{j,\sigma'(j)}\Big)^e,$$

(32)

where $\Pi_{L,J}$ is the group of permutations between the two sets L and J (L and J are ordered in increasing order), and element of order h in L is mapped to element of order h in J) and $sgn(\sigma)$ refers to the sign of a permutation σ. For example, if $L = \{1,2,3\}$, $J = \{2,3,4\}$ and $\sigma(1) = 3, \sigma(2) = 4, \sigma(3) = 2$. Then, $sgn(\sigma)=1$. One can check that the formulas in Theorems 6, 7 and 8 satisfy (32). A general proof of (32) is still open.

Example 3. *Consider the IA code with $n = 8, k = 4, d = 7$, $\alpha = 4, \beta = 1$. The code is defined over the finite field $\mathbb{F}_{2^5}$ and let g be the generator of its multiplicative group. Let P be a Vandermonde matrix given by*

$$
P = \begin{bmatrix}
1 & 1 & 1 & 1 \\
1 & g & g^2 & g^3 \\
1 & g^2 & g^4 & g^6 \\
1 & g^3 & g^6 & g^9
\end{bmatrix}.
$$

Using Theorems 6, 7 and 8, one can check that any two, three and four erasures can be repaired optimally using our repair framework.

In the following theorem, we provide an existence proof of IA MSMR codes for multiple erasures.

Theorem 9. *There exists $(2k, k, 2k - e, e, k, e)$ interference alignment MSMR codes, defined over a large enough finite field, such that any $e \leq k$ erasures can be optimally repaired.*

Proof: From Theorem 6, we know that if the errors are all either systematic or parity nodes, then efficient repair is possible. Thus, we only need to analyze the case of a mixture of systematic and parity failures.

Consider $e \leq k$ failures consisting of q systematic nodes and p parties nodes, indexed by the sets $\mathcal{Q}$ and $\mathcal{P}$. W.lo.g, assume that $\mathcal{Q} = [q]$ and $\mathcal{P} = [p]$. Let $\mathbf{s}$ denote the vector of unknowns such that pairs $(r_{i,j}, r_{j,i})$, $(\bar{r}_{i,j}, \bar{r}_{j,i})$ and $(s_{i,j}, \bar{s}_{j,i})$ are grouped together. Using (28), (29), (30) and (31), we construct A as in (19). Denote the determinant of A as $F(\kappa, P_{i,j}, P'_{i,j}, i \in \mathcal{Q}, j \in \mathcal{P}) \triangleq |A|$. The rows and columns of A are indexed by $\{r_{i,j}, s_{i,j}, \bar{r}_{i,j}, \bar{s}_{i,j}\}$. Let $M_{i,j}$ denote the minor in A corresponding to $A_{i,j}$. Similarity, $N_{i,j}$ denotes the minor in P corresponding to $P_{i,j}$. As $P' = (P^{-1})^t$, $P'_{i,j} = \frac{(-1)^{i+j}N_{j,i}}{|P|}$, one concludes that F is a rational function in $(\kappa, P_{i,j}, i, j \in [k])$.

Claim 2. F is not identically zero for any $q, p \geq 0, q + p = e \leq k$.

If Claim 2 holds, then the theorem is proved due to the following argument. By symmetry among the systematic (respectively parity) nodes of IA codes, any e-erasure pattern corresponds to a non-zero rational function F. Recall from [8] that the reconstruction process requires that every submatrix of P is invertible. This can be translated into a polynomial constraint given by $g(P_{i,j}, i \in \mathcal{Q}, j \in \mathcal{P}) \neq 0$. Let $T \triangleq g \prod_{e \text{ erasures}} F$. Here the product is over all possible e erasures, and the rational function F depend on the erasure pattern. Then, it follows that T is a non-zero rational polynomial in $(\kappa, P_{i,j}, (i,j) \in [k] \times [n-k])$. By Combinatorial Nullstellensatz [47], we can find assignments of the variables $(\kappa, \{P_{i,j}\})$ over a large enough finite field, such that the code guarantees optimal recovery of any set of e erasures.

Next, we prove Claim 2. We assume first that $q \leq \frac{k}{2}$. Let

$$P_{i,j} = 0 \ \forall \ (i,j) \in \mathcal{Q} \times \mathcal{P}. \tag{33}$$

Note that one can always construct a (normalized) invertible matrix P satisfying (33), so we can assume $|P| = 1$. Thus F is a polynomial. We will show

$$
\begin{aligned}
&F(\kappa, P_{i,j} = 0, P'_{i,j}, (i,j) \in \mathcal{Q} \times \mathcal{P}) \\
&= F(\kappa, P_{i,j} = 0, P'_{i,j} = 0, (i,j) \in \mathcal{Q} \times \mathcal{P}) \neq 0,
\end{aligned}
$$

which implies

$$F(\kappa, P_{i,j}, P'_{i,j}, (i,j) \in \mathcal{Q} \times \mathcal{P}) \neq 0.$$

To this end, we first prove that $F(\kappa, P_{i,j} = 0, P'_{i,j}, (i,j) \in \mathcal{Q} \times \mathcal{P})$, viewed as a polynomial of $(\kappa, \{P'_{i,j}\})$, does not depend on $\{P'_{i,j}\}$. From (29) and (30), one can check that $P'_{i,j}$ appears in A at entries given by

- $A_{r_{l,i}, \bar{s}_{j,l}}$ for $l \in \mathcal{Q}\backslash\{i\}$,
- $A_{\bar{s}_{m,i}, \bar{r}_{j,m}}$ for $m \in \mathcal{P}\backslash\{j\}$.

For any $l \in \mathcal{Q}\backslash\{i\}$, consider the two columns in A indexed by $r_{l,i}$ and $r_{i,l}$. Both columns have non-zero entries only at rows indexed with $r_{l,i}$ and $r_{i,l}$. Then, after removing entries at row $r_{l,i}$, it follows that both columns become linearly dependent, as both columns are scalar multiples of the same standard basis vector. Thus, $M_{r_{l,i}, \bar{s}_{j,l}} = 0$.

The example in (34) illustrates the case of two systematic failures, given by systematic nodes 1 and 2, and one parity failure, given by parity node 1. In this case $s = \left[s_{11}, \bar{s}_{11}, r_{12}, r_{21}, s_{21}, \bar{s}_{12}\right]$. Setting $P_{i,j} = 0$ for all $i = 1, 2, j = 1$ and looking at the submatrix of A by removing row $r_{1,2}$ and column $\bar{s}_{2,1}$ in (34), it can be seen that columns $r_{1,2}, r_{2,1}$ are dependent, hence its corresponding minor $M_{r_{1,2}, \bar{s}_{2,1}} = 0$. Similarly, for any $m \in \mathcal{P}\backslash\{j\}$, consider the two rows in A indexed by $\bar{r}_{j,m}$ and $\bar{r}_{m,j}$. Both rows have non-zero entries only at columns indexed with $\bar{r}_{j,m}$ and $\bar{r}_{m,j}$. Then, after removing entries at column $\bar{r}_{j,m}$, it follows that both rows become linearly dependent. Thus, $M_{\bar{s}_{m,i}, \bar{r}_{j,m}} = 0$.

The minors in A of all terms corresponding to $P'_{i,j}$ are thus equal to zero. Therefore, w.l.o.g, one can assume that $P'_{i,j} = 0, \forall (i,j) \in \mathcal{Q} \times \mathcal{P}$. It follows that A is block-diagonal matrix such that

- Row/column pairs $(r_{i,j}, r_{j,i})$ correspond to $\begin{bmatrix} -1 & -\kappa \\ -\kappa & -1 \end{bmatrix}$,
- Row/column pairs $(\bar{r}_{i,j}, \bar{r}_{j,i})$ correspond to $\begin{bmatrix} -1 & \kappa \\ \kappa & -1 \end{bmatrix}$,
- Row/column pairs $(s_{i,j}, \bar{s}_{j,i})$ correspond to $\begin{bmatrix} -1 & 1 \\ 1-\kappa^2 & -1 \end{bmatrix}$.
- Other entries are 0.

Therefore, $|A| = \kappa^{2qp}(1-\kappa^2)^{\binom{q}{2}+\binom{p}{2}} \neq 0$, as $\kappa \neq 0$ and $\kappa^2 \neq 1$.

Assume now that $q > \frac{k}{2}$. Then, $p \leq \frac{k}{2}$. Proceeding similarly, one can show that if $P_{ij} = 0, \forall (i,j) \in \mathcal{Q} \times \mathcal{P}$, then, all terms P_{ij} have no impact on $|A|$ and one obtains similarly $|A| = \kappa^{2rp}(1-\kappa^2)^{\binom{r}{2}+\binom{p}{2}}$. ■

IV. NON-EXISTENCE OF EXACT MBMR REGENERATING CODES

Recall that the MBMR point is defined as the minimum bandwidth point on the functional tradeoff. In this section, we explore the existence of linear exact MBMR regenerating codes for $1 < e < k$. Unlike the single erasure repair problem [43] and the cooperative repair problem [29], we prove that linear exact regenerating codes do not exist. Following [29], [43], we proceed by investigating subspace properties that linear exact MBMR codes should satisfy. Then, we prove that the derived properties over-constrain the system.

A. Subspace viewpoint

Linear exact regenerating codes can be analyzed from a viewpoint based on subspaces. A linear storage code is a code in which every stored symbol is a linear combination of the $\mathcal{M}$ symbols of the file. Let $\mathbf{f}$ denote an $\mathcal{M}$-dimensional vector containing the source symbols. Then, any symbol x can be represented by a vector $\mathbf{h}$ satisfying $x = \mathbf{f}^t \mathbf{h}$ such that $\mathbf{h} \in \mathbb{F}^{\mathcal{M}}$, $\mathbb{F}$ being the underlying finite field. The vectors $\mathbf{h}$ define the code. A node storing α symbols can be considered as storing α vectors. Node i stores $\mathbf{h}_1^{(i)} \dots \mathbf{h}_\alpha^{(i)}$. It is easy to see that linear operations performed on the stored symbols are equivalent to the same operations performed on the these vectors: $\sum \gamma_i \mathbf{f}^t \mathbf{h}_i = \mathbf{f}^t(\sum \gamma_i \mathbf{h}_i)$, $\gamma_i \in \mathbb{F}$. Thus, each node is said to store a subspace of dimension at most α. We write W_A to denote the subspace stored by all nodes in the set A, $A \subseteq [n]$. For repair, each helper node passes β symbols. Equivalently, each node passes a subspace of dimension at most β. We denote the subspace passed by node j to repair a set R of e nodes by S_j^R. The subspace passed by a set of nodes A to repair a set R of e nodes is denoted by $S_A^R = \sum_{j \in A} S_j^R$, where the sum denotes the sum of subspaces.

Notation. The notation $\bigoplus_j X_j$ denotes the direct sum of subspaces $\{X_j\}$. For a general exact regenerating code, which can be nonlinear, we use by abuse of notation W_A, S_A^R to represent the random variables of the stored information in nodes A, and of the transmitted information from helpers A to failed nodes R. Properties that hold using entropic quantities for a general code do hold when considering linear codes. For instance, consider two sets A and B. Then, we note the following

$$H(W_A) \rightarrow \dim(W_A),$$
$$H(W_A|W_B) \rightarrow \dim(W_A) - \dim(W_A \cap W_B),$$
$$I(W_A, W_B) \rightarrow \dim(W_A \cap W_B),$$

where the symbol $\rightarrow$ means *translates to*. When results hold for general codes, we only prove for the entropy properties, and the proof for the subspace properties of linear codes is omitted. All results on entropic quantities are for general codes, and all results on subspaces are for linear codes. Moreover, all results in this section refer to properties of *optimal exact* multi-node repair codes with $k > e$ (constructions for $k \leq e$ are presented in Section III-A), some of which are specific to MBMR codes and will be noted.

In this section, we assume that the codes are *symmetric*. Namely, the entropy (or subspace) properties do not depend on the indices of the nodes. Note that one can always construct a symmetric code from a non-symmetric code [49], hence our assumption does not lose generality. We now start by proving some properties that exact regenerating codes, satisfying the optimal functional tradeoff, should satisfy. We note that the following property is also presented in [35, Lemma 4].

Lemma 5. *Let $B \subseteq [n]$ be a subset of nodes of size e, then for an arbitrary set of nodes A, such that $0 \leq |A| \leq d, B \cap A = \emptyset$,*

$$H(W_B|W_A) \leq H(W_B|S_A^B) \leq \min(e\alpha, (d-|A|)\beta).$$

Proof: If nodes B are erased, consider the case of having

$$A = \begin{bmatrix} -1 & 1 - \frac{\kappa P_{11} P_{11}^{'}}{\kappa+1} & 0 & -P_{21} & 0 & 0 \\ 1 - \kappa^2 + \kappa(\kappa+1)P_{11}P_{11}^{'} & -1 & 0 & 0 & \kappa(\kappa+1)P_{21}P_{11}^{'} & 0 \\ 0 & \kappa P_{21}^{'} & -1 & -\kappa & 0 & 0 \\ 0 & 0 & -\kappa & -1 & 0 & \kappa P_{11}^{'} \\ 0 & 0 & -P_{11} & 0 & -1 & 1 - \frac{\kappa P_{21} P_{21}^{'}}{\kappa+1} \\ \kappa(\kappa+1)P_{11}P_{21}^{'} & 0 & 0 & 0 & 1 - \kappa^2 + \kappa(\kappa+1)P_{21}P_{21}^{'} & -1 \end{bmatrix} \begin{matrix} s_{1,1} \\ \bar{s}_{1,1} \\ r_{1,2} \\ r_{2,1} \\ s_{2,1} \\ \bar{s}_{2,1} \end{matrix} \qquad (34)$$

with column labels $s_{1,1}$, $\bar{s}_{1,1}$, $r_{1,2}$, $r_{2,1}$, $s_{2,1}$, $\bar{s}_{2,1}$.

nodes A and nodes C as helper nodes, $|C| = d - |A|$. Then, the exact repair condition requires

$$0 = H(W_B | S_A^B, S_C^B)$$
$$= H(W_B | S_A^B) - I(W_B, S_C^B | S_A^B)$$
$$\geq H(W_B | S_A^B) - H(S_C^B)$$
$$\geq H(W_B | S_A^B) - (d - |A|)\beta.$$

Moreover, we have $H(W_B | S_A^B) \leq H(W_B) \leq |B|\alpha$, $H(W_B | W_A) \leq H(W_B | S_A^B)$, and the results follows. ∎

In the next two subsections, we focus on the cases where $e \mid k$ and $e \nmid k$, respectively.

B. Case $e \mid k$

Note that in this case since $e < k$, we have $k \geq 2e$. Recall from Theorem 2 that points on the optimal tradeoff satisfy

$$\mathcal{M} = \sum_{j=0}^{\eta-1} \min(e\alpha, (d - je)\beta).$$

Points between and including MSMR and MBMR satisfy

$$\frac{d - k + e}{e}\beta \leq \alpha \leq \frac{d}{e}\beta.$$

Lemma 6. *(Entropy of data stored): Consider points on the optimal tradeoff. For an arbitrary set L of storage nodes of size e, and a disjoint set A such that $|A| = em < k$ for some integer m,*

$$H(W_L) = e\alpha,$$
$$H(W_L | W_A) = \min(e\alpha, (d - em)\beta).$$

For linear codes,

$$\dim(W_L) = e\alpha,$$
$$\dim(W_L) - \dim(W_L \cap W_A) = \min(e\alpha, (d - em)\beta).$$

Hence, the contents of any group of e nodes are independent. In particular, for a set A of nodes, $1 \leq |A| \leq e$, $H(W_A) = |A|\alpha$.

Proof: By reconstruction requirement, we write

$$\mathcal{M} = H(W_{[k]})$$
$$= H(W_{[e]}) + \sum_{j=1}^{\eta-1} H(W_{\{ej+1,\ldots,e(j+1)\}} | W_{[je]})$$

$$\leq \min(e\alpha, d\beta) + \sum_{j=1}^{\eta-1} \min(e\alpha, (d - ej)\beta)$$
$$= \mathcal{M},$$

where the inequality follows from Lemma 5. Thus, all inequalities must be satisfied with equality. ∎

Corollary 1. *At the MBMR point, for any set L of size e and disjoint set A of size $|A| = em < k$, we have*

$$\dim(W_L \cap W_A) = em\beta.$$

Proof: By Lemma 6 and $e\alpha = d\beta$,

$$\dim(W_L) - \dim(W_L \cap W_A) = \min(e\alpha, (d - em)\beta)$$
$$= (d - em)\beta.$$

Using the fact that $\dim(W_L) = e\alpha = d\beta$, we obtain the result. ∎

Lemma 7. *For any set E of size e, and a disjoint set A of size d, the MBMR point satisfies*

$$W_E = \bigoplus_{j \in A} S_j^E, \dim(S_j^E) = \beta.$$

Hence, the subspaces S_j^E and $S_{j'}^E$ are linearly independent. For every set $Q \subseteq A$, $\dim(S_Q^E) = |Q|\beta$. Moreover, each subspace S_j^E has to be in W_E, namely, $S_j^E \subseteq W_E$.

Proof: For exact repair, we need $W_E \subseteq \sum_j S_j^E$. Thus,

$$d\beta = e\alpha = \dim(W_E) \leq \dim(\sum_j S_j^E) \leq d\beta.$$

Thus, every inequality has to be satisfied with equality. ∎

Lemma 8. *At the MBMR point, for any set E of e nodes and any other disjoint set Q of size $|Q| \leq k - e$, we have*

$$S_Q^E = W_E \cap W_Q, \dim(W_E \cap W_Q) = \dim(S_Q^E) = |Q|\beta.$$

Proof: Consider Q nodes such that $|Q| \leq k - e$ helping in the repair of a set E of e nodes. Let J contains Q such that $|J| = k - e$. Denote $Q^c = J \backslash Q$. From Corollary 1, we have $\dim(W_E \cap W_J) = (k - e)\beta$. On the other hand, from Lemma 7, we have $\dim(S_J^E) = (k - e)\beta$ and $S_J^E \subseteq W_E$. Moreover, by definition, $S_J^E \subseteq W_J$. Thus, $S_J^E \subseteq W_E \cap W_J$. As the dimensions match, it follows that $S_J^E = W_E \cap W_J$. Note that $S_A^E \subseteq W_E \cap W_A$ holds for any subset A of size

$|A| \leq d$. Now, we write

$$
\begin{aligned}
S_J^E = W_E \cap W_J &= W_E \cap (W_Q + W_{Q^c}) \\
&\supseteq W_E \cap W_Q + W_E \cap W_{Q^c} \\
&\supseteq S_Q^E + S_{Q^c}^E = S_J^E.
\end{aligned}
$$

This implies that all inclusion inequalities have to be satisfied with equality and the result follows. ∎

The next lemma plays an important role in establishing the non-existence of exact MBMR codes. It only holds true when $e \geq 2$, which conforms with the existence of single erasure MBMR codes.

Lemma 9. *Consider the MBMR point. When $e \geq 2$, for any set of $e + 2 \leq k$ nodes, labeled 1 through $e + 2$, it holds that*

$$
\dim(W_{e+2} \cap W_{[e+1]}) = \dim(W_{e+2} \cap W_{[e]}) = \beta.
$$

Proof: We have

$$
\begin{aligned}
\dim(W_{[e+2]}) = \dim(W_{[e]}) &+ \dim(W_{e+1} + W_{e+2}) \\
&- \dim(W_{[e]} \cap (W_{e+1} + W_{e+2})) \\
= e\alpha + 2\alpha - 2\beta,
\end{aligned}
$$

where the second equality follows from Lemma 6, Lemma 8. On the other hand, we write

$$
\begin{aligned}
\dim(W_{[e+2]}) = \dim(W_{[e]}) \\
+ \dim(W_{e+1}) - \dim(W_{e+1} \cap W_{[e]}) \\
+ \dim(W_{e+2}) - \dim(W_{e+2} \cap W_{[e+1]}) \\
= e\alpha + 2\alpha - \beta - \dim(W_{e+2} \cap W_{[e+1]}).
\end{aligned}
$$

The lemma follows from equating both equations. ∎

Theorem 10. *Exact linear regenerating MBMR codes do not exist when $2 \leq e < k$ and $e \mid k$.*

Proof: Assuming that there exists an exact-repair regenerating code, we consider the first e nodes. Then, these nodes store linearly independent vectors. We write, for $i = 1, \ldots, e$, $W_i = \begin{bmatrix} V_{i1} & V_{i2} \end{bmatrix}$ where $V_{i,1}$ contains β linearly independent columns and $V_{i,2}$ contains the remaining $(\alpha - \beta)$ basis vectors for node i. Now, consider node $e + 1$. We have $\dim(W_{e+1} \cap W_{[e]}) = \beta$ by Lemma 9. That means that node $e + 1$ contains β columns, linearly dependent on the columns from the first e nodes. Since the first e nodes should be linearly independent, w.l.o.g, we can assume that the β dependent vectors of node $e + 1$, denoted by $V_{e+1,1}$, is of the form

$$
V_{e+1,1} = \sum_{i=1}^{e} V_{i,1}\mathbf{x}_i, \tag{35}
$$

such that $\mathbf{x}_i \neq \mathbf{0}_{\beta \times 1} \ \forall i = 1, \ldots, e$. Now, consider node $e + 2$. From Lemma 9, node $e + 2$ contains $(\alpha - \beta)$ vectors linearly independent from vectors in nodes 1 through $e + 1$. The remaining basis vectors of node $e + 2$ (which are linearly independent of the $(\alpha - \beta)$ vectors) are denoted by $V_{e+2,1}$. Now, to repair any set of e nodes out of the set of first $e + 1$ nodes, node $e + 2$ can only pass $V_{e+2,1}$. Otherwise, Lemma 8 will be violated. Then, this implies that $V_{e+2,1} \subseteq W_J$, for all $J \subseteq \{1, \ldots, e+1\}$ such that $|J| = e$. Then, it can be seen

that $V_{e+2,1}$ can only be of the same form in (35)

$$
V_{e+2,1} = \sum_{i=1}^{e} V_{i,1}\mathbf{y}_i, \text{ such that } \mathbf{y}_i \neq \mathbf{0}_{\beta \times 1} \ \forall i = 1, \ldots, e.
$$

Similar reasoning applies to node i for $i = e + 3, \ldots, k + 1$ to conclude that $V_{i,1}$ can be written as in (35).

Now, assume the first e nodes fail. Then, node i can only pass $V_{i,1}$ for $i = e + 1, \ldots k + 1$. We recall from Lemma 8 that $S_i^{[e]} = W_i \cap W_{[e]}$. The total number of vectors passed by these nodes is $(k - e + 1)\beta \geq (e + 1)\beta$. On the other hand, from (35), all $V_{i,1}$ are generated by $e\beta$ vectors. Thus, the set $\{V_{i,1}, i = e + 1, \ldots, k + 1\}$ must be linearly dependent, which contradicts the linear independence property of the passed subspaces passed for repair, as stated by Lemma 7. ∎

C. Case $e \nmid k$

Recall that from the analysis of Theorem 2, for $k = \eta e + r$, $1 \leq r \leq e - 1$, at the MBMR point, two scenarios generate the same minimum cut:

$$
\mathbf{u}_1 = [r, e, \ldots, e] \text{ and } \mathbf{u}_2 = [e, \ldots, e, r].
$$

Equivalently, we have

$$
\mathcal{M} = f(\mathbf{u}_1) = f(\mathbf{u}_2), \tag{36}
$$

where $f()$ is defined as in (4). Moreover, all points between and including MSMR and MBMR on the tradeoff satisfy

$$
\mathcal{M} = \min(r\alpha, d\beta) + \sum_{i=0}^{\eta-1} \min(e\alpha, (d - r - ie)\beta) = f(\mathbf{u}_1),
$$
$$
\frac{d - k + e}{e}\beta \leq \alpha \leq \frac{d + \eta r - \eta e}{r}\beta. \tag{37}
$$

Properties satisfied by exact regenerating codes developed in the previous section extend to the case $e \nmid k$ with slight modifications. We state the properties without detailed proofs as the techniques are the same.

Lemma 10. *Consider points on the optimal tradeoff. For an arbitrary set R of storage nodes of size r, and a set A such that $|A| = je + r < k$ for some integer $j \leq \eta - 1$, for all exact-regenerating codes operating on the functional tradeoff, it holds that*

$$
H(W_R) = r\alpha,
$$
$$
H(W_E | W_A) = \min(e\alpha, (d - r - je)\beta).
$$

For linear codes,

$$
\dim(W_R) = r\alpha,
$$
$$
\dim(W_E) - \dim(W_E \cap W_A) = \min(e\alpha, (d - r - je)\beta).
$$

Proof: The result can be derived by proceeding as in Lemma 6 and using the fact that $\mathcal{M} = f(\mathbf{u}_1)$ from (37). ∎

Remark 8. *In the case of $e \nmid k$, a set of e nodes are no longer linearly independent. This is expected as $e\alpha > d\beta$. Instead, it can be seen from Lemma 10 that any set of r nodes are linearly independent.*

Lemma 11. *For exact-regenerating codes operating at the MBMR point, given sets E, A, R and B such that $|E| = e$, E and A are disjoint, R and B are disjoint, $|A| = je$ with $j \leq \eta - 1$, $|R| = r$ and $|B| = \eta e$, it holds that*

$$H(W_E) = d\beta,$$
$$H(W_E|W_A) = (d - je)\beta,$$
$$H(W_R|W_B) = (d - \eta e)\beta.$$

For linear codes,

$$\dim(W_E) = d\beta,$$
$$\dim(W_E) - \dim(W_E \cap W_A) = (d - je)\beta,$$
$$\dim(W_R) - \dim(W_R \cap W_B) = (d - \eta e)\beta.$$

Proof: The result can be derived by proceeding as in Lemma 6 and using the fact that $\mathcal{M} = f(\mathbf{u}_2)$ from (36) and $e\alpha \geq d\beta, r\alpha \geq (d - ae)\beta$. ∎

It is easy to see that Lemma 7 and Lemma 8 hold true for the case $e \nmid k$, and for conciseness we do not repeat these lemmas. The following lemma is used to derive the contradiction in our non-achievability result.

Lemma 12. *Let $k = \eta e + r$, then exact linear MBMR point is not achievable when $d > k$. When $d = k$, for any set of $r + 1$ nodes, it holds that*

$$\dim(W_{r+1} \cap W_{[r]}) = \beta.$$

Proof: We have

$$d\beta = \dim(W_{[e]})$$
$$= \sum_{i=1}^{e} \left(\dim(W_i) - \dim(W_i \cap W_{[i-1]}) \right)$$
$$= e\alpha - \sum_{i=r+1}^{e} \dim(W_i \cap W_{[i-1]}),$$

where the last equality follows from the fact that the first r nodes are linearly independent. Thus, it follows that

$$\sum_{i=r+1}^{e} \dim(W_i \cap W_{[i-1]}) = e\alpha - d\beta = (e - r)(\alpha - \eta\beta), \tag{38}$$

where we used $\alpha = (d + r\eta - e\eta)\beta/r$ for MBMR point. Now we write

$$(e - r)(\alpha - \eta\beta) = \sum_{i=r+1}^{e} \dim(W_i \cap W_{[i-1]})$$
$$\geq \sum_{i=r+1}^{e} \dim(W_i \cap W_{[r]})$$
$$= (e - r) \dim(W_{r+1} \cap W_{[r]}),$$

where the last equality follows using symmetry. Then, it follows that

$$\dim(W_{r+1} \cap W_{[r]}) \leq \alpha - \eta\beta. \tag{39}$$

Combining (38) and (39), we obtain

$$\sum_{i=r+2}^{e} \dim(W_i \cap W_{[i-1]}) \geq (e - r - 1)(\alpha - \eta\beta). \tag{40}$$

On the other hand, we have

$$\sum_{i=r+2}^{e} \dim(W_i \cap W_{[i-1]}) \leq \sum_{i=r+2}^{e} \dim(W_i \cap W_{\mathcal{E}_i})$$
$$= (e - r - 1)\beta, \tag{41}$$

where $\mathcal{E}_i$ is a set of e nodes containing the first $i-1$ nodes and arbitrary $e - i + 1$ nodes, excluding node i, and the equality follows from Lemma 8. Combining (40) and (41), it follows

$$(e - r - 1)(\alpha - \eta\beta) \leq \sum_{i=r+2}^{e} \dim(W_i \cap W_{[i-1]}) \leq (e - r - 1)\beta. \tag{42}$$

It follows that $\alpha - \eta\beta = \frac{d - \eta e}{r}\beta \leq \beta$. The inequality holds only when $d = k$ and $\alpha - \eta\beta = \beta$. Indeed, when $d > k$, we have $\alpha - \eta\beta > \beta$. Therefore, we only consider the case $d = k$. Hence, it follows from (42) that

$$\sum_{i=r+2}^{e} \dim(W_i \cap W_{[i-1]}) = (e - r - 1)\beta.$$

Using (38), we obtain $\dim(W_{r+1} \cap W_{[r]}) = \beta$. ∎

Theorem 11. *Exact linear regenerating MBMR codes do not exist when $e < k$ and $e \nmid k$.*

Proof: From Lemma 12, exact linear MBMR codes may only be feasible when $d = k$. Next we show that in fact such codes do not exist. Consider repair of the set of nodes E containing nodes 1 through e. Consider helper node i. As $\dim(W_i \cap W_{[r]}) = \dim(W_i \cap W_{[e]}) = \beta$, it follows that $W_i \cap W_{[r]} = W_i \cap W_{[e]} = S_i^E$. Then, each helper node sends vectors in the span of $W_{[r]}$. Thus, the span of all sub-spaces $S_i^{[e]}$ is included in the span of $W_{[r]}$: $\sum_i S_i^E \subseteq W_{[r]}$. This implies that $\dim(\sum_i S_i^E) \leq \dim(W_{[r]})$. Namely, we should have $d\beta \leq r\alpha$: this is a contradiction as $d\beta > r\alpha$. ∎

D. Minimum bandwidth cooperative regenerating codes as centralized multi-node repair codes

In [23], the authors argued that MBCR codes can be used as centralized multi-node repair regenerating codes. We recall that MBCR codes are characterized with

$$(\alpha_{\text{MBCR}}, \gamma_{\text{MBCR}}) = \left(\frac{\mathcal{M}(2d + e - 1)}{k(2d - k + e)}, \frac{\mathcal{M}2de}{k(2d - k + e)} \right).$$

In the case of $e \mid k$, it is shown that MBCR codes achieve the MBMR bandwidth, i.e, $\gamma_{\text{MBCR}} = \gamma_{\text{MBMR}}$. In the case of $e \nmid k$, by imposing a certain entropy accumulation property on the entropy of any group of r nodes, [23] showed that the bandwidth achieved by MBCR codes is optimal. It is important to note here that, from (16), it can be checked that the entropy constraint condition results in $\gamma_{\text{MBCR}} > \gamma_{\text{MBMR}}$ for $e \nmid k$. Moreover, in both cases, it is not clear whether MBCR lies on the exact tradeoff of centralized repair. The next
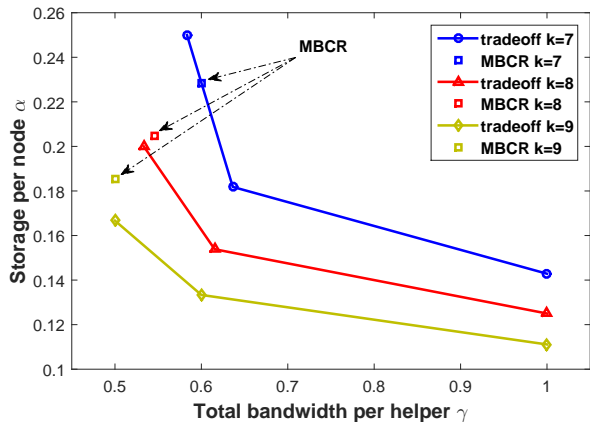
Fig. 4: Optimal functional repair tradeoffs for fixed $e = 3$ and different $k \in \{7, 8, 9\}$. The MBCR point lies on the tradeoff only in the case of $k = 7$.

theorem determines the cases in which MBCR codes meet the centralized functional repair tradeoff (but does not correspond to the minimum bandwidth point on the functional curve). As a consequence, for such cases, MBCR codes meet the exact repair tradeoff as well.

Theorem 12. *Assume $1 < e \leq k$, then, minimum bandwidth cooperative regenerating codes meet the centralized functional repair tradeoff if and only if $k \equiv 1 \mod e$.*

Proof: When $e \mid k$, from (14) and (15), it follows that $\gamma_{\text{MBMR}} = \gamma_{\text{MBCR}}$ and $\alpha_{\text{MBCR}} = \alpha_{\text{MBMR}} + \frac{\mathcal{M}(e-1)}{k(2d-k+e)}$. Thus, $\alpha_{\text{MBMR}} < \alpha_{\text{MBCR}}$. When $e \nmid k$, from (12) and (17), one can check that $\gamma_{\text{MBMR}} < \gamma_{\text{MBCR}} < f_r(0)$. Using (11), it follows that the optimal storage size corresponding to γ_{MBCR} and achieving the centralized functional repair tradeoff is given by

$$\alpha^*(\gamma_{\text{MBCR}}) = \frac{\mathcal{M} - \gamma_{\text{MBCR}} g_r(0)}{r} = \frac{\mathcal{M}(2d+e-r)}{k(2d-k+e)}$$
$$= \alpha_{\text{MBCR}} - \frac{\mathcal{M}(r-1)}{k(2d-k+e)}.$$

Therefore, $\alpha^*(\gamma_{\text{MBCR}}) \leq \alpha_{\text{MBCR}}$ with equality if and only if $r = 1$. ∎

Figure 4 illustrates the functional tradeoff for fixed $e = 3, \mathcal{M} = 1$ and multiple values of $k \in \{7, 8, 9\}$ such that $d = k$. As proved in Theorem 12, MBCR codes are optimal centralized repair codes only when $r = 1$, which corresponds to $k = 7$ in Figure 4. When $e \mid k$, MBCR codes achieve the same bandwidth as MBMR codes, but have a higher storage cost.

Remark 9. *Theorem 12 proves that, when $e \nmid k, r = 1$, MBCR codes achieve an interior point on the functional tradeoff that lies near the MBMR point. We note that the existence of this exact-repair interior point does not contradict the infeasibility result in Section V, where we assume $e \mid k$.*

V. INFEASIBILITY OF THE EXACT-REPAIR INTERIOR POINTS

In this section, we study the infeasibility of the interior points on the optimal *functional-repair* tradeoff for $e \mid k, e \mid d, 2e < k$, similarly to [17]. We note that all interior points satisfy $(d - k + e)\beta \leq e\alpha \leq d\beta$. This can be written as $(d' - \eta + 1)\beta \leq \alpha \leq d'\beta$, where $d' = \frac{d}{e}$ and $\eta = \frac{k}{e}$. This is similar to the single erasure case with reduced parameters. The proof techniques in this section follow along similar lines as [17] and some of the proofs are relegated to the appendix.

Parameterization of the interior points. Let $\alpha = (d' - p)\beta - \theta$, namely $e\alpha = (d - ep)\beta - e\theta$ with $p \in \{0, 1, \ldots, \eta-1\}$, $\theta \in [0, \beta)$ such that $\theta = 0$ if $p = \eta - 1$. Points on the functional tradeoff satisfy

$$\mathcal{M} = e \sum_{i=0}^{\eta-1} \min(\alpha, (d' - i)\beta).$$

A. Properties of exact-repair codes

We present a set of properties that exact-repair codes, satisfying the optimal functional tradeoff, must satisfy.

Lemma 13. *For a set A of arbitrary nodes of size ej, a set L of nodes of size e such that $L \cap A = \emptyset$, we have*

$$I(W_L, W_A) = \begin{cases} 0, & j \leq p, \\ e((j - p)\beta - \theta), & p < j < \eta, \\ e\alpha, & j \geq \eta. \end{cases}$$

Proof: See Appendix D. ∎

Corollary 2. *For an arbitrary set L of size e, and a disjoint set A such that $|A| = em < k$ for some integer m, we have*

$$H(W_L|S_A^L) = H(W_L|W_A) = \min(e\alpha, (d - em)\beta).$$

Proof: From Lemma 5, we have $H(W_L|S_A^L) \leq \min(e\alpha, (d - em)\beta)$. On the other hand, from Lemma 6,

$$H(W_L|S_A^L) \geq H(W_L|W_A) = \min(e\alpha, (d - em)\beta).$$

Thus, $H(W_L|S_A^L) = H(W_L|W_A) = \min(e\alpha, (d - em)\beta)$. ∎

Lemma 14. *In the situation where node m is an arbitrary helper node assisting in the repair of a second set of arbitrary nodes L of size e, we have*

$$H(S_m^L) = \beta,$$

irrespective of the identity of the other $d - 1$ helper nodes. Moreover, for set B of size $|B| \leq d - k + e$ with $B \cap L = \emptyset$, we have

$$H(S_B^L) = |B|\beta.$$

Proof: See Appendix E. ∎

Helper node pooling. Consider a set F consisting of a collection of $f \leq d + e$ nodes (f is a multiple of e), and a subset R of the set F consisting of er' nodes, $r' \geq 1$. A helper node pooling scenario is a scenario where upon failure of any e nodes $L \subseteq R$, the d helper nodes include all the $f - e$ remaining nodes in F. The remaining $d - f + e$ helper nodes are

fixed given L. Consider a subset of nodes $M \subseteq F \backslash R$. Partition the nodes in R into arbitrary but fixed sets $R_1, R_2, \ldots, R_{r'}$, each of size e. Denote by $S_M^R = (S_M^{R_1}, \ldots S_M^{R_{r'}})$ the collective transmitted information from helper nodes M to repair $R_1, \ldots, R_{r'}$, respectively.

Lemma 15. *In the helper node pooling scenario where* $\min(\eta, \frac{f}{e}) > p + 2 \geq r'$, *for any set of e arbitrary nodes* $M \subseteq F \backslash R$, *we have*

$$H(S_M^R) \leq e(2\beta - \theta).$$

Proof: See Appendix F. ∎

Lemma 16. *In the helper node scenario where* $\min\{\eta, \frac{f}{e}\} > p + 1 \geq r'$, *for an arbitrary set of e nodes* $M \subseteq F \backslash R$, *and an arbitrary pair of set of e nodes* L_1, L_2, *it must be that*

$$H(S_M^{L_1} | S_M^{L_2}) \leq e\theta,$$

and hence

$$H(S_M^R) \leq e(\beta + (r' - 1)\theta).$$

Proof: See Appendix G. ∎

B. Non-existence proof

For interior points, $1 \leq p \leq \eta - 2$. First, we consider the interior points for which $e\alpha$ is a multiple of β. That is: $e\alpha = (d - ep)\beta, \theta = 0$.

Theorem 13. *Exact-repair codes do not exist for the interior points with $\theta = 0$.*

Proof: Consider a sub-network F consisting of $d + e$ nodes. The parameters satisfy the condition in Lemma 16. Note that by the regeneration property for any set of e nodes $L \subseteq F$, $H(W_L | S_{F-L}^L) = 0$. Moreover, for distinct $M, L_1, L_2 \subseteq F$, with $\theta = 0$, we have $H(S_M^{L_1} | S_M^{L_2}) = 0$. We partition the nodes in F into groups of size e, denoted $L_i, i = 1, 2, \ldots, d' + 1$. Then, we write

$$
\begin{aligned}
\mathcal{M} \leq H(W_F) &\leq H(S_{F-L_1}^{L_1}, \ldots, S_{F-L_{d'+1}}^{L_{d'+1}}) \\
&= H(S_{L_1}^{F-L_1}, \ldots, S_{L_{d'+1}}^{F-L_{d'+1}}) \\
&\leq \sum_{i=1}^{d'+1} H(S_{L_i}^{F-L_i}) \\
&\leq \sum_{i=1}^{d'+1} e\beta \qquad (43) \\
&= (d + e)\beta,
\end{aligned}
$$

where the inequality (43) follows from Lemma 16. On the other hand,

$$
\begin{aligned}
\mathcal{M} &= \sum_{i=0}^{d'-1} \min(e\alpha, (d - ie)\beta) \\
&= \sum_{i=0}^{d'-1} \min((d - ep)\beta, (d - ie)\beta) \\
&= 2(d - ep)\beta + \sum_{i=2}^{d'-1} \min((d - ep)\beta, (d - ie)\beta)
\end{aligned}
$$

$$
\begin{aligned}
&\geq 2(d - ep)\beta + (\eta - 2)e\beta \\
&\geq 2e\beta + (d - ep)\beta + (\eta - 2)e\beta \\
&= (d - 2e)\beta + (k - 2e - ep)\beta \\
&\geq (d - 2e)\beta,
\end{aligned}
$$

where we assume $1 \leq p \leq \eta - 2$ (non-MSMR point). Thus, $ep + 2e \leq k \leq d$. Both bounds are contradictory, thus proving the impossibility result in the case of $\theta = 0$. ∎

Theorem 14. *For any given values of $\mathcal{M}$, exact-repair regenerating codes do not exist for the parameters lying in the interior of the storage-bandwidth tradeoff when $\theta \neq 0$, except possibly for the case $p + 2 = \eta$ and $\theta \geq \frac{d - ep - e}{d - ep}\beta$.*

Proof: See Appendix H. ∎

VI. ADAPTIVE MULTI-NODE REPAIR FOR MBR CODES

In this section, we study multi-node repair for MBR codes, allowing a varying number of helpers and a varying number of failures. In Section IV, we proved that MBMR codes are not achievable for linear exact repair codes, when $2 \leq e < k$. When $e = 1$, exact MBMR codes are MBR codes and their existence is well established in the literature [43]. *Adaptive* regenerating codes possess the extra feature that the number of helpers involved in the repair process can be adaptively selected, which provides the storage system with robustness to the network varying conditions [25], [50]. Adaptive MSR codes have been constructed in [36]. On the other hand, adaptive MBR codes have been investigated in [51], in which case optimal repair means that the total repair bandwidth for each number of helpers d is the lowest possible, and is given by $\gamma = \alpha, \forall d_{\min} \leq d \leq d_{\max}$ (assuming the storage per node contains no redundancy). Here $d_{\min}, d_{\max}$ are between k and $n - 1$. It is shown in [51] that adaptive MBR codes, designed for arbitrary d, $d_{\min} \leq d \leq d_{\max}$, are equivalent to MBR codes that are designed for the worst-case number of helpers $d_{\min}$, and they satisfy optimal repair for arbitrary number of helpers $d_{\min} \leq d \leq d_{\max}$. Namely, adaptive MBR codes satisfy for any $d_{\min} \leq d \leq d_{\max}$,

$$d\beta = \alpha,$$

$$\alpha = \frac{2d_{\min}\mathcal{M}}{-k^2 + k + 2kd_{\min}} = \frac{d_{\min}\mathcal{M}}{d_{\min}k - \binom{k}{2}},$$

where the storage size α corresponds to the MBR code with $d_{\min}$ helpers.

A natural question of interest is whether there exists an MBR code that efficiently recover from varying number of failures simultaneously. In this section, we investigate the problem of repairing multiple failures in MBR codes under exact repair, for *varying number of helpers d and varying number of failures e*, such that $d_{\min} \leq d \leq d_{\max}$, $1 \leq e \leq k$, $e + d \leq n$. First, we derive a lower bound on the multi-node repair bandwidth for MBR codes, which applies to exact and functional codes. We assume that an MBR code is designed for d helpers, and we want to repair e failures. To emphasize the dependency on e, denote the total repair bandwidth by $\gamma_{\text{MBR}}(e)$.

Theorem 15. *Consider an (n, k, d, α, β) MBR regenerating code, the total repair bandwidth $\gamma_{MBR}(e)$ needed to repair any set of $1 \leq e \leq k$ nodes satisfies*

$$\gamma_{MBR}(e) \geq e\alpha - \binom{e}{2}\frac{\alpha}{d}. \tag{44}$$

Proof: Assume w.l.o.g that the first e nodes are to be repaired. From [52], at MBR point, for any set of A nodes of size $m < k$ and for $i \notin A$, we have $H(W_i|W_A) = (d-m)\beta$. Therefore,

$$
\begin{aligned}
H(W_{[e]}) &= \sum_{i=1}^{e} H(H_i|W_{[i-1]}) \\
&= \sum_{i=1}^{e}(d-i+1)\beta = (ed - \binom{e}{2})\beta.
\end{aligned}
$$

Noting that at the MBR, $\alpha = d\beta$, (44) follows. ∎

We now briefly describe a construction of adaptive MBR codes that simultaneously and efficiently repair single node failures, presented in [51]. Then, we show how to optimally repair multiple failures in this construction.

A. Adaptive single-failure MBR construction

The construction is based on product matrix codes [43], [51]. Let $\alpha = \prod_{d=d_{\min}}^{d_{\max}} d$. Define $z = \frac{\alpha}{d_{\min}}$ and construct the $(\alpha \times \alpha)$ data matrix M as

$$
M = \begin{bmatrix}
M_1 & O & \cdots & O \\
O & M_2 & \cdots & O \\
\vdots & & \ddots & \vdots \\
O & \cdots & O & M_z
\end{bmatrix},
$$

where O is a $(d_{\min} \times d_{\min})$ zero matrix and each of the submatrices M_i is filled with information symbols, and is symmetric and satisfies the structural properties of a product-matrix MBR code for parameters k and $d_{\min}$. For instance, M_i is given by

$$
M_i = \begin{bmatrix}
N_i & L_i \\
L_i^t & O'
\end{bmatrix},
$$

where N_i is a symmetric $(k \times k)$ matrix, L_i is $(k \times (d_{\min} - k))$ matrix, and O' is $(d_{\min} - k) \times (d_{\min} - k)$ zero matrix. let Ψ be an $(zn \times d_{\min})$ Vandermonde matrix, with rows denoted by ψ_j^t, for $1 \leq j \leq zn$. Then, storage node l is associated with

$$\mathbf{w}_l^t = \left[\psi_{(l-1)z+1}^t, \ldots, \psi_{lz}^t\right] M = \left[\psi_{(l-1)z+1}^t M_1, \ldots, \psi_{lz}^t M_z\right].$$

Single node repair. Denote the set of helpers by $\mathcal{H}$ such that $|\mathcal{H}| = d$ and $d_{\min} \leq d \leq d_{\max}$. Let Ω be an $(z \times z)$ matrix such that Ω^t is a Vandermonde matrix. Assume that node f fails. Let Ω_d be a $(\frac{\alpha}{d} \times z)$ matrix containing the first $\frac{\alpha}{d}$ rows of Ω. Moreover, let Φ_i be an $(\alpha \times z)$ matrix

$$
\Phi_i = \begin{bmatrix}
\psi_{(i-1)z+1} & & \\
& \ddots & \\
& & \psi_{iz}
\end{bmatrix}.
$$

Each helper node $i_j \in \mathcal{H}$ transmits $\mathbf{s}_{i_j,f}^t = \mathbf{w}_{i_j}^t \Phi_f \Omega_d^t$. After simplification, the replacement node obtains

$$\mathbf{w}_f^t \left[\Phi_{i_1}\Omega_d^t, \ldots, \Phi_{i_d}\Omega_d^t\right] = \mathbf{w}_f^t \Theta_{\mathcal{H}}.$$

Noting that $\Theta_{\mathcal{H}}$ is invertible [51], the replacement node can thus recover $\mathbf{w}_f^t$.

B. Adaptive multi-node repair in MBR codes

We state our result in the following theorem.

Theorem 16. *Adaptive single-failure MBR regenerating codes with storage per node α and arbitrary number of helpers $d_{\min} \leq d \leq d_{\max}$, presented in [51], can simultaneously and optimally repair e failures with d helpers, for all $d_{\min} \leq d \leq d_{\max}$, $1 \leq e \leq k$, $e + d \leq n$.*

Proof: Assume w.l.o.g that the first e nodes failed and d helpers are used, where $d_{\min} \leq d \leq d_{\max}$, $1 \leq e \leq k$, $e + d \leq n$. Denote the helpers by the set $\mathcal{H} = \{i_1, \ldots, i_d\}$. First, the repair of node 1 is done by contacting all the d helpers and downloading $\frac{\alpha}{d}$ symbols from each one of them, using the procedure described for single node repair. Node 2 is then repaired using only $d_{\min}$ helpers, comprising repaired node 1 and any other $d_{\min} - 1$ helpers in $\mathcal{H}$, such that each helper provides $\frac{\alpha}{d_{\min}} = z$ symbols. The same procedure is then applied repeatedly until recovering the last node e by contacting any $d_{\min} - e + 1$ helpers in $\mathcal{H}$ and using contributions from the $e-1$ already repaired nodes. The overall repair bandwidth is given by

$$d\frac{\alpha}{d} + \sum_{i=1}^{e-1} z(d_{\min} - i) = e\alpha - \binom{e}{2}\frac{\alpha}{d_{\min}},$$

which matches the bound in (44), establishing the optimality of the repair procedure. ∎

Remark 10. *Repairing e failures in an $(n, k, d_{\min}, \alpha, \beta)$ MBR code separately requires a bandwidth of size $e\alpha$. However, simultaneously repairing e failures using $d \geq d_{\min}$ reduces the bandwidth by $\binom{e}{2}\frac{\alpha}{d_{\min}}$.*

Remark 11. *The repair procedure of multiple erasures in Theorem 16 is asymmetric. However, one can always duplicate the code a sufficient number of times to achieve a symmetric repair strategy (e.g. [49]).*

VII. CONCLUSION

We studied the problem of centralized repair of multiple erasures in distributed storage systems. We explicitly characterized the optimal functional tradeoff between the repair bandwidth and the storage size per node. For instance, we obtained the expressions of the extreme points on the tradeoff, namely the minimum storage multi-node repair (MSMR) and the minimum bandwidth multi-node repair (MBMR) points. In the case of $e \geq k$, we showed that the tradeoff reduces to a single point, for which we provided a code construction achieving it. We described a general framework for converting single erasure minimum storage regenerating codes to MSMR codes. Then we applied the framework to product-matrix codes

and interference alignment codes. Furthermore, we proved that the functional MBMR point is not achievable for linear exact repair codes for $1 < e < k$. We also showed that the functional repair tradeoff is not achievable under exact repair, except for maybe a small portion near the MSMR point for $e < k, e \mid k, e \mid d$. Finally, we presented an MBR code that can adaptively and optimally repair varying number of failures with varying number of helpers.

Open problems include the generalization of the non-existence proof of linear exact-repair MBMR regenerating codes to non-linear codes. It is interesting to determine the storage and bandwidth values of an exact minimum bandwidth regenerating code. Moreover, characterization of the storage-bandwidth tradeoff for exact repair for the interior points is still not known.

APPENDIX

A. Proof of Lemma 4

We first state the following lemma which will be useful in the proof.

Lemma 17. *For fixed β, the scenario $\mathbf{u} = [e, \ldots, e, r]$ achieves the lowest final value of minimum cut:*

$$\lim_{\alpha \to +\infty} f(\mathbf{u}) \geq \lim_{\alpha \to +\infty} f([e, \ldots, e, r]), \forall \mathbf{u} \in \mathcal{P},$$

where $f(\mathbf{u})$ and $\mathcal{P}$ are defined in (4) and (5), respectively.

Proof: for a specific cut $\mathbf{u}$, we have

$$\lim_{\alpha \to +\infty} f(\mathbf{u}) = \sum_{i=1}^{g} (d - \sum_{j=1}^{i-1} u_j)\beta$$

$$= d\beta g - \beta \sum_{i=1}^{g} \sum_{j=1}^{i-1} u_j = g d\beta - \beta \sum_{i=1}^{g-1} u_i(g-i)$$

$$= \beta(dg - g \sum_{i=1}^{g-1} u_i + \sum_{i=1}^{g-1} i u_i) = \beta((d-k)g + \sum_{i=1}^{g} i u_i).$$

To obtain the smallest minimum cut value, we need to solve the following problem

$$\begin{aligned}
\underset{\mathbf{u},g}{\text{minimize}} \quad & (d-k)g + \sum_{i=1}^{g} i u_i \\
\text{subject to} \quad & 1 \leq u_i \leq e, \quad\quad\quad\quad (45) \\
& \sum_{i=1}^{g} u_i = k.
\end{aligned}$$

It can be seen that the solution to (45) is given by $\mathbf{u} = [e, \ldots, e, r]$. ∎

We now study the different functions $C_j(\alpha)$ for $j = 0, \ldots, \eta$. An example of the different functions to be analyzed is given in Figure 5, with $k = 9, d = 10, \beta = 1$. It is observed that $\mathbf{u} = [1, 3, 3, 3]$ generates the lowest cut before some threshold $\alpha^* = 5$, after which the lowest cut is generated by $\mathbf{u} = [3, 3, 3, 1]$. In the following, by analyzing $C_j(\alpha)$ for $j = 0, \ldots, \eta$, we prove that the above observation holds true in general.
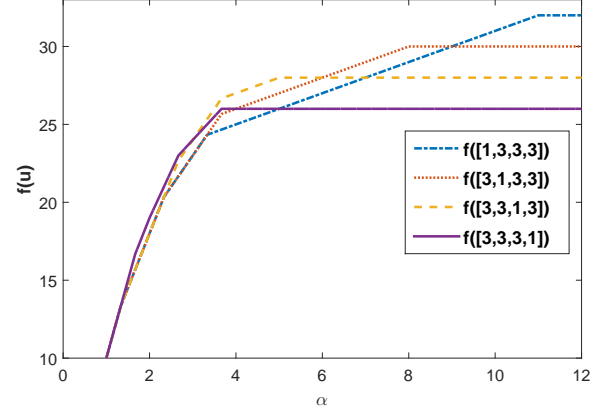

Fig. 5: Values of the cut function for different vectors $\mathbf{u}$ for $k = 9, d = 10, \beta = 1$

a) j=0: we have

$$C_0(\alpha) = \min(r\alpha, d\beta) + \sum_{i=0}^{\eta-1} \min(e\alpha, (d-r-ie)\beta)$$

$$= r \min(\alpha, \frac{d\beta}{r}) + \sum_{i=0}^{\eta-1} e \min(\alpha, \frac{(d-r-ie)\beta}{e}).$$

$C_0(\alpha)$ is a piecewise linear function with breakpoints given by $\{\frac{d-r-(\eta-1)e}{e}\beta, \frac{d-r-(\eta-2)e}{e}\beta, \ldots, \frac{d-r}{e}\beta, \frac{d}{r}\beta\}$. C_0 increases from 0 at a slope of k. Its slope is then reduced by e by the successive breakpoints and then finally by r until it levels off.

b) $1 \leq j \leq \eta$: for each j, we have

$$C_j(\alpha) = \sum_{i=0}^{j-1} \min(e\alpha, (d-ie)\beta) + \min(r\alpha, (d-je)\beta)$$

$$+ \sum_{i=j}^{\eta-1} \min(e\alpha, (d-r-ie)\beta)$$

$$= \sum_{i=0}^{j-1} e \min(\alpha, \frac{(d-ie)\beta}{e})$$

$$+ r \min(\alpha, \frac{(d-je)\beta}{r}) + \sum_{i=j}^{\eta-1} e \min(\alpha, \frac{(d-r-ie)\beta}{e}).$$

$C_j(\alpha)$ is also piecewise-linear function with non-increasing successive slopes. Its breakpoints are given by $\{\frac{d-r-(\eta-1)e}{e}\beta, \ldots, \frac{d-r-je}{e}\beta, \frac{d-(j-1)e}{e}\beta, \ldots, \frac{d}{e}\beta\} \cup \{\frac{d-je}{r}\beta\}$. The exact relative position of the breakpoint $\frac{d-je}{r}\beta$ with respect to the other breakpoints of $C_j(\alpha)$ depends on the system's parameters. However, we give a lower bound on $\frac{d-je}{r}\beta$.

$$\begin{aligned}
\frac{d-je}{r} - \frac{d-r-(j-1)e}{e} &= \frac{ed - rd - re + r^2 - j(e^2 - re)}{re} \\
&\geq \frac{(e-r)d - re + r^2 - \eta(e^2 - re)}{re} \\
&\geq \frac{(e-r)k - re + r^2 - \eta(e^2 - re)}{re} \\
&= 0,
\end{aligned}$$

where the first inequality follows by noticing that the expression is decreasing in j and letting $j = \eta$, and the second inequality follows as the corresponding expression is increasing d.

Figure 6 illustrates the relative positions of all the breakpoints of $C_0(\alpha)$ and $C_j(\alpha), j \geq 1$, where for example $\frac{d-je}{r} \in [\frac{d-r-(j-1)e}{e}, \frac{d-r-(j-2)e}{e}]$. We denote by $C_j(\infty) = \lim_{\alpha \to +\infty} C_j(\alpha)$.

Lemma 18. *For $1 \leq j \leq \eta$, there exists a point $\alpha_c(j) \in [\frac{d}{e}, \frac{d}{r}]$ such that*

$$
\begin{aligned}
C_0(\alpha_c(j)) &= C_j(\alpha_c(j)), \\
C_0(\alpha) &\leq C_j(\alpha) && \text{if } \alpha \leq \alpha_c(j), \\
C_0(\alpha) &\geq C_j(\alpha) && \text{if } \alpha \geq \alpha_c(j), \\
C_j(\alpha) &= C_j(\infty) && \text{if } \alpha \geq \alpha_c(j).
\end{aligned}
\tag{46}
$$

Proof: W.l.o.g, assume $\beta = 1$. First, we note that

$$
C_0(\alpha) = C_j(\alpha) = k\alpha \text{ for } \alpha \leq \frac{d-r-(j-1)e}{e}.
$$

Next, we analyze the behavior of each of the functions $C_0(\alpha)$ and $C_j(\alpha)$ over the successive intervals $I_i \triangleq (\frac{d-r-ie}{e}, \frac{d-r-(i-1)e}{e}]$ for $i \in \{j-1, j-2, \ldots, 1\}$. Let $x_i = \frac{d-r-ie}{e}$ and define $s_j(I_i)$ as the slope of $C_j(\alpha)$ just before $\alpha = x_i$. Consider a given interval $I_i = (x_i, x_{i-1}]$, we have

- $C_0(\alpha)$ has no breakpoint inside I_i. Thus, $C_0(\alpha)$ increases by

$$
C_0(x_{i-1}) - C_0(x_i) = s_0(I_i) - e.
$$

- $C_j(\alpha)$ has either one or two breakpoints inside I_i.
  1) in the case of $C_j(\alpha)$ has a single breakpoint inside I_i (at $\alpha = \frac{d-ie}{e}$), $C_j(\alpha)$ increases by

$$
\begin{aligned}
C_j(x_{i-1}) - C_j(x_i) &= s_j(I_i)\frac{r}{e} + (s_j(I_i) - e)\frac{e-r}{e} \\
&= s_j(I_i) - e + r.
\end{aligned}
$$

  2) in the case of $C_j(\alpha)$ has two breakpoints inside I_i, namely at $\alpha = \frac{d-je}{r}$ and $\alpha = \frac{d-ie}{e}$. Let $\Delta = \frac{d-je}{r} - \frac{d-r-ie}{e}$ (c.f. Figure 6). Assuming $\frac{d-je}{r} \leq \frac{d-ie}{e}$, then, $C_j(\alpha)$ increases by

$$
\begin{aligned}
C_j(x_{i-1}) - C_j(x_i) &= (s_j(I_i) - r)(1 - \Delta - \frac{e-r}{e}) \\
&\quad + \frac{e-r}{e}(s_j(I_i) - r - e) + s_j(I_i)\Delta \\
&= s_j(I_i) - e + \Delta r.
\end{aligned}
$$

Assuming $\frac{d-je}{r} \geq \frac{d-ie}{e}$, then, $C_j(\alpha)$ increases by

$$
\begin{aligned}
C_j(x_{i-1}) - C_j(x_i) &= \frac{r}{e}s_j(I_i) + (k-e)(\Delta - \frac{r}{e}) \\
&\quad + (s_j(I_i) - r - e)(1 - \Delta) \\
&= s_j(I_i) - e + \Delta r,
\end{aligned}
$$

which shows that the increase does not depend on the relative position of the two breakpoints.

Now that we have computed the increase increment of each C_j over I_i, we proceed to compare $C_0(\alpha)$ and $C_j(\alpha)$ for

$1 \leq j \leq \eta$.

We discuss two cases:

Case 1: Assume $\frac{d-je}{r} \in I_{j_0}$ for some $j_0 \in [1, j-1]$. j_0 may not exist, which will be discussed in the second case. Based on the above discussion, it can be seen that

$$
C_j(\alpha) \geq C_0(\alpha), \text{ for } \alpha \leq x_{j_0}.
$$

This can be seen by noticing that $\forall i < j_0, s_0(I_i) = s_j(I_i)$ and that

$$
(C_j(x_{i-1}) - C_j(x_i)) - (C_0(x_{i-1}) - C_0(x_i)) = r \geq 0.
$$

Over I_{j_0}, C_j also dominates C_0 at every point as $s_0(I_{j_0}) = s_j(I_{j_0})$ and

$$
(C_j(x_{i-1}) - C_j(x_i)) - (C_0(x_{i-1}) - C_0(x_i)) = \Delta r \geq 0.
$$

For $i > j_0$, we have $s_0(I_i) - s_j(I_i) = r$. Moreover, over each $I_i, i > j_0$, we have

$$
\begin{aligned}
&(C_j(x_{i-1}) - C_j(x_i)) - (C_0(x_{i-1}) - C_0(x_i)) \\
&= (s_j(I_i) - e + r) - (s_0(I_i) - e) = 0.
\end{aligned}
$$

Combining the last equation and the observation that $C_j(x_{j_0-1}) \geq C_j(x_{j_0-1})$, it follows that C_j continues to dominate C_0 over the successive intervals $I_i, i > j_0$. So far, we have shown that

$$
C_j(\alpha) \geq C_0(\alpha), \text{ for } \alpha \leq \frac{d-r}{e}.
$$

For $\alpha \geq \frac{d-r}{e}$, we observe that C_j increases with a slope of e and levels off at $\frac{d}{e}$ while C_0 increases at smaller slope given by r and levels off at $\frac{d}{r} > \frac{d}{e}$. Moreover, we know from Lemma 17 that C_0 levels off at a higher value than that of C_j. Thus, there exists $\alpha_c(j) \in [\frac{d}{e}, \frac{d}{r}]$ that satisfies (46).

Case 2: Assume $\frac{d-r}{e} < \frac{d-je}{r} \leq \frac{d}{r}$, then, using similar arguments as in the first case, it follows that for $\alpha \leq \frac{d-r}{e}$, $C_j(\frac{d-r}{e}) \geq C_0(\frac{d-r}{e})$. At $\alpha = \frac{d-r}{e}$, $C_j(\alpha)$ has a slope of $r + e$, which is higher than that of C_0, given by r. Thus, the slope of C_j remains higher than than of C_0 until C_j levels off. Combining these observations with the fact that C_0 levels off at a higher value, it follows that both curves will intersect only once. Moreover, the intersection at a point at which C_j has leveled off i.e., we have $\alpha_c(j) \geq \max(\frac{d}{e}, \frac{d-je}{r})$. Therefore, (46) holds also in this case. ∎

Using Lemma 18 and the fact that C_η achieves the smallest final value from Lemma 17, that is $C_\eta(\infty) \leq C_j(\infty)$, $j \in [0, \eta-1]$, it follows that (9) holds for any $j \in [0, \eta]$. Moreover, as $\alpha_c(\eta) \in [\frac{d}{e}, \frac{d}{r}]$, $\alpha_c(\eta)$ satisfies

$$
r\alpha_c(\eta) + \sum_{i=0}^{\eta-1}(d-r-ie)\beta = (\eta+1)\beta d - e\beta\frac{\eta^2+\eta}{2},
$$

which implies that

$$
r\alpha_c(\eta) + \eta(d-r-\frac{e\eta}{2}+\frac{e}{2})\beta = (\eta+1)\beta d - e\beta\frac{\eta^2+\eta}{2}.
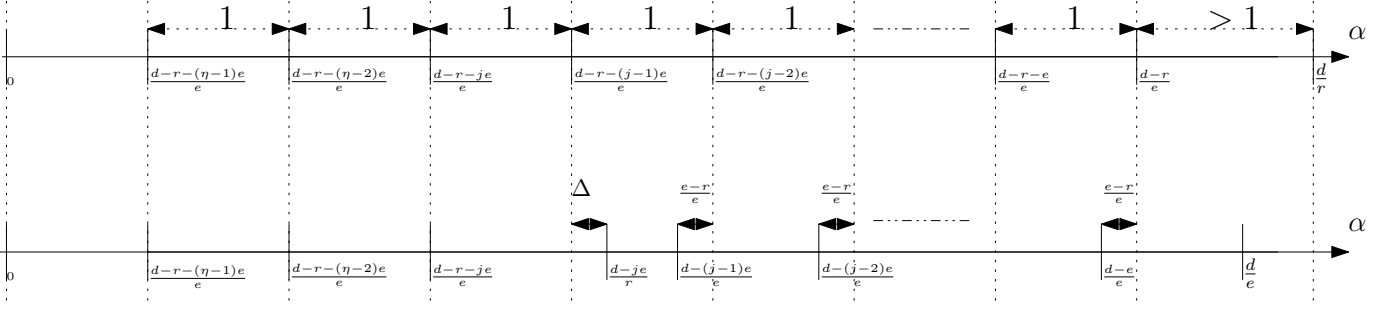$$

Simplifying the last equation yields (10).

Fig. 6: relative positions of the breakpoints of $C_0(\alpha)$ and $C_j(\alpha)$ (with $\beta = 1$)

.

B. Storage-bandwidth tradeoff expression

We start with the case $k = \eta e + r$. The optimization trade-off is

$$\begin{aligned} \underset{\alpha \geq 0}{\text{minimize}} \quad & \alpha \\ \text{subject to} \quad & C(\alpha) \geq \mathcal{M}. \end{aligned} \tag{47}$$

The constraint is a piece-wise linear function $C(\alpha)$ is given by

$$C(\alpha) = \begin{cases} (\eta+1)\beta d - e\beta\eta(\eta+1)/2, \alpha \geq \alpha_c, \\ r\alpha + \sum_{j=0}^{\eta-1} b_j, \alpha \in [\frac{b_0}{e}, \alpha_c], \\ (r+ie)\alpha + \sum_{j=i}^{\eta-1} b_j, \alpha \in [\frac{b_i}{e}, \frac{b_{i-1}}{e}], i = 1, \dots, \eta-1, \\ k\alpha, \alpha \leq \frac{b_{\eta-1}}{e}, \end{cases} \tag{48}$$

with $\alpha_c = \frac{d+\eta r - \eta e}{r}\beta$, $b_i = (d-r-ie)\beta$ and

$$\sum_{j=i}^{\eta-1} b_j = \beta(\eta-i)(d-r-\frac{e(\eta-1+i)}{2}) \tag{49}$$

$$= \gamma \frac{(\eta-i)(-2r+e+2d-\eta e - ei)}{2d} \triangleq \gamma g_r(i), \tag{50}$$

such that

$$g_r(i) = \frac{(\eta-i)(-2r+e+2d-\eta e - ei)}{2d}.$$

The expression $C(\alpha)$ increases from 0 to a maximum value given by $\beta((\eta+1)d - \binom{\eta+1}{2})$. To solve (47), we let $\alpha^* = C^{-1}(\mathcal{M})$ under the condition $\mathcal{M} \leq \beta((\eta+1)d - \binom{\eta+1}{2})$. Therefore, we obtain,

$$\alpha^* = \begin{cases} \frac{\mathcal{M}}{k}, & \mathcal{M} \in [0, \frac{kb_{\eta-1}}{e}] \\ \frac{\mathcal{M} - \sum_{j=i}^{\eta-1} b_j}{r+ie}, & \mathcal{M} \in [(r+ie)\frac{b_i}{e} + \sum_{j=i}^{\eta-1} b_j, (r+ie)\frac{b_{i-1}}{e} + \sum_{j=i}^{\eta-1} b_j], \\ & \qquad \text{for } i = \eta-1, \dots 1, \\ \frac{\mathcal{M} - \sum_{j=0}^{\eta-1} b_j}{r}, & \mathcal{M} \in [\frac{b_0 r}{e} + \sum_{j=0}^{\eta-1} b_j, r\alpha_c + \sum_{j=0}^{\eta-1} b_j], \end{cases}$$

with

$$\frac{rb_i}{e} + ib_i + \sum_{j=i}^{\eta-1} b_j$$
$$= \frac{-\eta^2 e^2 + \eta e^2 - 2aer + 2dae - e^2 i^2 - e^2 i - 2eir - 2r^2 + 2dr)}{2de}\gamma$$

$$= \frac{-k^2 - r^2 + e(k-r) + 2kd - e^2(i^2+i) - 2ier}{2ed}\gamma$$

$$\triangleq \gamma \frac{\mathcal{M}}{f(i)},$$

such that

$$f_r(i) = \frac{2ed\mathcal{M}}{-k^2 - r^2 + e(k-r) + 2kd - e^2(i^2+i) - 2ier}.$$

Therefore, fixing $\mathcal{M}$ and varying γ, we write

$$\alpha^* = \begin{cases} \frac{\mathcal{M}}{k}, & \mathcal{M} \in [0, \frac{kg_r(\eta-1)\gamma}{e}], \\ \frac{\mathcal{M} - \gamma g_r(i)}{r+ie}, & \mathcal{M} \in [\frac{\gamma \mathcal{M}}{f_r(i)}, \frac{\gamma \mathcal{M}}{f_r(i-1)}], i = \eta-1, \dots 1, \\ \frac{\mathcal{M} - \gamma g_r(0)}{r}, & \mathcal{M} \in [\frac{\gamma \mathcal{M}}{f_r(0)}, (g_r(0) + \frac{d+ar-ae}{d})\gamma]. \end{cases} \tag{51}$$

As a function of γ, after simplifications, we obtain the expression of α^* as in Theorem 3. We note that there are η piecewise linear portions on the curve. Moreover, the minimum bandwidth point γ_{MBMR} is given by

$$\gamma_{\text{MBMR}} = \frac{\mathcal{M}}{g_r(0) + \frac{d+ar-ae}{d}} = \frac{d\mathcal{M}}{d(\eta+1) - e\binom{\eta+1}{2}}. \tag{52}$$

The expression of α_{MBMR} is given by

$$\alpha_{\text{MBMR}} = \frac{\mathcal{M} - \gamma_{\text{MBMR}} g(0)}{r} = \gamma_{\text{MBMR}} \frac{d+\eta r - e\eta}{rd}.$$

in the case of $e \mid k$, we have $r = 0$. The expression of the tradeoff is obtained from (51) by setting $r = 0$ and eliminating the last line. We note that in this case, there are $\eta - 1$ piecewise linear portions on the trade-off curve.

C. Derivations of (29), (30) and (31) for IA codes

Consider two systematic nodes $l_1, l_2 \in [k], l_1 \neq l_2$, starting from (25) and noting that $V'^t U' = \kappa P'$, we obtain after simplification

$$r_{l_1,l_2} = \mathbf{v}_{l_2}'^t \mathbf{w}_{l_1}$$

$$= \mathbf{v}_{l_2}'^t (U' - \frac{\kappa^2}{1+\kappa} V \mathbf{e}_{l_1} \mathbf{e}_{l_1}^t P') \begin{bmatrix} \bar{s}_{1,l_1} - \sum_{j \neq l_1} p_{j,1} r_{j,l_1} \\ \vdots \\ \bar{s}_{k,l_1} - \sum_{j \neq l_1} p_{j,k} r_{j,l_1} \end{bmatrix}$$

$$= \kappa \mathbf{e}_{l_2}^t P' \begin{bmatrix} \bar{s}_{1,l_1} - \sum_{j \neq l_1} p_{j,1} r_{j,l_1} \\ \vdots \\ \bar{s}_{k,l_1} - \sum_{j \neq l_1} p_{j,k} r_{j,l_1} \end{bmatrix}$$

$$= \sum_{j\in[k]}(\kappa P'_{l_2,j})\bar{s}_{j,l_1} - \kappa r_{l_2,l_1}.$$

Proceeding in a similar way, for a systematic node $l \in [k]$ and a parity node $m \in [k]$, starting from (26), we obtain

$$\bar{s}_{m,l} = \mathbf{v}'^{t}_{l}\bar{\mathbf{w}}_m$$

$$= \mathbf{v}'^{t}_{l}((1-\kappa^2)V + (1+\kappa)U'\mathbf{e}_m\mathbf{e}_m^t P^t)\begin{bmatrix} s_{1,m} + \frac{\kappa^2}{1-\kappa^2}\sum_{j\neq m}P'_{1,j}\bar{r}_{j,m} \\ \vdots \\ s_{k,m} + \frac{\kappa^2}{1-\kappa^2}\sum_{j\neq m}P'_{k,j}\bar{r}_{j,m} \end{bmatrix}$$

$$= ((1-\kappa^2)\mathbf{e}_l^t + (1+\kappa)\kappa P'_{l,m}\mathbf{e}_m^t P^t)\begin{bmatrix} s_{1,m} + \frac{\kappa^2}{1-\kappa^2}\sum_{j\neq m}P'_{1,j}\bar{r}_{j,m} \\ \vdots \\ s_{k,m} + \frac{\kappa^2}{1-\kappa^2}\sum_{j\neq m}P'_{k,j}\bar{r}_{j,m} \end{bmatrix}$$

$$= (1 - \kappa^2 + \kappa(1+\kappa)P'_{l,m}P_{l,m})s_{l,m}$$
$$+ \sum_{j\in[k]\setminus l}(\kappa(1+\kappa)P'_{l,m}P_{j,m})s_{j,m} + \sum_{j\in[k]}(\kappa^2 P'_{l,j})\bar{r}_{j,m}.$$

Finally, consider two parity nodes $m_1, m_2 \in [k], m_1 \neq m_2$, starting from (26), we obtain

$$\bar{r}_{m_1,m_2} = \mathbf{u}^t_{m_2}\bar{\mathbf{w}}_{m_1}$$
$$= \mathbf{u}^t_{m_2}((1-\kappa^2)V$$

$$+ (1+\kappa)U'\mathbf{e}_{m_1}\mathbf{e}_{m_1}^t P^t)\begin{bmatrix} s_{1,m_1} + \frac{\kappa^2}{1-\kappa^2}\sum_{j\neq m_1}P'_{1,j}\bar{r}_{j,m_1} \\ \vdots \\ s_{k,m_1} + \frac{\kappa^2}{1-\kappa^2}\sum_{j\neq m_1}P'_{k,j}\bar{r}_{j,m_1} \end{bmatrix}$$

$$= \frac{1-\kappa^2}{\kappa}\mathbf{e}_{m_2}^t P^t\begin{bmatrix} s_{1,m_1} + \frac{\kappa^2}{1-\kappa^2}\sum_{j\neq m_1}P'_{1,j}\bar{r}_{j,m_1} \\ \vdots \\ s_{k,m_1} + \frac{\kappa^2}{1-\kappa^2}\sum_{j\neq m_1}P'_{k,j}\bar{r}_{j,m_1} \end{bmatrix}$$

$$= \sum_{j\in[k]}(\frac{1-\kappa^2}{\kappa}P_{j,m_2})s_{j,m_1} + \kappa\bar{r}_{m_2,m_1}.$$

D. Proof of Lemma 13

Proof: First, we note that when $j \geq \eta$, $I(W_L, W_A) = H(W_L) - H(W_L|A) = H(W_L) = e\alpha$. In the following, we assume $j < \eta$. We write

$$I(W_L, W_A) = H(W_L) - H(W_L|A)$$
$$= e\alpha - \min(e\alpha, (d - je)\beta) \qquad (53)$$
$$= e(\alpha - \min(\alpha, (d' - j)\beta))$$
$$= e(\alpha - (d' - j)\beta)^+$$
$$= e((j - p)\beta - \theta)^+,$$

where we use the notation $(x)^+ \triangleq \max(x, 0)$. Here (53) follows from Lemma 6. ∎

E. Proof of Lemma 14

Proof: Partition the set of d helpers into A and B such that $|A| = k - e$ and $|B| = d - k + e$, such that $m \in B$. We have $H(W_L|S_A^L) = \min(e\alpha, (d - k + e)\beta) = (d - k + e)\beta$, as $e\alpha \geq (d - k + e)\beta$ for all points on the tradeoff. Moreover, exact repair requires $H(W_L|S_A^L, S_B^L) = 0$. Thus, $H(S_B^L) \geq$

$(d - k + e)\beta$. This implies $H(S_B^L) = (d - k + e)\beta$. Moreover, it must hold that $H(S_m^L) = \beta$ in addition to S_m^L and $S_{m'}^L$ being independent if $m \neq m'$. Moreover, by choosing $M \subseteq B$, one obtains $H(S_M^L) = e\beta$. ∎

F. Proof of Lemma 15

Proof: If the statement holds true for some f, r', then it also holds true for all $f' \geq f$ and $r'' \leq r'$. Thus, for the proof, we only need to consider $F = R \cup M, |F| = f = e(p+3), |R| = r'e = (p+2)e, |M| = e$.

Consider repair of an arbitrary set of e nodes $L \subseteq R$, where the set of helpers include M and the $e(p+1)$ remaining nodes in R. Then, we write

$$I(S_M^L; W_R) = I(S_M^L; W_L, W_{R-L})$$
$$= I(S_M^L; W_{R-L}) + I(S_M^L; W_L|W_{R-L})$$
$$\geq I(S_M^L; W_L|W_{R-L})$$
$$= H(W_L|W_{R-L}) - H(W_L|W_{R-L}, S_M^L)$$
$$\geq H(W_L|W_{R-L}) - H(W_L|S_{R-L}^L, S_M^L)$$
$$= \min(e\alpha, (d - e(p+1))\beta)$$
$$\quad - \min(e\alpha, (d - e(p+2))\beta) \qquad (54)$$
$$= (d - e(p+1))\beta - (d - e(p+2))\beta = e\beta.$$

Here (54) follows from Lemma 6 and Corollary 2. Then, we obtain

$$H(S_M^L|W_R) = H(S_M^L) - I(S_M^L; W_R) \leq e\beta - e\beta = 0.$$

Hence, $H(S_M^L|W_R) = 0$. Since L is arbitrary, it follows that $H(S_M^R|W_R) = 0$. It follows from Lemma 13 that

$$H(S_M^R) = I(S_M^R; W_R) \leq I(W_M; W_R) = e(2\beta - \theta).$$

Hence the proof is completed. ∎

G. Proof of Lemma 16

Proof: The set is R assumed to consist of $|R| = er' = e(p+1)$ nodes, and the set F is such that $F = R \cup \{M\}$, $|M| = e$. Similar to Lemma 15,

$$I(S_M^L; W_R) \geq H(W_L|W_{R-L}) - H(W_L|S_{R-L}^L, S_M^L)$$
$$= \min(e\alpha, (d - (r' - 1)e)\beta) - \min(e\alpha, (d - r'e)\beta)$$
$$= (d - pe)\beta - e\theta - (d - (p+1)e)\beta$$
$$= e(\beta - \theta).$$

Then, it must be that

$$H(S_M^L|W_R) = H(S_M^L) - I(S_M^L; W_R) \leq e\beta - e(\beta - \theta) = e\theta. \qquad (55)$$

Note that the last inequality holds for any set $L \subseteq R$. Next, consider $L_1, L_2 \subseteq R$. For this, consider

$$H(S_M^{L_1}, S_M^{L_2}) = I(W_R; S_M^{L_1}, S_M^{L_2}) + H(S_M^{L_1}, S_M^{L_2}|W_R)$$
$$\leq I(W_R; W_M) + H(S_M^{L_1}, S_M^{L_2}|W_R)$$
$$= I(W_R; W_M) + H(S_M^{L_1}|W_R) + H(S_M^{L_2}|W_R, S_M^{L_1})$$
$$\leq e(\beta - \theta) + e\theta + e\theta = e(\beta + \theta), \qquad (56)$$

where the last inequality follows from Lemma 13 and (55). Then, we have

$$H(S_M^{L_1}|S_M^{L_2}) = H(S_M^{L_1}, S_M^{L_2}) - H(S_M^{L_2})$$
$$= H(S_M^{L_1}, S_M^{L_2}) - e\beta$$
$$\leq e(\beta + \theta) - e\beta = e\theta,$$

where the first equality follows from Lemma 14. Finally, partitioning the nodes in R into sets $R_1, R_2, \ldots, R_{r'}$ of size e, it follows

$$H(S_M^R) \leq H(S_M^{R_1}) + \sum_{i=2}^{r'} H(S_M^{R_i}|S_M^{R_{i-1}}) \leq e\beta + e(r'-1)\theta.$$

Thus the proof is completed. ∎

H. Proof of Theorem 14

Proof: Take a subnetwork F of $d+e$ nodes. Let $L, M \subseteq F$ be two disjoint groups of e nodes. Partition the $d-e$ remaining nodes into two sets, A of cardinality ep and B of cardinality $d - ep - e$. Exact repair requires

$$H(W_L|S_A^L, S_B^L, S_M^L) = 0,$$
$$H(W_M|S_A^M, S_B^M, S_L^M) = 0.$$

It follows that

$$H(W_L, W_M|W_A, S_B^L, S_B^M, S_M^L)$$
$$= H(W_L|W_A, S_B^L, S_B^M, S_M^L) + H(W_M|W_L, W_A, S_B^L, S_B^M, S_M^L)$$
$$= 0.$$

Therefore, we have

$$H(S_B^L, S_B^M, S_M^L) \geq H(W_L, W_M|W_A)$$
$$= H(W_L|W_A) + H(W_M|W_A, W_L)$$
$$= H(W_L) - I(W_L; W_A) + H(W_M)$$
$$\quad - I(W_M; W_A, W_L)$$
$$= e\alpha - 0 + e\alpha - e(\beta - \theta) \quad (57)$$
$$= 2e\alpha - e\beta + e\theta$$
$$= 2((d-ep)\beta - e\theta) - e\beta + e\theta$$
$$= (2d - 2ep - e)\beta - e\theta. \quad (58)$$

Here (57) follows from Lemma 13. We note that the lower bound does not depend on whether d is a multiple of e. Next, we obtain an an upper bound on the same quantity.

Partition B into sets of size e, denoted by L_i. We will use $R = L \cup M, r' = 2$, in the helper node pooling.

case: $p + 2 < \eta$: In this case, the parameters satisfy the condition in Lemma 15.

$$H(S_B^L, S_B^M, S_M^L) \leq \sum_{L_i \in B} H(S_{L_i}^L, S_{L_i}^M) + H(S_M^L)$$
$$\leq \sum_{L_i \in B} e(2\beta - \theta) + e\beta \quad (59)$$
$$= (d - pe - e)(2\beta - \theta) + e\beta$$
$$= (2d - 2ep - e)\beta - (d - ep - e)\theta, \quad (60)$$

where the inequality (59) is obtained using Lemma 14 and Lemma 15. Equations (58) and (60) are in contradiction if

$d - ep - e > e \iff d > e(p+2)$, which is true as $d \geq k = ae > (p+2)e$.

case: $p + 2 = \eta$: In this case, Lemma 16 is used to derive an upper bound on $H(S_B^L, S_B^M, S_M^L)$. Lemma 16 does not hold if $\eta = 2$. It holds for $\eta > 2 \iff k > 2e$. Thus, we consider $k > 2e$. We have

$$H(S_B^L, S_B^M, S_M^L) \leq \sum_{L_i \in B} H(S_{L_i}^L, S_{L_i}^M) + H(S_M^L)$$
$$\leq \sum_{L_i \in B} e(\beta + \theta) + e\beta$$
$$= (d - ep)\beta + (d - ep - e)\theta. \quad (61)$$

Equations (58) and (61) are in contradiction when $\theta < \frac{d-ep-e}{d-ep}\beta$. ∎

References

[1] M. Zorgui and Z. Wang, "Centralized multi-node repair in distributed storage," in *2016 54th Annual Allerton Conference on Communication, Control, and Computing (Allerton)*, Sept 2016, pp. 617–624.

[2] ——, "Centralized multi-node repair for minimum storage regenerating codes," in *IEEE International Symposium on Information Theory (ISIT), 2017*, 2017, pp. 2213–2217.

[3] S. Ghemawat, H. Gobioff, and S.-T. Leung, "The google file system," in *ACM SIGOPS operating systems review*, vol. 37, no. 5. ACM, 2003, pp. 29–43.

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

[5] R. Ahlswede, N. Cai, S.-Y. Li, and R. W. Yeung, "Network information flow," *IEEE Trans. Inf. Theory*, vol. 46, no. 4, pp. 1204–1216, 2000.

[6] T. Ho, M. Médard, R. Koetter, D. R. Karger, M. Effros, J. Shi, and B. Leong, "A random linear network coding approach to multicast," *IEEE Trans. Inf. Theory*, vol. 52, no. 10, pp. 4413–4430, 2006.

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

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

[9] Y. Wu and A. G. Dimakis, "Reducing repair traffic for erasure coding-based storage via interference alignment," in *IEEE International Symposium on Information Theory*, 2009, pp. 2276–2280.

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

[11] Z. Wang, I. Tamo, and J. Bruck, "Explicit minimum storage regenerating codes," *IEEE Trans. Inf. Theory*, vol. 62, no. 8, pp. 4466–4480, Aug 2016.

[12] A. S. Rawat, O. O. Koyluoglu, and S. Vishwanath, "Progress on high-rate MSR codes: Enabling arbitrary number of helper nodes," in *Information Theory and Applications Workshop (ITA), 2016*. IEEE, 2016, pp. 1–6.

[13] S. Goparaju, A. Fazeli, and A. Vardy, "Minimum storage regenerating codes for all parameters," *IEEE Trans. Inf. Theory*, vol. 63, no. 10, pp. 6318–6328, 2017.

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

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

[16] M. Ye and A. Barg, "Explicit constructions of optimal-access MDS codes with nearly optimal sub-packetization," *IEEE Trans. Inf. Theory*, vol. 63, no. 10, pp. 6307–6317, 2017.

[17] N. B. Shah, K. V. Rashmi, P. V. Kumar, and K. Ramchandran, "Distributed storage codes with repair-by-transfer and nonachievability of interior points on the storage-bandwidth tradeoff," *IEEE Trans. Inf. Theory*, vol. 58, no. 3, pp. 1837–1852, March 2012.

[18] I. M. Duursma, "Outer bounds for exact repair codes," *arXiv preprint arXiv:1406.4852*, 2014.

[19] ——, "Shortened regenerating codes," *IEEE Trans. Inf. Theory*, 2018.

[20] S. Mohajer and R. Tandon, "New bounds on the (n, k, d) storage systems with exact repair," in *IEEE International Symposium on Information Theory (ISIT)*, 2015, pp. 2056–2060.

[21] B. Sasidharan, K. Senthoor, and P. V. Kumar, "An improved outer bound on the storage-repair-bandwidth tradeoff of exact-repair regenerating codes," in *IEEE International Symposium on Information Theory (ISIT)*, 2014, pp. 2430–2434.

[22] R. Bhagwan, K. Tati, Y. Cheng, S. Savage, and G. M. Voelker, "Total recall: System support for automated availability management." in *NSDI*, vol. 4, 2004, pp. 25–25.

[23] A. S. Rawat, O. O. Koyluoglu, and S. Vishwanath, "Centralized repair of multiple node failures with applications to communication efficient secret sharing," *IEEE Trans. Inf. Theory*, vol. 64, no. 12, pp. 7529–7550, 2018.

[24] P. Hu, C. W. Sung, and T. H. Chan, "Broadcast repair for wireless distributed storage systems," in *2015 10th International Conference on Information, Communications and Signal Processing (ICICS)*. IEEE, 2015, pp. 1–5.

[25] A.-M. Kermarrec, N. Le Scouarnec, and G. Straub, "Repairing multiple failures with coordinated and adaptive regenerating codes," in *International Symposium on Network Coding (NetCod), 2011*. IEEE, 2011, pp. 1–6.

[26] K. W. Shum and Y. Hu, "Cooperative regenerating codes," *IEEE Trans. Inf. Theory*, vol. 59, no. 11, pp. 7229–7258, Nov 2013.

[27] J. Li and B. Li, "Cooperative repair with minimum-storage regenerating codes for distributed storage," in *INFOCOM, 2014 Proceedings IEEE*. IEEE, 2014, pp. 316–324.

[28] M. Ye and A. Barg, "Cooperative repair: Constructions of optimal mds codes for all admissible parameters," *IEEE Trans. Inf. Theory*, pp. 1–1, 2018.

[29] A. Wang and Z. Zhang, "Exact cooperative regenerating codes with minimum-repair-bandwidth for distributed storage," in *INFOCOM, 2013 Proceedings IEEE*. IEEE, 2013, pp. 400–404.

[30] J. Chen and K. W. Shum, "Repairing multiple failures in the Suh-Ramchandran regenerating codes," in *IEEE International Symposium on Information Theory (ISIT)*, July 2013, pp. 1441–1445.

[31] P. Gopalan, C. Huang, H. Simitci, and S. Yekhanin, "On the locality of codeword symbols," *IEEE Trans. Inf. Theory*, vol. 58, no. 11, pp. 6925–6934, Nov 2012.

[32] 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.

[33] N. Prakash, V. Lalitha, and P. V. Kumar, "Codes with locality for two erasures," in *IEEE International Symposium on Information Theory (ISIT)*, June 2014, pp. 1962–1966.

[34] W. Song and C. Yuen, "Locally repairable codes with functional repair and multiple erasure tolerance," *arXiv preprint arXiv:1507.02796*, 2015.

[35] Z. Wang, I. Tamo, and J. Bruck, "Optimal rebuilding of multiple erasures in MDS codes," *IEEE Trans. Inf. Theory*, vol. 63, no. 2, pp. 1084–1101, 2017.

[36] M. Ye and A. Barg, "Explicit constructions of high-rate MDS array codes with optimal repair bandwidth," *IEEE Trans. Inf. Theory*, vol. 63, no. 4, pp. 2001–2014, 2017.

[37] R. Li, J. Lin, and P. P. Lee, "Enabling concurrent failure recovery for regenerating-coding-based storage systems: From theory to practice," *IEEE Trans. Comput.*, vol. 64, no. 7, pp. 1898–1911, 2015.

[38] V. Guruswami and M. Wootters, "Repairing reed-solomon codes," *IEEE Trans. Inf. Theory*, vol. 63, no. 9, pp. 5684–5698, 2017.

[39] H. Dau, I. Duursma, H. M. Kiah, and O. Milenkovic, "Repairing reed-solomon codes with multiple erasures," *IEEE Trans. Inf. Theory*, 2018.

[40] ——, "Repairing reed-solomon codes with two erasures," in *IEEE International Symposium on Information Theory (ISIT)*, 2017, pp. 351–355.

[41] J. Mardia, B. Bartan, and M. Wootters, "Repairing multiple failures for scalar mds codes," *IEEE Trans. Inf. Theory*, 2018.

[42] M. Ye and A. Barg, "Repairing Reed-Solomon codes: Universally achieving the cut-set bound for any number of erasures," *arXiv preprint arXiv:1710.07216*, 2017.

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

[44] Y. Wu, A. G. Dimakis, and K. Ramchandran, "Deterministic regenerating codes for distributed storage," in *Allerton Conference on Control, Computing, and Communication*. Citeseer, 2007, pp. 1–5.

[45] H. Zhang, H. Li, K. W. Shum, H. Hou, and S. R. Li, "Concurrent regenerating codes," *IET Communications*, vol. 11, no. 3, pp. 362–369, 2017.

[46] D. E. Knuth, *The Art of Computer Programming, Volume 1 (3rd Ed.): Fundamental Algorithms*. Redwood City, CA, USA: Addison Wesley Longman Publishing Co., Inc., 1997.

[47] N. Alon and M. Tarsi, "Combinatorial nullstellensatz," *Combinatorics Probability and Computing*, vol. 8, no. 1, pp. 7–30, 1999.

[48] M. Zorgui and Z. Wang, "Centralized multi-node repair regenerating codes," in *IEEE Trans. Inf. Theory*, 2019. [Online]. Available: https://github.com/Marwen-Zorgui/Centralized_repair_IA

[49] M. Elyasi, S. Mohajer, and R. Tandon, "Linear exact repair rate region of (k+ 1, k, k) distributed storage systems: A new approach," in *IEEE International Symposium on Information Theory (ISIT)*, 2015, pp. 2061–2065.

[50] V. Aggarwal, C. Tian, V. A. Vaishampayan, and Y.-F. R. Chen, "Distributed data storage systems with opportunistic repair," in *INFOCOM, 2014 Proceedings IEEE*. IEEE, 2014, pp. 1833–1841.

[51] K. Mahdaviani, A. Khisti, and S. Mohajer, "Bandwidth adaptive & error resilient regenerating codes with minimum repair bandwidth," in *IEEE International Symposium on Information Theory (ISIT)*, 2016, pp. 235–239.

[52] K. Rashmi, N. B. Shah, P. V. Kumar, and K. Ramchandran, "Explicit construction of optimal exact regenerating codes for distributed storage," in *47th Annual Allerton Conference on Communication, Control, and Computing, 2009. Allerton 2009*. IEEE, 2009, pp. 1243–1249.

Marwen Zorgui (S'13) received the Engineering degree from the Ecole Polytechnique de Tunisie, Tunisia, in 2013, the M.S. degree in Electrical Engineering from King Abdullah University of Science and Technology, Saudi Arabia, in 2015. He is now pursuing his Ph.D. at the University of California, Irvine, CA, USA. His research interests include coding theory and information theory.

Zhiying Wang (S'09–M'13) received the B.Sc. degree in Information Electronics and Engineering from Tsinghua University in 2007, M. Sc. and Ph.D degrees in Electrical Engineering from California Institute of Technology in 2009 and 2013, respectively. She was a postdoctoral fellow in Department of Electrical Engineering, Stanford University. She is currently Assistant Professor at Center for Pervasive Communications and Computing, University of California, Irvine. Dr. Wang is the recipient of NSF Center for Science of Information (CSoI) Postdoctoral Research Fellow, 2013. She received IEEE Communication Society Data Storage Best Paper Award. Her research focuses on information theory, coding theory, with an emphasis on coding for data storage.