Storage-Repair Bandwidth Trade-off for Wireless Caching with Partial Failure and Broadcast Repair

Nitish Mital*, Katina Kralevska†, Cong Ling*, and Deniz Gündüz*
*Department of Electrical Electronics Engineering, Imperial College London
†Dep. of Information Security and Communication Technology, NTNU, Norwegian University of Science and Technology
Email: {n.mital,d.gunduz,c.ling}@imperial.ac.uk, katinak@ntnu.no

***Abstract*—Repair of multiple partially failed cache nodes is studied in a distributed wireless content caching system, where r out of a total of n cache nodes lose part of their cached data. Broadcast repair of failed cache contents at the network edge is studied; that is, the surviving cache nodes transmit broadcast messages to the failed ones, which are then used, together with the surviving data in their local cache memories, to recover the lost content. The trade-off between the storage capacity and the repair bandwidth is derived. It is shown that utilizing the broadcast nature of the wireless medium and the surviving cache contents at partially failed nodes significantly reduces the required repair bandwidth per node.**

I. Introduction

Caching popular contents closer to end-users, particularly in the available storage space at the wireless network edge, is attracting a lot of attention in the recent years as a promising method to alleviate the increasing burden on the backhaul links of wireless access points, e.g., small cell base stations, and to improve the quality of service, particularly by reducing the latency [1], [2], or energy consumption [3]. The literature on distributed coded caching systems focuses mostly on the code design or resource allocation for efficient storage of popular contents, assuming reliable cache nodes. However, storage devices are often unreliable and prone to failures; thus, efficient repair techniques that guarantee continuous data availability are essential for a successful implementation of distributed caching and content delivery techniques in practice.

Maximum distance separable (MDS) codes are typically used for distributed caching of contents at multiple access points [1], [2], [4]. MDS codes provide flexibility for storage so that users with different connectivity or mobility patterns can download a file from only a subset of the access points. In particular, an (n, k) MDS code encodes a file of size M bits by splitting it into k equal-size fragments, and encoding them into n fragments which are stored at n cache nodes. The original file can be recovered by accessing any k out of n fragments from k distinct access points. When some nodes partially or fully fail, their cache contents need to be regenerated to be able to continue serving users. An important objective of edge caching in wireless networks is to reduce the backhaul link loads; therefore, we will consider *cache recovery at the edge*; that is, rather than updating the failed cache contents from a central server through backhaul links, the failed cache contents are regenerated with the help of surviving cache nodes. The total amount of data transferred from the surviving nodes to repair the failed nodes is called the *repair bandwidth*. Traditional MDS codes have high storage efficiency, but their repair bandwidth is large [5]. The data of one node is repaired by accessing and transferring data from k nodes, i.e., by recovering the whole content library.

Dimakis et al. showed in [5] that there is a fundamental trade-off between the storage and repair bandwidth by mapping the repair problem in a distributed storage system to a multicasting problem over an information flow graph. The analysis focuses on a single node repair; that is, losing one of the nodes triggers the repair process. Regenerating codes achieve any point on the optimal trade-off curve, while minimum-storage regenerating (MSR) codes and minimum-bandwidth regenerating (MBR) codes operate on the two extremes of this trade-off curve.

It was observed in [6] that multiple node repair; that is, the repair process starts only after r nodes fail, is more efficient in terms of the repair bandwidth per node, compared to repairing each node as it fails. In [7] and [8], the authors introduce cooperative regenerating codes, which repair multiple failures cooperatively by allowing each of the r nodes being repaired to collect data from the $n - r$ non-failed nodes, and then to cooperate with the other $r - 1$ nodes being repaired. Instead, similarly to [9], we will consider broadcast repair; that is, transmissions from each node are received in an error-free manner by all the other nodes. The storage-repair bandwidth trade-off for the broadcast repair of multiple fully failed nodes is investigated in [10], while [11] considers an equivalent centralized repair model.

In this paper, we consider the *partial repair* problem, studied in [9], in which the repair process starts after multiple node failures, but each of the failing cache nodes loses only a part of its contents, and the remainder of the cache contents should be used along with the transmissions from the surviving nodes to make sure that the repaired nodes can still continue to serve user requests (i.e., *functional repair*). We will see that partial repair further reduces the repair bandwidth.

In [9], the authors derive a lower bound on the number of packet transmissions at the MSR point for error-free partial broadcast repair, and provide an explicit code construction for

This work was supported in part by the European Union's H2020 research and innovation programme under the Marie Sklodowska-Curie Action SCAVENGE (grant agreement no. 675891), and by the European Research Council (ERC) Starting Grant BEACON (grant agreement no. 677854).

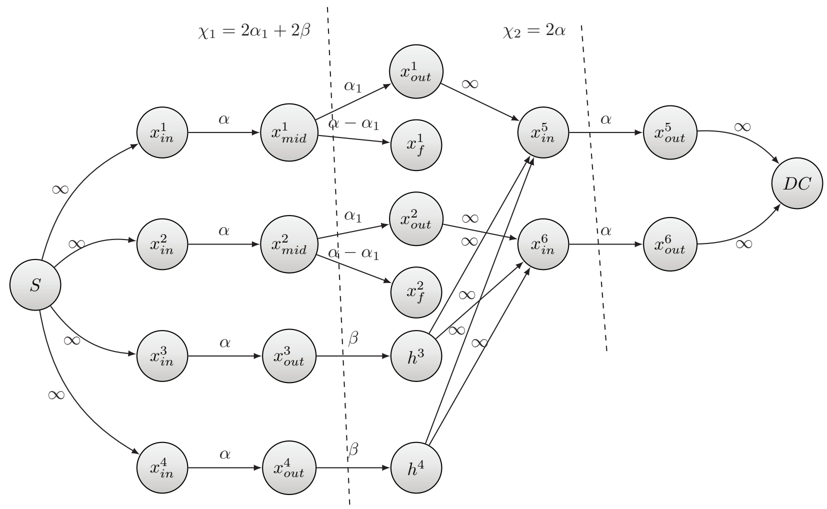

Fig. 1: Information flow graph $\mathcal{G}$ with $n = 4, k = 2, r = 2$, one repair round and two cuts χ_1 and χ_2.

a special case. The information flow graph construction in [9] does not capture the relation between storage capacity per node and the repair bandwidth, thus focusing only on the MSR point on the storage-repair bandwidth trade-off curve. In this paper, we study the entire optimal trade-off curve. Through specific examples we show that the MSR point from [9], where each node stores M/k bits, is not feasible with a finite repair bandwidth for all cases.

In [12], the authors derive the storage-repair bandwidth trade-off considering partial node failure in a wired distributed storage system. They show that the repair bandwidth can be reduced by two-layer coding; however, this comes at the expense of increased storage redundancy. Reference [13] investigates the storage-repair bandwidth trade-off for clustered storage networks, where multiple nodes within a cluster fail. This is close to the partial failure model that is considered here, since each cluster could model multiple memory units within a node, and multiple memory units failing is equivalent to partial failure in a node. However, we consider partial failures at multiple nodes, i.e., multiple clusters, and broadcast transmissions from the surviving nodes.

Notations. For two integers $i < j$, we denote the set $\{i, i+1, \ldots, j\}$ by $[i : j]$, while the set $[1 : j]$ is denoted in short by $[j]$. Sets are denoted with the calligraphic font, while vectors and matrices are denoted with bold letters.

II. System Model

Consider a wireless caching network with n nodes, denoted by $\mathcal{N} \triangleq \{1, \ldots, n\}$, each equipped with a storage capacity of α bits. The nodes collaboratively store a file of size M bits; such that, a user connected to any k of these n nodes can recover the file. Here the file represents a popular content cached by n access points at the wireless network edge, which a mobile user should be able to recover after connecting to an arbitrary subset of the k access points. The nodes, corresponding to the cache-equipped Access Points (APs), are assumed to be fully connected wirelessly; that is, broadcast transmissions from any node are assumed to be correctly received by all the other nodes in the network. We also assume orthogonal channels for data transmission; that is, no interference among these broadcast transmissions.

We consider a scenario in which a portion of the stored bits in some storage nodes is subject to being lost. We refer to these nodes as the *faulty nodes*, and to the surviving nodes that do not experience any losses as the *complete nodes*. We assume that the repair occurs in rounds, and a repair round is initiated when r nodes experience partial failure; i.e., each of these r nodes loses $\alpha - \alpha_1$ of it s stored bits, where $\alpha_1 \triangleq \rho\alpha$, $\rho \in [0, 1]$. Thus, a single repair round repairs r partially faulty nodes. There is no loss in a repair round, during which the lost bits in the faulty nodes are repaired with the help of transmitted bits from the complete nodes and the remaining bits that have not been lost in each of the faulty nodes. In general, the repair is *functional*, i.e., the repaired portion may not be the same as the original, but it satisfies the same property that any k nodes are sufficient to reconstruct the whole file.

A. Information flow graph

The repair dynamics of the network can be represented by an information flow graph that evolves over time. See Fig. 1 and Fig. 2 for illustrations. It is a directed acyclic graph consisting of six types of nodes: a single source node S, storage nodes $x^i_{in}, x^i_{mid}, x^i_{out}$, helpers h_i, and a data collector node DC. Each complete storage node $x^i, i \in [n]$, is represented by two vertices: an input vertex x^i_{in} and an output vertex x^i_{out}, which are connected by a directed edge $x^i_{in} \rightarrow x^i_{out}$ with capacity α. A faulty node is represented by four vertices: an input vertex x^i_{in}, an intermediate vertex x^i_{mid} that is connected to x^i_{in} by a directed edge $x^i_{in} \rightarrow x^i_{mid}$ of capacity α, an output vertex x^i_{out} that is connected to x^i_{mid} by a directed edge $x^i_{mid} \rightarrow x^i_{out}$ of capacity α_1, and a failed vertex x^i_f that is connected to x^i_{mid} by a directed edge $x^i_{mid} \rightarrow x^i_f$ of capacity $\alpha - \alpha_1$. The failed vertex represents the corrupted portion of the data in the storage node. The justification for such a representation is that any node can be represented arbitrarily as the sum of two virtual nodes of capacities α_1 and $\alpha - \alpha_1$, without loss of generality. Thus, there are $2n$ virtual storage nodes at each round, n with storage capacity α_1 and n with capacity $\alpha - \alpha_1$. The way in which the division into the virtual nodes is done does not affect the flow arguments.

Each vertex in the graph at any given time has two modes, active or inactive, depending on its availability. Initially, the source node S is active and it transmits data to n storage nodes such that the DC can retrieve the file from any k nodes. This is modeled by adding edges of capacity ∞ from S to the input vertices of all the storage nodes, $S \to x^i_{in}, i \in [n]$. . From this point onwards, the source node becomes inactive, and the n storage nodes become active.

When r nodes experience partial failure of $\alpha - \alpha_1$ bits each, in the s-th round, the repair process is triggered and r newcomers join the system. Note that a newcomer represents the corresponding node being repaired. A newcomer x^i where $i = sn + j, j \in [n]$, represents the node x^j after the s-th round. The lost data is regenerated at the newcomers by receiving functions of the stored data from the $n - r$ complete storage nodes through the helper nodes. The $n - r$ complete storage nodes are connected to the corresponding helper nodes with a directed edge $x^i_{out} \to h^i$ of capacity β, which denotes the number of bits broadcasted by x^i. Each helper node h^i is connected with infinite capacity links to all the newcomers. This represents the broadcast nature of the transmission medium.

Definition 1. *The repair bandwidth $\gamma = (n-r)\beta$ is defined as the total number of bits the complete storage nodes broadcast in a repair round.*

We model a newcomer with two vertices x^i_{in} and x^i_{out}, and a directed edge $x^i_{in} \to x^i_{out}$ with capacity α. The newcomer $x^i, i = sn + j$, uses the α_1 bits from the corresponding node being repaired. This is captured in the flow graph by edges $x^p_{mid} \to x^p_{out}$, where $p = (s-1)n+j$, of capacity α_1, followed by edges $x^p_{out} \to x^i_{in}$ with infinite capacity between the output vertices of the nodes being repaired and the input vertices of corresponding newcomers.

A data collector node DC, corresponds to a request to reconstruct the file. Data collectors connect to any subset of k active nodes and retrieve all the stored data in these nodes, represented with edges with infinite capacity.

A *cut* in the information flow graph is a subset of edges such that there is no path from the source node S to the data collector DC that does not go through any of the edges in the cut. We define the *capacity* of a cut as the sum of its edge capacities, and the *min-cut* of a graph as the cut with the minimum capacity among all possible data collectors and all possible information flow graphs for different failure patterns. The following proposition from [5] characterizes the fundamental performance limits of a distributed storage system with node failures using its information flow graph representation, modeling it as a multicast network coding problem and by finding the min-cut as the capacity of the network which ensures that there is enough information flow from the source to any data collector to reconstruct the file. However, [5] only presents the result for bounded number of failure/repair rounds (finite information flow graph). A later paper by the same authors [14] extends the proof for unbounded number of failure/repair rounds, represented by an infinite information flow graph.

Proposition 1. *[5] Consider any given finite information flow graph $\mathcal{G}$, with a finite set of data collectors. If the min-cut separating the source from each data collector is larger than or equal to the file size M, then there exists a linear network code defined over a sufficiently large finite field $\mathbb{F}$ (whose size depends on the graph) such that all data collectors can recover the original file. Further, randomized network coding guarantees that all collectors can recover the file with probability that can be driven arbitrarily close to 1 by increasing the field size.*

The existence of such linear codes can be shown by proving that the MDS property of the code is preserved even after an unbounded number of repairs (demonstrated in [14]). This can be shown by induction and by exploiting the linearity of the code and the repair process.

Following Proposition 1, using the information flow graph construction in this paper, we will find the minimum cut of the information flow graph over all possible failure combinations. We enumerate cuts as $\chi_1, \chi_2, \dots$ (see Fig. 1 and 2). In Section 3, we demonstrate how to find the min-cut for a specific example.

III. Storage-Bandwidth Trade-off for Partial Repair

Consider the scenario illustrated in Fig. 1, where $n = 4, k = 2$ and $r = 2$. The capacity of cut χ_1 is $2\alpha_1 + 2\beta$, while the capacity of cut χ_2 is 2α. Then the min-cut is $\min\{2\alpha_1 + 2\beta, 2\alpha\}$. From Proposition 1, to ensure that the file can be reconstructed by the data collector, $\min\{2\alpha_1 + 2\beta, 2\alpha\} \geq M$.

Next we consider a scenario represented in Fig. 2 with $n = 4, k = 3$ and $r = 2$, where two repair rounds are required to determine the min-cut. The number of repair rounds is determined by ensuring that each of the k nodes serving the DC go through at least one repair round, so that all the different capacity edges occur at least once in the path, from S to DC, through each node. Therefore, the minimum number of repair rounds required is $\lceil k/r \rceil$. It is possible to show that the min-cut is then given by $\min\{3\alpha_1 + 2\beta, 2\alpha + \alpha_1\}$, and the sufficient condition for the reconstruction of the file by the DC is $\min\{3\alpha_1 + 2\beta, 2\alpha + \alpha_1\} \geq M$ [15].

For each set of parameters $(n, k, \gamma, \alpha, r, \rho)$, there is a family of information flow graphs, each of which corresponds to a particular evolution of node failures/repairs. We denote this family of directed acyclic graphs by $\mathcal{G}(n, k, \gamma, \alpha, r, \rho)$. An

Note that adding an edge with capacity ∞ means that all the information in the node sending the data is available in the input vertices of the nodes receiving the data.

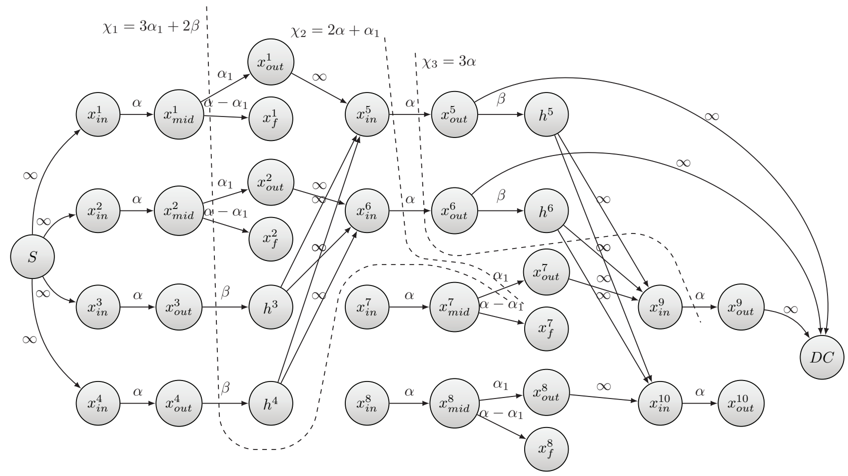

Fig. 2: Information flow graph $\mathcal{G}$ with $n = 4, k = 3, r = 2$, two repair rounds and three cuts χ_1, χ_2 and χ_3.

$(n, k, \gamma, \alpha, r, \rho)$ tuple is feasible, if a code with storage α and repair bandwidth γ exists.

Theorem 1. *For any $\alpha \geq \alpha^*(n, k, \gamma, r, \rho)$, the points $(n, k, \gamma, \alpha, r, \rho)$ are feasible, and linear network codes suffice to achieve them. It is information theoretically impossible to achieve points with $\alpha < \alpha^*(n, k, \gamma, r, \rho)$. If r divides k, the threshold function $\alpha^*(n, k, \gamma, r, \rho)$ is given by:*

$$\alpha^*(n, k, \gamma, r, \rho) = \begin{cases} \frac{M}{k} & \gamma \in [f(0), \infty) \\ \frac{M - g(i)\gamma}{k - ir(1-\rho)} & \gamma \in [f(i), f(i-1)] \end{cases} \tag{1}$$

where, for $i = 1, 2, \ldots, \frac{k}{r} - 1$,

$$f(i) \triangleq \frac{2M(1-\rho)(n-r)}{(2k - r(i+1)(1-\rho))i + \frac{2k}{r}(n-k)}, \tag{2}$$

$$g(i) \triangleq \frac{1}{2}\left(2n - 2k - r + ir\right)\frac{ir}{n-r}. \tag{3}$$

Proof. Refer to [15]. ∎

Corollary 1. *The minimum storage point is achieved by the pair $(\alpha_{MSR}, \gamma_{MSR}) = \left(\frac{M}{k}, \frac{Mr(n-r)(1-\rho)}{k(n-k)}\right)$.*

Corollary 2. *The minimum repair bandwidth point is achieved by the pair $(\alpha_{MBR}, \gamma^*_{MBR}) = \left(\frac{M - g'\gamma^*_{MBR}}{k\rho + r(1-\rho)}, \frac{2Mr(n-r)(1-\rho)}{k(2n - k(1-\rho) - r(1+\rho))}\right)$ where $g' = \frac{1}{2}\frac{(k-r)(2n-k-2r)}{n-r}$.*

Minimum-storage regenerating (MSR) and minimum-bandwidth regenerating (MBR) codes attain the points in Corollary 1 and Corollary 2, respectively.

Remark 1. *For $\rho = 0$ and $r = 1$, i.e., complete failure of exactly one node, the model is equivalent to that in [5], and the trade-off curve from Theorem 1 coincides with the trade-off curve in [5]. Similarly, for $\rho = 0$ and $r > 1$, i.e., multiple complete failures, the trade-off curve from Theorem 1 coincides with the one in [10].*

Theorem 2. *In the same context as in Theorem 1, if r does not divide k, let $p \triangleq \lfloor k/r \rfloor$ and $k_0 \triangleq pr$. Assume $\frac{n-k_0+(z-1)r}{r} \leq \frac{n-k_0-r}{k-k_0} \leq \frac{n-k_0+zr}{r}$, for some $z \in [p-2]$, or $0 \leq \frac{n-k_0-r}{k-k_0} \leq \frac{n-k_0}{r}$ for $z = 0$. Also define $k' \triangleq k\rho + (1-\rho)k_0$. Then the threshold function $\alpha^*(n, k, \gamma, r, \rho)$ is given by:*

$$\alpha^* = \begin{cases} \frac{M - g(i)\gamma}{k - ir(1-\rho)} & \gamma \in [f(i), f(i-1)], \\ & 0 \leq i \leq z-1 \\ \frac{M - g(z)\gamma}{k - zr(1-\rho)} & \gamma \in [f', f(z-1)] \\ \frac{M - [g(z) + \frac{n-k_0-r}{n-r}]\gamma}{k' - zr(1-\rho)} & \gamma \in [f(z), f'] \\ \frac{M - [g(i) + \frac{n-k_0-r}{n-r}]\gamma}{k' - ir(1-\rho)} & i \geq z+1, \\ & \gamma \in [f(i), f(i-1)] \end{cases} \tag{4}$$

where $i = 0, 1, \ldots, \frac{k}{r} - 1$, and f, g and f' are defined as

$$f(i) \triangleq \begin{cases} \infty & i = -1 \\ \frac{2M(1-\rho)(n-r)}{(2k - r(i+1)(1-\rho))i + \frac{2k}{r}(n-k)} & i \leq z-1 \\ \frac{2M(1-\rho)(n-r)}{(2k' - r(i+1)(1-\rho))i + \frac{2k'(n-k_0)}{r} + n - k_0 - r} & \\ & i \geq z \end{cases} \tag{5}$$

$$g(i) \triangleq \frac{1}{2}\left(2n - 2k_0 - r + ir\right)\frac{ir}{n-r} \tag{6}$$

$$f' \triangleq \begin{cases} \frac{2M(n-r)}{\left[\frac{2(n-k_0)(k-k_0-r)+2r^2}{k-k_0} + (z-1)r\right]z + \frac{2k(n-k_0-r)}{(k-k_0)(1-\rho)}} & \\ & \textit{if } z > 0 \\ \frac{M(k-k_0)(n-r)(1-\rho)}{k'(n-k_0-r)} & \textit{if } z = 0. \end{cases} \tag{7}$$

Proof. The proof follows the same principle as that of Theorem 1 [15], except that now there is a small offset in the thresholds due to the non-divisibility of k by r. ∎

IV. Special Cases

Example 1. Consider a network with the following parameters: $n = 4$, $k = 3$, $r = 2$ and $\alpha_1 = \frac{\alpha}{2}$ (Fig. 2).

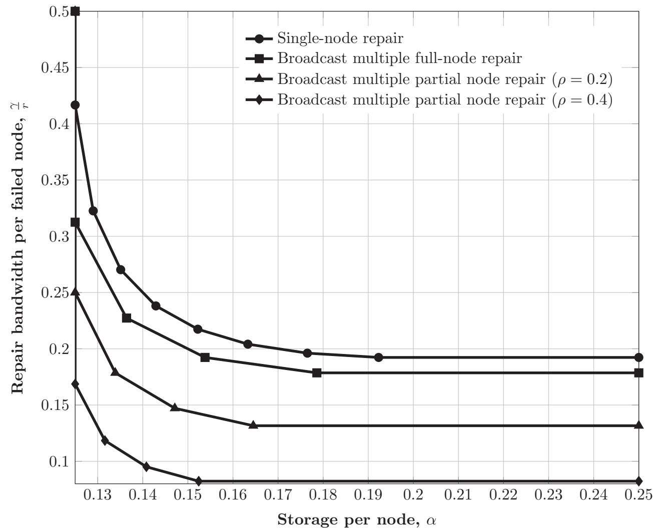

Fig. 3: Trade-off curve between the repair bandwidth and storage, $M = 1, k = 8$ and 10 helper nodes. For single node failure $r = 1$ and for multiple node failures $r = 2$.

According to Eq. (6) in [9], the MSR point is achieved by the pair $(\alpha, \gamma) = (\frac{M}{3}, M)$. However, by constructing the information flow graph as shown in Fig. 2 and finding the minimum cutset, we find that the storage point of $\frac{M}{3}$ is not feasible, and instead the storage point $\alpha = \frac{2M}{5}$ is feasible, and the corresponding repair bandwidth point is $\gamma = \frac{2M}{5}$. This is verified by Theorem 2, where $z = 0$ for this example, and therefore $f' = f(z-1) = \infty$. Therefore $\alpha = \frac{M}{k}$ is not possible. Interestingly, when we substitute the value of $\alpha = \frac{2M}{5}$ into Eq. (6) of [9], the repair bandwidth in [9] coincides with ours.

Example 2. Consider the parameters: $n = 4, k = 2, r = 1, \alpha_1 = \frac{\alpha}{2}$. According to Eq. (6) in [9], the MSR point is achieved by the pair $(\alpha, \gamma) = (\frac{M}{2}, M)$. Our approach achieves $\gamma = \frac{3M}{8}$, which is achievable using linear network codes in $GF(q)$, assuming q is large enough.

Example 3. Consider a network with $n = 4$, $k = 2$, $r = 2$ and $\alpha_1 = \frac{\alpha}{2}$. Reference [9] proposed a coding scheme for this example. It can be verified from Corollary 2 that this scheme achieves the minimum repair bandwidth, i.e., $(\alpha, \gamma) = (\frac{M}{2}, \frac{M}{2})$.

V. Results and discussion

In Fig. 3, we plot the storage-repair bandwidth (per failed node) trade-off for single-node repair with full node failure [5], broadcast repair of multiple full node failures [10], and the broadcast repair of multiple partial node failures (as derived in Theorem 1). The gap between the top two curves in Fig. 3 is due to multiple node repair as well as the benefit from broadcast repair. The further reduction in the repair bandwidth in the bottom two curves in Fig. 3 comes from utilizing the remaining portion of data that is not lost on a failed node. We observe that the repair bandwidth reduces quickly for small values of storage capacity α, and saturates at a fixed value beyond a particular threshold value of α, which corresponds to the MBR point. This threshold value of α becomes smaller for larger values of ρ. There is another threshold value of α below which a finite repair bandwidth is not feasible any more, which corresponds to the MSR point.

VI. Conclusions

We have considered the repair of multiple partial failures through broadcast transmissions in a wireless distributed edge caching system. We have derived the optimal storage-repair bandwidth trade-off curve by constructing an information flow graph to represent the evolution of the system over time, and finding the min-cut across all failure combinations. Our results show that partial failures can reduce the repair bandwidth significantly thanks to the remaining correct bits in failed nodes. We have also shown that some storage and repair bandwidth pairs reported in the literature may not be feasible in general. Future work will focus on designing explicit codes achieving this trade-off, while also taking into account the heterogeneity of wireless networks.

References

[1] N. Golrezaei, A. Molisch, A. Dimakis, and G. Caire, "Femtocaching and device-to-device collaboration: A new architecture for wireless video distribution," *IEEE Comm. Magazine*, vol. 51, no. 4, 2013.

[2] E. Ozfatura and D. Gündüz, "Mobility and popularity-aware coded small-cell caching," *IEEE Comm. Letters*, vol. 22, no. 2, 2018.

[3] M. Gregori, J. Gómez-Vilardebó, J. Matamoros, and D. Gündüz, "Wireless content caching for small cell and d2d networks," *IEEE Journal on Selected Areas in Comm.*, vol. 34, no. 5, pp. 1222–1234, May 2016.

[4] N. Mital, D. Gündüz, and C. Ling, "Coded caching in a multi-server system with random topology," in *IEEE Wireless Communications and Networking Conference (WCNC)*, April 2018, pp. 1–6.

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

[6] Y. Hu, Y. Xu, X. Wang, C. Zhan, and P. Li, "Cooperative recovery of distributed storage systems from multiple losses with network coding," *IEEE Journal Selected Areas in Comms.*, vol. 28, no. 2, 2010.

[7] A. M. Kermarrec, N. L. Scouarnec, and G. Straub, "Repairing multiple failures with coordinated and adaptive regenerating codes," in *2011 International Symposium on Networking Coding*, July 2011, pp. 1–6.

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

[9] M. Gerami, M. Xiao, and M. Skoglund, "Partial repair for wireless caching networks with broadcast channels," *IEEE Wireless Communications Letters*, vol. 4, no. 2, pp. 145–148, April 2015.

[10] P. Hu, C. W. Sung, and T. H. Chan, "Broadcast repair for wireless distributed storage systems," in *Int. Conf. on Inf., Comms. and Signal Proc.*, Dec 2015, pp. 1–5.

[11] A. Rawat, O. Koyluoglu, and S. Vishwanath, "Centralized repair of multiple node failures," in *IEEE Int. Symp. Inf. Theory*, 2016.

[12] M. Gerami, M. Xiao, and M. Skoglund, "Two-layer coding in distributed storage systems with partial node failure/repair," *IEEE Communications Letters*, vol. 21, no. 4, pp. 726–729, April 2017.

[13] V. Abdrashitov, N. Prakash, and M. Medard, "The storage vs repair bandwidth trade-off for multiple failures in clustered storage networks," in *IEEE Inf. Theory Workshop (ITW)*, Nov 2017, pp. 46–50.

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

[15] N. Mital, K. Kralevska, D. Gunduz, and C. Ling, "Storage-Repair Bandwidth Trade-off for Wireless Caching with Partial Failure and Broadcast Repair," *ArXiv e-prints*, Jun. 2018.