Repair for Distributed Storage Systems With Packet Erasure Channels and Dedicated Nodes for Repair

Majid Gerami, *Student Member, IEEE*, Ming Xiao, *Senior Member, IEEE*, Jun Li, *Member, IEEE*, Carlo Fischione, *Member, IEEE*, and Zihuai Lin, *Senior Member, IEEE*

***Abstract*—We study the repair problem in distributed storage systems where storage nodes are connected through packet erasure channels and some nodes are dedicated to repair [termed as dedicated-for-repair (DR) storage nodes]. We first investigate the minimum required repair-bandwidth in an asymptotic setup, in which the stored file is assumed to have an infinite size. The result shows that the asymptotic repair-bandwidth over packet erasure channels with a fixed erasure probability has a closed-form relation to the repair-bandwidth in lossless networks. Next, we show the benefits of DR storage nodes in reducing the repair bandwidth, and then we derive the necessary minimal storage space of DR storage nodes. Finally, we study the repair in a nonasymptotic setup, where the stored file size is finite. We study the minimum practical-repair-bandwidth, i.e., the repair-bandwidth for achieving a given probability of successful repair. A combinatorial optimization problem is formulated to provide the optimal practical-repair-bandwidth for a given packet erasure probability. We show the gain of our proposed approaches in reducing the repair-bandwidth.**

I. Introduction

DISTRIBUTED storage systems have recently attracted significant research interests for many applications in data centers and peer-to-peer storage networks, e.g., OceanStore, Total Recall and DHash++ [1]. While existing research results are mostly for distributed storage systems over wired networks, distributed storage systems can also be applied over wireless networks. The use of wireless transmission between storage nodes has been suggested in [2] to combat congestion (oversubscription) in data centers. Distributed storage systems can also be used in wireless caching networks with device-to-device communication [3]–[6]. These systems can also have applications in *delay tolerant networks (DTNs)* [7].

In distributed storage systems, node failure and repair have been studied [1], [3]–[12]. More specifically, if a storage node fails or leaves the system, in a mechanism, called repair, a new node is regenerated by transmitting sufficient data from the surviving nodes to the new node. In the most of the existing results, it is assumed that the links between storage nodes are error-free, without any error or erasure during repairing information transmission. However, in the distributed storage systems over wired networks, the transmitting packets may be lost due to e.g., congestion and buffer overflow in intermediate switches/routers, or due to protocol and load balancer issues, as it is reported in [13] for long-term measurement analysis over the links in the current data-centers. In wireless networks, repairing packets may be lost due to channel fading or interference [14], [15].

Based on above observations, we will study the repair problem in a distributed storage system with lossy channels. We derive the optimal storage-bandwidth tradeoffs for distributed storage systems with packet erasure channels. We show that the optimal tradeoffs are asymptotically achievable (when the file size is infinitely large). More specifically, consider a source file that contains M fragments, where each fragment contains ξ packets (thus, the file contains $M\xi$ packets). Suppose that the surviving nodes totally transmit $\gamma'(\xi)$[1] packets to the new node. Asymptotic analysis gives the minimum bandwidth $\gamma'(\xi)/M\xi$ when ξ tends to infinity, which may be used as the performance bound for a finite ξ.

As node failure in distributed storage systems may frequently happen [16], [17], it is valuable to have some nodes dedicated to repair to reduce the repair bandwidth. We term these nodes dedicated-for-repair (DR) nodes. We investigate the benefits of DR storage nodes. For the storage nodes participating both file recovery and repair, we call them the *complete storage nodes*. In a distributed storage system, the repair-bandwidth could be reduced by using more complete storage nodes in repair [1], if the system allows. However, in some scenarios, the system may not allow to increase the number of complete storage nodes in repair due to e.g., limitations in storage capacity and the repair-bandwidth. Yet, if we have additional storage nodes with smaller storage capacity and smaller repair-bandwidth (than complete storage node) and they function the same as complete storage nodes for repairing, then the repairing costs could be saved for storage space and repair-bandwidth. Note that DR storage nodes are designated only for repair. That is, DR nodes will not participate source file recovery by data collectors. We will study the minimal storage space of DR storage nodes and will show the gain of using DR storage nodes in reducing the repair bandwidth.

Though asymptotic analysis can serve as performance bounds and is valid especially for very large file size, in practice the file size might be limited, namely, a finite

Manuscript received June 9, 2015; revised November 27, 2015 and February 10, 2016; accepted February 11, 2016. Date of publication February 29, 2016; date of current version April 13, 2016. The associate editor coordinating the review of this paper and approving it for publication was M. Ardakani.

M. Gerami, M. Xiao, J. Li, and C. Fischione are with the Department of Electrical Engineering, KTH Royal Institute of Technology, Stockholm 10044, Sweden (e-mail: gerami@kth.se).

Z. Lin is with the Department of Electrical Engineering, University of Sydney, Sydney, N.S.W. 2006, Australia.

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TCOMM.2016.2532879

[1] γ' is a function of ξ.

number of repairing packets. For this case, we show that the repair-bandwidth can be much larger than that of the asymptotic analysis. We term the repair bandwidth where the number of repairing packets is finite as the *practical-repair-bandwidth*. Then, we propose a method to reduce the practical-repair-bandwidth by studying the probability of successful repair. The results show that a method with smaller asymptotic repair bandwidth is not necessarily a better approach in reducing the practical-repair-bandwidth. We also formulate a combinatorial optimization problem to find the optimized scheme (in terms of practical-repair-bandwidth) based on the packet erasure probabilities. We also show that the results for non-asymptotic analysis converges to the asymptotic analysis results when ξ tends to infinity.

The rest of the paper is organized as follows. In Section II, the background and the related works are discussed. The system model is given in Section III. Then in Section IV, we study the optimal storage-bandwidth tradeoffs in packet erasure networks. In Section V, we propose using DR storage nodes to reduce the repair bandwidth. Next, in Section VI, we analyze the repair bandwidth for a finite number of repairing packets (finite file-size) and then propose methods to reduce the practical repair-bandwidth. Finally, we conclude the paper in Section VII. Most of the proofs are deferred to appendices in Section VIII.

II. Background and Related Works

In a distributed storage system, when a node fails, to maintain the reliability of the system, a node is regenerated. In the regenerating process, the surviving nodes transmit sufficient data to the new node such that the system with the new node still maintains the reconstruction property (any k out of n complete storage nodes can reconstruct the stored file). Yet, the new node may have different coded packets compared to the failed one. This is termed as *functional repair*. On the other side, for *exact repair*, the coded packets of the new node are exactly the same as those in the failed node. In this paper, we mainly consider functional repair.

Reference [1] modeled the repair process by an information flow graph, and mapped the repair problem into a multicast problem in lossless networks (no transmission errors). Cut-set bound analysis on the information flow graph showed that a sink (or a data collector) can reconstruct the original file of size $M\xi$ packets if and only if

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

where each storage node stores α fragments such that any set of k storage nodes can reconstruct the original file, and in repair d ($d \geq k$) number of surviving nodes transmit $d\beta$ fragments to the new node. The above fundamental bound leads us to the capacity of such systems. In what follows, we will provide two useful definitions in distributed storage systems.

Definition 1 (Information Rate in a Distributed Storage System): The information rate in a distributed storage system under the dynamic of node failure/repair is defined as the amount of information that a data collector obtains by connecting to k or more (complete) storage nodes in one unit of time.

Here, the dynamic of node failure/repair means the process in which (different) nodes continuously fail and are repaired.

Definition 2 (Capacity of a Distributed Storage System): The maximum amount of information that a data-collector can obtain in one unit of time by connecting to k or more (complete) storage nodes is defined as the capacity. The term $\sum_{i=0}^{k-1} \min\{\alpha\xi, (d-i)\beta\xi\}$ equals to the capacity of a distributed storage system in lossless networks [1]. For the optimal codes, $\sum_{i=0}^{k-1} \min\{\alpha\xi, (d-i)\beta\xi\} = M\xi$.

The authors in [1] derived an explicit relation between α, γ, d, M and k for the optimal storage-bandwidth tradeoff. The codes achieving the optimal tradeoff are called *regenerating codes*. The codes achieving two extreme points on the optimal storage-bandwidth tradeoff are called *minimum storage regenerating (MSR)* and *minimum bandwidth regenerating (MBR)* codes, respectively. The optimal repair-bandwidth at these two extreme points can also be derived by two sequential optimization processes under the constraint (1). MSR codes are achieved by first minimizing the storage and then minimizing the repair-bandwidth. The minimum storage capacity required for the reconstruction property is M/k fragments. Thus, nodes for MSR codes store the same amount of data as the MDS (maximum distance separable, optimal error correction codes) codes. From (1), we can derive the minimum repair-bandwidth for an MSR code as

$$\begin{aligned} \alpha_{\text{MSR}} &= \frac{M}{k}, \\ \gamma_{\text{MSR}} &= \frac{Md}{k(d-k+1)}. \end{aligned} \tag{2}$$

In the optimization process, if we first minimize the repair-bandwidth and then storage per node, the other extreme point, namely, MBR is achieved. It can be easily verified that in general $\gamma \geq \alpha$ [11]. For the MBR codes $\gamma = \alpha$. Therefore, setting $\gamma = d\beta = \alpha$ on the optimum bound $\sum_{i=0}^{k-1} \min\{\alpha, (d-i)\beta\} = M$ yields

$$\begin{aligned} \alpha_{\text{MBR}} &= \frac{2Md}{k(2d-k+1)}, \\ \gamma_{\text{MBR}} &= \frac{2Md}{k(2d-k+1)}. \end{aligned} \tag{3}$$

The code construction and the achievablity of the functional and exact repair have been studied in [18], [8], [19]. In [20], cooperative regenerating codes are proposed to reduce the bandwidth in the scenario of multiple-node failure. Surviving node cooperation is suggested in [12], [21] in order to minimize the cost of repair in multi-hop storage networks. Yet, in most of the previous works of regenerating codes, it is assumed that the links between storage nodes are perfect. That is, there is no error or erasure. Recently Rashmi et al. [10] proposed a regenerating code which is resistant to a specific number of path failures by requesting more nodes to join the repair process. Particularly, for the code resistant to d_2 path failures, it requires to transmit from $d_{\text{tot}} = d_1 + d_2$ surviving nodes instead of d_1 nodes

(d_1 nodes are assumed to be sufficient for repair with perfect channels). Compared to [10], we consider the probability of successful repair. A regenerating process is successful when the new node together with the surviving nodes has the reconstruction property. We show that optimal d_1 and d_2, in which the probability of successful repair is maximized, depend on the erasure probability of the links. In addition, we study the capacity of distributed storage networks where the channels between the surviving nodes and the new node are packet erasure channels, and propose a method to reduce the repair-bandwidth by DR storage nodes.

Other related results on network coding for erasure networks are as follows. In [22], it is shown that random linear codes can achieve the capacity of packet erasure networks. The capacity of wireless erasure networks has been studied in [23]. In [24], [25], the probability of successful reconstruction of a source file is studied in packet erasure networks. In contrast to [24], [25], we study the repair problem.

III. System Model

Consider that a distributed storage system stores a file of M fragments, where each fragment contains ξ packets. We refer to ξ as *subpacketization order*. We assume that repairing information from the surviving nodes to the new node is packetized. A packet here is considered as the basic information unit and is denoted by a vector of ρ symbols taken from the finite field $GF(q)$, where q is the alphabet size. Hence, a packet has $\rho \log_2(q)$ bits of information [22]. We assume that channel coding has been used for each packet and perfect error detection is also used. Therefore, a packet in any channel is either received error-free or is dropped and each link can be regarded as a packet erasure channel [23]. When a packet has only one bit of information, then the channel can also be regarded as a binary erasure channel. We also assume packets are erased independently with a given probability p and channels are memoryless. We remark that in this paper, all the communication channels are point to point, and thus the effect of broadcast and interference in wireless channels are not studied here.

In our system, there are n storage nodes with a storage capacity α fragments and every k nodes can reconstruct the source file. Moreover there are $h (h \geq 0)$ DR storage nodes, each of which with storage capacity α' fragments. To repair a failed node, $d (d \geq k)$ surviving nodes among $n-1$ complete storage nodes plus h DR storage nodes equally send $\gamma\xi = (d+h)\beta\xi$ packets (in total) to the new node. For simplifying illustration, we denote the distributed storage system with packet erasure channels by $\mathrm{DSS}(n, k, d, h, \alpha, \alpha', \gamma = d\beta, M\xi, p)$. We note that, in this system, the reconstruction property is relaxed for DR storage nodes. That is, k storage nodes including at least one DR storage node cannot reconstruct the original file.

In this paper, we extend the regenerating codes in two senses: firstly we extend the regenerating code to lossy networks, and secondly we extend the regenerating code by adding the DR storage nodes to the repair process. Therefore, it would be useful to define the extended MSR (EMSR) and extended MBR (EMBR) codes as follows.

Definition 3 (EMSR$_{h,p}$ Codes): Consider a distributed storage system that stores a file of size M fragments by an $(n, k, d) - \mathrm{EMSR}_{h,p}$ codes. Then each complete storage node stores $\alpha = M/k$ fragments, and every k out of n complete storage nodes can reconstruct the source file. In repair, d complete storage nodes plus h DR storage nodes help generating a new node. The repair packets over the channels between the helper nodes and the new node are lost with a probability p. These codes have the minimum repair-bandwidth for the extended regenerating codes where $\alpha = M/k$.

Definition 4 (EMBR$_{h,p}$ Codes): Consider a distributed storage system that stores a file of size M fragments by an $(n, k, d) - \mathrm{EMBR}_{h,p}$ codes. Then each complete storage node stores $\alpha > M/k$ fragments, and every k out of n complete storage nodes can reconstruct the source file. In repair, d complete storage nodes plus h DR storage nodes help generating a new node. The repair packets over the channels between the helper nodes and the new node are lost with a probability p. These codes have the minimum repair-bandwidth among all the extended regenerating codes.

IV. Asymptotic Analysis

A. Information Flow Graph

To study the fundamental performance limits, we will first give the analysis when the file size is asymptotically large. Our analysis is based on a graphical representation of a distributed storage system, namely, the information flow graph $\mathcal{G}$, which was first adopted in [1] to analyze the repair problem in lossless networks. We modify the graph in [1] to analyze the repair problem in a distributed storage system with packet erasure channels and in the presence of DR storage nodes. Then the information flow graph is a directed acyclic graph denoted by $\mathcal{G}(\mathcal{N}, \mathcal{A})$, where $\mathcal{N}$ denotes the set of nodes and $\mathcal{A}$ denotes the set of edges in the network. Each complete storage node is modeled by two nodes, *in* and *out* nodes, which are connected by a link of capacity $\alpha\xi/\tau$ packets per unit of time. The source is connected to *in* nodes of complete storage nodes by links of capacity $\alpha\xi/\tau$ packets per unit of time. Each DR storage node is modeled by one node. The source is connected to a DR storage nodes by a link of capacity $\alpha'\xi/\tau$ packets per unit of time. When a node fails, d complete storage nodes plus h DR storage nodes transmit repairing packets to the new node. Each of $d+h$ helper nodes sends packets to the new node by a rate $\beta\xi/\tau$ packets per unit of time. The transmitted packets on the lossy links might be erased with a packet erasure probability p. In such a graph, a data collector (DC) connects to k out nodes of complete storage nodes and reconstructs the original file in τ units of time. We say that repair is successful if a data-collector by connecting any k *out* nodes of complete storage nodes can reconstruct the original file in τ units of time. In asymptotic analysis, ξ and τ tend to infinity. An information flow graph with a number of node failure/repair is shown in Fig. 1. We note that in the modified information flow graph we introduced the time, τ for analysis, in which data-collectors reconstruct the file. This is not required in lossless networks in [1].

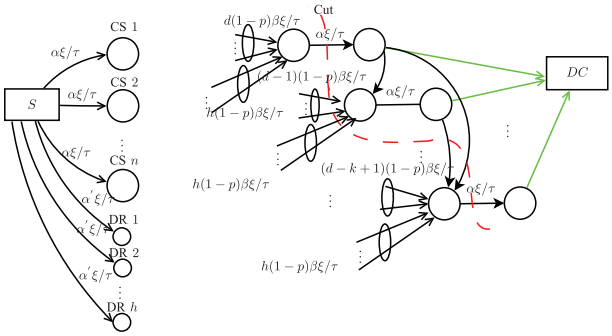


Fig. 1. Information flow graph for distributed storage systems with packet erasure channels and DR storage nodes. CS stands for a complete storage node and DR stands for a DR storage node.

A cut-set bound analysis over the above information flow graph leads us to the following upper bound of the information rate. We later discuss about the achievable bounds.

Lemma 1: An upper bound of the information rate in the above information flow graph is

$$\sum_{i=0}^{k-1} \min\{\alpha\xi/\tau, (d-i+h)\beta(1-p)\xi/\tau\}. \tag{4}$$

Proof: The proof is inspired by the cut-set bound analysis in [1] and the cut analysis in network information theory [26]. The detailed proof can be found in Appendix A. ■

B. Optimal Storage-Bandwidth Tradeoff in Packet Erasure Networks

In what follows, we will first investigate the fundamental performance limits on the repair in packet erasure networks for $h = 0$. In the next section, we will discuss more details about DR storage nodes and then extend the results for $h > 0$. For $h = 0$, when a node fails, a new node is generated by the help of d surviving nodes (among $n-1$ complete storage nodes). Each of d surviving nodes transmits packets by a rate $\beta\xi/\tau$. The transmitted packets from the surviving nodes to the new node might be erased with an erasure probability p. We note that we restrict our analysis to the case where all the links have equal packet erasure probability, p. Where the links between d surviving nodes and the new node have different packet erasure probabilities (like in heterogenous networks), then the approach that requires d nodes equally transmit β fragments to the new node is suboptimal. The following theorem states the capacity of our distributed storage systems.

Theorem 1: Consider a distributed storage system $\mathrm{DSS}(n, k, d, h = 0, \alpha, \alpha', \gamma = d\beta, M\xi, p)$ with a finite number of node failure/repair processes. Suppose T denotes the set of data-collectors in the system, and τ denotes the units of time for reconstructing the original file by data-collectors. There exists a linear code over $GF(q)$ and for $\delta > 0$ and $q > |T|/\delta$ such that data-collectors can achieve, with arbitrarily small error probability, the rate

$$R_0 = (1-\delta)\sum_{i=0}^{k-1} \min\{\alpha\xi/\tau, (d-i)(1-p)\beta\xi/\tau\}. \tag{5}$$

Hence, for large q and ξ, and δ close to zero, there exist linear codes if storage capacity of each node, α, is greater than $\alpha^*(n, k, d, \gamma)$. It is information-theoretically impossible to find a code for $\alpha < \alpha^*(n, k, d, \gamma)$. The function of normalized $\alpha^*(n, k, d, \gamma)$ (α^*/M) over normalized γ (γ/M) is given as follows.

$$\alpha^*/M = \begin{cases} \frac{1}{k} & \text{if } \frac{\gamma}{M} \in [h(0), +\infty), \\ \frac{1-g(i)(1-p)\gamma/M}{k-i} & \text{if } \frac{\gamma}{M} \in [h(i), h(i-1)), \end{cases}$$

where $i = 1, \ldots, k-1$, and

$$h(i) \triangleq \frac{2d}{[(2k-i-1)i + 2k(d-k+1)](1-p)}, \tag{6}$$

and

$$g(i) \triangleq \frac{(2d+2k+i+1)i}{2d}. \tag{7}$$

The complete proof of Theorem 1 is provided in Appendix B. We use infinite length codewords (infinite number of packet transmissions) to achieve the optimal storage-bandwidth tradeoff. We note that infinite length codewords might not be practical, as it may lead to infinite delay in repair. However the analysis is still useful since it provides us insights on the repair problem in packet erasure networks and it provide performance bounds. The result of asymptotic analysis shows a closed-form relation between the storage-bandwidth tradeoff in packet erasure networks with the tradeoff in lossless networks. That is, for a given packet erasure probability p and for a given storage capacity α, the asymptotic repair-bandwidth in the packet erasure network is $1/(1-p)$ times larger than the corresponding repair-bandwidth in the lossless network. Fig. 2 shows the optimal storage-bandwidth tradeoffs in packet erasure networks with $p = 0.1, 0.2, 0.3$. Expectedly, a larger packet erasure probability leads to a higher repair traffic. Two extreme points on the storage-bandwidth tradeoff can be computed by the following equations. For the EMSR codes,

$$\begin{aligned} \alpha_{\mathrm{EMSR}_{h=0,p}} &= \frac{M}{k}, \\ \gamma_{\mathrm{EMSR}_{h=0,p}} &= \frac{Md}{k(d-k+1)(1-p)}. \end{aligned} \tag{8}$$

For the EMBR codes,

$$\begin{aligned} \alpha_{\mathrm{EMBR}_{h=0,p}} &= \frac{2Md}{k(2d-k+1)}, \\ \gamma_{\mathrm{EMBR}_{h=0,p}} &= \frac{2Md}{k(2d-k+1)(1-p)}. \end{aligned} \tag{9}$$

V. Reducing Repair-Bandwidth by DR Storage Nodes

In what follows, we will discuss more details on DR storage nodes and their benefits in repair. We first discuss the benefits of DR storage nodes in lossless networks (i.e., for $p = 0$). Then we will extend the results to a general form ($p > 0$).

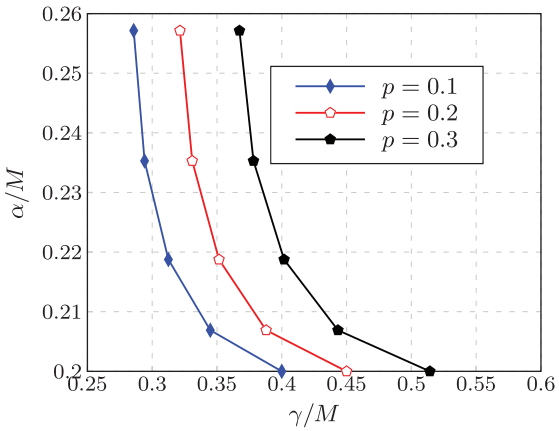


Fig. 2. Fundamental storage-bandwidth tradeoffs in packet erasure networks, for $n = 10$, $k = 5$, $d = 9$, $h = 0$, and for different packet erasure probabilities. The higher packet erasure probability (p), the higher repair-bandwidth.

A. Motivation

We illustrate the use of DR storage nodes by two examples.

Example 1 (DR Storage Nodes for EMSR Codes). Consider a distributed storage system depicted in Fig. 3. Suppose, a file of size 4 fragments is encoded by a $(3, 2)$-MDS code over $GF(13)$, and distributed among 3 complete storage nodes such that any complete storage node stores 2 fragments. When there is no a DR storage node, if any storage node fails, then repairing a failed node requires 4 fragment transmissions (substituting $d = n - 1 = 2, k = 2, M = 4$ in (2)). Now, assume that there exists a DR storage node with the storage capacity of one fragment and storing a coded fragment $a_1 + a_2 + b_1 + b_2$. We describe how the DR storage node help in repairing nodes 1, 2 and 3. If node 1 fails, nodes 2 sends $a_2 + b_2$, node 3 sends $8a_2 + 8b_2 + 12a_1 + 9b_1$, and the DR storage node transmits $a_1 + a_2 + b_1 + b_2$, as shown in Fig. 4a. The new node can generate fragments a_1, b_1 by some linear operations on the received fragments. This is accomplished by removing the *interfering* term $a_2 + b_2$ and then solving a two-dimensional linear equation (for more details about the method of interference alignments in repair please see [8], [27], [28]). Similarly if nodes 2 or 3 fails, we can show that with the DR storage node, repairing any failed node requires 3 fragment transmissions. In node 3, the fragments in the new node is generated by exact repair through these linear operations over the new node's received fragments; $3a_1 + 2a_2 = (4a_1 + b_1) + (3a_2 + b_2) - (a_1 + b_1 + a_2 + b_2)$; $9b_1 + 8b_2 = 12(a_1 + b_1 + a_2 + b_2) - 3(4a_1 + b_1) - 4(3a_2 + b_2)$. Thus, in each repair process, one unit of (fragment) transmission is saved.

Example 2 (DR Storage Nodes for EMBR Codes): Consider a distributed storage system depicted in Fig. 4. Here, the distributed storage system stores a file using a $(4, 3, 3) - \text{EMBR}_{h=1,p=0}$ over $GF(2)$. A source file contains nine fragments $c_1, \ldots, c_9$. A fragment c_{10} is encoded by $c_{10} = c_4 + c_7 + c_9$. Nodes 1, 2, 3 and 4 store the source file such that any 3 nodes can reconstruct the source file. A DR storage node storing c_4, c_7, c_9 can help all the other storage nodes in repair. This is shown in Fig. 4.

In general, solving (4) in Lemma 1 for a given M and $p = 0$ leads to a lower bound of repair-bandwidth. For two extreme points, the lower bound of the repair bandwidth are as follows. For EMSR codes, we have,

$$\gamma_{\text{EMSR}_{h,p=0}} = \frac{M(d+h)}{k(d-k+h+1)}. \tag{10}$$

And for EMBR, we have

$$\alpha_{\text{EMBR}_{h,p=0}} = \frac{2M(d+h)}{k(2(d+h)-k+1)} \tag{11}$$

$$\gamma_{\text{EMBR}_{h,p=0}} = \frac{2M(d+h)}{k(2(d+h)-k+1)}. \tag{12}$$

The above results show potential reduction in the repair-bandwidth by a factor of $d(d+h-k+1)/((d+h)(d-k+1))$ for EMSR codes, and by a factor of $d(2(d+h)-k+1)/((d+h)(2d-k+1))$ for EMBR codes. We note that we could achieve these gains by adding h complete storage nodes. However DR storage nodes have smaller storage space (we will show later) and require less repair-bandwidth compared to complete storage nodes. The minimal α' will be discussed as follows.

B. Minimum DR Storage Capacity for the EMSR Codes

We show that there exist linear codes for repairing in a DSS satisfying the bound in (10) when each complete storage node stores ($\alpha = M/k$) fragments, and each DR storage node stores $\alpha' = \beta$ fragments.

Lemma 2 (Achievability for EMSR codes with $\alpha' = \beta$): For a repair process in a DSS with parameters $(n, k, d, h \leq \lfloor M/\beta \rfloor, \alpha = M/k, \alpha' = \beta, \beta, M)$ of EMSR codes, there exist linear codes if each of $d + h$ helper nodes, including the DR storage nodes, transmit $\beta = M/(k(d+h-k+1))$ fragments to the new node. In addition, the functional repair is always possible after any number of node failure/repair processes.

Proof: See Appendix C. ■

Using Lemma 2 and the fact that we design $\alpha' \geq \beta$, we can deduce the optimal bound of α' for EMSR codes.

Theorem 2: For a repair process in a DSS with parameters $(n, k, d, h \leq \lfloor M/\beta \rfloor, \alpha = M/k, \alpha', \beta, M)$ of EMSR codes, the optimal storage space for DR storage nodes is $\alpha' = \beta$.

Theorem 2 shows that reducing the repair-bandwidth for the EMSR codes can be achieved by adding a DR storage node with storage capacity of β instead of using a complete storage node with capacity $\alpha = M/k = \beta(d+h-k+1)$. Hence, it requires less storage space by the ratio of $(d+h-k+1)$.

In the next subsection, we investigate the minimum storage required for a DR storage node for EMBR codes.

C. Minimum Repairing Storage Capacity for EMBR Codes

A DR storage node requires more storage space for EMBR codes compared to EMSR codes (where it was $\alpha' = \beta$). We will present in the following lemma that for the EMBR codes it is necessary to have $\alpha' \geq k\beta$.

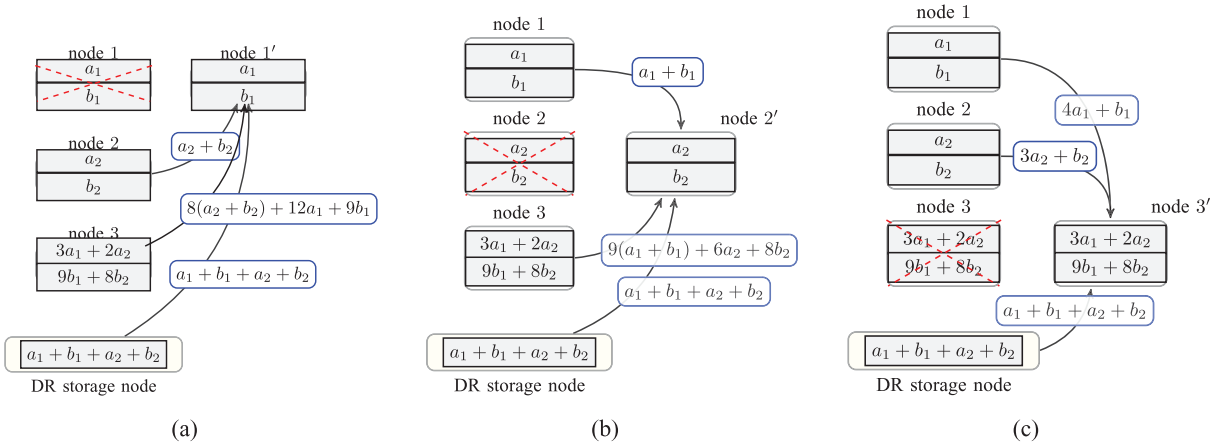


Fig. 3. A distributed storage system with a DR storage node. Nodes store a file based on a $(3, 2, 2) - \text{EMSR}_{h=1,p=0}$ code. The DR storage nodes helps the repair of nodes 1, 2, and 3 respectively in (a), (b), (c). The finite field here is GF(13). This example shows a DR storage node, with a half storage space compared to a complete storage node, functions similar to a complete storage node in repair. Please refer to Fig. 5 to see the process of repair for the DR storage node.

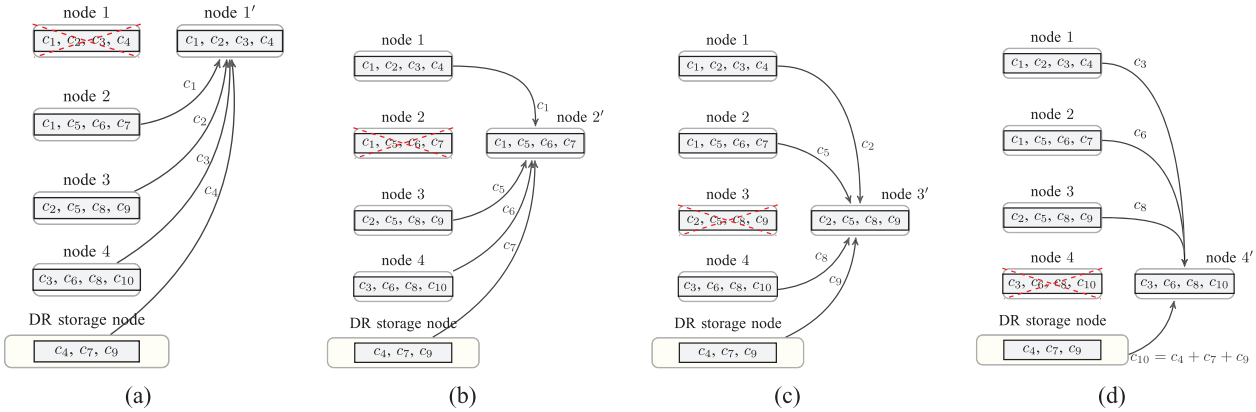


Fig. 4. A DR storage node when nodes 1, 2, 3, 4, and 5 store a file based on a $(4, 3, 3) - \text{EMBR}_{h=1,p=0}$. The DR storage nodes helps the repair of nodes 1, 2, 3, 4 respectively in (a), (b), (c), (d). The finite field here is GF(2). This example shows a DR storage node, with 3/4 storage space of a complete storage node, functions similar to a complete storage node in repair. Please refer to Fig. 6 to see the process of repair for the DR storage node.

Lemma 3 (Non-achievability for EMBR codes with $\alpha' < k\beta$): For the EMBR codes in a DSS with parameters $(n, k, d, h, \alpha = (d + h)\beta, \alpha', \beta, M)$, $\alpha' \geq k\beta$.

Proof: See Appendix D. ■

The next lemma shows that it is sufficient for the DR storage node to have $\alpha' = k\beta$.

Lemma 4 (Achievability for EMBR codes with $\alpha' = k\beta$): For the repair process of EMBR codes with DR storage nodes, each of which has storage capacity $\alpha' = k\beta$, there exists a linear code if each of $d + h$ nodes, including $h \leq \lfloor M/k\beta \rfloor$ DR storage nodes, transmits $\beta = 2M/(k(2(d + h) - k + 1))$ fragments to the new node. In addition, the functional repair is always possible after any number of node failure/repair processes.

Proof: See Appendix E. ■

Using Lemma 3 and 4, we can deduce the optimal bound of α' for EMBR codes.

Theorem 3: For a repair process in a DSS with parameters $(n, k, d, h \leq \lfloor M/k\beta \rfloor, \alpha, \alpha', \beta, M)$ of EMBR codes, the optimal storage space for DR storage nodes is $\alpha' = k\beta$.

Theorem 3 shows that reducing the repair-bandwidth for the EMBR codes can be achieved by adding a DR storage node with storage space of $\alpha' = k\beta$ instead of $\alpha = (d + h)\beta$. Note that in some scenarios (if not most), $d = n - 1$. Thus, using DR storage nodes can reduce storage space.

D. Repairing a DR Storage Node

Since DR storage nodes can also fail, we investigate the repair-bandwidth of DR storage nodes. We first show the repair of a DR storage node in Fig. 5 for the EMSR code of our running example (Example 1) and in Fig. 6 for the EMBR code of our running example (Example 2). Repairing a DR storage node in each of these examples requires one unit fewer (fragment) transmissions than the repair bandwidth of a complete storage node. Next, we study the problem in a more general case.

Similar to a complete storage node, the repair for a DR storage node can be functional or exact. In the functional repair (which includes the exact repair also) of a DR storage node, a number of complete storage nodes plus the surviving DR storage nodes transmit sufficient data to the new DR storage node. Let $\gamma_R = (d_R + h_R)\beta_R$ be the total repair bandwidth for a DR storage node, where d_R denotes the number of complete storage nodes in repair, h_R denotes the number of DR storage

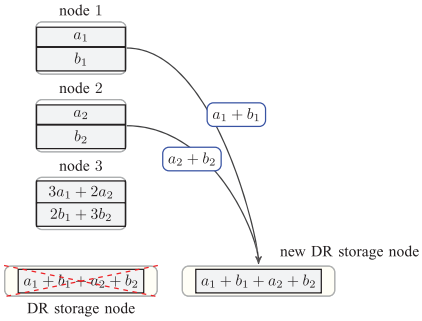


Fig. 5. Repair of a DR storage node for EMSR codes. The new DR storage node is generated by transmitting two fragments from nodes 1 and 2. The repair-bandwidth of a DR storage node is smaller than the repair-bandwidth for a complete storage node.

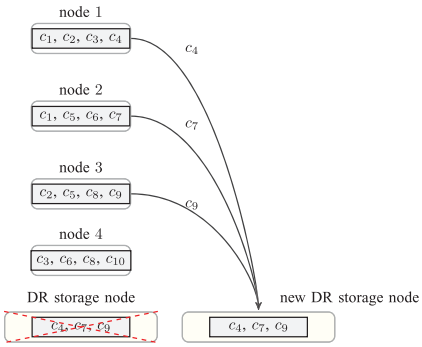


Fig. 6. Repair of a DR storage node for EMBR codes. The new DR storage node is generated by transmitting three fragments from nodes 1, 2 and 3. The repair-bandwidth of a DR storage node is smaller than the repair-bandwidth for a complete storage node.

nodes (among $h-1$ surviving DR nodes) in repair, β_R denotes the number of fragments each helper node sends in repair. At this stage, the optimal repair-bandwidth of a DR storage node is still not known. In this paper, we will show in the following proposition that the bandwidth $\gamma_R = k\beta$ is sufficient for the (exact) repair of a DR storage node. We note that $\gamma_R = k\beta$ may be suboptimal, but it still requires less repair-bandwidth than a complete storage node.

Proposition 1: Repair-bandwidth $\gamma_R = k\beta$ is sufficient for regenerating a new DR storage node for EMSR and EMBR codes.

Proof: See Appendix F. ■

Proposition 1 shows that the repair-bandwidth for a DR storage node is not greater than the repair-bandwidth for a complete storage nodes ($k\beta \leq d\beta$).

E. DR Storage Nodes for non-EMSR and non-EMBR Codes

In the previous subsections, we studied the DR storage node for two extreme points on the storage-bandwidth tradeoff, namely EMSR and EMBR codes. Using the same approach, we can extend the results to all the points on the storage-bandwidth tradeoff. A formal result is given in the following conjecture.

Conjecture 1: Consider a distributed storage system $(n, k, d, h, \alpha, \alpha^{'}, \beta, M, p = 0)$ with a code on the optimal storage-bandwidth tradeoff point $(\sum_{i=0}^{k-1} \min\{\alpha, (d+h-i)\beta\} = M)$. Let the points on the optimal tradeoff that are between EMSR and EMBR (two extreme) points are denoted as an the interior points. If we present an interior point on the storage-bandwidth tradeoff by subdividing the set of points on the optimal storage-bandwidth tradeoff into $(k-1)$ subsets as follows: $S_\theta = \{(\alpha, \beta) | \beta(d-\theta) < \alpha \leq \beta(d-\theta+1)\}$ for $\theta \in \{1, 2, \cdots, k-2\}$. Then the repairing storage capacity $\alpha^{'}_\theta = (k-\theta)\beta$.

Note that in this setting for EMSR codes we have $(d+1-(k-1)) \leq \alpha$, and for EMBR codes $\alpha \leq (d+1-(0))\beta$, which corresponds respectively to $\theta_{EMSR} = k-1$, and $\theta_{EMBR} = 0$. Thus in general $\alpha^{'}_\theta = (k-\theta)\beta$ is the optimal required repairing storage space for the *extended regenerating codes*.

Remark 1: We remark that the above conjecture is for functional repair and the optimal bound for exact repair is still not known.

F. Repair With DR Storage Nodes in Packet Erasure Networks

The above results show that repair bandwidth can be reduced if DR storage nodes are used for a distributed storage system with error-free channels. In what follows, we shall show that DR storage nodes can also reduce the repair-bandwidth in packet erasure networks. The following two corollaries illustrate more formally the effect of DR storage nodes in packet erasure networks. Since the analysis and code construction are similar to Section IV, we skip the proof.

Corollary 1: For a DSS with parameters $(n, k, d, h, \alpha = M/k, \alpha^{'} = \beta, \gamma, M\xi, p)$ in a packet erasure network with channels having packet erasure probability p, the asymptotic optimal repair-bandwidth for EMSR codes is $\gamma_{\text{EMSR}_{h,p}}/M = (d+h)/(k(d+h-k+1)(1-p))$.

Similarly, for EMBR codes, we have the following corollary.

Corollary 2: For a DSS with parameters $(n, k, d, h, \alpha, \alpha^{'} = k\beta, \gamma, M\xi, p)$ in a packet erasure network with packet erasure probability p for all channels, the asymptotic optimal repair-bandwidth for EMBR codes is $\gamma_{\text{EMBR}_{h,p}}/M = 2(d+h)/(k(2d+2h-k+1)(1-p))$.

Remark 2: We note that in this paper, a DR storage nodes behaves the same as a complete storage node in repair and transmits the same amount of repairing data as a complete storage node. That is, if β' denotes the amount of data transmitted by a DR storage node in repair, $\beta' = \beta$. The analysis for $\beta' \neq \beta$ can be studied as future work, following a similar approach as the study of flexible regenerating codes in [29], in which the surviving nodes are allowed to transmit different amount of repairing data.

VI. Finite File Size Analysis

A. The Probability of Successful Repair

Above, we investigated the optimal repair-bandwidth in packet erasure networks under the assumption that the stored file contains an infinite number of packets (infinite order of subpacketization). In practice, a large order of subpacketization leads to high traffic load for packet headers and high complexity in decoding (e.g., by Gaussian elimination method, it requires

the inverse of an $M\xi \times M\xi$ matrix). Meanwhile in some networks the packet size is fixed (or finite). Thus the number of packets in repair traffic might be finite. It is interesting to study the repair-bandwidth in the case of a finite order of subpacketization. Since the channel between the surviving nodes and the new node is lossy, then the repair may fail (the new node may not receive sufficient repair packets). We will first analyze the probability of successfully receiving β fragments from a surviving node, denoted by P_β. Then, we will study the probability of successful repair (PSR). We note that the asymptotic optimal repair-bandwidth in Section IV can serve as a lower-bound of the repair-bandwidth for a finite subpacketization order.

To successfully repair, the new node must receive β fragments (that equals to $\beta\,\xi$ packets) from each of d surviving nodes. Due to erasures, a surviving node transmits $t\,\xi$ packets produced by linear combinations (in GF(q)) of $\beta\,\xi$ repair packets). For a packet erasure channel, $t = \beta/(1-p)$ if $\xi \to \infty$, which means infinite subpacketization order. For finite ξ, the total number of packet transmissions, t, may be much larger than $\beta/(1-p)$, and it depends on parameters q and ξ. The successful receiving of β fragments from a surviving node depends on two conditions. Firstly, i packets must be received for $i \geq \beta\,\xi$. Secondly, $\beta\,\xi$ packets with independent encoding vectors among i successfully received packets are available at the new node. The first condition forms a binomial distribution and the second condition is derived by the probability of spanning $\beta\,\xi$ dimensions by i vectors [30], [31]. Thus, we have (more details are also provided in Appendix G):

$$P_\beta = \sum_{i=\beta\,\xi}^{t\,\xi} \binom{t\,\xi}{i} (1-p)^i\, p^{(t\,\xi - i)}\, \frac{\prod_{l=0}^{\beta\,\xi-1} \left(q^i - q^l\right)}{q^{\beta\,\xi\,i}} \text{ if } t \geq \beta. \tag{13}$$

When d surviving nodes are sending repairing packets, the probability of successful repair, denoted as P_s, is obtained by

$$P_s = P_\beta^d. \tag{14}$$

Figure 7 shows P_β for a distributed storage system using MBR codes with parameters $(n = 10, k = 5, d = 9, \beta = 2, M = 70)$, $p = 0.3$ and $q = 5$. We observe that the required repair-bandwidth is larger than the optimum (asymptotic) repair-bandwidth. That is, the bandwidth overhead ratio t/β might be several times larger than the optimal value $1/(1-p) \approx 1.43$ due to a finite subpacketization order.

B. Practical-Repair-Bandwidth

We formally define the practical-repair-bandwidth as the minimum required bandwidth to achieve $P_s \geq 1-\delta$, where δ denotes as a parameter indicating how close the PSR is to 1 and $(0 \leq \delta \leq 1)$. Let $\widehat{\gamma}(\delta, d)$ denote the practical-repair-bandwidth for given δ and d, then

$$\widehat{\gamma}(\delta, d) = \min_t d\,t, \quad \text{subject to: } P_s \geq 1-\delta, \tag{15}$$

where t is the number of packets transmitted by each surviving node.

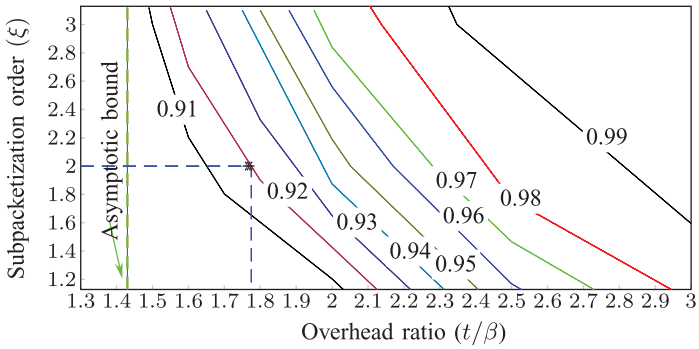


Fig. 7. P_β (numbers in the curves) in erasure networks for different values of ξ and bandwidth overhead ratio t/β. The field size here is $q = 5$ and $p = 0.3$. The larger ξ, the closer the bandwidth overhead ratio to the optimal value $1/(1-p) \approx 1.43$. As an example, the $*$ symbol in the figure shows that for achieving $P_\beta \geq 0.92$, and for a given $\xi = 2$, each surviving node must transmit at least 1.77β fragments.

C. Reducing Practical-Repair-Bandwidth

We have shown that DR storage nodes can reduce the repairing bandwidth for the scenario of infinite repair packets. Similarly, we can show that DR storage nodes reduce the bandwidth for finite repair packets. However, in what follows, we propose another method to reduce the practical repair-bandwidth. Consider a repair process in a distributed storage system. Packets on the links are erased i.i.d. with probability p. Successful receiving $\beta(d_1)^2\,\xi$ packets from d_1 surviving nodes guarantees successful repair. Suppose that the repair packets from each surviving node are independent of the set of d_1 nodes. This can happen for example by the use of product-matrix regenerating codes in [19]. Then the new node by receiving from any set of d_1 nodes can successfully be generated. Thus, to increase the PSR, $d_{\text{tot}} = d_1 + d_2$, $(d_2 \geq 0)$ surviving nodes transmit repairing packets. Each surviving node transmits $t\,\xi$ packets, each of which is formed by a linear combination of $\beta(d_1)\,\xi$ repair packets over $GF(q)$. Hence, repair is successful if the new node receives $\beta(d_1)$ fragments from at least d_1 links. We also note that the packets transmitted from one surviving node can only be used to decode the repairing information of that surviving node. Hence, the PSR is the probability by which the new node receives $\beta(d_1)$ fragments from at least d_1 out of $d_1 + d_2$ surviving nodes. It is calculated by

$$P_s = \sum_{i=d_1}^{d_1+d_2} \binom{d_1+d_2}{i} (P_\beta)^i (1-P_\beta)^{d_1+d_2-i}. \tag{16}$$

The PSR for two schemes when $d_1 = 9, d_2 = 0$ and $d_1 = 7$, $d_2 = 2$ have been compared in Fig. 8. For this example on both schemes MBR codes with parameters $(n = 10, k = 5, M = 70)$ are used and other parameters are set as $\xi = 5$, $q = 5$ and $p = 0.3$. Moreover, the practical repair-bandwidth for $P_s \geq 0.99$ (i.e., $\delta = 0.01$) has been compared between these two methods. We see that the scheme with smaller asymptotic optimal repair-bandwidth has almost twice larger practical-repair-bandwidth than the other scheme. This motivates the

[2] β is a function of d_1 helper nodes

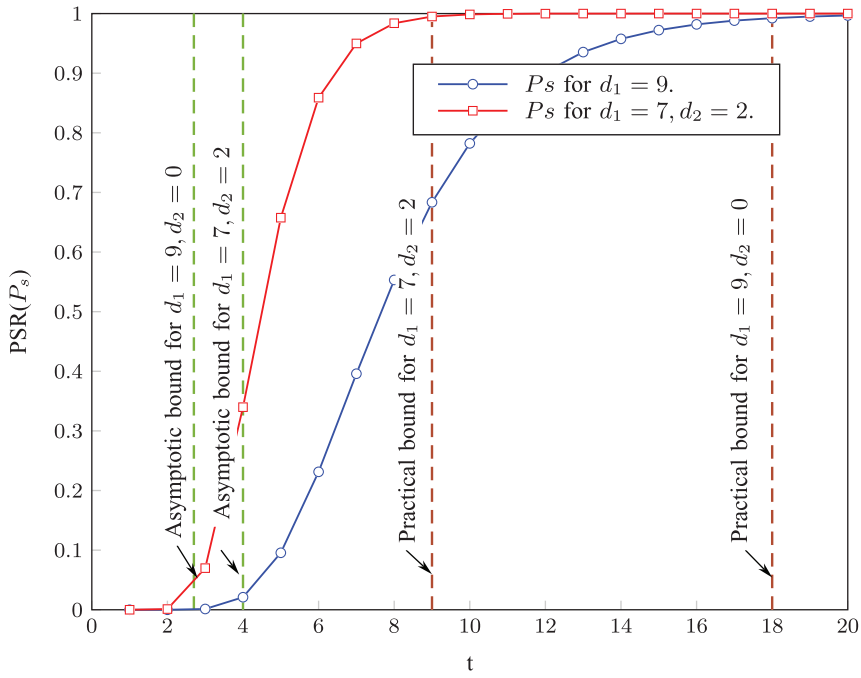


Fig. 8. Probability of successful repair in packet erasure networks for two schemes over the repair-traffic from each node. The first scheme aims at reducing asymptotic repair bandwidth and use $d_1 = 9, d_2 = 0$. The second scheme uses $d_1 = 7, d_2 = 2$. The figure (vertical lines) also compares practical repair-bandwidth regarding $PSR \geq 0.99$ versus optimal asymptotic repair-bandwidth. Here, MBR codes are used and parameters are set as $n = 10, k = 5, M = 70, \xi = 5, q = 5$.

following optimization problem. Given the constraint that the PSR is greater than $1 - \delta$, we aim to minimize the practical-repair-bandwidth $\widehat{\gamma}(\delta, d_1 + d_2)$ with variables d_1 and d_2. The optimization problem can be formulated as follows,

$$\begin{aligned} \min_{d_1,d_2} \quad & \widehat{\gamma}(\delta, d_1 + d_2) \\ \text{subject to:} \quad & P_s \geq 1 - \delta, \\ & d_1 + d_2 \leq d_{\text{tot}}. \end{aligned} \tag{17}$$

It is not straightforward to find an analytical solution for the optimization problem. The optimal practical-repair-bandwidth solution depends on the probability of packet erasure on the links, the finite field size and the subpacketization order. We use the optimization problem in the previous example and find the corresponding d_1 and d_2 for the given finite field size $q = 5$ and subpacketization order $\xi = 5$ over the different packet erasure probabilities. d_1 and d_2 that minimize the repair-bandwidth are shown in Fig. 9. We can conclude that for the network with higher erasure probabilities, more redundant data (larger d_2) should be used to reduce the practical repair-bandwidth. Conversely, for the network with lower erasure probabilities, less redundant data is needed and the optimal practical repair-bandwidth is closer to the optimal asymptotic repair-bandwidth.

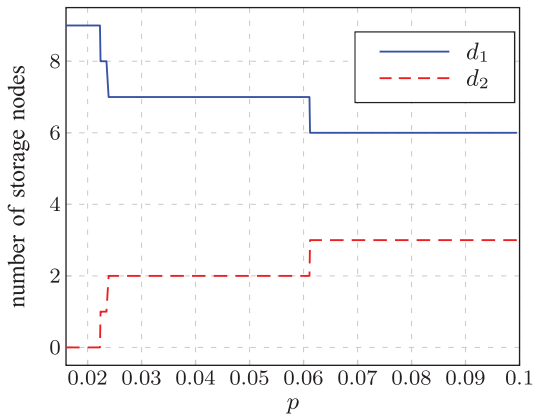


Fig. 9. The value of d_1 and d_2 that minimize the practical repair-bandwidth over different values of links packet erasure probability, p. Here, $\delta = 0.01$, and p changes from 0.01 to 0.1. For a network with higher p the reliability of repair becomes more critical and then repair-bandwidth is minimized when the redundant information is increased (d_2 increases). Here, $q = 5, \xi = 5$.

VII. Conclusions

We studied the repair problem for distributed storage systems with packet erasure channels and dedicated-for-repair (DR) storage nodes. We investigated the optimal storage-bandwidth tradeoffs for packet erasure networks. We proposed DR storage nodes to reduce the repair-bandwidth. A benefit of DR storage nodes is smaller storage capacity and repair-bandwidth, compared to complete storage nodes. We also studied the necessary minimal storage capacity for the DR storage nodes. We note that the benefit of DR storage nodes is not only limited to lossy networks, but they are also valid in reducing the repair-bandwidth in lossless networks. Then, we investigated the probability of successful repair and then proposed an approach to reduce the repair-bandwidth when the file size is finite. To reduce the practical repair-bandwidth, we formulated an optimization problem and showed that the optimal solution for that depends on the channel erasure probability.

APPENDIX

A. Proof of Lemma 1

Inspired by the analysis in [1] for distributed storage systems in lossless networks, we provide the proof for distributed storage systems with packet erasure channels and DR storage nodes. Another useful information-theoretic tool in the proof is the cut-set upper bound analysis introduced by Cover and Thomas [26]. Given a subset $\mathcal{V}_x$ of nodes in a network (i.e., $\mathcal{V}_x \subset \mathcal{N}$, where $\mathcal{N}$ is the set of nodes in the network), when all the nodes in $\mathcal{V}_x$ perfectly cooperate and also all the nodes in $\mathcal{V}_x^c$ [3] perfectly cooperate, then the network can be modeled to a multiple-input multiple-output point to point erasure channel. Since the channels are independent and memoryless, the upper bound of the information transfer between $\mathcal{V}_x$ and $\mathcal{V}_x^c$ is when inputs to channels are independent and have uniform distribution among the set of alphabets. Thus the upper bound of information transfer is the sum of capacities of edges between $\mathcal{V}_x$ and $\mathcal{V}_x^c$.

We now show that the information rate in the modified information flow graph is upper bounded by $\sum_{i=0}^{k-1} \min\{\alpha\xi/\tau, (d+h-i)(1-p)\beta\xi/\tau\}$. That is equal to show that the min-cut in the information flow graph equals to $\sum_{i=0}^{k-1} \min\{\alpha\xi/\tau, (d+h-i)(1-p)\beta\xi/\tau\}$ [32]. The information flow graph after k failure/repair processes can be shown in k subsequent stages, as depicted in Fig. 1. We first prove there is a cut in the network with capacity $\sum_{i=0}^{k-1} \min\{\alpha\xi/\tau, (d+h-i)(1-p)\beta\xi/\tau\}$. For the purpose, consider a cut that passes a route with a minimum capacity at any stage of repair. For example at stage 1, the cut selects a route between $\alpha\xi/\tau$ and $(d+h)(1-p)\beta\xi/\tau$, where $(1-p)\beta\xi/\tau$ is the capacity of the packet erasure link between one of the helper nodes and the new node. At stage 2, since the new node can get $(1-p)\beta\xi/\tau$ packets per unit time from the previously generated node, then the cut selects between $\alpha\xi/\tau$ and $(d+h-1)(1-p)\beta\xi/\tau$, and so on. Finally, there will be a graph with a cut capacity equivalent to $\sum_{i=0}^{k-1} \min\{\alpha\xi/\tau, (d+h-i)(1-p)\beta\xi/\tau\}$. Any other cut has a capacity larger than, or equal to, $\sum_{i=0}^{k-1} \min\{\alpha\xi/\tau, (d+h-i)(1-p)\beta\xi/\tau\}$. This concludes that $\sum_{i=0}^{k-1} \min\{\alpha\xi/\tau, (d+h-i)(1-p)\beta\xi/\tau\}$ is the min-cut, and consequently it is the upper bound of the information transfer rate between the source and a data-collector in this network.

B. Proof of Theorem 1

We use random linear codes to prove the achievability. The converse is proved as a corollary of Lemma 1, where we substitute $h = 0$ in the derived upper-bound.

Achievability: Let us first prove for the case that there is only one data-collector. Suppose packets $w_1, w_2, \ldots, w_{M\xi}$ are the source data. There is a data collector which is connected to a subset of k out nodes of complete storage nodes. We apply the random linear codes proposed in [22], [33] for achieving the capacity of multicasting over lossy packet networks.

Encoding: Packets $v_1, v_2, \ldots, v_\Omega$ are constructed by random linear combinations of the original packets $w_1, w_2, \ldots, w_{M\xi}$. That is, v_l for $l = 1, \ldots, v_\Omega$ is

$$v_l = \sum_{i=1}^{M\xi} \eta_{li} w_i, \tag{18}$$

where η_{li} is selected randomly and uniformly from $GF(q)$. The packets $v_1, v_2, \ldots, v_\Omega$ are then injected to the source node according to a Poisson distribution by a constant rate $R_0 = (1-\delta)\sum_{i=0}^{k-1} \min\{\alpha\xi/\tau, (d-i)(1-p)\beta\xi/\tau$, for some $\delta > 0$. For a node, in an occasion of a packet transmission, the packet is formed by a linear combinations of the packets that previously have been received by the node. Thus, a packet x, the input of channels, can be written as a linear combination of $v_, \ldots, v_\Omega$, as

$$x = \sum_{j=1}^{\Omega} \pi_i v_i \tag{19}$$

$$= \sum_{j=1}^{\Omega} \pi_i \sum_{i=1}^{M\xi} \eta_{li} w_i \tag{20}$$

$$= \sum_{i=1}^{M\xi} \left(\sum_{j=1}^{\Omega} \pi_i \eta_{ji} \right) w_i \tag{21}$$

$$= \sum_{i=1}^{M\xi} \phi_i w_i. \tag{22}$$

Here, vector $\pi = [\pi_j]_{j=1}^{\Omega}$ is called the *auxiliary encoding vectors* associated with packet x and vector $\phi = [\phi_i]_{i=1}^{M\xi}$ is called the *global encoding vectors* associated with packet x. We say packet x is *innovative* to node i, if the auxiliary encoding vector of packet x is not in the span of the auxiliary encoding vectors of the previously received packets in node i. The global encoding vector of each packet is placed in the header of the packet.

Decoding: A data-collector by receiving $M\xi$ packets with independent global encoding vectors can successfully decode the source data.

Side Information at Data Collector: A data-collector for successfully decoding the file needs global encoding vectors of the received packets. The global encoding vector of each packet can be placed inside each packet's header. The overhead of side information can be made negligible if the packet size is large enough.

Analyzing the Probability of Error: Using the above encoding and decoding, the error in decoding may happen due to two error events: the first event (E_1) is that the decoder receives fewer than $M\xi$ innovative packets; second event (E_2) is that decoder receives greater than or equal to $M\xi$ innovative packets, but there is no $M\xi$ independent packets (packets having independent global encoding vectors) such that $M\xi \times M\xi$ global encoding matrix become full rank.

The probability of the first error event can be derived by the analyzing the propagation of innovative packets in the network. The results in [22] shows that if random network coding, as described above, is used to achieve the capacity of multicast packet erasure networks, then the propagation of innovative packets in network follows the same as queuing network where

[3] $\mathcal{V}_x^c$ denotes the complement set of $\mathcal{V}_x$.

each node in the network, works like a $M/M/1$ queuing system. That is, if φ_i is the arrival rate of innovative packets to node i, the node i has service rate $\varphi_i(1-1/q)/(1-\delta)$. This queue to be stable it requires $\varphi_i \leq \varphi_i(1-1/q)/(1-\delta)$. Thus we must have $q \geq 1/\delta$. When this queuing networks works for a long time, then the number of innovative packets in the network (denoted as N_i) is time-invariant random variable [22]. Therefore, the data collector receives fewer than $M\xi$ innovative packets if $\Omega - N_i < M\xi$. The probability that $N_i > \Omega - M\xi$ can be made arbitrarily small, by selecting $\Omega = \lfloor M\xi/R_c \rfloor$, for $R_c < 1$, and then selecting large ξ. (For more details, please refer to the proof of Theorem 1 in [22].)

The probability of the second error event can be bounded by

$$\begin{aligned} P(E_2) &= Prob((\text{No. of indep. glob. vectors} < M\xi)| \\ &\quad (\text{No. of innov. vec.} \geq M\xi)) \quad (23) \\ &\leq Prob((\text{No. of indep. glob. vectors} < M\xi)| \\ &\quad (\text{No. of innov. vec.} = M\xi)) \quad (24) \\ &\leq \prod_{i=1}^{M\xi}(1-1/q^i) \quad (25) \end{aligned}$$

This can be made arbitrarily small by selecting large q and $\xi \to \infty$.

When there are a set T of data collectors (a multicast network) the proof follows the same except that for having stable queue, we must have $\varphi_i \leq \varphi_i(1-|T|/q)/(1-\delta)$. (For more details, please refer to the proof of Theorem 2 in [22]). This gives us a condition on the required finite field size as $q > |T|/\delta$.

Achievable Rate: Finally, by the above random network coding the rate is $M\xi/\tau$, where $M\xi$ is the total information transmitted and τ is the time taken for packets $v_1, \ldots, v_\Omega$ to reach to source node S by rate $R_0 = (1-\delta)\sum_{i=0}^{k-1}\min\{\alpha\xi/\tau, (d-i)(1-p)\beta\xi/\tau$. For ξ sufficiently large, Ω would be sufficiently large, then with a probability of error not exceeding ε the rate would be,

$$\begin{aligned} \frac{M\xi}{\tau} &> \frac{M\xi}{\Omega(1/R_0+\varepsilon)} \quad (26) \\ &\geq \frac{Rc\,R_0}{1+\varepsilon R_0} = \frac{Rc\,R(1-\delta)}{1+\varepsilon R(1-\delta)}, \quad (27) \end{aligned}$$

which can be made arbitrarily close to $R = \sum_{i=0}^{k-1}\min\{\alpha\xi/\tau, (d-i)(1-p)\beta\xi/\tau\}$. In the above equations, (26) holds because of Fano's inequality, and (27) holds because $\Omega = \lfloor M\xi/R_c \rfloor$. By Lemma 1, it is impossible to achieve a rate greater than R. Thus, the rate $R = \sum_{i=0}^{k-1}\min\{\alpha\xi/\tau, (d-i)(1-p)\beta\xi/\tau\}$ is the optimal achievable bound for this distributed storage system. Hence, for the optimal code we have $M\xi/\tau = \sum_{i=0}^{k-1}\min\{\alpha\xi/\tau, (d-i)(1-p)\beta\xi/\tau\}$. Then, for finite but very large ξ and τ, and δ very close to zero, we define $C_0(\alpha) = \sum_{i=0}^{k-1}\min\{\alpha, (d-i)(1-p)\beta\} = M$. Then, the optimal repair-bandwidth for varying individual node storage capacities can thus be derived by the similar approach adopted in [1]. Then,

$$C_0(\alpha) = \begin{cases} k\alpha & \alpha \in [0, b_0] \\ b_0 + (k-1)\alpha & \alpha \in (b_0, b_1] \\ \vdots & \\ b_0 + b_1 + \cdots + b_{k-1} & \alpha \in (b_{k-1}, \infty] \end{cases} \quad (28)$$

where,

$$b_i \triangleq \left(1 - \frac{k-1-i}{d}\right)(1-p)\gamma. \quad (29)$$

Then, we find the minimum α, denoted as $\alpha* = C_0^{-1}(M)$. Hence,

$$\alpha^* = \begin{cases} M/k & M \in [0, kb_0] \\ \frac{M-b_0}{(k-1)} & M \in (kb_0, b_0 + (k-1)b_1] \\ \vdots & \\ M - \sum_{j=0}^{k-1} b_j & M \in \left(\sum_{j=0}^{k-2} b_j + b_{k-2}, \sum_{j=0}^{k-1} b_j\right) \end{cases} \quad (30)$$

In general we have,

$$\alpha^* = \frac{M - \sum_{j=0}^{i-1} b_j}{k-i}, \quad (31)$$

where,

$$M \in \left(\sum_{j=0}^{i-1} b_j + (k-i)b_{i-1}, \sum_{j=0}^{i} b_j + (k-i-1)b_i\right]. \quad (32)$$

Now from the definition of b_i, we can verify that,

$$\sum_{j=0}^{i-1} b_j = (1-p)\gamma g(i), \quad (33)$$

and

$$\sum_{j=0}^{i} b_j + (k-i-1)b_i = \gamma\frac{M}{h(i)}. \quad (34)$$

Thus, we have,

$$\alpha^* = \frac{M - (1-p)\gamma g(i)}{k-i}, \quad (35)$$

where,

$$M \in \left(\frac{\gamma M}{h(i-1)}, \frac{\gamma M}{h(i)}\right]. \quad (36)$$

By deriving γ based on M, the result as in (6) is obtained.

C. Proof of Lemma 2

The proof is based on using random linear codes for storing the original file in storage nodes and in the repair process. Let $\mathbf{Q}_i$ denotes the encoding matrix of a complete storage node i such that each row of the $\mathbf{Q}_i$ represents the encoding vector

of a fragment stored in node i. Similarly, let $\mathbf{Q}_{h_j}$ denotes the encoding matrix of DR storage node j. Each row of the $\mathbf{Q}_{h_j}$ represents the encoding vector of a fragment stored in DR storage node j. By exploiting sparse-zero lemma [34], we show for a large finite field size there exist codes $\mathbf{Q}_i$ for $i = 1, \ldots, n$ and $\mathbf{Q}_{h_j}$ for $j = 1, \ldots, h$ such that the new node is generated by linear codes and the repair-bandwidth is optimal.

For the code construction, we split the source file of a size M into $k(d + h - k + 1)$ fragments. We denote the source file by vector $\mathbf{x} = [x_1, x_2, \cdots, x_{k(d+h-k+1)}]^T$. Substituting these set of parameters $(d, h, M = k(d + h - k + 1))$ in Eq. (10), we have $\beta = 1$. We construct an $(n, k, d) - \text{EMSR}_h$ code (with a little abuse of notation $(n, k, d) - \text{EMSR}_h$ denotes $(n, k, d) - \text{EMSR}_{p=0,h}$ codes) requiring the minimum repair-bandwidth. Before that, we define a property, denoted as the *extended MDS property*, as follows.

Definition 5 (Extended MDS Property): For a set of n complete storage nodes and h DR storage nodes, we say it has the extended MDS property if: I) for any subset of k complete storage nodes the matrix of their encoding vectors has full rank; II) for any set of one complete storage node plus h DR storage nodes the matrix of their encoding vectors has full rank; III) for any set of $k - 1$ complete storage nodes plus h DR storage nodes the matrix of their encoding vectors has full rank.

Finding a code with conditions I and II is not difficult. However, finding a code satisfying Condition III may look challenging. We note that in our setting, we can find such a code because we always have $h = h\beta < M - (k - 1)\alpha = d + h - k + 1$. That is, there are always some vectors orthogonal to the span of encoding vectors in the set of $k - 1$ complete storage nodes plus h DR storage nodes (because $h\beta + (k - 1)\alpha < M$).

Lemma 5 (Sparse-Zero Lemma): Consider a multi-variable polynomial $g(\alpha_1, \alpha_2, \ldots, \alpha_n)$ which is not identically zero, and has the maximum degree in each variable at most d_0. Then, there exist variables $\gamma_1, \gamma_2, \ldots, \gamma_n$ in the finite field GF(q), for $q \geq d_0$, such that $g(\gamma_1, \gamma_2, \ldots, \gamma_n) \neq 0$.

Proof: See proof of Lemma 19.17 in [34]. ■

Lemma 6: Consider an $(n, k, d) - \text{EMSR}_h$ code satisfying the Extended MDS property. For any set of selecting $k - 1$ complete storage nodes, it is possible to select one vector from each of h DR storage node and one vector from $d - k + 1$ nodes such that resulting matrix of $(k - 1)(d + h - k + 1) + h + (d - k + 1) = k(d + h - k + 1)$ encoding vectors has full rank.

Proof: For an encoding matrix $\mathbf{Q}_i$, let span($\mathbf{Q}_i$) denotes the span of their row vectors. For encoding vectors of a set of $k - 1$ nodes, let say $\mathbf{Q}_{s_1}, \ldots, \mathbf{Q}_{s_{k-1}}$ and h DR storage nodes, we prove there exists an encoding vector in $\mathbf{Q}_{s_i}$ that is not in span($\mathbf{Q}_{s_1}, \ldots, \mathbf{Q}_{s_{k-1}}, \mathbf{Q}_{h_1}, \ldots, \mathbf{Q}_{h_h}$). The proof is based on a contradiction argument.

If there is not an encoding vector in $\mathbf{Q}_{s_i}$, for $i \notin \{1, \ldots, k-1\}$, then the code cannot have the Extended MDS property. This is because, by code construction $\mathbf{Q}_{s_i}$ is not in the span of $Q_{h_1}, \ldots, Q_{h_h}$. Therefore, if there exists no encoding vector in $\mathbf{Q}_{s_i}$ that is out of span($\mathbf{Q}_{s_1}, \ldots, \mathbf{Q}_{s_{k-1}}, \mathbf{Q}_{h_1}, \ldots, \mathbf{Q}_{h_h}$), then $\mathbf{Q}_{s_i}$ is in the span of $\mathbf{Q}_{s_1}, \ldots, \mathbf{Q}_{s_{k-1}}$. This means $\mathbf{Q}_{s_1}, \ldots, \mathbf{Q}_{s_{k-1}}, \mathbf{Q}_{s_k}$ is not full rank. This is a contradiction to the Extended MDS property. ■

Theorem 4: Consider a distributed storage system storing a file by an $(n, k, d) - \text{EMSR}_h$ code and having Extended MDS property. For any finite field $GF(q)$, with

$$q \geq d_0 = (d + h - k + 1)k\left(\binom{n}{k} + 2\binom{n-1}{k-1} + 2n\right), \tag{37}$$

there exists a linear code for storing a file in the system such that it maintains the Extended MDS property before/after repair in which it requires the minimum repair-bandwidth.

Proof: The proof is by induction. We construct a code that has the Extended MDS property and maintains that property after repair. To this aim, we first find coefficients in $n(d + h - k + 1)$ encoding vectors of $Q_1, \ldots, Q_n$ (codes in complete storage nodes), and coefficients in h encoding vectors of $Q_{h_1}, \ldots, Q_{h_h}$, codes in DR storage nodes such that:

$$\prod_{\{s_1,\ldots,s_k\}\subseteq 1,\ldots,n} \det\left[Q_{s_1}, \ldots, Q_{s_k}\right] \times$$
$$\prod_{\{s_1\}\subseteq 1,\ldots,n} \det\left([\underbrace{Q_{s_1}, Q_{h_1} \ldots, Q_{h_h}}_{A_1}][A_1]^T\right) \times$$
$$\prod_{\{s_1,..,s_{k-1}\}\subseteq 1,\ldots,n} \det\left([\underbrace{Q_{s_1}, \ldots, Q_{s_{k-1}}, Q_{h_1}, \ldots, Q_{h_h}}_{A_2}][A_2]^T\right) \neq 0. \tag{38}$$

As $h\beta + (k - 1)\alpha < M$, then there exist some assignments for the coefficients which makes the left-hand side in (38) as a non-zero value. Thus we can use sparse-zero Lemma. The maximum degree of each coefficient is not greater than d_0. This implies that there exist some coefficients in $GF(q)$, for $q > d_0$, which make the left-hand side of (38) a non-zero value.

Now, suppose that we have a code with the Extended MDS property in the storage nodes. We show that the repair is possible with the minimum repair-bandwidth if $q > d_0$. Let $\mathbf{Q}'_1$ be the code on the new nodes which is generated by

$$\mathbf{Q}'_1 = [\mathbf{Q}_2\mathbf{b}_2, \ldots, \mathbf{Q}_{d-k+1}\mathbf{b}_{d-k+1}, \mathbf{Q}_{h_1}\mathbf{b}_{h_1}, \ldots, \mathbf{Q}_{h_h}\mathbf{b}_{h_h}]\mathbf{Z}, \tag{39}$$

where $\{\mathbf{b}_2, \ldots, \mathbf{b}_{d-k+1}\}$, $\{\mathbf{b}_{h_1}, \ldots, \mathbf{b}_{h_h}\}$, and $\mathbf{Z}$ are proper vectors with arrays (coefficients) that we can freely select such that the new code has the Extended MDS property. That is we select these free coefficients such that

$$\prod_{\{s_1,\ldots,s_{k-1}\}\subseteq 2,\ldots,n} \det\left[Q'_1, Q_{s_1}, \ldots, Q_{s_{k-1}}\right]$$
$$\times \det\left([\underbrace{Q'_1, Q_{h_1} \ldots, Q_{h_h}}_{A_3}][A_3]^T\right)$$
$$\times \prod_{\{s_1,\ldots,s_{k-2}\}\subseteq 2,\ldots,n}$$
$$\det\left([\underbrace{Q'_1, Q_{s_1}, \ldots, Q_{s_{k-2}}, Q_{h_1}, \ldots, Q_{h_h}}_{A_4}][A_4]^T\right) \neq 0. \tag{40}$$

Using Lemma 6 there will be some assignment for coefficients which make the term in the left-hand side of the above equation a non-zero value. Hence the left-hand side of the above equation is a non-zero polynomial. Therefore, we can use sparse-zero Lemma. Again, since the maximum degree of each free coefficient is

$$(d+h-k+1)k\left(\binom{n-1}{k-1}+2\binom{n-2}{k-2}+2\right), \tag{41}$$

which is smaller than d_0, then there exist some coefficients in $GF(q)$, for $q > d_0$, which make the left-hand side of (40) as a non-zero value. ■

D. Proof of Lemma 3

We prove for $h = 1$. For other values of h, the proof follows the same. For an $\text{EMBR}_{h=1}$ code, we have $\sum_{i=0}^{k-1}(d+1-i)\beta = M$. We now prove by a contradiction argument. Suppose $\alpha^{'} < k\beta$. Let us first consider a set of $\alpha^{'}$ that $(k-1)\beta \le \alpha^{'} < k\beta$. Let S denote the random variable representing the source file. Suppose S takes a value randomly and uniformly taken from a set $\mathbb{S} = \{1, \cdots, 2^M\}$. That is the the source file contains M bits of information, and we have $H(S) = M$, where $H(X)$[4] refers to the binary entropy of the random variable X. Next, let W_l denote the random variable representing the content of node l for $l \in [n]$. Assume W_{n+i} denotes the corresponding random variable for the content of the new node in i-th stage of repair. Since every k nodes have to reconstruct the original file, we have,

$$\begin{aligned}
M &= H(W_{n+1}, W_{n+2}, \cdots, W_{n+k}) && (42)\\
&= H(W_{n+1}) + H(W_{n+2}|W_{n+1}) + \cdots \\
&\quad + H(W_{n+k}|W_{n+1}, \cdots, W_{n+k-1}) && (43)\\
&= \underbrace{\overbrace{d\beta}^{\text{fr. CS s}} + \overbrace{\beta}^{\text{fr. DR}}}_{\text{stage 1}} + \underbrace{\overbrace{(d-1)\beta}^{\text{fr. CS s}} + \overbrace{\beta}^{\text{fr. DR}}}_{\text{stage 2}} + \cdots \\
&\quad + \underbrace{\overbrace{(d-(k-2))\beta}^{\text{fr. CS s}} + \overbrace{\beta}^{\text{fr. DR}}}_{\text{stage } k-1} + && (44)\\
&\quad + \underbrace{\overbrace{(d-(k-1))\beta}^{\text{fr. CS s}} + \overbrace{(\alpha'-(k-1)\beta)}^{\text{fr. DR}}}_{\text{stage } k} && (45)\\
&= (d+1)\beta + (d)\beta + \cdots \\
&\quad + (d+1-(k-1))\beta + (\alpha' - k\beta) && (46)\\
&= \sum_{i=0}^{k-1}(d+1-i)\beta + (\alpha' - k\beta), && (47)\\
&= M + (\alpha' - k\beta), && (48)\\
&< M. && (49)
\end{aligned}$$

which is a contradiction.

In the above equations, CSs stands for the complete storage nodes and DR stands for the DR storage node. In the proof, (42) follows by the chain rule of entropy, and (43) follows from the fact that each new node in stage i, conditioning on knowing information in previous stages, receives at most $(d+1-i)\beta$ information. Also (45) follows from the fact that at stage k, the DR storage node can send at most $\alpha' - (k-1)\beta$ (new) information to node $n+k$. Also (48) follows from the fact that $\sum_{i=0}^{k-1}(d+1-i)\beta\} = M$. Finally, (49) follows from $\alpha' - k\beta < 0$. The same argument can be followed for $\alpha^{'} < (k-1)\beta$. This finalizes the proof.

[4] $H(X) = \mathbb{E}\{-\log_2 P(X)$.

E. Proof of Lemma 4

We propose linear codes for achieving the bound $\sum_{i=0}^{k-1}(d+h-i)\beta = M$ where there are h DR storage nodes and d complete storage nodes in the repair process. Our proof is inspired by the argument adopted in [35]. For the code construction we first define a test, denoted as the *extended rank test*.

Definition 6 (Extended Rank Test): We say that a linear code $\mathbf{Q}_{s_1}, \ldots, \mathbf{Q}_{s_k}, \mathbf{Q}_{h_1}, \ldots, \mathbf{Q}_{h_h}$ in $GF(q)$ passes the extended rank test by a test vector $[h_1, \ldots, h_k, h_{h_1}, \ldots, h_{h_h}, h_{k+1}, \ldots, h_n]$, for $0 \le h_i \le \alpha$ and $0 \le h_{h_i} \le k\beta$, if matrix $[\mathbf{Q}_1\mathbf{E}_1, \ldots, \mathbf{Q}_k\mathbf{E}_k, \mathbf{Q}_{h_1}\mathbf{E}_{h_1}, \ldots, \mathbf{Q}_{h_h}\mathbf{E}_{h_h}, \mathbf{Q}_{k+1}\mathbf{E}_{k+1}\ldots\mathbf{Q}_n\mathbf{E}_n]$ has full rank M, where $\mathbf{Q}_i\mathbf{E}_i$ denotes the first h_i vectors of $\mathbf{Q}_i$.

We may also say that a code passes an extended rank test *weakly* by test vector h, if matrix $[\mathbf{Q}_1\mathbf{E}_1, \ldots, \mathbf{Q}_k\mathbf{E}_k, \mathbf{Q}_{h_1}\mathbf{E}_{h_1}, \ldots, \mathbf{Q}_{h_h}\mathbf{E}_{h_h}, \mathbf{Q}_{k+1}\mathbf{E}_{k+1}\ldots\mathbf{Q}_n\mathbf{E}_n]$ has full rank M, conditioning that we allow in each node to linearly combine the code vectors in $\mathbf{Q}_i$ and send h_i coded vectors. A test vector has $1 \times (n+h)$ dimension based on the following pattern:

$$\mathbf{h} = (\underbrace{h_1, \ldots, h_k}_{k}, \underbrace{h_{h_1}, h_{h_2}, \ldots, h_{h_h}}_{h}, \underbrace{h_{k+1}, \ldots, h_n}_{n-k}). \tag{50}$$

Note that the positions $k+1, k+2, \ldots, k+h$ in the extended rank test vector are dedicated for DR storage nodes. Now our objective is to design a code such that the code maintains the property of passing extended rank test over the following set of vectors:

$$\mathcal{H}_0 \triangleq \{\mathbf{h} | \mathbf{h} \text{ is a permutation of } \mathbf{h}^{(0)}\}, \tag{51}$$

where

$$\mathbf{h}^{(0)} = [\underbrace{\alpha, \ldots, \alpha}_{k}, \underbrace{0, \ldots, 0}_{n-k+h}]\}. \tag{52}$$

Here, the permutation of a $n+h$-dimensional vector $\mathbf{h}$ is defined in this way that the arrays whose positions in the vector belong to complete storage nodes can permute between themselves, and arrays whose positions in the vector belong to DR storage nodes can permute between themselves. For constructing $(n,k,d) - \text{EMBR}_h$ codes, we find a set $\mathcal{H} \supseteq \mathcal{H}_0$, such that if the code before any node failure (let say old code) passes the extended rank test by vectors in $\mathcal{H}$, the code after repair (let say new code) can also pass the test. Suppose such $\mathcal{H}$ is given, by the following theorem we states that it is always possible to construct a linear $(n,k,d) - \text{EMBR}_h$ code over sufficiently large finite field size. More formally, we state in the following theorem.

Theorem 5: Suppose a set of test vectors $\mathcal{H}$ with the following properties is given:

- $\mathcal{H} \supseteq \mathcal{H}_0$;
- $\forall \mathbf{h} \in \mathcal{H}$, where $\mathbf{h} = [h_1, \ldots, h_k, h_{h_1}, \ldots, h_{h_h}, h_{k+1}, \ldots, h_n]$, we have $0 \le h_i \le \alpha$ and $0 \le h_{h_i} \le k\beta$;
- $\forall \mathbf{h} \in \mathcal{H}$, $w(\mathbf{h}) \triangleq \sum_{i=1}^{n+h} h_i \ge M$;
- For any $h \in H$ and any permutation of $1, \ldots, n$, let say $s_1, \ldots, s_n$, there exists a vector $h' \in H$ that

$$h'_{s_1} = 0 \tag{53}$$

$$h_{s_1} \ge \sum_{i=2}^{d+h+1} \tag{54}$$

$$h'_{s_1} \in [h_{s_1}, h_{s_1} + \beta], \text{ for } i = 2, \ldots, d+h+1, \tag{55}$$

$$h'_{s_1} = h_{s_1} \text{ for } i = d+h+2, \ldots, n+h. \tag{56}$$

Then there exists a linear code for constructing an $(n, k, d) - \text{EMBR}_h$ code, provided that the finite field size q is greater than

$$q \ge \max\left\{\binom{n\alpha + h\,k\,\beta}{M}, 3|\mathcal{H}|M\right\}. \tag{57}$$

Proof: The proof is again based on induction. The proof is inspired by the proof of Theorem 3 in [35]. We remark that the set $\mathcal{H}$ for EMBR code is different from the analysis in [35]. We maintain the property that

$$\prod_{\mathbf{h}\in\mathcal{H}} \det\left([\mathbf{Q}_1\mathbf{E}_{h_1}, \ldots, \mathbf{Q}_n\mathbf{E}_{h_n}\,\mathbf{Q}_{h_{h_1}}\mathbf{E}_{h_{h_1}}, \ldots, \mathbf{Q}_{h_{h_h}}\mathbf{E}_{h_{h_h}}]\right) \ne 0. \tag{58}$$

Let us initialize the system by constructing a code which passes the extended rank test by the set $\mathcal{H}$. For that, we first construct a matrix $\mathbf{G}$, as a generator matrix of an $(n\alpha + h\,k\beta, M)$ MDS code. For this construction the finite field size $q > \binom{n\alpha+h\,k\,\beta}{M}$ is sufficient. From the MDS code property, we know that every set containing M or greater than M, row vectors of $\mathbf{G}$ has rank M. Now, if we assign the row vectors as the code vectors of complete storage nodes and h DR storage nodes, we see that the constructed code passes the extended rank test.

Now, suppose that node 1 fails and we want to generate a new node such that the new code also passes the extended rank test. Suppose the code on the new node is denoted as $\mathbf{Q}'_1$, then

$$\mathbf{Q}'_1 = [\mathbf{Q}_{s_2}\mathbf{B}_{s_2}, \ldots, \mathbf{Q}_{s_{d+1}}\mathbf{B}_{s_2}, \mathbf{Q}_{s_{h_1}}\mathbf{B}_{s_{h_1}}, \ldots, \mathbf{Q}_{s_{h_h}}\mathbf{B}_{s_{h_h}}]\mathbf{Z}. \tag{59}$$

Now, let the new node download $h'_{s_i} - h_{s_i}$ from the complete storage node i, for $i \in \{2, \ldots, d+1\}$ and also download $h'_{s_{h_i}} - h_{s_{h_i}}$ from the DR storage node i, for $i \in \{1, \ldots, h\}$. Since $\alpha \ge h_{s_1} \ge \sum_{i=2}^{d+1} h'_{s_i} - h_{s_i} + \sum_{i=1}^{h} h'_{s_{h_i}} - h_{s_{h_i}}$, the new node can store this data in its nodes such that resulted code passes the rank test by the vectors in set $\mathcal{H}$. Therefore, there exist an assignment such that

$$\det\left([\mathbf{Q}_1\mathbf{E}_{h_1}, \ldots, \mathbf{Q}_n\mathbf{E}_{h_n}, \mathbf{Q}_{h_{h_1}}\mathbf{E}_{h_{h_1}}, \ldots, \mathbf{Q}_{h_{h_h}}\mathbf{E}_{h_{h_h}}]\right) \ne 0 \tag{60}$$

Thus, the polynomial is non-zero polynomial and we can use sparse-zero Lemma. We can verify that the maximum degree of a coefficient is not greater than $3|\mathcal{H}|M$. Hence, there exist coefficients in $GF(q)$ that make all the products non-zero. ■

Now, we describe how to construct the set $\mathcal{H}$ given in Theorem 5. We construct for $d = n-1$. For other values of d, it needs further investigation. We construct $\mathcal{H}$ containing vectors $h^{(j)}$, for $j = 0, 1, \ldots, k$ and their permutations, where

$$h^{(j)} \triangleq (\underbrace{\alpha, \alpha, \ldots, \alpha}_{k-j}, \underbrace{j\beta, j\beta, \ldots, j\beta}_{n+h-k}, \underbrace{(j-1)\beta, \ldots, 2\beta, \beta, 0}_{j}). \tag{61}$$

Next theorem states that the above construction satisfy the requirements for the test set $\mathcal{H}$.

Theorem 6. The set of test vectors constructed by the above method satisfies the following conditions:

- $\mathcal{H} \supseteq \mathcal{H}_0$;
- $\forall \mathbf{h} \in \mathcal{H}$, where $\mathbf{h} = [h_1, \ldots, h_k, h_{h_1}, \ldots, h_{h_h}, h_{k+1}, \ldots, h_n]$, we have $0 \le h_i \le \alpha$ and $0 \le h_{h_i} \le k\beta$;
- $\forall \mathbf{h} \in \mathcal{H}$, $w(\mathbf{h}) \triangleq \sum_{i=1}^{n+h} h_i \ge M$;
- For $j = 0, \ldots, k$, $h^{(j)}$ consists of $(k-j)\alpha$'s followed by a $\beta-$gradually-decreasing vector ending in 0. Here, a vector $(x_1, x_2, \ldots, x_n)$ is $\beta-$gradually-decreasing vector if $x_1 \ge x_2 \ge \cdots \ge x_n$, and the difference between two consecutive elements is not larger than β.
- For any $h \in \mathcal{H}$ and any permutation of $1, \ldots, n$, let say $s_1, \ldots, s_n$, there exists a vector $h' \in \mathcal{H}$ that

$$h'_{s_1} = 0 \tag{62}$$

$$h_{s_1} \ge \sum_{i=2}^{d+h+1} \tag{63}$$

$$h'_{s_1} \in [h_{s_1}, h_{s_1} + \beta], \text{ for } i = 2, \ldots, d+h+1, \tag{64}$$

$$h'_{s_1} = h_{s_1} \text{ for } i = d+h+2, \ldots, n+h. \tag{65}$$

It is straightforward to verify that $\mathcal{H}$ satisfies Properties 1, 2, 3, and 4. We next provide the proof for Property 5. Note that for the EMBR code, we have test vectors $\mathbf{h}^j$'s and their permutations, for $j \in \{1, \ldots, k\}$. Also note that in our construction positions $k+1, \ldots, k+h+1$ are dedicated to DR storage nodes and thus $s_1 \notin \{k+1, \ldots, k+h+1\}$. For the proof, we study different cases, as follows.

Case 1: If $j = k$, or $s_1 > k - j$, to satisfy Property 5, we choose $\mathbf{h}'$ as the following vector:

$$\mathbf{h}' = (h_1, \ldots, h_{s_1-1}, h_n = 0, h_{s_1}, \ldots, h_{n-1}), \tag{66}$$

which is a permutation of $\mathbf{h}$ and hence is in $\mathcal{H}$.

Case 2: If $j \in 0, \ldots, k-1$, and $s_1 \le k - j$, then

$$\mathbf{h}^{(j)} = (\underbrace{\alpha, \ldots, \alpha}_{k-j}, \underbrace{j\beta, j\beta, \ldots, j\beta}_{n+h-k}, \underbrace{(j-1)\beta, \ldots, 2\beta, \beta, 0}_{j})). \tag{67}$$

To satisfy Property 5, we choose $\mathbf{h}'$ as the following vector:

$$\mathbf{h}' = (\underbrace{\alpha, \ldots, \alpha}_{s_1-1}, 0, \underbrace{\alpha, \ldots, \alpha}_{k-j-s_1}, \underbrace{(j+1)\beta, \ldots, (j+1)\beta}_{n+h-k}, \underbrace{j\beta, \ldots, \beta}_{j})), \tag{68}$$

which is a permutation of $\mathbf{h^{(j+1)}}$ and hence is in $\mathcal{H}$.

F. Proof of Proposition 1

For EMSR codes: For simplicity, and without loss of generality [11], assume $\beta = 1$. By Theorem 2, we know $\alpha' = \beta = 1$. We show that there is a linear combination of content of the k complete storage node that makes the content of the new DR storage nodes. Assume the global encoding vector of the fragment of the new node is ν_1. Assume the global encoding vectors of M/k fragments of the complete storage node i is $[\omega_{ij}]_{j=1}^{M/k}$, in which each column represents the global encoding vector of a fragment. Then, we show that there is a vector $[\gamma_1 \dots \gamma_M]$ that

$$[\gamma_1 \dots \gamma_M] \underbrace{\left([\omega_{1j}]_{j=1}^{M/k} \; [\omega_{2j}]_{j=1}^{M/k} \dots [\omega_{kj}]_{j=1}^{M/k} \right)^T}_{\mathbf{A}} = \nu_1. \quad (69)$$

Since every k complete storage nodes can reconstruct the original file, then matrix $\mathbf{A}$ is invertible. Therefore, there is a solution to the problem. After finding $[\gamma_1 \dots \gamma_M]$, each node sends a fragment which is a linear combination of its stored fragment. For example node 1 sends $[\gamma_1 \dots \gamma_{M/k}][\omega_{1j}]_{j=1}^{M/k\;T}$ and so on. Then, the new DR storage node generate its content by

$$\begin{aligned} \nu_1 = & \underbrace{[\gamma_1 \dots \gamma_{M/k}][\omega_{1j}]_{j=1}^{M/k\;T}}_{\text{from node 1}} + \dots \\ & + \underbrace{[\gamma_{M(k-1)/k+1} \dots \gamma_M][\omega_{kj}]_{j=1}^{M/k\;T}}_{\text{from node } k}. \end{aligned} \quad (70)$$

Therefore, $k\beta$ fragments are sufficient to generate the new DR storage node.

For EMBR codes: Again, assume $\beta = 1$. By Theorem 3, we know $\alpha' = k\beta = k$. Since the DR storage node could help regenerating a complete storage nodes by sending $\beta = 1$ fragments and since for EMBR codes all the information received in repair by a complete storage node is saved in the storage (i.e., $(d + h)\beta = \alpha$) then we can conclude that each DR storage node and a complete storage node have β fragments of information in common. Thus, in repairing a DR storage node, k complete storage nodes equally send $k\beta$ fragments to the new DR storage node. More formally, assume $\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_k$ are the encoding vectors of k stored fragments in the (failed) DR storage node. The DR storage node in repair of node $i_1, i_2, \dots, i_k$ could send respectively $\mathbf{a}_1\mathbf{x_1}, \mathbf{a}_2\mathbf{x_2}, \dots, \mathbf{a}_k\mathbf{x_k}$. Since the new node stores all the fragments that it receives, then $\mathbf{a}_1\mathbf{x_1}$ is available in node i_1, $\mathbf{a}_2\mathbf{x_2}$ is available in node i_2, and so on. In repairing a DR storage nodes, $i_1, i_2, \dots, i_k$ send $\mathbf{a}_1\mathbf{x_1}, \mathbf{a}_2\mathbf{x_2}, \dots, \mathbf{a}_k\mathbf{x_k}$. Since $\mathbf{a}_i$'s are independent vectors (if not, the data-collector could not reconstruct the original file by connecting to those k complete storage nodes) then the new DR storage node can recover lost fragments $\mathbf{x_1}, \mathbf{x_2}, \dots, \mathbf{x_k}$ by Gaussian elimination method. Thus, again $k\beta$ fragments are sufficient.

G. Probability of Successful Repair by Random Linear Codes

Suppose that a surviving node is transmitting a number of packets toward the new node. Each packet in our model is a random linear combination of $\beta\xi$ independent packets (having independent encoding vectors). Consider a case when the new node receives i packets from the surviving node. Let D_i be the random variable denoting the dimension of subspace spanned by all i vectors. Then, by random matrix arguments [30], for $i \geq d$, we have

$$\Pr(D_i = d) = \frac{\prod_{j=0}^{d-1}(q^i - q^j)\prod_{j=0}^{d-1}(q^{\beta\xi} - q^j)}{q^{i\beta\xi}\prod_{j=0}^{d-1}(q^d - q^j)}. \quad (71)$$

Hence, the probability of having $d = \beta\xi$ dimensions by receiving $i \geq \beta\xi$ vectors equals to

$$\Pr(D_i = \beta\xi) = \frac{\prod_{r=0}^{d-1}(q^i - q^r)}{q^{i\beta\xi}}. \quad (72)$$

That is the probability of having $\beta\xi$ independent packets by receiving $i \geq \beta\xi$ packets. Since packets are lost in the channel independently and with probability p, then the probability of receiving i packets from $t\xi$ transmitted packets has a binomial distribution and can be calculated as

$$\binom{t\,\xi}{i}(1-p)^i\, p^{(t\,\xi - i)}. \quad (73)$$

Combining this result with (72) gives us the probability of having $\beta\xi$ independent packets by transmitting $t\xi$ packets as

$$P_\beta = \sum_{i=\beta\,\xi}^{t\,\xi} \binom{t\,\xi}{i}(1-p)^i\, p^{(t\,\xi - i)} \frac{\prod_{l=0}^{\beta\,\xi - 1}(q^i - q^l)}{q^{\beta\,\xi\,i}} \text{ if } t \geq \beta. \quad (74)$$

Thus, the formula in Section VI-A is resulted.

REFERENCES

[1] A. G. Dimakis, P. B. 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, Sep. 2010.

[2] Y. Cui, H. Wang, X. Cheng, and B. Chen, "Wireless data center networking," *IEEE Wireless Commun.*, vol. 18, no. 6, pp. 46–53, Dec. 2011.

[3] M. Gerami, M. Xiao, and M. Skoglund, "Partial repair for wireless caching networks with broadcast channels," *IEEE Wireless Commun. Lett.*, vol. 4, no. 2, pp. 145–148, Apr. 2015.

[4] M. Gerami, M. Xiao, S. Salimi, and M. Skoglund, "Secure partial repair in wireless caching networks with broadcast channels," in *Proc. IEEE Commun. Netw. Security (CNS)*, 2015, pp. 353–360.

[5] J. Pääkkönen, C. Hollanti, and O. Tirkkonen, "Device-to-device data storage for mobile cellular systems," in *Proc. IEEE Globecom Workshops*, 2013, pp. 671–676.

[6] C. Hollanti, D. Karpuk, A. Barreal, and H.-F. Lu, "Space-time storage codes for wireless distributed storage systems," in *Proc. 4th Int. Conf. Wireless Commun. Veh. Technol. Inf. Theory Aerosp. Electron. Syst. (VITAE)*, 2014, pp. 1–5.

[7] M. Gerami and M. Xiao, "Repair for distributed storage systems with erasure channels," in *Proc. Int. Conf. Commun. (ICC)*, 2013, pp. 4058–4062.

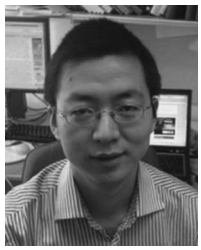
[8] Y. Wu and A. G. Dimakis, "Reducing repair traffic for erasure coding-based storage via interference alignment," in *Proc. IEEE Symp. Inf. Theory*, 2009, pp. 2276–2280.

[9] A.-M. Kermarrec, N. Le Scouarnec, and G. Straub, "Repairing multiple failures with coordinated and adaptive regenerating codes," in *Proc. Workshop Netw. Coding Theory Appl.*, 2011, pp. 1–6.

[10] K. Rashmi, N. B. Shah, K. Ramchandran, and P. Kumar, "Regenerating codes for errors and erasures in distributed storage," in *Proc. IEEE Symp. Inf. Theory*, 2012, pp. 1202–1206.

[11] N. B. Shah, K. Rashmi, P. V. Kumar, and K. Ramchandran, "Distributed storage codes with repair-by-transfer and nonachievability of interior points on the storage-bandwidth tradeoff," *IEEE Trans. Inf. Theory*, vol. 58, no. 3, pp. 1837–1852, Mar. 2012.
[12] M. Gerami, M. Xiao, and M. Skoglund, "Optimal-cost repair in multi-hop distributed storage systems," in *Proc. IEEE Symp. Inf. Theory*, 2011, pp. 1437–1441.
[13] P. Gill, N. Jain, and N. Nagappan, "Understanding network failures in data centers: Measurement, analysis, and implications," *ACM SIGCOMM Comput. Commun. Rev.*, vol. 41, no. 4, pp. 350–361, 2011.
[14] U. Madhow, *Fundamentals of Digital Communication*. Cambridge, U.K.: Cambridge Univ. Press, 2008.
[15] L. Ahlin, J. Zander, and S. Ben Slimane, "Principles of wireless communications," Lund, Sweden: Student literature, 2006.
[16] A. G. Dimakis, K. Ramchandran, Y. Wu, and C. Suh, "A survey on network codes for distributed storage," *Proc. IEEE*, vol. 99, no. 3, pp. 476–489, Mar. 2011.
[17] M. Sathiamoorthy et al., "XORing elephants: Novel erasure codes for big data," in *Proc. 39th Int. Conf. Very Large Data Bases (VLDB Endowment)*, 2013, pp. 325–336.
[18] Y. Wu, "Existence and construction of capacity-achieving network codes for distributed storage," *IEEE J. Sel. Areas Commun.*, vol. 28, no. 2, pp. 277–288, Feb. 2010.
[19] K. Rashmi, N. B. Shah, P. V. Kumar, and K. Ramchandran, "Explicit construction of optimal exact regenerating codes for distributed storage," in *Proc. Allerton Conf. Commun. Control Comput.*, 2009, pp. 1243–1249.
[20] S. Jain, K. Fall, and R. Patra, *Routing in a Delay Tolerant Network*, vol. 34. New York, NY, USA: ACM, 2004.
[21] M. Gerami, M. Xiao, C. Fischione, and M. Skoglund, "Decentralized minimum-cost repair for distributed storage systems," in *Proc. Int. Conf. Commun. (ICC)*, 2013, pp. 1910–1914.
[22] D. S. Lun, M. Médard, R. Koetter, and M. Effros, "On coding for reliable communication over packet networks," *Phys. Commun.*, vol. 1, no. 1, pp. 3–20, 2008.
[23] A. F. Dana, R. Gowaikar, R. Palanki, B. Hassibi, and M. Effros, "Capacity of wireless erasure networks," *IEEE Trans. Inf. Theory*, vol. 52, no. 3, pp. 789–804, Mar. 2006.
[24] S. Jain, M. Demmer, R. Patra, and K. Fall, "Using redundancy to cope with failures in a delay tolerant network," *ACM SIGCOMM Comput. Commun. Rev.*, vol. 35, no. 4, pp. 109–120, 2005.
[25] D. Leong, A. G. Dimakis, and T. Ho, "Distributed storage allocation problems," in *Proc. Workshop Netw. Coding Theory Appl.*, 2009, pp. 86–91.
[26] T. M. Cover and J. A. Thomas, *Elements of Information Theory*. Hoboken, NJ, USA: Wiley, 2012.
[27] C. Suh and K. Ramchandran, "Exact-repair MDS codes for distributed storage using interference alignment," in *Proc. IEEE Symp. Inf. Theory*, 2010, pp. 161–165.
[28] 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.
[29] N. B. Shah, K. Rashmi, and P. V. Kumar, "A flexible class of regenerating codes for distributed storage," in *Proc. IEEE Int. Symp. Inf. Theory (ISIT)*, 2010, pp. 1943–1947.
[30] S. Acedanski, S. Deb, M. Médard, and R. Koetter, "How good is random linear coding based distributed networked storage," in *Proc. Workshop Netw. Coding Theory Appl.*, 2005, pp. 1–6.
[31] M. Martalò, M. Picone, M. Amoretti, G. Ferrari, and R. Raheli, "Randomized network coding in distributed storage systems with layered overlay," in *Proc. Inf. Theory Appl. Workshop (ITA)*, 2011, pp. 1–7.
[32] D. P. Bertsekas, *Network Optimization: Continuous and Discrete Methods*, vol. 8. Belmont, MA, USA: Athena Scientific, 1998.
[33] D. S. Lun, M. Médard, R. Koetter, and M. Effros, "Further results on coding for reliable communication over packet networks," in *Proc. Int. Symp. Inf. Theory (ISIT)*, 2005, pp. 1848–1852.
[34] R. W. Yeung, *Information Theory and Network Coding*. New York, NY, USA: Springer, 2008.
[35] Y. Wu, A. G. Dimakis, and K. Ramchandran, "Deterministic regenerating codes for distributed storage," in *Proc. Allerton Conf. Control Comput. Commun.*, 2007, pp. 1–5.

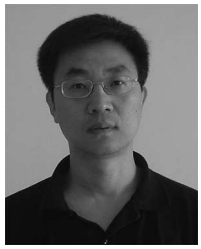
Majid Gerami (S'13) was born in Minab, Hormozgan, Iran. He received the Master of Science degree in electrical engineering from Sharif University of Technology, Tehran, Iran. He is currently pursuing toward his Ph.D. degree at the School of Electrical Engineering, KTH (The Royal Institute of Technology), Stockholm, Sweden. He worked for some years as a Researcher and Network Designer with the Electronic Research Center (ERC), Sharif University, Iran Telecommunication Research Center (ITRC), Mobile Communication Center of Iran (MCCI), and Ericsson. He was a summer internship researcher with ABB Corporate Research Center, Västerås, Sweden, in the summer of 2013. Since December 2015, he has been a Visiting Researcher at Bell Laboratories, Murray Hill, NJ, USA. His research interests include information theory, communication theory, network coding, network optimization, distributed storage networks, and wireless caching networks.



Ming Xiao (S'02–M'07–SM'12) received bachelor's and master's degrees in engineering from the University of Electronic Science and Technology of China, Chengdu, China, in 1997 and 2002, respectively, and the Ph.D degree from Chalmers University of Technology, Gothenburg, Sweden, in November 2007. From 1997 to 1999, he worked as a Network and Software Engineer with China Telecom. From 2000 to 2002, he also held an administrative position with SiChuan Communications. Since November 2007, he has been with the Department of Communication Theory, School of Electrical Engineering, Royal Institute of Technology, Stockholm, Sweden, where he is currently an Associate Professor of communications theory. Since 2012, he has been an Associate Editor for the IEEE TRANSACTIONS ON COMMUNICATIONS, the IEEE COMMUNICATIONS LETTERS (Senior Editor since January 2015), and the IEEE WIRELESS COMMUNICATIONS LETTERS. He received the Hans Werthen Grant from the Royal Swedish Academy of Engineering Science (IVA) in March 2006, and Ericsson Research Funding from Ericsson in 2010. He was the recipient of best paper awards in IC-WCSP (International Conference on Wireless Communications and Signal Processing) in 2010 and IEEE ICCCN (International Conference on Computer Communication Networks) in 2011, and the Chinese Government Award for Outstanding Self-Financed Students Studying Aborad in March 2007.

Jun Li (M'09) received the Ph.D. degree in electronic engineering from Shanghai Jiao Tong University, Shanghai, China, in 2009. From January 2009 to June 2009, he worked with the Department of Research and Innovation, Alcatel Lucent Shanghai Bell as a Research Scientist. From June 2009 to April 2012, he was a Postdoctoral Fellow with the School of Electrical Engineering and Telecommunications, University of New South Wales, Sydney, N.S.W., Australia. From April 2012 to June 2015, he was a Research Fellow with the School of Electrical Engineering, The University of Sydney, Sydney, N.S.W., Australia. Since June 2015, he has been a Professor with the School of Electronic and Optical Engineering, Nanjing University of Science and Technology, Nanjing, China. His research interests include network information theory, channel coding theory, wireless network coding, and cooperative communications.

Carlo Fischione (M'05) received the Laurea degree in electronic engineering (*Laurea*, *summa cum laude*, 5/5 years) and the Ph.D. degree in electrical and information engineering (3/3 years) from the University of L'Aquila, L'Aquila, Italy, in 2001 and 2005, respectively. He is currently a tenured Associate Professor with KTH Royal Institute of Technology, Electrical Engineering and ACCESS Linnaeus Center, Stockholm, Sweden. He has held research positions with Massachusetts Institute of Technology, Cambridge, MA, USA (2015, Visiting Professor), Harvard University, Cambridge, MA, USA (2015, Associate), University of California at Berkeley, Berkeley, CA, USA (2004–2005, Visiting Scholar, and 2007–2008, Research Associate), and Royal Institute of Technology, Stockholm, Sweden (2005–2007, Research Associate). He has coauthored over 100 publications, including a book, book chapters, international journals and conferences, and international patents. His research interests include optimization with applications to wireless sensor networks, networked control systems, wireless networks, security, and privacy. He is an Associate Editor of *Automatica* (Elsevier), has chaired or served as a technical member of program committees of several international conferences, and is serving as a referee for technical journals. Meanwhile, he also has offered his advice as a Consultant to numerous technology companies such as Berkeley Wireless Sensor Network Lab, Ericsson Research, Synopsys, and United Technology Research Center. He is a co-funder and CTO of the sensor networks start-up companies Aukoti (Internet of Things indoor navigation) and MIND (ancient and modern musical instruments networked). He is an Ordinary Member of DASP (the academy of history Deputazione Abruzzese di Storia Patria). He was the recipient or corecipient of a number of awards including the Best Paper Award from the IEEE Transactions on Industrial Informatics (2007), the Best Paper Awards at the IEEE International Conference on Mobile Ad-hoc and Sensor System 05 and 09 (IEEE MASS 2005 and IEEE MASS 2009), the Best Paper Award of the IEEE Sweden VT-COM-IT Chapter (2014), the Best Business Idea Awards from VentureCup East Sweden (2010) and from Stockholm Innovation and Growth (STING) Life Science in Sweden (2014), the Ferdinando Filauro Award from the University of L'Aquila, Italy (2003), the Higher Education Award from Abruzzo Region Government, Italy (2004), the Junior Research Award from Swedish Research Council (2007), and the Silver Ear of Wheat Award in history from the Municipality of Tornimparte, Italy (2012).



Zihuai Lin (S'98–M'99–SM'11) received the Ph.D. degree in electrical engineering from Chalmers University of Technology, Gothenburg, Sweden, in 2006. Prior to this, he has held positions with Ericsson Research, Stockholm, Sweden. Following Ph.D. graduation, he worked as a Research Associate Professor with Aalborg University, Aalborg, Denmark, and currently with the School of Electrical and Information Engineering, The University of Sydney, Sydney, N.S.W., Australia. His research interests include source/channel/network coding, coded modulation, MIMO, OFDMA, SC-FDMA, radio resource management, cooperative communications, small-cell networks, and 5G cellular systems.