Centralized Multi-Node Repair in Distributed Storage

Marwen Zorgui, and Zhiying Wang
Center for Pervasive Communications and Computing (CPCC)
University of California, Irvine, USA
{mzorgui,zhiying}@uci.edu

***Abstract*—In distributed storage systems, multiple storage node failures are frequent and efficiently recovering them is crucial for high system performance. In this work, we consider the problem of repairing multiple failures in a centralized way, which can be desirable in many data storage configurations. We first establish the tradeoff between the repair bandwidth and the storage size for functional repair. Using a graph-theoretic approach, the optimal tradeoff is identified as the solution to an integer optimization problem, for which we derive a closed-form expression. 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. Expressions of the extreme points, namely the minimum storage multi-node repair (MSMR) and minimum bandwidth multi-node repair (MBMR) points, are also derived. Furthermore, we prove that functional MBMR point is not achievable for linear exact repair codes. Finally, for $e \mid k$ 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 repair.**

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 [1]. However, for the same redundancy factor, replication systems fall short on providing the highest level of reliability. On the other hand, erasure codes are optimal in terms of the redundancy-reliability tradeoff. In erasure codes, a file of size $\mathcal{M}$ is divided into k pieces, 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 data reliability for the given redundancy. However, traditional erasure codes suffer from high repair bandwidth. In case of a single node erasure, they require to download 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 not practical 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 [2] proposed a new class of erasure codes, called regenerating codes, that optimally solve the repair bandwidth problem. Interestingly, the authors in [2] 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, are participating 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

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

The equation 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.

Following the seminal work in [2], there has been a flurry of interest in designing practical regenerating codes achieving the optimal tradeoff, focusing mainly on the extreme MSR and MBR points [3]–[11]. The authors in [12] presented a product-matrix framework that allows design of MBR codes for any value of d and design of MSR codes for $d \geq 2k-2$. The product-matrix construction enjoys simple encoding and decoding and ensures optimal repair of all nodes.

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 erasure. Moreover, many systems [13] apply a lazy repair strategy, which seeks to limit the repair cost of erasure codes: instead of immediately repairing every single failure, one waits until e erasures occur, then, the repair is done by downloading the equivalent of the total information in the system to regenerate the erased nodes.

In this work, we consider the repair problem of multiple erasures in a centralized manner. The framework requires the content of any k out of n nodes in the system to be sufficient to reconstruct the entire data. Upon failure of e nodes in the system, the repair is carried out by contacting any d nodes (helpers) out of the $n-e$ available nodes, and downloading β amount of information from each of the d helpers. Our 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.

The centralized repair framework is interesting in many practical situations. 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 [13]. For instance, one can think of a rack-based node placement architecture [14] in which failures frequently occur to nodes corresponding to a particular rack. In this scenario, a centralized repair of the entire rack is favorable to repairing the rack on per-node basis. Furthermore, [14] showed that a centralized repair framework can have interesting applications to 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. [15]). For the above reasons, we believe that characterizing the repair-bandwidth tradeoff under the centralized repair framework is important from both, an information-theoretic and also a practical perspective.

A. Related work

Cooperative regenerating codes (also known as coordinated regenerating codes) have been studied to address the repair of multiple erasures [16], [17]. 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. The repair is carried out in a distributed way. Cooperative regenerating codes achieving the extreme points on the cooperative tradeoff have been developed: minimum storage cooperative regenerating (MSCR) codes [18] and minimum bandwidth cooperative regeneration (MBCR) codes [19].

The problem of centralized repair has been considered in [20], in which the authors restricted themselves to MDS codes, corresponding to the point of minimum required storage per node. [20] 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 [21], the authors proved that Zigzag codes, which are MDS code designed initially for repairing optimally single erasures [11], can also be used to optimally repair multiple erasures in a centralized manner. In [14], the authors independently proved that multiple failures can be repaired in Zigzag codes with optimal bandwidth. Moreover, [14] 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. Based on that, the authors derived lower bound on β for systems having a certain entropy accumulation property and then showed achievability of the minimum bandwidth using MBCR codes. However, the optimal storage per node size α is not known under these codes. In [22], the authors provided an explicit MDS code construction that provide optimal repair for all $e \leq n-k$ and $k \leq d \leq n-k$. The authors in [15] studied the problem of broadcast repair for wireless distributed storage which is equivalent to the model we study in this paper.

B. Contributions of the paper

In this paper, we first establish the tradeoff between the repair bandwidth and the storage size for functional repair where the repaired nodes are not necessarily the same as the failed nodes. We obtain the tradeoff using information flow graphs. 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. Furthermore, we prove that functional minimum bandwidth multi-node repair point is not achievable for linear exact repair codes, while linear codes achieve such point for single erasure [12]. Finally, we show that the functional repair tradeoff is not achievable under exact repair, except for maybe a small portion near the minimum storage multi-node repair point, which parallels the results for single erasure repair [23], for $e \mid k, e \mid d$.

The remainder of the paper is organized as follows. A description of the system model is provided in Section II. The analysis of the functional tradeoff is detailed in Section III. Section IV describes our code construction in case $e \geq k$. We prove the non-achievability of MBMR codes under linear exact repair in Section V. The non-achievabilty of the interior points under exact repair is investigated in Section VI. Section VII draws conclusions. Finally, some of the proofs are relegated to Section VIII.

II. System model

The notations $e \mid k$ and $e \nmid k$ are used to denote whether k is a multiple of e, or not, respectively. $\mathbf{u} = [u_1, \ldots, u_m]$ denotes a vector of length m. For a set A, $|A|$ denotes the size of A. We write $[i] = \{1, \ldots, i\}$ for any integer $i \geq 1$.

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. 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 $d \geq k$ helpers 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. The repair bandwidth is given by β. 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. 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 tradeoff, the minimum bandwidth mutli-node repair (MBMR) point has the minimum possible β, and the minimum storage mutli-node repair (MSMR) point has the minimum possible α.

III. Functional storage-bandwidth tradeoff

A. Information flow graphs

Similar to [2], 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. An IFG has different kinds of nodes. It contains a single source node s that represents the source of the data object. Each storage node i 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 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 nodes, each with storage size α connected to the source node with edges of capacity ∞. The IFG evolves with time. Upon failure of e nodes, e new nodes join simultaneously the system. Each of the replacement nodes x^j 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 $x^i_{virtual}$ that links the d helpers to the new storage nodes. Likewise, the virtual node consists of an input node $x^i_{virtual,in}$ and an output node $x^i_{virtual,out}$. The input node $x^i_{virtual,in}$ is connected to the d helpers with edges each of capacity β. The output node $x^i_{virtual,out}$ is connected to the input node $x^i_{virtual,in}$ with an edge of capacity $e\alpha$, reflecting to the overall size of the data to be stored in the new replacement nodes. The output node $x^i_{virtual,out}$ is then connected to the input nodes x^j_{in} of the replacement nodes, with edges of capacity ∞.

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 nodes in G. The set of all data collector nodes in a graph G is denoted by $\mathrm{DC}(G)$. For an IFG $G \in \mathcal{G}$ and a data collector $t \in \mathrm{DC}(G)$, the minimum min-cut value separating the source node s and the data collector t is denoted by $\mathrm{mincut}_G(s,t)$.

B. 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 [2]. Celebrated results from network coding [24], [25] are then invoked to establish the fundamental limits of the repair problem.

Determining the functional tradeoff of the centralized repair problem follows along the same idea as the single erasure tradeoff and the cooperative regenerating codes [2], [17].

According to the max-flow bound of network coding [24], for a data collector to be able to reconstruct the data, the minimum cut (min-cut) separating the source to the data collector should be larger or equal to the data object size $\mathcal{M}$. Considering all possible data collectors and all possible failure patterns, the following condition is necessary and sufficient for the existence of regenerating codes[1] satisfying the reliability constraint:

$$\min_{G\in\mathcal{G}} \min_{t\in \mathrm{DC}(G)} \mathrm{mincut}_G(s,t) \geq \mathcal{M}. \tag{1}$$

[1]Strictly speaking, this is only valid when the number of failures/repairs is bounded. A rigorous proof is required to drop the boundedness assumption as [17], [26]

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

Theorem 1. For fixed system parameters $(\mathcal{M},n,k,d,e,\alpha,\beta)$, regenerating codes satisfying the centralized multi-node repair condition exist if and only if

$$\mathcal{M} \leq \min_{\mathbf{u}\in\mathcal{P}} \left(\sum_{i=1}^{g} \min(u_i\alpha, (d - \sum_{j=1}^{i-1} u_j)\beta) \right) \triangleq \min_{\mathbf{u}\in\mathcal{P}} f(\mathbf{u}), \tag{2}$$

where

$$f(\mathbf{u}) = \sum_{i\geq 1} \min(u_i\alpha, (d - \sum_{j=1}^{i-1} u_j)\beta), \tag{3}$$

$$\mathcal{P} = \{\mathbf{u} : 1 \leq u_i \leq e \text{ such that } \sum_{i=1}^{g} u_i = k, g \leq k\}. \tag{4}$$

We note that (2) was also independently developed in [14].

Proof: Consider a recovery scenario $\mathbf{u} \in \mathcal{P}$ in which a data collector DC connects to a subset of k nodes $\{x^i_{out} : i \in I\}$, where I is the subset of contacted nodes. 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.

As all incoming edges of DC have infinite capacity, we only examine cuts $(\bar{U}, U)$ with $S \in U$ and $\{x^i_{out} : i \in I\} \subseteq \bar{U}$. 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. Since nodes are sorted, nodes considered at the $i-$th step cannot depend on nodes considered at $j-$th step with $j > i$.

Consider the i-th group, consider the case $|\left\{x^i_{\text{in}} \in U\right\}| = m$ and the remaining nodes are such that $x^i_{\text{in}} \in \bar{U}$.

- if $x^i_{\text{in}} \in U$, then the contribution of each node is α. The overall contribution of these nodes is $m\alpha$.
- else: $x^i_{\text{in}} \in \bar{U}$, then if $x^i_{\text{virtual,out}} \in U$, the contribution of this node is ∞. Thus, we only consider the case $x^i_{\text{virtual,out}} \in \bar{U}$. Then, we discuss two cases
  - if $x^i_{\text{virtual,in}} \in U$, the contribution to the cut is $e\alpha$.
  - else, since the i-th group is the topologically i-th repair group, at most $\sum_{j=1}^{i-1} u_j$ edge come from output nodes in $\bar{U}$. Thus, the contribution is $(d - \sum_{j=1}^{i-1} u_j)\beta$. Thus, the contribution of this node is $\min(e\alpha, (d - \sum_{j=1}^{i-1} u_j)\beta)$.
    As for these nodes, $x^i_{\text{virtual,out}} \in \bar{U}$, we do not need to account for other similar nodes.

Thus, if $m = u_i$, the contribution of the i-th repair group is $u_i\alpha$. If $m < u_i$, the contribution is $m\alpha + \min(e\alpha, (d - \sum_{j=1}^{i-1} u_j)\beta)$, which can be reduced to $\min(e\alpha, (d - \sum_{j=1}^{i-1} u_j)\beta)$ if $m = 0$. Thus, to lower the cut capacity, either $m = u_i$ in case $(d - \sum_{j=1}^{i-1} u_j)\beta > u_i\alpha$ or $m = 0$ otherwise. Thus, the total

contribution of the i-th repair group is

$$\min(u_i\alpha, (d - \sum_{j=1}^{i-1} u_j)\beta).$$

Finally, summing all contributions from different repair groups and considering the worst case for $\mathbf{u} \in \mathcal{P}$ implies that

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

with $\mathcal{P}$ defined as in (4). Therefore, the existence of regenerating codes is guaranteed by [24] as long as $\mathcal{M} \leq \min_{G\in\mathcal{G}} \min_{t\in \mathrm{DC}(G)} \mathrm{mincut}_G(s,t)$. ∎

In the sequel, we will use the notation $k = ae + r$, such that $a = \lfloor \frac{k}{e} \rfloor$ and $r = k \bmod e$.

C. Solving the minimum cut problem

In this section, we derive the structure of the optimal configuration $\mathbf{u}$ in (2) for any set of parameters (α, β). For instance, we show that for $(i-1)e < k \leq ie$, the number of optimal repair groups g^* (the support of $\mathbf{u}$) is equal to i. The result is formalized in the following proposition.

Theorem 2. For an $(\mathcal{M}, n, k, d, e, \alpha, \beta)$ storage system, the scenario $\mathbf{u}^*$ corresponding to the minimum cut over all information flow graphs (cf. (2)) is characterized as follows:

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

where $0 < r < e$.

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

Lemma 3. Let $\alpha, \beta, u_1, u_2, d, e, l$ be non-negative reals 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 (3).

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

$$\begin{aligned} \underset{\mathbf{u}=[u_1,u_2]}{\text{minimize}} \quad & \min(u_1\alpha, d\beta) + \sum_{i=0}^{l-1} \min(e\alpha, (d - ie - u_1)\beta) \\ & + \min(u_2\alpha, (d - (l+1)e - u_1)\beta) \\ \text{subject to} \quad & 0 \leq u_1 \leq s, \\ & 0 \leq u_2 \leq e, \\ & u_1 + u_2 = s. \end{aligned} \tag{5}$$

Substituting u_2 by $s - u_1$ in (5), using the identity $\min(a,b) = \frac{a+b-|a-b|}{2}$ and after eliminating constant terms, (5) becomes equivalent to

$$\begin{aligned} \underset{u_1}{\text{minimize}} \quad & -u_1 l\beta - |u_1\alpha - d\beta| - \sum_{i=0}^{l-1} |e\alpha - d\beta + ie\beta + u_1\beta| \\ & - |s\alpha - u_1(\alpha - \beta) - (d - le)\beta| \\ \text{subject to} \quad & 0 \leq u_1 \leq s. \end{aligned} \tag{6}$$

The objective function in (6), as function of u_1, is concave on 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 $u_1^* = 0$. ∎

1) Case $k \leq e$: In this scenario, 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 n_1 and n_2 nodes respectively, we have $u_1 + u_2 \leq e$. One can group these two groups into a single repair group to achieve a lower cut value. Indeed, from the cut expression in (2), 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 l. After grouping 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 3, 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 (2) is simply characterized by $\mathcal{M} \leq \min(k\alpha, d\beta)$. Moreover, $\alpha_{MSMR} = \alpha_{MBMR} = \frac{\mathcal{M}}{k}$ and $\beta_{MSMR} = \beta_{MBMR} = \frac{\mathcal{M}}{d}$. Equivalently, the functional storage bandwidth tradeoff reduces to a single point given by $(\alpha_{MSMR}, \beta_{MSMR}) = (\alpha_{MBMR}, \beta_{MBMR}) = (\frac{\mathcal{M}}{k}, \frac{\mathcal{M}}{d})$.

2) Case $e < k$: Motivated by the previous case, the intuition is that, 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 equal to r, are equal to e. Lemma 3 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.

Corollary 4. Assume that $u_1 + u_2 = e + s$. Then, the following inequality holds

$$f([u_1, \underbrace{e, \ldots, e}_{l \text{ times}}, u_2]) \geq \min(f([s, \underbrace{e, \ldots, e}_{l+1 \text{ times}}]), f([\underbrace{e, \ldots, e}_{l+1 \text{ times}}, s])), \tag{7}$$

where $f(\mathbf{u})$ is defined as in (3).

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 3 by replacing the constraint in (6) by $s \leq u_1 \leq e$. ∎

For a fixed β, as a function of α, we denote the min-cut corresponding to $\mathbf{u} = [\underbrace{e, \ldots, e}_{j \text{times}}, r, \underbrace{e, \ldots, e}_{a-j \text{ times}}]$ by $C_j(\alpha), j =$

$0, \dots, a$. As will be shown later in the proof of Theorem 2, a careful analysis of the behavior of the $a+1$ different configurations $C_j(\alpha)$ is needed to determine the overall optimal scenario leading the lowest minimum cut. We state the result in the following lemma, whose proof is relegated to Appendix VIII-A.

Lemma 5. There exists a point $\alpha_c(a) \in [\frac{d}{e}\beta, \frac{d}{r}\beta]$ such that, for any $0 \leq j \leq a$,

$$C_j(\alpha) \begin{cases} \geq C_0(\alpha), & \text{if } \alpha \leq \alpha_c(a), \\ \geq C_a(\alpha), & \text{if } \alpha \geq \alpha_c(a), \end{cases} \tag{8}$$

with

$$\alpha_c(a) = \frac{d + ar - ae}{r}\beta. \tag{9}$$

Proof of Theorem 2: Now that we have the necessary machinery, we proceed as follows: given any configuration $\mathbf{u}$, we keep combining and/or changing repair groups by means of successive applications of Lemma 3 and Corollary 4 until we can no longer reduce the minimum cut. The algorithm converges 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 a, and as the number of repair groups of full size e is upper bounded by a, the algorithm must converge after a finite number of steps. It can be seen then that the above reduction procedure has a finite number of outcomes, given by

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

Therefore, if $e \mid k$, then the optimal scenario corresponds to considering exactly a repair groups. On the other hand, if $e \nmid k$, then, it is optimal to consider exactly $a+1$ repair groups. However, the optimal position of the repair group with r nodes needs to be determined. Then, using the Lemma 5, 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 set $[2, 3, 2]$. Reducing this set 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.

D. 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.

Theorem 6. For an $(\mathcal{M}, n, k, d, e, \alpha, \beta)$ storage system, there exists a threshold function $\alpha^*(\mathcal{M}, n, k, d, \gamma, e)$ such that for any $\alpha \geq \alpha^*(\mathcal{M}, n, k, d, e, \beta)$, regenerating codes exist. For any $\alpha < \alpha^*(\mathcal{M}, n, k, d, e, \beta)$, it is impossible to construct codes achieving the target parameters. The threshold function $\alpha^*(\mathcal{M}, n, k, d, \gamma, e)$ is defined as follows:

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

$$\alpha^* = \begin{cases} \frac{\mathcal{M}}{k}, & \gamma \in [f_0(a-1), +\infty), \\ \frac{\mathcal{M} - \gamma g_0(i)}{ie}, & \gamma \in [f_0(i-1), f_0(i)], i = a-1, \dots 1, \end{cases} \tag{10}$$

- else: $k = ae + r$ with $1 \leq r \leq e-1$, then:

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

where

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

$$g_r(i) = \frac{(a-i)(-2r + e + 2d - ae - ei)}{2d}. \tag{13}$$

Proof: See Appendix VIII-B. ∎

Remark 1. In case e divides k, the following equality holds for all points on the tradeoff

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

Therefore, the tradeoff between α and β is the same as the single erasure tradeoff of a system with reduced parameters given by $\frac{\mathcal{M}}{e}$, $\frac{k}{e} = a$ and $\frac{d}{e}$. The expression of the tradeoff in this case can be recovered from [2] with the appropriate parameters.

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

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

$$\alpha_{\text{MSMR}} = \frac{\mathcal{M}}{k}, \gamma_{\text{MSMR}} = \frac{\mathcal{M}}{k}\frac{ed}{d - k + e}. \tag{14}$$

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

- If $k = ae$, we obtain

$$\gamma_{\text{MBMR}} = \frac{2ed\mathcal{M}}{-k^2 + ek + 2kd} = \frac{d\mathcal{M}}{da - e\binom{a}{2}}, \tag{15}$$

$$\alpha_{\text{MBMR}} = \frac{\gamma_{\text{MBMR}}}{e}. \tag{16}$$

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 [2] and also the minimum bandwidth cooperative repair point [17].

- If $k = ae + r$, we obtain

$$\gamma_{\text{MBMR}} = \frac{2\mathcal{M}e}{(k - r + e)(2d - k + r)} = \frac{d\mathcal{M}}{d(a+1) - e\binom{a+1}{2}}, \tag{17}$$

$$\alpha_{\text{MBMR}} = \gamma_{\text{MBMR}}\frac{d + ar - ea}{rd}. \tag{18}$$

Comparing the amount of information stored to the total bandwidth, we have

$$\frac{e\alpha_{\text{MBMR}}}{\gamma_{\text{MBMR}}} = 1 + \frac{(e-r)(d-ea)}{rd} > 1. \tag{19}$$

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.

IV. Construction when $k \leq e$

In case $k \leq e$, 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 and the reconstruction bandwidth are the same. Therefore, one can achieve α and γ, by dividing the data into k packets and encoding them using (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 contacting $d-k$ nodes. Such repair is asymmetric in nature. We describe one approach for achieving the repair with equal contribution from d helpers.

1) Divide the original file into kd symbols (i.e., packets) (that is $\mathcal{M} = kd$) and encode them using an (nd, kd) MDS code.
2) Store the encoded packets at n nodes, such that each node is storing $\alpha = d$ encoded packets.
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 $\beta = \frac{M}{d} = k$ symbols. The replacement nodes receive dk different coded symbols, which are sufficient to reconstruct the whole data and thus regenerate the missing symbols.

Remark 2. 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 $\delta = \text{lcm}(k, k+1, k+2, \ldots, n-e)$ (lcm denotes the least common multiple). The file of size $\mathcal{M}$ is then divided into $k\delta$ packets and encoded into $n\delta$ with an MDS code. Each node then stores $\alpha = \frac{\mathcal{M}}{k} = \frac{k\delta}{k} = \delta$ coded symbols. For repair with a specific d, each node transmits any $\beta = \frac{\mathcal{M}}{d} = k\frac{\delta}{d}$ coded symbols for his node. Similarly, it can be seen that reconstruction and exact repair is always feasible for any $k \leq d \leq n - e$. Note that the constraint of the field size arises from the need for an $(n\delta, k\delta)$ MDS code. The field size needs to be no less than $n\delta$, e.g. Reed Solomon codes.

V. Non-existence of exact MBMR regenerating codes

Exact regenerating codes are of interest in practice. Exact regenerating codes achieving the MSMR point have been constructed [14], [21], [22], [27]. In this section, we explore the existence of linear exact MBMR regenerating codes. Unlike the single erasure repair problem [12] and the cooperative repair problem [19], we prove that linear exact regenerating codes do not exist. Following [12], [19], we proceed by investigating subspace properties linear exact MBMR codes should satisfy. Then, we prove that the derived properties over-constrain the system.

A. Subspace viewpoint

Linear exact regenerating codes for the MBMR point 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 $\mathcal{M}$ source symbols. 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 $\mathbf{h} \in \mathcal{F}^{\mathcal{M}}$, $\mathcal{F}$ being the underlying finite field. The vectors $\mathbf{h}$ define the code. A node storing α symbols can be considered as storing α vectors of the code. Node i stores $\mathbf{h}_1^{(i)} \ldots \mathbf{h}_\alpha^{(i)}$. It is easy to see that linear operations performed on the stored symbols are equivalent to the same operations performed on the these vectors: $\sum \gamma_i \mathbf{f}^t \mathbf{h}_i = \mathbf{f}^t(\sum \gamma_i \mathbf{h}_i)$. 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. For regeneration, each 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 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. The symbol $\bigoplus_j A_j$ denotes the direct sum of subspaces A_j.

Notation. For a general exact regenerating code, which can be nolinear, 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

$$H(W_A) \to \dim(W_A), \tag{20}$$

$$H(W_A|W_B) \to \dim(W_A) - \dim(W_A \cap W_B), \tag{21}$$

$$I(W_A, W_B) \to \dim(W_A \cap W_B), \tag{22}$$

where the symbol $\to$ 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.

In this section, we focus on *s*ymmetric codes. Namely, the results do not depend on the indices of the nodes. Note that one can always construct a symmetric code form a non-symmetric code [28]. We now start by proving some properties exact regenerating codes should satisfy. The following property [21, Lemma 4] is valid for all optimal exact regenerating codes, not necessarily MBMR codes.

Lemma 7. Let $B \subseteq [n]$ be a subset of nodes of size e, then for an arbitrary set of nodes A, such that $|A| \leq d, B \cap A = \emptyset$,

$$H(W_B|W_A) \leq \min(|B|\alpha, (d - |A|)\beta). \tag{23}$$

Proof: If nodes B are erased, consider the case of having nodes A and nodes C as helper nodes, $|C| = d - |A|$. Then,

the exact repair condition requires

$$\begin{aligned}0 =&H(W_B|S_A^B, S_C^B)\\ &= H(W_B|S_A^B) - I(W_B, S_C^B|S_A^B)\\ &\geq H(W_B|S_A^B) - H(S_C^B)\\ &\geq H(W_B|S_A^B) - (d-|A|)\beta\\ &\geq H(W_B|W_A) - (d-|A|)\beta.\end{aligned}$$

Moreover, we have $H(W_B|W_A) \leq H(W_B) \leq |B|\alpha$ and the results follows. The proof was also given in [21]. ∎

In the next subsection, we focus on the case where $e \mid k$.

B. Case $e \mid k$

Points on the tradeoff satisfy

$$\mathcal{M} = \sum_{j=0}^{a-1} \min(e\alpha, (d-je)\beta), \frac{d-k+e}{e}\beta \leq \alpha \leq \frac{d}{e}\beta.$$

Lemma 8. *(Entropy of data stored):* 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

$$H(W_L) = e\alpha, \tag{24}$$

$$H(W_L|W_A) = \min(e\alpha, (d-em)\beta). \tag{25}$$

For linear codes,

$$\dim(W_L) = e\alpha, \tag{26}$$

$$\dim(W_L) - \dim(W_L \cap W_A) = \min(e\alpha, (d-em)\beta). \tag{27}$$

Proof: By reconstruction requirement, we write

$$\mathcal{M} = H(W_{[k]}) \tag{28}$$

$$= H(W_{[e]}) + \sum_{j=1}^{a-1} H(W_{\{ej+1,\ldots,e(j+1)\}}|W_{[je]}) \tag{29}$$

$$\leq e\alpha + \sum_{j=1}^{a-1} \min(e\alpha, (d-ej)\beta) \tag{30}$$

$$\leq \sum_{j=0}^{a-1} \min(e\alpha, (d-ej)\beta) \tag{31}$$

$$= \mathcal{M}, \tag{32}$$

where (30) uses Lemma 7 and (31) follows as $e\alpha \leq d\beta$ for all points on the tradeoff. Thus, all inequalities must be satisfied with equality. ∎

Remark 3. Lemma 8 states that the contents of any group of e nodes are independent. In particular, for each node i, we have $H(W_i) = \alpha$.

Corollary 9. 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:

$$\begin{aligned}\dim(W_L) - \dim(W_L \cap W_A) &= \min(e\alpha, (d-em)\beta)\\ &= (d-em)\beta.\end{aligned}$$

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

Lemma 10. For any set E of size e, the MBMR point satisfies

$$W_E = \bigoplus_j S_j^E, \dim(S_j^E) = \beta.$$

The subspaces S_j^E and $S_{j'}^E$ are linearly independent. Moreover, each subspace has to be in the span of W_E: $S_j^E \subseteq W_E$.

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

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

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

Lemma 11. 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

$$S_Q^E = W_E \cap W_Q, \dim(S_Q^E) = |Q|\beta. \tag{33}$$

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 $|J| = k-e$. From Corollary 9, we have $\dim(W_E \cap W_J) = (k-e)\beta$. On the other hand, from Lemma 10 , we have $S_J^E \subseteq W_E$. 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 12. 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

$$\dim(W_{e+2} \cap (W_{[e+1]})) = \dim(W_{e+2} \cap (W_{[e]}) = \beta. \tag{34}$$

Proof: We have

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

where the second equality follows from Lemma 8, Lemma 11 and the fact that any set of e nodes are linearly independent. On the other hand, we write

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

The lemma follows from equating both equations. ∎

Theorem 13. Exact linear regenerating MBMR codes do not exist when $2 \leq e < k$ and $e \mid k$.

Proof: Assuming there exists an exact-repair regenerating code satisfying the constraints, we consider the first e nodes. Then, these nodes store linearly independent vectors. We write, for $i = 1, \ldots, e$, $W_i = \begin{pmatrix} V_{i1} & V_{i2} \end{pmatrix}$ where V_{i1} contains β linearly independent columns and V_{i2} contains the remaining $(\alpha - \beta)$ basis vectors for node i. Now, consider node $e+1$. We have $\dim(W_{e+1} \cap W_1^e) = \beta$. That means that node $e+1$ contains β columns, linearly dependent on the columns from the first e nodes. Since any set of e nodes among the first $e+1$ nodes should be linearly independent, w.l.o.g, we can assume that the β dependent nodes of node $e+1$, $V_{e+1,1}$ 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 12, node $e+2$ contains $(\alpha - \beta)$ vectors linearly independent form 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 contained in $V_{e+2,1}$. Now, to repair any set of e nodes from the set of first $e+1$ nodes, node $e+2$ can only pass $V_{e+2,1}$. Otherwise, Lemma 11 will be violated. Then, this implies that $V_{e+2,1} \subseteq W_{\mathcal{J}}$, where $\mathcal{J} \subseteq \{1, \ldots, e+1\}$ such that $|\mathcal{J}| = e$. Them it can be seen that $V_{e+2,1}$ can only be of the same form in (35)

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

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 11 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 $e\beta$ nodes. 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 10. ■

C. Case $e \nmid k$

In this case, all points on the tradeoff satisfy

$$\mathcal{M} = \min(r\alpha, d\beta) + \sum_{i=0}^{a-1} \min(e\alpha, (d - r - ie)\beta), \tag{36}$$

$$\frac{d-k+e}{e}\beta \leq \alpha \leq \frac{d+ar-ae}{r}\beta. \tag{37}$$

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

Lemma 14. For an arbitrary set R of storage nodes of size r, and a set A such that $|A| = je + r < k$ for some integer $j \leq a-1$, for all exact-regenerating codes operating on the functional tradeoff, it holds that

$$\dim(W_R) = r\alpha, \tag{38}$$

$$\dim(W_E) - \dim(W_E \cap W_A) = \min(e\alpha, (d - r - je)\beta). \tag{39}$$

Remark 4. In case $e \nmid k$, a set of e are no longer linearly independent. This is expected as $e\alpha > d\beta$. Instead, it can be seen from Lemma 14 that any set of r nodes are linearly independent.

Recall that from the analysis of Theorem 2, at the MBMR point, two scenarios generate the same minimum cut: $\mathbf{u}_1 = [r, e, \ldots, e]$ and $\mathbf{u}_2 = [e, \ldots, e, r]$. Equivalently, we have

$$\mathcal{M} = f(\mathbf{u}_1) = f(\mathbf{u}_2), \tag{40}$$

where $f(\mathbf{u})$ is defined as in (3).

Lemma 15. 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 $j \leq a-1$, $|R| = r$ and $|B| = ae$, it holds that

$$\dim(W_E) = d\beta, \tag{41}$$

$$\dim(W_E) - \dim(W_E \cap W_A) = (d - je)\beta, \tag{42}$$

$$\dim(W_R) - \dim(W_R \cap W_B) = (d - ae)\beta. \tag{43}$$

Proof: The result can be derived by proceeding as in Lemma 14 and using the fact that $\mathcal{M} = f(\mathbf{u}_2)$ from (40). Equations (42) and (43) follow by noticing that $e\alpha \geq d\beta$. ■

Lemma 10 and Lemma 11 hold true case $e \nmid k$. The following lemma is used to derive the contradiction.

Lemma 16. let $k = ae + r$, then at the MBMR point, for any set of $r+1$ nodes, it holds that

$$\dim(W_{r+1} \cap W_{[r]}) = \beta. \tag{44}$$

Proof: We have

$$d\beta = \dim(W_{[e]}) \tag{45}$$

$$= \sum_{i=1}^{e} \dim(W_i) - \dim(W_i \cap W_{[i-1]}) \tag{46}$$

$$= e\alpha - \sum_{i=r+1}^{e} \dim(W_i \cap W_{[i-1]}), \tag{47}$$

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

$$\sum_{i=r+1}^{e} \dim(W_i \cap W_{[i-1]}) = e\alpha - d\beta = (e-r)(\alpha - a\beta). \tag{48}$$

Now we write

$$(e-r)(\alpha - a\beta) = \sum_{i=r+1}^{e} \dim(W_i \cap W_{[i-1]}) \tag{49}$$

$$\geq \sum_{i=r+1}^{e} \dim(W_i \cap W_{[r]}) \tag{50}$$

$$= (e-r)\dim(W_{r+1} \cap W_{[r]}), \tag{51}$$

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

$$\dim(W_{r+1} \cap W_{[r]}) \leq \alpha - a\beta. \tag{52}$$

Combining (48) and (52), we obtain

$$\sum_{i=r+2}^{e} \dim(W_i \cap W_{[i-1]}) \geq (e-r-1)(\alpha - a\beta). \tag{53}$$

On the other hand, we have

$$\sum_{i=r+2}^{e} \dim(W_i \cap W_{[i-1]}) \leq \sum_{i=r+2}^{e} \dim(W_i \cap W_{\mathcal{E}_i}) \quad (54)$$

$$= (e-r-1)\beta, \quad (55)$$

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 11. Combining (53) and (55), it follows

$$(e-r-1)(\alpha - a\beta) \leq \sum_{i=r+2}^{e} \dim(W_i \cap W_{[i-1]}) \leq (e-r-1)\beta. \quad (56)$$

It follows that $\alpha - a\beta = \frac{d-ae}{r}\beta \leq \beta$. The last equality holds only when $d = k$. Otherwise, $\alpha - a\beta > \beta$. Therefore, we only consider the case $d = k$ for which $\alpha = (a+1)\beta$ and $\alpha - a\beta = \beta$. Moreover, it follows from (56) that

$$\sum_{i=r+2}^{e} \dim(W_i \cap W_{[i-1]}) = (e-r-1)\beta. \quad (57)$$

Using (48), we obtain $\dim(W_{r+1} \cap W_{[r]}) = \beta$. ∎

Theorem 17. Exact linear regenerating MBMR codes do not exist when $e < k$ and $e \nmid k$.

Proof: 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 $d\beta > r\alpha$. ∎

VI. Non-feasibility of exact-interior point

In this section, we study the non-feasibility of the interior points for $e \mid k, e \mid d$, similarly to [23]. We note that all interior points satisfy $(d-k+e)\beta \leq e\alpha \leq d\beta$. This can be written as $(d'-a+1)\beta \leq \alpha \leq d'\beta$, where $d' = \frac{d}{e}$ and $a = \frac{k}{e}$. This is similar to the single erasure case with reduced parameters.

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

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

A. Properties of Exact-Repair Codes

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

Lemma 18. 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

$$I(W_L, W_A) = \begin{cases} 0 & j \leq p, \\ e((j-p)\beta - \theta) & p < j < a \\ e\alpha & j \geq a. \end{cases} \quad (58)$$

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

$$I(W_L, W_A) = H(W_L) - H(W_L|A) \quad (59)$$

$$= e\alpha - \min(e\alpha, (d-je)\beta) \quad (60)$$

$$= e(\alpha - \min(\alpha, (d'-j)\beta)) = e(\alpha - (d'-j)\beta)^+ \quad (61)$$

$$= e((j-p)\beta - \theta)^+. \quad (62)$$

∎

Corollary 19. For an arbitrary set L size e, and a disjoint set A such that $|A| = em < k$ for some integer m, we have

$$H(W_L|S_A^L) = H(W_L|W_A) = \min(e\alpha, (d-ej)\beta). \quad (63)$$

Proof: From Lemma 7, we have $H(W_L|S_A^L) \leq \min(e\alpha, (d-ej)\beta)$. On the other hand, from Lemma 8,

$$H(W_L|S_A^L) \geq H(W_L|W_A) = \min(e\alpha, (d-ej)\beta). \quad (64)$$

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

Lemma 20. 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

$$H(S_m^L) = \beta, \quad (65)$$

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

$$H(S_B^L) = |B|\beta. \quad (66)$$

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 $H(S_M^L) = e\beta$. ∎

a) 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. A helper node pooling scenario is a scenario where on failure on any e nodes $L \subseteq R$, the d helper nodes assisting in its repair include all the $f-e$ remaining nodes in F. The remaining helper nodes are denoted by $\mathcal{V}(L)$. Let $|R| = r'e$.

Lemma 21. In the helper node pooling scenario where $\min(a, \frac{f}{e}) > p+2 \geq r'$, for any set of e arbitrary node $M \subseteq F - R$, we have

$$H(S_M^R) \leq e(2\beta - \theta). \quad (67)$$

Proof: The statement holds true for all $f' \geq f$ and $r'' \leq r'$. Then, for the proof, consider $r' = p+2$ and $F = R \cup M$, $f = e(p+3)$.

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

$$
\begin{aligned}
I(S_M^L; W_R) &= I(S_M^L; W_L, W_{R-L}) && (68)\\
&= I(S_M^L; W_{R-L}) + I(S_M^L; W_L|W_{R-L}) && (69)\\
&\geq I(S_M^L; W_L|W_{R-L}) && (70)\\
&= H(W_L|W_{R-L}) - H(W_L|W_{R-L}, S_M^L) && (71)\\
&\geq H(W_L|W_{R-L}) - H(W_L|S_{R-L}^L, S_M^L) && (72)\\
&= \min(e\alpha, (d-e(p+1))\beta) - \\
&\quad \min(e\alpha, (d-e(p+2))\beta) && (73)\\
&= (d-e(p+1))\beta - (d-e(p+2))\beta = e\beta. && (74)
\end{aligned}
$$

Then, we obtain

$$
H(S_M^L|W_R) = H(S_M^L) - I(S_M^L; W_R) \leq e\beta - e\beta = 0. \quad (75)
$$

Hence, $H(S_M^L|W_R) = 0$. Since, the choice of the set L from R was arbitrary, it follows $H(S_M^R|W_R) = 0$.

It follows from Lemma 18 that

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

■

Lemma 22. In the helper node scenario where $\min\{a, \frac{f}{e}\} > p+1 \geq r'$, for an arbitrary set of e nodes $M \subseteq F - R$, and an arbitrary pair of set of e nodes L_1 and L_2, it must be that

$$
H(S_M^{L_1}|S_M^{L_2}) \leq e\theta, \quad (76)
$$

and hence

$$
H(S_M^R) \leq e(\beta + (r'-1)\theta). \quad (77)
$$

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

$$
\begin{aligned}
I(S_M^L; W_R) &= I(S_M^L; W_{R-L}, W_L) && (78)\\
&= I(S_M^L; W_{R-L}) + I(S_M^L; W_L|W_{R-L}) && (79)\\
&\geq I(S_M^L; W_L|W_{R-L}) && (80)\\
&= H(W_L|W_{R-L}) - H(W_L|W_{R-L}, S_M^L) && (81)\\
&\geq H(W_L|W_{R-L}) - H(W_L|S_{R-L}^L, S_M^L) && (82)\\
&= \min(e\alpha, (d-(r'-1)e)\beta) \\
&\quad - \min(e\alpha, (d-re)\beta) && (83)\\
&= (d-pe)\beta - e\theta - (d-(p+1)e)\beta && (84)\\
&= e(\beta - \theta). && (85)
\end{aligned}
$$

Then, it must be that

$$
H(S_M^L|W_R) = H(S_M^L) - I(S_M^L; W_R) \leq e\beta - e(\beta-\theta) = e\theta. \quad (86)
$$

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

$$
\begin{aligned}
&H(S_M^{L_1}, S_M^{L_2}) \\
&= I(\mathcal{W}_\mathcal{R}; S_M^{L_1}, S_M^{L_2}) + H(S_M^{L_1}, S_M^{L_2}|W_R) && (87)\\
&\leq I(W_R; W_M) + H(S_M^{L_1}, S_M^{L_2}|W_R) && (88)\\
&= I(W_R; W_M) + H(S_M^{L_1}|W_R) + H(S_M^{L_2}|W_R, S_M^{L_1}) && (89)\\
&\leq e(\beta-\theta) + e\theta + e\theta = e(\beta+\theta), && (90)
\end{aligned}
$$

where the last inequality follows from Lemma 18. Then, we have

$$
\begin{aligned}
H(S_M^{L_1}|S_M^{L_2}) &= H(S_M^{L_1}, S_M^{L_2}) - H(S_M^{L_2}) && (91)\\
&= H(S_M^{L_1}, S_M^{L_2}) - e\beta && (92)\\
&\leq e(\beta+\theta) - e\beta = e\theta, && (93)
\end{aligned}
$$

where the first equality follows from (66).

Finally, partitioning the nodes in R in arbitrary sets $L_1, L_2, \ldots, L_{r'}$, it follows

$$
H(S_M^R) \leq H(S_M^{L_1}) + \sum_{i=2}^{r'} H(S_M^{L_i}|S_M^{L_{i-1}}) \leq e\beta + e(r'-1)\theta. \quad (94)
$$

■

B. Non-existence proof

First, we consider the interior points that are multiple of β. That is: $e\alpha = (d-ep)\beta, \theta = 0$, with p lying in the range $1 \leq p \leq a-2$.

Theorem 23. Exact-repair codes do not exist for the interior points with $\theta = 0$.

Proof: Consider a sub-network F consisting of $d+e$ nodes. Note that for any set $L \subseteq F$, $H(W_L|S_{F-L}^L) = 0$. 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. Then, we write

$$
\begin{aligned}
\mathcal{M} \leq H(W_F) &\leq H(\{S_{F-L}^L\}_{L_i}) && (95)\\
&= H(\{S_L^{F-L}\}_{L_i}) && (96)\\
&\leq \sum_{L_i} H(S_{L_i}^{F-L_i}) && (97)\\
&\leq \sum_{L_i} e\beta = (d+e)\beta, && (98)
\end{aligned}
$$

where the inequality follows from Lemma 22. On the other hand,

$$
\begin{aligned}
\mathcal{M} &= \sum_i \min(e\alpha, (d-ie)\beta) && (99)\\
&= \sum_i \min((d-ep)\beta, (d-ie)\beta) && (100)\\
&= 2(d-ep)\beta + \sum_{i\geq 2} \min((d-ep)\beta, (d-ie)\beta) && (101)\\
&\geq 2(d-ep)\beta + (a-2)e\beta && (102)\\
&\geq 2e\beta + (d-ep)\beta + (a-2)e\beta && (103)\\
&\geq 2e\beta + (d-ep)\beta + (a-2)e\beta && (104)\\
&= (d-2e)\beta + (k-2e-ep)\beta \geq (d-2e)\beta, && (105)
\end{aligned}
$$

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

Theorem 24. 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 $p+2 = a$ and $\theta \geq \frac{d-ep-e}{d-ep}\beta$.

Proof: See Appendix VIII-C. ■

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 case $e \geq k$, we showed that the tradeoff reduces to a single point, for which we have provided a code construction achieving it. Furthermore, we proved that the functional MBMR point is not achievable for linear exact repair codes. Similarly, we have shown that the functional repair tradeoff is not achievable under exact repair, except for maybe a small portion near the MSMR point. Open problems in this topic include achievability of non-linear exact MBMR codes, reducing the subpacketization size for exact MSMR codes, and characterization of interior points for exact repair.

VIII. Appendices

A. proof of Lemma 5

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

Lemma 25. The scenario $\mathbf{u} = [e, \ldots, e, r]$ achieves the lowest final value of minimum cut:

$$\lim_{\alpha \to +\infty} f(\mathbf{u}) \geq \lim_{\alpha \to +\infty} f([e, \ldots, e, r]), \forall \mathbf{u} \in \mathcal{P}, \tag{106}$$

where $f(\mathbf{u})$ and $\mathcal{P}$ are defined in (3) and (4), respectively.

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

$$\begin{aligned}
&\lim_{\alpha \to +\infty} f(\mathbf{u}) \\
&= \sum_{i=1}^{g} (d - \sum_{j=1}^{i-1} u_j)\beta \\
&= d\beta g - \beta \sum_{i=1}^{g} \sum_{j=1}^{i-1} u_j = gd\beta - \beta \sum_{i=1}^{g-1} u_i (g-i) \\
&= \beta(dg - g\sum_{i=1}^{g-1} u_i + \sum_{i=1}^{g-1} i u_i) = \beta((d-k)g + \sum_{i=1}^{g} i u_i).
\end{aligned} \tag{107}$$

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

$$\begin{aligned}
\underset{\mathbf{u},g}{\text{minimize}} \quad & (d-k)g + \sum_{i=1}^{g} i u_i \\
\text{subject to} \quad & 1 \leq u_i \leq e, \\
& \sum_{i=1}^{g} u_i = k.
\end{aligned} \tag{108}$$

It can be seen that the solution to (108) is given by $\mathbf{u} = [e, \ldots, e, r]$. ∎

We now study the different functions $C_j(\alpha)$ for $j = 0, \ldots, a$.

a) j=0: we have

$$\begin{aligned}
C_0(\alpha) &= \min(r\alpha, d\beta) + \sum_{i=0}^{a-1} \min(e\alpha, (d - r - ie)\beta) \\
&= r \min(\alpha, \frac{d\beta}{r}) + \sum_{i=0}^{a-1} e \min(\alpha, \frac{(d - r - ie)\beta}{e}).
\end{aligned}$$

$C_0(\alpha)$ is a piecewise linear function with breakpoints given by $\{\frac{d-r-(a-1)e}{e}\beta, \frac{d-r-(a-2)e}{e}\beta, \ldots, \frac{d-r}{e}\beta, \frac{d}{r}\beta\}$. 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) $1 \leq j \leq a$*:* for each j, we have

$$\begin{aligned}
C_j(\alpha) &= \sum_{i=0}^{j-1} \min(e\alpha, (d - ie)\beta) + \min(r\alpha, (d - je)\beta) \\
&\quad + \sum_{i=j}^{a-1} \min(e\alpha, (d - r - ie)\beta) \\
&= \sum_{i=0}^{j-1} e \min(\alpha, \frac{(d - ie)\beta}{e}) \\
&\quad + r \min(\alpha, \frac{(d - je)\beta}{r}) + \sum_{i=j}^{a-1} e \min(\alpha, \frac{(d - r - ie)\beta}{e}).
\end{aligned}$$

$C_j(\alpha)$ is also piecewise-linear function with non-increasing successive slopes. Its breakpoints are given by $\{\frac{d-r-(a-1)e}{e}\beta, \ldots, \frac{d-r-je}{e}\beta, \frac{d-(j-1)e}{e}\beta, \ldots, \frac{d}{e}\beta\} \cup \{\frac{d-je}{r}\beta\}$. 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 $\frac{d-je}{r}\beta$.

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

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

Figure 1 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 26. For $1 \leq j \leq a$, there exists a point $\alpha_c(j) \in [\frac{d}{e}, \frac{d}{r}]$ such that

$$\begin{aligned}
&C_0(\alpha_c(j)) = C_j(\alpha_c(j)), \\
&\quad C_0(\alpha) \leq C_j(\alpha) \qquad \text{if } \alpha \leq \alpha_c(j), \\
&\quad C_0(\alpha) \geq C_j(\alpha) \qquad \text{if } \alpha \geq \alpha_c(j), \\
&\quad C_j(\alpha) = C_j(\infty) \qquad \text{if } \alpha \geq \alpha_c(j).
\end{aligned} \tag{109}$$

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 $I_i \triangleq$

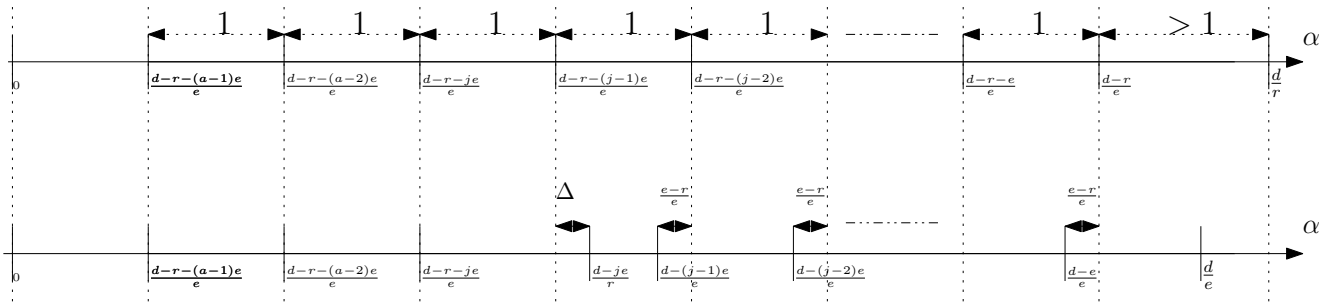

Fig. 1: relative positions of the breakpoints (with $\beta = 1$)

$(\frac{d-r-ie}{e}, \frac{d-r-(i-1)e}{e}]$ 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 case $C_j(\alpha)$ 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 case $C_j(\alpha)$ 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 1).
  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

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

  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 $1 \leq j \leq a$.

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 seen by noticing that $s_0(I_i) = s_j(I_i)$ and that

$$\begin{aligned} &(C_j(x_{i-1}) - C_j(x_i)) - (C_0(x_{i-1}) - C_0(x_i)) \\ &= r \geq 0, \ \forall i < j_0. \end{aligned}$$

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

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

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 continue 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 25 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 (109).

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, (109) holds also in this case. ■

Using Lemma 26 and the fact that C_a achieves the smallest final value from Lemma 25, that is $C_a(\infty) \leq C_j(\infty), j \in [0, a-1]$, it follows that (8) holds for any $j \in [0, a]$. Moreover, as $\alpha_c(a) \in [\frac{d}{e}, \frac{d}{r}]$, $\alpha_c(a)$ satisfies

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

which implies that

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

Simplifying the last equation yields (9).

B. Storage-bandwidth tradeoff expression

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

$$\begin{aligned} \underset{\alpha}{\text{minimize}} \quad & \alpha \\ \text{subject to} \quad & C(\alpha) \geq \mathcal{M}. \end{aligned} \tag{110}$$

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

$$C(\alpha) = \begin{cases} (a+1)\beta d - e\beta a(a+1)/2, & \alpha \geq \alpha_c, \\ r\alpha + \sum_{j=0}^{a-1} b_j, & \alpha \in [\frac{b_0}{e}, \alpha_c], \\ (r+ie)\alpha + \sum_{j=i}^{a-1} b_j, & \alpha \in [\frac{b_i}{e}, \frac{b_{i-1}}{e}], \\ & \text{for } i = 1, \ldots, a-1, \\ k\alpha, & \alpha \leq \frac{b_{a-1}}{e}, \end{cases} \tag{111}$$

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

$$\begin{aligned} \sum_{j=i}^{a-1} b_j &= \beta(a-i)(d - r - \frac{e(a-1+i)}{2}) \\ &= \gamma\frac{(a-i)(-2r+e+2d-ae-ei)}{2d} \triangleq \gamma g_r(i), \end{aligned}$$

such that

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

The expression $C(\alpha)$ increases from 0 to a maximum value given by $\beta((a+1)d - \binom{a+1}{2})$. To solve (110), we let $\alpha^* = C^{-1}(\mathcal{M})$ under the condition $\mathcal{M} \leq \beta((a+1)d - \binom{a+1}{2})$. Therefore, we obtain,

$$\alpha^* = \begin{cases} \frac{\mathcal{M}}{k}, & \mathcal{M} \in [0, \frac{kb_{a-1}}{e}] \\ \frac{\mathcal{M} - \sum_{j=i}^{a-1} b_j}{r+ie}, & \mathcal{M} \in [(r+ie)\frac{b_i}{e} + \sum_{j=i}^{a-1} b_j, (r+ie)\frac{b_{i-1}}{e} + \sum_{j=i}^{a-1} b_j] \\ & \text{for } i = a-1, \ldots 1, \\ \frac{\mathcal{M} - \sum_{j=0}^{a-1} b_j}{r}, & \mathcal{M} \in [\frac{b_0 r}{e} + \sum_{j=0}^{a-1} b_j, r\alpha_c + \sum_{j=0}^{a-1} b_j], \end{cases} \tag{112}$$

with

$$\begin{aligned} &\frac{rb_i}{e} + ib_i + \sum_{j=i}^{a-1} b_j \\ &= \frac{-a^2e^2 + ae^2 - 2aer + 2dae - e^2i^2 - e^2i - 2eir - 2r^2 + 2dr)}{2de}\gamma \\ &= \frac{-k^2 - r^2 + e(k-r) + 2kd - e^2(i^2+i) - 2ier}{2ed}\gamma \\ &\triangleq \gamma\frac{\mathcal{M}}{f(i)}, \end{aligned}$$

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

$$\alpha^* = \begin{cases} \frac{\mathcal{M}}{k}, & \mathcal{M} \in [0, \frac{kg_r(a-1)\gamma}{e}], \\ \frac{\mathcal{M} - \gamma g_r(i)}{r+ie}, & \mathcal{M} \in [\frac{\gamma\mathcal{M}}{f_r(i)}, \frac{\gamma\mathcal{M}}{f_r(i-1)}], \text{ for } i = a-1, \ldots 1, \\ \frac{\mathcal{M} - \gamma g_r(0)}{r}, & \mathcal{M} \in [\frac{\gamma\mathcal{M}}{f_r(0)}, (g_r(0) + \frac{d+ar-ae}{d})\gamma]. \end{cases} \tag{113}$$

As a function of γ, after simplifications, we obtain the expression of α^* as in Theorem 6. We note that there are a piece-wise linear portions on the curve. Moreover, the minimum bandwidth point γ_{MBMR} is given by

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

The expression of α_{MBMR} is given by

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

In case $e \mid k$, we have $r = 0$. The expression of the tradeoff is obtained from (113) by setting $r = 0$ and eliminating the last line. We note that in this case, there are $a-1$ piece-wise linear portions on the trade-off curve.

C. Proof of Theorem 24

Proof: Take a subnetwork of $d+e$ nodes. Let L and M be two 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

$$H(W_L|S_A^L, S_B^L, S_M^L) = 0, \tag{114}$$

$$H(W_M|S_A^M, S_B^M, S_L^M) = 0. \tag{115}$$

It follows that

$$H(W_L, W_M|W_A, S_B^L, S_B^M, S_M^L) \tag{116}$$

$$= H(W_L|W_A, S_B^L, S_B^M, S_M^L) + H(W_M|W_L, W_A, S_B^L, S_B^M, S_M^L) \tag{117}$$

$$= 0. \tag{118}$$

Therefore, we have

$$H(S_B^L, S_B^M, S_M^L) \geq H(W_L, W_M|W_A) \tag{119}$$

$$= H(W_L|W_A) + H(W_M|W_A W_L) \tag{120}$$

$$= H(W_L) - I(W_L; W_A) + H(W_M) - I(W_M; W_A W_L) \tag{121}$$

$$= e\alpha - 0 + e\alpha - e(\beta - \theta) \tag{122}$$

$$= 2e\alpha - e\beta + e\theta \tag{123}$$

$$= 2((d-ep)\beta - e\theta) - e\beta + e\theta \tag{124}$$

$$= (2d - 2ep - e)\beta - e\theta. \tag{125}$$

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

case: $p+2 < a$:

$$H(S_B^L, S_B^M, S_M^L) \leq \sum_{L_i \in B} H(S_{L_i}^L, S_{L_i}^M) + H(S_M^L) \tag{126}$$

$$\leq \sum_{L_i \in B} e(2\beta - \theta) + e\beta \tag{127}$$

$$= (d - pe - e)(2\beta - \theta) + e\beta \tag{128}$$

$$= (2d - 2ep - e)\beta - (d - ep - e)\theta, \tag{129}$$

where the first inequality is obtained using Lemma 21. Equations (125) and (129) are in contradiction if $d - ep - e > e \iff d > e(p+2)$, which is true as $d \geq k > p+2$ and $\theta \neq 0$.

case: $p+2 = a$*:* In this case, Lemma 22 is used to derive an upper bound on $H(S_B^L, S_B^M, S_M^L)$. Lemma 22 does not hold if $a = 2$. It holds for $a > 2 \iff k > 2e$. Thus, we consider $k > 2e$. We have

$$H(S_B^L, S_B^M, S_M^L) \leq \sum_{L_i \in B} H(S_{L_i}^L, S_{L_i}^M) + H(S_M^L) \quad (130)$$

$$\leq \sum_{L_i \in B} e(\beta + \theta) + \beta \quad (131)$$

$$= (d - ep)\beta + (d - ep - e)\theta. \quad (132)$$

Equations (125) and (129) are in contradiction when

$$\theta < \frac{d - ep - e}{d - ep}\beta. \quad (133)$$

■

REFERENCES

[1] 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.

[2] 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.

[3] 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.

[4] 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.

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

[6] 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.

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

[8] A. S. Rawat, O. O. Koyluoglu, and S. Vishwanath, "Progress on high-rate MSR codes: Enabling arbitrary number of helper nodes," *arXiv preprint arXiv:1601.06362*, 2016.

[9] S. Goparaju, A. Fazeli, and A. Vardy, "Minimum storage regenerating codes for all parameters," *arXiv preprint arXiv:1602.04496*, 2016.

[10] V. R. Cadambe, C. Huang, S. A. Jafar, and J. Li, "Optimal repair of MDS codes in distributed storage via subspace interference alignment," *arXiv preprint arXiv:1106.1250*, 2011.

[11] 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.

[12] 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.

[13] 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.

[14] A. S. Rawat, O. O. Koyluoglu, and S. Vishwanath, "Centralized repair of multiple node failures with applications to communication efficient secret sharing," *arXiv preprint arXiv:1603.04822*, 2016.

[15] 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.

[16] 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.

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

[18] 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.

[19] 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.

[20] 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.

[21] Z. Wang, I. Tamo, and J. Bruck, "Optimal rebuilding of multiple erasures in MDS codes," *arXiv preprint arXiv:1603.01213*, 2016.

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

[23] 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.

[24] 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.

[25] 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.

[26] 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.

[27] 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, May 2013.

[28] 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 2015(ISIT'15)*, June 2015, pp. 2061–2065.