Distributed Storage in Mobile Wireless Networks with Device-to-Device Communication

Distributed Storage in Mobile Wireless Networks with Device-to-Device Communication

Jesper Pedersen, Alexandre Graell i Amat, *Senior Member, IEEE*,
Iryna Andriyanova, *Member, IEEE*, and Fredrik Brännström, *Member, IEEE*

***Abstract*—We consider the use of distributed storage (DS) to reduce the communication cost of content delivery in wireless networks. Content is stored (*cached*) in a number of mobile devices using an erasure correcting code. Users retrieve content from other devices using device-to-device communication or from the base station (BS), at the expense of higher communication cost. We address the repair problem when a device storing data leaves the cell. We introduce a repair scheduling where repair is performed periodically and derive analytical expressions for the overall communication cost of content download and data repair as a function of the repair interval. The derived expressions are then used to evaluate the communication cost entailed by DS using several erasure correcting codes. Our results show that DS can reduce the communication cost with respect to the case where content is downloaded only from the BS, provided that repairs are performed frequently enough. If devices storing content arrive to the cell, the communication cost using DS is further reduced and, for large enough arrival rate, it is always beneficial. Interestingly, we show that MDS codes, which do not perform well for classical DS, can yield a low overall communication cost in wireless DS.**

***Index Terms*—Caching, content delivery, device-to-device communication, distributed storage, erasure correcting codes.**

I. Introduction

It IS predicted that the global mobile data traffic will exceed 30 exabytes per month by 2020, nearly a tenfold increase compared to the traffic in 2015 [1]. This dramatic increase threatens to completely congest the already burdened wireless networks. One popular approach to reduce peak traffic is to store popular content closer to the end users, a technique known as *caching*. The idea is to deploy a number of access points (called helpers) with large storage capacity, but low-rate wireless backhaul, and store data across them [2], [3]. Users can then download content from the helpers, resulting in a higher throughput per user. In [4] it was suggested to store content directly in the mobile devices, taking advantage of the high storage capacity of modern smart phones and tablets. The requested content can then be directly retrieved from neighbouring mobile devices, using *device-to-device* (D2D) *communication*. This allows for a more efficient content delivery at no additional infrastructure cost. Caching in the mobile devices to alleviate the wireless bottleneck has attracted a significant interest in the research community in the recent years [5]–[8]. In all these works, simple content caching and/or replication (i.e., a number of copies of a content are stored in the network) is considered. Additionally, the use of maximum distance separable (MDS) codes to facilitate decentralized random caching was investigated in [8].

A relevant problem in D2D-assisted mobile caching networks is the repairing of the lost data when a storage device is unavailable, e.g., when a storage device fails or leaves the network. Repairing of the lost data was considered in [9], where the communication cost incurred by data download and repair was analyzed for a caching scheme where data is stored in the mobile devices using replication and regenerating codes [10]. A strong assumption in [9] is that the repair of the lost content is performed instantaneously. As a result, content can always be downloaded from the mobile devices. Under the assumption of instantaneous repair, the caching strategy that minimizes the overall communication cost is 2-replication.

In this paper, we consider content caching in a wireless network scenario using erasure correcting codes. When using erasure correcting codes to cache content, caching bears strong ties with the concept of distributed storage (DS) for reliable data storage. Indeed, the set of mobile devices storing content can be seen as a distributed storage network. The fundamental difference with respect to DS for reliable data storage is that data download can be done not only from the storage nodes, but the base station (BS) can also assist to deliver the data. Therefore, the strict guarantees on fault tolerance can be relaxed, which brings new and interesting degrees of freedom with respect to erasure-correcting coding for DS for reliable data storage. Here, to avoid confusion with standard (uncoded) caching, we will use the term *wireless distributed storage*, highlighting the resemblance with DS using erasure correcting codes for reliable data storage in, e.g., data centers. Similar to the scenario in [9], we consider a cellular system where mobile devices roam in and out of a cell according to a Poisson random process and request content at random times. The cell is served by a BS, which always has access to the content. Content is also stored across a limited number of mobile devices using an erasure correcting code. Our main focus is on the repair problem when a device that stores data leaves the network. In particular, we introduce a more realistic repair scheduling than the one in [9] where lost content is repaired (from storage devices using D2D communication or from the BS) at periodic times.

We derive analytical, closed-form expressions for the overall communication cost of content download and data repair as a function of the repair interval. The derived expressions are

This paper was presented in part at the IEEE Information Theory Workshop, Jeju Island, Korea, October 2015.

This work was partially funded by the Swedish Research Council under grants 2011-5961 and 2011-5950, and by the European Research Council under Grant No. 258418 (COOPNET).

J. Pedersen, A. Graell i Amat, and F. Brännström are with the Department of Signals and Systems, Chalmers University of Technology, SE-41296 Gothenburg, Sweden (e-mail: {jesper.pedersen, alexandre.graell, fredrik.brannstrom}@chalmers.se).

I. Andriyanova is with the ETIS-UMR8051 group, ENSEA/University of Cergy-Pontoise/CNRS, 95015 Cergy, France (e-mail: iryna.andriyanova@ensea.fr).

general and can be used to analyze the overall communication cost incurred by any erasure correcting code for DS. As an example of the application of the proposed framework, we analyze the overall communication cost incurred by MDS codes, regenerating codes [10], and locally repairable codes (LRCs) [11]. We show that wireless DS can reduce the overall communication cost as compared to the basic scenario where content is only downloaded from the BS. However, this is provided that repairs can be performed frequently enough. Moreover, in the case when nodes storing content arrive to the cell, the communication cost using DS is further reduced and, for large enough arrival rate, it is always beneficial as compared to BS download. The repair interval that minimizes the overall communication cost depends on the network parameters and the underlying erasure correcting code. We show that, in general, instantaneous repair is not optimal. The derived expressions can also be used to find, for a given repair interval, the erasure correcting code yielding the lowest overall communication cost.

Non-instantaneous repairs, the so-called "lazy" repairs, have already been proposed for DS in data centers [12], [13] to reduce the amount of data that has to be transmitted within the storage network during the repair process, known as the *repair bandwidth*. However, contrary to [12], [13], in the wireless scenario considered here the non-instantaneous repairs impact both data repair and download. We show that, somewhat interestingly, erasure correcting codes achieving a low repair bandwidth do not always perform well in a wireless DS setting. On the other hand, MDS codes, which entail a high repair bandwidth, can yield a low overall communication cost for some repair intervals.

Notation: The probability density function (pdf) of a random variable X is denoted by $f_X(\cdot)$. Expectation and probability are denoted by $\mathbb{E}[\cdot]$ and $\mathbb{P}(\cdot)$, respectively. We use bold lowercase letters $\boldsymbol{x}$ to denote vectors and bold uppercase letters $\boldsymbol{X}$ for matrices.

II. System Model

We consider a single cell in a cellular network, served by a BS, where mobile devices (referred to as nodes) arrive and depart according to a Poisson random process. The initial number of nodes in the network is M. Nodes wish to download content from the network. For simplicity, we assume that there is a single object (file), of size F bits, stored at the BS. We further assume that nodes can store data and communicate between them using D2D communication. The considered scenario is depicted in Fig. 1.

Arrival-departure model. Nodes arrive according to a Poisson process with exponential independent, identically distributed (i.i.d.) random inter-arrival times T_{a} with pdf

$$f_{T_{\text{a}}}(t) = M\lambda e^{-M\lambda t}, \quad \lambda \geq 0,\ t \geq 0, \tag{1}$$

where $M\lambda$ is the expected arrival rate of a node and t is time, measured in time units (t.u.).

The nodes stay in the cell for an i.i.d. exponential random lifetime T_{l} with pdf

$$f_{T_{\text{l}}}(t) = \mu e^{-\mu t}, \quad \mu \geq 0,\ t \geq 0, \tag{2}$$

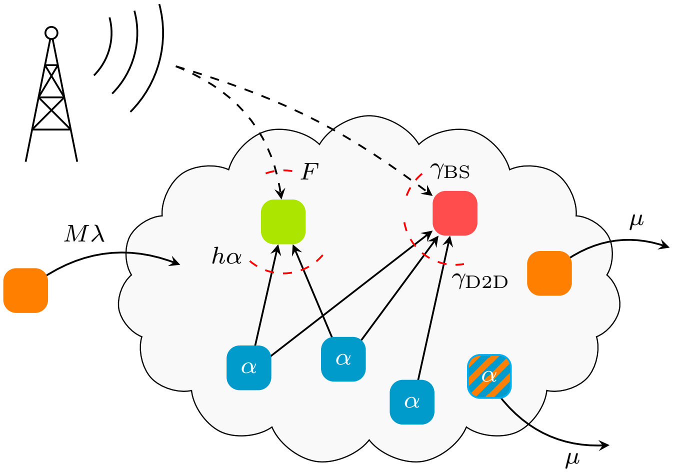

Figure 1. A wireless network with data storage in the mobile devices (nodes). A new node arrives to the network at rate $M\lambda$. The departure rate per node is μ. Blue nodes store exactly α bits each. The green node requests the file and downloads it from the storage nodes (solid arrows), or from the BS (dashed arrow). The repair onto a node (in red) is carried out by transmitting γ_{D2D} bits from storage nodes (solid arrows) or γ_{BS} bits from the BS (dashed arrow).

where μ is the expected departure rate of a node. The number of nodes in the cell can be described by an $\mathsf{M}/\mathsf{M}/\infty$ queuing model where the probability that there are i nodes in the cell is [14]

$$\pi(i) = \frac{(M\lambda/\mu)^i}{i!} e^{-(M\lambda/\mu)}. \tag{3}$$

For simplicity, we assume that $\mu = \lambda$, i.e., the flow in and out from the cell is the same and the expected number of nodes in the cell stays constant (equal to M).

Data storage. The file is partitioned into k packets, called symbols, of size $\frac{F}{k}$ bits and is encoded into n coded symbols, $n \geq 2$, using an (n, k) erasure correcting code of rate $R = k/n < 1$. The encoded data is stored in m nodes, $2 \leq m \leq n$, referred to as *storage nodes*. Note that $m \leq n$ implies that a storage node may store multiple coded symbols. For some of the considered erasure correcting codes, this is the case (see Section VI). To simplify the analysis in Sections III and IV, we set $m \ll M$. This guarantees that the probability that the number of nodes in the cell is smaller than m is negligibly small, i.e.,

$$\sum_{i=0}^{m-1} \pi(i) \ll 1, \tag{4}$$

using (3). For example, for $m \leq 10$ and $M = 30$, (4) is less than $7.2 \cdot 10^{-6}$. Therefore, with high probability the file can be stored in the cell. In the results section we show that this simplification has negligible impact and that the analytical expressions match closely with the simulation results.

Each storage node stores exactly α bits, i.e., we consider a symmetric allocation [15]. Hence[1],

$$\alpha = \frac{1}{m} \cdot \frac{F}{R} \geq \frac{F}{k}. \tag{5}$$

Incoming process. Nodes arriving to the cell may bring cached content. The expected arrival rate of nodes storing content is $m\lambda_{\text{c}}$, $\lambda_{\text{c}} \leq \mu$. We also assume that the expected

[1] Without loss of generality, we assume $\alpha \in \mathbb{N}$.

arrival rate of nodes not carrying content is $M\lambda - m\lambda_c$, so that the expected arrival rate of a node (with or without content) is $M\lambda$ and the expected number of nodes in the cell is M (see above). The incoming process is discussed in more detail in Section V.

Data delivery. Nodes request the file at random times with i.i.d. random inter-request time T_r with pdf

$$f_{T_r}(t) = \omega e^{-\omega t}, \quad \omega \geq 0,\ t \geq 0, \tag{6}$$

where ω is the expected request rate per node. Whenever possible, the file is downloaded from the storage nodes using D2D communication, referred to as D2D download. In particular, we assume that data can be downloaded from any subset of h storage nodes, $1 \leq h < m$, which we will refer to as the *download locality*. In other words, D2D download is possible if h or more storage nodes remain in the cell. In this case, the amount of downloaded data is $h\alpha \geq F$ bits.[2] In the case where there are less than h storage nodes in the cell, the file is downloaded from the BS, which we refer to as BS download. In this case, F bits are downloaded.

Communication cost. We assume that transmission from the BS and from a storage node (in D2D communication) have different costs. We denote by ρ_{BS} and ρ_{D2D} the cost (in cost units (c.u.) per bit, [c.u./bit]) of transmitting one bit from the BS and from a storage node, respectively. Therefore, the cost of downloading a file from the BS and the storage nodes is $\rho_{\text{BS}}F$ and $\rho_{\text{D2D}}h\alpha$, respectively. Furthermore, we define $\rho \triangleq \rho_{\text{BS}}/\rho_{\text{D2D}} > 0$, where $\rho > 1$ corresponds to a high traffic load in the BS-to-device link and $\rho < 1$ reflects a scenario where the battery of the devices is the main constraint.

A. Repair Process

When a storage node leaves the cell, its stored data is lost (see blue node with orange stripes in Fig. 1). Therefore, another node needs to be populated with data to maintain the initial state of reliability of the DS network, i.e., m storage nodes. The restore (repair) of the lost data onto another node, chosen uniformly at random from all nodes in the cell that do not store any content, will be referred to as the repair process. We introduce a scheduled repair scheme where the repair process is run periodically. We denote the interval between two repairs by Δ (in t.u.), $\Delta \geq 0$. Note that $\Delta = 0$ corresponds to the case of instantaneous repair, considered in [9].

Similar to the download, repair can be accomplished from the storage nodes (D2D repair) or from the BS (BS repair), with cost per bit ρ_{D2D} and ρ_{BS}, respectively. The amount of data (in bits) that needs to be retrieved from the network to repair a single failed node is referred to as the *repair bandwidth*, denoted by γ. For simplicity, we assume that each repair is handled independently of the others. In particular, we assume that D2D repair can be performed from any subset of r storage nodes, $1 \leq r < m$, by retrieving $\beta \leq \alpha$ bits from each node. In other words, D2D repair is possible if there are at least r storage nodes in the cell at the moment of repair. In this case, $\gamma_{\text{D2D}} = r\beta \geq \alpha$, and the corresponding communication cost is $\rho_{\text{D2D}}\gamma_{\text{D2D}}$. Parameter r is usually referred to as the *repair locality* in the DS literature. If there are less than r storage nodes in the cell at the moment of repair, then the repair is carried out by the BS. In this case, $\gamma_{\text{BS}} = \alpha$, with communication cost $\rho_{\text{BS}}\gamma_{\text{BS}}$. Note that $\gamma_{\text{D2D}}/\gamma_{\text{BS}} \geq 1$. For both repair and download, we assume error-free transmission.

Parameters m, h, r, α and β, and subsequently γ_{D2D} and γ_{BS}, depend on the erasure correcting code used for storage. Since m, h and r are very important parameters, an erasure correcting code in DS is typically defined with the triple $[m, h, r]$. This will be further explained in Section VI.

III. Repair and Download Cost

In this section, we derive analytical expressions for the repair and download cost, and subsequently for the overall communication cost, as a function of the repair interval Δ. For analysis purposes, we initially disregard the incoming process, i.e., set $\lambda_c = 0$. The case $\lambda_c > 0$ is then addressed in Section V building upon the results in this section. We denote by $\bar{C}_r$ the average communication cost of repairing lost data, and refer to it as the repair cost. Also, we denote by $\bar{C}_d$ the average communication cost of downloading the file, and refer to it as the download cost. The (average) overall communication cost is denoted by $\bar{C}$, where $\bar{C} \triangleq \bar{C}_r + \bar{C}_d$. The costs are defined in cost units per bit and time unit, [c.u./(bit×t.u.)].

For later use, we denote by $b_i(m,p)$ the probability mass function (pmf) of the binomial distribution with parameters m and p,

$$b_i(m,p) \triangleq \binom{m}{i} p^i (1-p)^{m-i}, \quad 0 \leq i \leq m. \tag{7}$$

A. Repair Cost

The repair cost $\bar{C}_r$ has two contributions, corresponding to the cases of BS repair and D2D repair. Denote by m_r^{D2D} and m_r^{BS} the average number of nodes repaired from the storage nodes and from the BS, respectively, in one repair interval. Then, $\bar{C}_r$ (in [c.u./(bit×t.u.)]) is given by

$$\bar{C}_r = \frac{1}{F\Delta}\left(\rho_{\text{BS}}\gamma_{\text{BS}}m_r^{\text{BS}} + \rho_{\text{D2D}}\gamma_{\text{D2D}}m_r^{\text{D2D}}\right), \tag{8}$$

where $\rho_{\text{BS}}\gamma_{\text{BS}}$ and $\rho_{\text{D2D}}\gamma_{\text{D2D}}$ (in c.u.) are the cost of repairing a single storage node from the BS and from storage nodes, respectively (see Section II-A), and we normalize by F such that $\bar{C}_r$ does not depend on the file size.

The repair cost, $\bar{C}_r$, is given in the following theorem.

Theorem 1. *Consider the DS network in Section II with departure rate μ, communication costs ρ_{BS} and ρ_{D2D}, BS repair bandwidth γ_{BS}, file size F, repair interval Δ, and probability p that a node has not left the network during a time Δ. Furthermore, consider the use of an $[m, h, r]$ erasure*

[2] To simplify the analysis in Sections III and IV, we assume that the download bandwidth is the same irrespective of whether the request comes from a storage node itself or not, i.e., users do not have access to their own stored data. This is a reasonable approximation if $m \ll M$. Furthermore, this may be a practical assumption. Due to concerns about security in systems that allow for D2D connectivity, it has been proposed to isolate part of the memory in the mobile devices to be used only for DS, so that devices cannot have access to their own cached data [16].

correcting code with D2D repair bandwidth γ_{D2D}. The repair cost is given by

$$\bar{C}_{\mathrm{r}} = \frac{1}{F\Delta}\left(\rho_{\mathrm{BS}}\gamma_{\mathrm{BS}}\sum_{i=0}^{r-1}(m-i)b_i(m,p)\right.$$
$$\left.+\rho_{\mathrm{D2D}}\gamma_{\mathrm{D2D}}\sum_{i=r}^{m}(m-i)b_i(m,p)\right). \qquad (9)$$

Proof: As the inter-departure times are exponentially distributed, the probability that a storage node has not left the network during a time Δ and is available for repair is

$$p = \mathbb{P}(T_1 > \Delta) = e^{-\mu\Delta}.$$

Hence, the probability that i storage nodes are available for repair is $b_i(m,p)$. If i storage nodes remain in the cell, then $m-i$ repairs need to be performed. D2D repair is performed if $i \geq r$, and BS repair is performed otherwise. Therefore,

$$m_{\mathrm{r}}^{\mathrm{D2D}} = \sum_{i=r}^{m}(m-i)b_i(m,p), \quad m_{\mathrm{r}}^{\mathrm{BS}} = \sum_{i=0}^{r-1}(m-i)b_i(m,p).$$

Using these expressions in (8), we obtain (9). ■

Remark 1. *We see from* (8) *that if $\rho_{\mathrm{BS}}\gamma_{\mathrm{BS}} < \rho_{\mathrm{D2D}}\gamma_{\mathrm{D2D}}$, i.e., $\rho < \frac{\gamma_{\mathrm{D2D}}}{\gamma_{\mathrm{BS}}}$, D2D repair should never be performed, as repairing always from the BS yields a lower repair cost. In this case the repair cost would be*

$$\bar{C}_{\mathrm{r}}^{\mathrm{BS}} = \frac{1}{F\Delta}\rho_{\mathrm{BS}}\gamma_{\mathrm{BS}}m(1-e^{-\mu\Delta}).$$

B. Download Cost

Similar to $\bar{C}_{\mathrm{r}}$, the download cost $\bar{C}_{\mathrm{d}}$ has two contributions, corresponding to the case where content is downloaded from the BS and from the storage nodes. Denote by p_{BS} and p_{D2D} the probability that, for a request, the file is downloaded from the BS and from the storage nodes, respectively. Then, $\bar{C}_{\mathrm{d}}$ can be written as

$$\bar{C}_{\mathrm{d}} = \frac{M\omega}{F}\left(\rho_{\mathrm{BS}}Fp_{\mathrm{BS}} + \rho_{\mathrm{D2D}}h\alpha p_{\mathrm{D2D}}\right), \qquad (10)$$

where $\rho_{\mathrm{BS}}F$ and $\rho_{\mathrm{D2D}}h\alpha$ are the cost of downloading the file from the BS and from the storage nodes, respectively (see Section II), and $M\omega$ is the overall request rate per t.u.. Again, we normalize by F so that the cost does not depend on the file size. The download cost is given in the following theorem.

Theorem 2. *Consider the DS network in Section II with expected number of nodes in the cell M, departure rate μ, request rate ω, communication costs ρ_{BS} and ρ_{D2D}, file size F, and repair interval Δ. Furthermore, consider the use of an $[m,h,r]$ erasure correcting code that stores α bits per node. Let $\mu_i = i\mu$ for $i = h,\dots,m$, and $p_i = e^{-\mu_i\Delta}$. The download cost is given by*

$$\bar{C}_{\mathrm{d}} = M\omega\left(\rho_{\mathrm{BS}} + \left(\rho_{\mathrm{D2D}}\frac{h\alpha}{F} - \rho_{\mathrm{BS}}\right)\frac{1}{\Delta}\sum_{i=h}^{m}\frac{1-p_i}{\mu_i}\prod_{\substack{j=h\\ j\neq i}}^{m}\frac{j}{j-i}\right). \qquad (11)$$

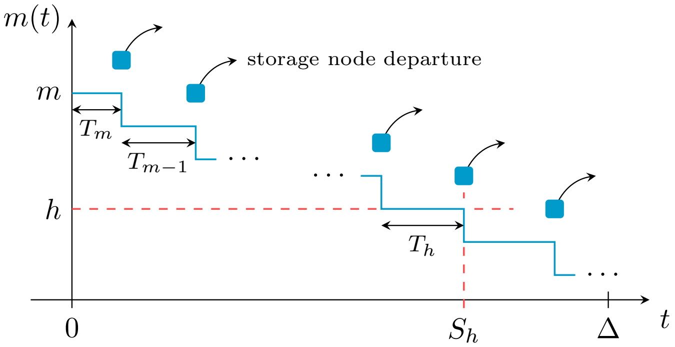

Figure 2. Number of available storage nodes within the repair interval Δ. At $t = 0$, there are m storage nodes available. S_h is the time after which less than h storage nodes are available, hence D2D download is no longer possible.

The proof is given in Appendix A. Here, for ease of understanding, we give an outline of the proof. Since $p_{\mathrm{D2D}} + p_{\mathrm{BS}} = 1$, it follows from (10) that to derive $\bar{C}_{\mathrm{d}}$ is sufficient to derive p_{D2D}. Let $m(t)$ be the number of storage nodes alive in the cell within a repair interval, i.e., for $t \in [0,\Delta)$, with $m(0) = m$. It is important to observe that $m(t)$ is described by a Poisson death process [14], since storage nodes may leave the cell, and no repair is attempted before a time Δ. This random process is illustrated in Fig. 2. At some point, too many storage nodes have left the network, such that the number of available storage nodes goes below h and D2D download is no longer possible. Denote the (random) time this occurs by S_h, i.e., $m(t) < h \ \forall\ t \geq S_h$, $t \in [0,\Delta)$ (see Fig. 2). Denote by $\tilde{W}_\ell$ the arrival time of the ℓth file request within a repair interval, $t \in [0,\Delta)$. The probability p_{D2D} can then be derived in two steps.

1) Find the pdf of the arrival time of the file requests within a repair interval Δ, $\tilde{W}_\ell$.
2) Find the probability that a request arrives before S_h, $p_{\mathrm{D2D}} = \mathbb{P}(\tilde{W}_\ell < S_h)$ (i.e., D2D download is possible).

Remark 2. *If $\rho_{\mathrm{BS}}F < \rho_{\mathrm{D2D}}h\alpha$, i.e., $\rho < \frac{h\alpha}{F}$, performing BS download only is optimal. The download cost is then*

$$\bar{C}_{\mathrm{d}}^{\mathrm{BS}} = M\omega\rho_{\mathrm{BS}}. \qquad (12)$$

We also have the following result about the behavior of $\bar{C}_{\mathrm{d}}$ in (11).

Corollary 1. *For $\mu > 0$, $\bar{C}_{\mathrm{d}}$ is monotonically increasing with Δ if $\rho > \frac{h\alpha}{F}$, monotonically decreasing with Δ if $\rho < \frac{h\alpha}{F}$, and constant otherwise.*

Proof: The proof follows directly from differentiating $\bar{C}_{\mathrm{d}}$ with respect to Δ and is therefore omitted. ■

C. Overall Communication Cost

Combining Theorems 1 and 2, one obtains the expression for the overall communication cost,

$$\bar{C} = \bar{C}_{\mathrm{r}} + \bar{C}_{\mathrm{d}}. \qquad (13)$$

Note that, in general, $\bar{C}$ is not monotone with Δ. We can derive the following result for $\Delta = 0$ (instantaneous repair) and $\Delta \to \infty$ (no repair).

Corollary 2.

$$\lim_{\Delta\to 0}\bar{C} = \frac{\rho_{\mathrm{D2D}}}{F}(\gamma_{\mathrm{D2D}}m\mu + M\omega h\alpha). \tag{14}$$

Moreover, for $\mu > 0$,

$$\lim_{\Delta\to\infty}\bar{C} = M\omega\rho_{\mathrm{BS}}. \tag{15}$$

Proof: See Appendix B. ■

For instantaneous repair ($\Delta = 0$), both repair and download are always performed from the storage nodes. Thus, the two terms in (14) correspond to the D2D repair and D2D download, and we recover the result in [9]. For $\Delta \to \infty$, data is never repaired (hence, $\bar{C}_{\mathrm{r}} = 0$). For $\mu > 0$, the number of storage nodes in the cell will become smaller than h at some point, and D2D download is no longer possible. Therefore, the overall communication cost in (15) is the BS download cost in (12).

IV. Hybrid Repair and Download

In the system model in Section II and the analysis in Section III we assumed that if repair (resp. download) cannot be completed from storage nodes (because there are less than r (resp. h) storage nodes available in the cell), BS repair (resp. download) is performed. Alternatively, for both repair and download, a node might retrieve data from the available storage nodes using D2D communication and retrieve the rest from the BS to complete the repair or the download. We will refer to this setup as partial D2D repair and partial D2D download, and the scheme that implements it as the *hybrid repair and download scheme*. In the following, we extend the analysis in Section III to the hybrid scheme.

A. Repair Cost

Assume that, at the time of repair, $i < r$ storage nodes are available, i.e., repair cannot be accomplished exclusively from the storage nodes. However, $i\beta$ bits could be retrieved from the i available storage nodes and the remaining $\gamma_{\mathrm{D2D}} - i\beta = (r-i)\beta$ bits to complete the repair from the BS. The corresponding communication cost is $(\rho_{\mathrm{BS}}(r-i) + \rho_{\mathrm{D2D}}i)\beta$. For the conventional scheme, D2D repair is not possible for $i < r$, and the repair cost corresponds to that of BS repair, i.e., $\rho_{\mathrm{BS}}\gamma_{\mathrm{BS}}$. This implies that, if $i < r$, partial repair leads to a reduced repair cost if $(\rho_{\mathrm{BS}}(r-i) + \rho_{\mathrm{D2D}}i)\beta < \rho_{\mathrm{BS}}\gamma_{\mathrm{BS}}$ or, equivalently, $i > \frac{\rho_{\mathrm{BS}}}{\rho_{\mathrm{BS}}-\rho_{\mathrm{D2D}}}\left(r - \frac{\gamma_{\mathrm{BS}}}{\beta}\right) \triangleq \mathsf{c}$. For $i < r$, the hybrid scheme performs partial D2D repair if $i > \mathsf{c}$ and BS repair otherwise. The repair cost is given in the following theorem.

Theorem 3. *Consider the DS network in Section II using the hybrid scheme. The repair cost is given by*

$$\begin{aligned}\bar{C}_{\mathrm{r}}^{\mathrm{hybrid}} = \frac{1}{F\Delta}\Bigg(&\rho_{\mathrm{BS}}\gamma_{\mathrm{BS}}\sum_{i=0}^{a}(m-i)b_i(m,p)\\ &+ \sum_{i=a+1}^{r-1}(m-i)(\rho_{\mathrm{BS}}(r-i) + i\rho_{\mathrm{D2D}})\beta b_i(m,p)\\ &+ \rho_{\mathrm{D2D}}\gamma_{\mathrm{D2D}}\sum_{i=r}^{m}(m-i)b_i(m,p)\Bigg),\end{aligned}$$

where $a = \min\left\{\left\lfloor\frac{\rho_{\mathrm{BS}}}{\rho_{\mathrm{BS}}-\rho_{\mathrm{D2D}}}\left(r - \frac{\gamma_{\mathrm{BS}}}{\beta}\right)\right\rfloor, r-1\right\}$, $\left(r - \frac{\gamma_{\mathrm{BS}}}{\beta}\right) \geq 0$ for all codes in Section VI, and $p = e^{-\mu\Delta}$.

Proof: It follows the same lines as the proof of Theorem 1. ■

B. Download Cost

Similar to the repair case, if $i < h$ storage nodes are available at the time of a file request, the file cannot be downloaded solely from the storage nodes. However, $i\alpha$ bits could be downloaded from the i available storage nodes and the remaining $(h-i)\alpha$ bits from the BS, with communication cost $(\rho_{\mathrm{BS}}(h-i) + \rho_{\mathrm{D2D}}i)\alpha$. For the conventional scheme, the download cost corresponds to that of BS download, i.e., $\rho_{\mathrm{BS}}F$. Hence, the hybrid scheme leads to a lower download cost if $(\rho_{\mathrm{BS}}(h-i) + \rho_{\mathrm{D2D}}i)\alpha < \rho_{\mathrm{BS}}F$, or equivalently, $i > \frac{\rho_{\mathrm{BS}}}{\rho_{\mathrm{BS}}-\rho_{\mathrm{D2D}}}\left(h - \frac{F}{\alpha}\right) \triangleq \mathsf{d}$. For $i < h$, the hybrid scheme performs partial D2D download if $i > \mathsf{d}$ and BS download otherwise. The download cost is given in the following theorem.

Theorem 4. *Consider the DS network in Section II using the hybrid scheme. Let $\mu_i = i\mu$ and $p_i = e^{-\mu_i\Delta}$, for $i = 1,\ldots,m$. The download cost is given by*

$$\begin{aligned}\bar{C}_{\mathrm{d}}^{\mathrm{hybrid}} = \frac{M\omega}{F}\Bigg(&\rho_{\mathrm{BS}}F\Bigg(1 - \frac{1}{\Delta}\sum_{i=1}^{m}\frac{1-p_i}{\mu_i}\prod_{\substack{j=1\\ j\neq i}}^{m}\frac{j}{j-i}\Bigg)\\ &+ \rho_{\mathrm{BS}}F\sum_{i=1}^{a}\mathsf{c}_i + \sum_{i=a+1}^{h-1}(\rho_{\mathrm{BS}}(h-i) + i\rho_{\mathrm{D2D}})\alpha\mathsf{c}_i\\ &+ \rho_{\mathrm{D2D}}h\alpha\frac{1}{\Delta}\sum_{i=h}^{m}\frac{1-p_i}{\mu_i}\prod_{\substack{j=h\\ j\neq i}}^{m}\frac{j}{j-i}\Bigg),\end{aligned} \tag{16}$$

where $a = \min\left\{\left\lfloor\frac{\rho_{\mathrm{BS}}}{\rho_{\mathrm{BS}}-\rho_{\mathrm{D2D}}}\left(h - \frac{F}{\alpha}\right)\right\rfloor, h-1\right\}$, $\left(h - \frac{F}{\alpha}\right) \geq 0$, and

$$\begin{aligned}\mathsf{c}_i = &\frac{1}{\Delta}\sum_{i'=i}^{m}\frac{1-p_{i'}}{\mu_{i'}}\prod_{\substack{j=i\\ j\neq i'}}^{m}\frac{j}{j-i'}\\ &- \frac{1}{\Delta}\sum_{i'=i+1}^{m}\frac{1-p_{i'}}{\mu_{i'}}\prod_{\substack{j=i+1\\ j\neq i'}}^{m}\frac{j}{j-i'}.\end{aligned}$$

Proof: See Appendix C. ■

V. Repair and Download Cost with an Incoming Process

The analysis in the preceding sections does not consider the possibility that nodes arriving to the cell may bring content. In a real scenario with neighboring cells, however, this may be the case. We will refer to the arrival of nodes with content as the *incoming process*. Considering an incoming process significantly complicates the analysis. This is due to the fact that arriving nodes may bring content that is not *directly useful*, in the sense that they may bring code symbols which are already available in another storage node. At a given time, it is likely that some symbols will be stored by more than one storage node, while other symbols will not be present in the storage network (due to node departures). As a result, the analysis needs to consider *storage node classes*, where a node class defines the set of storage nodes storing given code symbols. In general, for an $[m, h, r]$ erasure correcting code, there are m storage node classes, since all code symbols are different. The case of simply replicating the data (using a repetition code) is a bit different. Despite the fact that all code symbols are equal, for the analysis of m-replication we still need to consider m storage node classes, i.e., we treat each of the code symbols of the m-replication as they were different.

In this section, we extend the analysis in Sections III and IV to the scenario with an incoming process. In particular, we show that Theorem 1 and Theorem 2 can also be used to analyze the repair and download costs for this scenario by using different input parameters. More precisely, we consider the scenario where storage nodes of a given class arrive to the cell according to a Poisson process with expected arrival rate $\lambda_{\mathrm{c}} \leq \mu$. An incoming storage node brings a single code symbol of a given class. Furthermore, nodes not storing content arrive according to a Poisson process with expected arrival rate $M\lambda - m\lambda_{\mathrm{c}}$. The departure rate for all nodes is $\mu = \lambda$, i.e., as before, the average number of nodes in the cell is M. We assume the practical scenario where the BS maintains a list of the nodes storing content, which is communicated periodically to all nodes in the cell every Φ t.u.. For simplicity, we assume that $\Phi = \Delta$.

A. Repair Cost

Denote by $c_i(t)$ the number of class-i storage nodes in the cell at time t. Also, denote by $q_{i,0}(t)$ the probability that class i is empty at time t, i.e., $q_{i,0}(t) = \mathbb{P}(c_i(t) = 0)$. Since all storage node classes have the same arrival and departure rate, we can drop subindex i and write $q_{i,0}(t) = q_0(t)\ \forall i$. Also, let $\boldsymbol{q} = (q_0, q_1, \ldots)$ be the stationary distribution, where q_j is the probability that class i has j storage nodes. Equation (9) in Theorem 1 can then be used for the scenario with an incoming process by setting $p \leftarrow (1 - q_0)$.

The difficulty here lies in computing $\boldsymbol{q}$. Without repairs, the evolution of $c_i(t)$ is given by a Poisson birth-death process, which can be modeled by an $\mathsf{M/M/\infty}$ Markov chain model. In this case, the stationary distribution $\boldsymbol{q} = (q_0, q_1, \ldots)$ exists and can be computed. However, the repairs performed every Δ t.u. interfere with the stationarity of the process. Indeed, in the presence of repairs, the evolution of $c_i(t)$ does no longer correspond to a Poisson birth-death process. In this case, the analysis appears to be formidable.

Here, we propose the following two-step procedure to compute $\boldsymbol{q}$. Consider a single repair interval of duration Δ, where $c_i(\Delta)$ is the number of storage nodes in class i at time Δ. Within a repair interval $t \in [0, \Delta)$, $c_i(t)$ is described by a Poisson birth-death process[3]. Since storage node classes are independent of each other and have the same arrival and departure rates, we can focus on a single class. Hence, we will drop the subindex i in $c_i(t)$ and simply write $c(t)$.

Let $P_{ij}(t) = \mathbb{P}(c(t) = j | c(0) = i)$ denote the transition probability function of the continuous-time $\mathsf{M/M/\infty}$ Markov chain representing the Poisson birth-death process. $P_{ij}(t)$ can be computed by deriving a set of differential equations, called Kolmogorov's forward equations, whose solution can be computed as follows [17]. Let $\boldsymbol{P}(t)$ be the $S \times S$ matrix with (i, j)th entry $P_{ij}(t)$, where $S-1$ is the maximum number of storage nodes of one class. Also, let r_{ij} be the transition rates of the continuous-time Markov chain. Then $\boldsymbol{P}(t)$ can be computed as [17]

$$\boldsymbol{P}(t) = e^{t\boldsymbol{G}} \triangleq \sum_{\ell=0}^{\infty} \frac{(t\boldsymbol{G})^{\ell}}{\ell!}, \tag{17}$$

where $\boldsymbol{G}$ is the generator of the Markov chain, with entries g_{ij}, $i = 0, \ldots, S-1$ and $j = 0, \ldots, S-1$, given by

$$g_{ij} = r_{ij} \quad \text{for } i \neq j,$$
$$g_{ii} = -\sum_{j=0}^{S-1} r_{ij},$$

with

$$r_{ij} = \begin{cases} \lambda_{\mathrm{c}} & \text{if } j = i+1 \\ i\mu & \text{if } j = i-1 \\ 0 & \text{otherwise} \end{cases}. \tag{18}$$

The infinite power series in (17) converges for any square matrix $\boldsymbol{G}$, and can be efficiently computed using, e.g., the algorithm described in [18].

Note that in our scenario, S is not finite. However, if $\lambda_{\mathrm{c}} \leq \mu$ the probability of having $c(t) = j$ storage nodes of a given class at time t, $\mathbb{P}(c(t) = j)$, sharply decreases with j. Therefore, we can limit S to a sufficiently large value, and by solving (17) get a very good approximation of $\boldsymbol{P}(\Delta)$.

Given $\boldsymbol{P}(\Delta)$, we can estimate the stationary distribution $\boldsymbol{q}$ recursively. For a given distribution at time $t = 0$, $\boldsymbol{q}(0)$, we can compute $\boldsymbol{q}(i\Delta)$ as

$$q_j(i\Delta) = \sum_{\ell=0}^{S-1} P_{\ell j}(\Delta)\tilde{q}_{\ell}((i-1)\Delta), \quad j = 0, \ldots, S-1, \tag{19}$$

where $\tilde{q}_0(i\Delta) = 0$ and $\tilde{q}_1(i\Delta) = q_0(i\Delta) + q_1(i\Delta)$, due to the repair, and $\tilde{q}_{\ell}(i\Delta) = q_{\ell}(i\Delta)$ for $\ell = 2, \ldots, S-1$.

Equivalently, this recursion can be written in compact form as

$$\tilde{\boldsymbol{q}} = \lim_{N\to\infty} \boldsymbol{q}(0)(\boldsymbol{P}(\Delta)\boldsymbol{X})^N, \tag{20}$$
$$\boldsymbol{q} = \tilde{\boldsymbol{q}}\boldsymbol{P}(\Delta), \tag{21}$$

where $\boldsymbol{X}$ is an $S \times S$ matrix with entries $x_{00} = 0$, $x_{ii} = 1$ for $i > 0$, and $x_{01} = 1$. Note that $\boldsymbol{q}$ and $\tilde{\boldsymbol{q}}$ are the stationary distributions before and after repair, respectively.

Theorem 5. *Consider the DS network in Section II with departure rate μ, arrival rate of storage nodes of a given class λ_{c}, arrival rate of nodes not storing content $M\lambda - m\lambda_{\mathrm{c}}$, communication costs ρ_{BS} and ρ_{D2D}, BS repair bandwidth γ_{BS}, file size F, and repair interval Δ. Furthermore, consider the use of an $[m, h, r]$ erasure correcting code with D2D*

[3] This is contrast to the case with no incoming process, where the evolution of $c_i(t)$ for $t \in [0, \Delta)$ is described by a Poisson death process.

repair bandwidth γ_{D2D}. The repair cost is given by (9) *with $p \leftarrow (1-q_0)$, and q_0 is given by the first element of $\boldsymbol{q}$ in* (21).

Proof: The proof follows from the discussion above. ■

Theorem 6. *Consider the DS network in Section II with departure rate μ, arrival rate of storage nodes of a given class λ_{c}, and arrival rate of nodes not storing content $M\lambda - m\lambda_{\text{c}}$, using the hybrid scheme. The repair cost is given by the expression in Theorem 3 with $p \leftarrow (1-q_0)$, and q_0 is given by the first element of $\boldsymbol{q}$ in* (21).

Proof: The proof follows from the discussion above. ■

Remark 3. *It is important to remark that the analysis for the scenario with an incoming process does not consider the departure of individual storage nodes, but rather the departure of whole classes, i.e., all nodes of a given class. Thus, r and m in* (9) *should not be interpreted as r storage nodes and m storage nodes, respectively, but as r and m storage node classes.*

Remark 4. *Note that in the analysis above we have made the assumption that the stationary distribution $\boldsymbol{q}$ exists. While we do not have a formal proof for this, our numerical results suggest that it does exist. In fact, the recursion* (20) *and* (21) *converges to the same $\boldsymbol{q}$ independently of $\boldsymbol{q}(0)$.*

B. Download Cost

Assume that after repair there are ℓ storage nodes of a given class, say class i. With some abuse of notation, let $c_i(t,\ell)$ be the number of storage nodes of class i at time t, where parameter ℓ indicates that $c_i(t=0)=\ell$. The evolution of $c_i(t,\ell)$ for $t \in [0,\Delta)$ is given by a Poisson death process. Denote by U_ℓ the time instant at which the last of the ℓ storage nodes in class-i leaves the cell. U_ℓ is hypoexponentially distributed with pdf given by (31), with $h=1$ and $m=\ell$. The expected value of U_ℓ is [19, Sec. 1.3.1]

$$\mathbb{E}[U_\ell] = \sum_{j=1}^{\ell} \frac{1}{j\mu}. \tag{22}$$

Note that U_1 is exponentially distributed.

Let U be the time instant at which the last of the storage nodes in class i leaves the cell or, in other words, the time instant at which the whole class i leaves the cell. The pdf of U is a weighted sum of the pdfs U_ℓ, weighed by $\tilde{q}_\ell$, i.e., it is a weighted sum of hypoexponential distributions. The expected value of U is

$$\mathbb{E}[U] = \sum_{\ell=0}^{\infty} \tilde{q}_\ell \sum_{j=1}^{\ell} \frac{1}{j\mu}. \tag{23}$$

Let $b(t)$ be the number of nonempty storage node classes in the cell at time t. Computing $\bar{C}_{\text{d}}$ exactly requires to compute the distribution of the time instant at which $b(t)$ changes from h to $h-1$, denoted by S_h, similar to the case with no incoming process (see Appendix A). Unfortunately, due to the fact that the pdf of U is a weighted sum of hypoexponential distributions, computing the pdf of U seems unfeasible. Here, we propose to approximate the pdf of U by an exponential pdf. Indeed, it appears that $\tilde{q}_1$ is in general the largest element in $\tilde{\boldsymbol{q}}$, therefore the distribution of U has a large exponential component. Assuming that U is well approximated by an exponential distribution with mean $\tilde{\mu}^{-1}$, the download cost for the scenario with an incoming process can then be computed using (11) in Theorem 2 by setting $\mu \leftarrow \tilde{\mu}$, where now a storage node departure should be interpreted as a storage node class departure. We have observed that by approximating the pdf of U by an exponential distribution with mean $\tilde{\mu}^{-1} = \mathbb{E}[U]$, the analytical results match very well with the simulations for the whole range of interesting values of μ and λ_{c}, as shown in the results section. The download cost for the hybrid scheme is found by using (16) in Theorem 4 with $\mu \leftarrow \tilde{\mu}$.

VI. Erasure Correcting Codes in Distributed Storage

From Sections III–V, it can be seen that the overall communication cost $\bar{C}$ depends on the network parameters μ (λ), λ_{c}, M, and ω, and on the parameters m, h, r, α, and β (and subsequently on $\gamma_{\text{D2D}} = r\beta$ and $\gamma_{\text{BS}} = \alpha$), which are determined by the erasure correcting code used for DS. An erasure correcting code for DS is typically described in terms of the number of nodes used for storage, the download locality and the repair locality, and is defined using the notation $[m,h,r]$. In this section, we briefly describe MDS codes [20], regenerating codes [10] and LRCs [11] in the context of DS. We also connect the code parameters $[m,h,r]$ with the code parameters (n,k). In Section VII, we then evaluate the overall communication cost of DS using these three code families.

We remark that the analysis in the previous sections applies directly to MDS and regenerating codes. However, due to the specificities of LRCs, Theorem 1 needs to be slightly modified, as shown in Section VI-C below.

A. Maximum Distance Separable Codes

Assume the use of an (n,k) MDS code for DS. In this case, each storage node stores one coded symbol, hence $m=n$ and $\alpha_{\text{MDS}} = \frac{F}{k}$. Due to the MDS property, D2D repair and D2D download require to contact $r=h=k$ storage nodes. Therefore, an (n,k) MDS code in a DS context is described with the triple $[n,k,k]$. Moreover, $\beta_{\text{MDS}} = \alpha_{\text{MDS}} = \frac{F}{k}$, i.e., $\gamma_{\text{D2D}} = F$. The fact that an amount of information equal to the size of the entire file has to be retrieved to repair a single storage node is a known drawback of MDS codes [10]. The simplest MDS code is the n-replication scheme. In this case, each storage node stores the entire file, i.e., $\alpha_{\text{rep}} = F$ and $r=h=k=1$.

B. Regenerating Codes

A lower repair bandwidth γ_{D2D} (as compared to MDS codes) can be achieved by using regenerating codes [10], at the expense of increasing r [10]. Two main classes of regenerating codes are covered here, minimum storage regenerating (MSR) codes and minimum bandwidth regenerating (MBR) codes. MSR codes yield the minimum storage per node, i.e.,

α_{MSR} is minimum, while MBR codes achieve minimum D2D repair bandwidth. Regenerating codes have two repair models, *functional repair* and *exact repair* [21]. In exact repair, the lost data is regenerated exactly [21]. In functional repair, the lost data is regenerated such that the initial state of reliability in the DS system is restored [21], but the regenerated data does not need to be a replica of the lost data [21]. Here, we consider only exact repair, since it is of more practical interest [22].

An exact-repair $[m,h,r]$ MSR code in a DS system has $k = h(r-h+1)$ and $n = m(r-h+1)$, with $r = 2(h-1),\dots,m-1$ [22].[4] Hence, using (5),

$$\alpha_{\text{MSR}} = \frac{1}{m}\cdot\frac{F}{R} = \frac{F}{m}\cdot\frac{m(r-h+1)}{h(r-h+1)} = \frac{F}{h}.$$

Furthermore [22],

$$\beta_{\text{MSR}} = \frac{F}{k} = \frac{F}{h}\cdot\frac{1}{r-h+1} \leq \alpha_{\text{MSR}},$$

with equality only when $r = h$, which is only possible for $h = 1$ and $h = 2$ due to the restriction on the values for the repair locality. The repair bandwidth,

$$\gamma_{\text{D2D}} = r\beta_{\text{MSR}} = \frac{F}{h}\cdot\frac{r}{r-h+1} \leq F,$$

is minimized for $r = m-1$ [10]. We remark that the storage per node α (and hence the average download cost) for an $(m,h) \equiv [m,h,h]$ MDS code and an $[m,h,r]$ MSR code are equal.

An MBR code further reduces the repair bandwidth at the expense of increasing the storage per node. An exact-repair $[m,h,r]$ MBR code has $k = hr - \binom{h}{2}$ and $n = mr$ for $r = h,\dots,m-1$ [22]. Using (5), we have

$$\alpha_{\text{MBR}} = \frac{1}{m}\cdot\frac{F}{R} = \frac{F}{m}\cdot\frac{2mr}{h(2r-h+1)} = \frac{F}{h}\cdot\frac{2r}{2r-h+1}.$$

Furthermore [22],

$$\beta_{\text{MBR}} = \frac{F}{k} = \frac{F}{h}\cdot\frac{2}{2r-h+1} \leq \alpha_{\text{MBR}}.$$

Similar to the MSR codes, the repair bandwidth of an MBR code,

$$\gamma_{\text{D2D}} = r\beta_{\text{MBR}} = \frac{F}{h}\frac{2r}{2r-h+1} \leq F,$$

is minimized for $r = m-1$ [10].

Note that an $[m,1,r]$ regenerating code has exactly the same overall communication cost as an m-replication scheme.

C. Locally Repairable Codes

A lower repair locality r (as compared to MDS codes) is achieved by using LRCs [11]. An $[m,h,r]$ LRC has $k = rh$ and $n = m(r+1)$, where $r < h$ and $(r+1) \mid m$. Each node stores

$$\alpha_{\text{LRC}} = \frac{1}{m}\cdot\frac{F}{R} = \frac{F}{m}\cdot\frac{m(r+1)}{rh} = \frac{F}{h}\cdot\frac{r+1}{r}$$

bits. The storage nodes are arranged in $G \triangleq \frac{m}{r+1}$ disjoint repair groups with $r+1$ nodes in each group. Any single storage node can be repaired *locally* by retrieving $\gamma_{\text{D2D}} = r\beta_{\text{LRC}}$ bits from r nodes in the repair group [11]. A storage node involved in the repair process transmits all its stored data, i.e., $\beta_{\text{LRC}} = \alpha_{\text{LRC}}$, hence

$$\gamma_{\text{D2D}} = r\beta_{\text{LRC}} = \frac{F}{h}(r+1) \leq F.$$

If *local* D2D repair is not possible, repair can be carried out *globally* by retrieving $h\alpha_{\text{LRC}}$ bits from any subset of h storage nodes. Since it is necessary to distinguish between local and global repairs (as opposed to MDS and regenerating codes), the expression of the repair cost $\bar{C}_{\text{r}}$ in Theorem 1 does not apply to LRCs and needs to be modified. We denote by $m_{\text{r,l}}^{\text{D2D}}$ and $m_{\text{r,g}}^{\text{D2D}}$ the average number of nodes repaired from the storage nodes locally and globally, respectively, in one repair interval. We will also need the following definitions. Let $\boldsymbol{X} \triangleq (X_0, X_1, \dots, X_{r+1})$ be the random vector whose component X_i is the random variable giving the number of repair groups with i storage node departures in a repair interval Δ. Note that X_i takes values in $\{0,1,\dots,G\}$ and $\sum_i X_i = G$. The probability of i storage node departures in a repair group is $y_i \triangleq \binom{r+1}{i}p^{r+1-i}(1-p)^i$, where $p = e^{-\mu\Delta}$ is the probability that a storage node has not left the network during a time Δ. Let $\boldsymbol{x} \triangleq (x_0, x_1, \dots, x_{r+1})$ be a realization of $\boldsymbol{X}$ and let $\boldsymbol{y} \triangleq (y_0, y_1, \dots, y_{r+1})$. Then,

$$\mathbb{P}(\boldsymbol{X} = \boldsymbol{x}) = \sum_{\boldsymbol{x}:|\boldsymbol{x}|=G}\binom{G}{\boldsymbol{x}}\boldsymbol{y}^{\boldsymbol{x}}, \tag{24}$$

where $|\boldsymbol{x}| \triangleq \sum_i x_i$, $\binom{G}{\boldsymbol{x}} \triangleq \frac{G!}{x_0!x_1!\cdots x_{r+1}!}$ is the multinomial coefficient, and $\boldsymbol{y}^{\boldsymbol{x}} \triangleq \prod_i y_i^{x_i}$.

The repair cost for LRCs is given in the following theorem.

Theorem 7. *Consider the DS network in Section II with departure rate μ, communication costs ρ_{BS} and ρ_{D2D}, BS repair bandwidth γ_{BS}, file size F, and repair interval Δ. Furthermore, consider the use of an $[m,h,r]$ LRC with G disjoint repair groups and D2D repair bandwidth γ_{D2D}. The repair cost is given by*

$$\bar{C}_{\text{r}} = \frac{1}{F\Delta}\left(\rho_{\text{BS}}\gamma_{\text{BS}}m_{\text{r}}^{\text{BS}} + \rho_{\text{D2D}}\left(\gamma_{\text{D2D}}m_{\text{r,l}}^{\text{D2D}} + h\alpha_{\text{LRC}}m_{\text{r,g}}^{\text{D2D}}\right)\right), \tag{25}$$

where

$$\begin{aligned} m_{\text{r,l}}^{\text{D2D}} &= mp^r(1-p),\\ m_{\text{r,g}}^{\text{D2D}} &= \sum_{\boldsymbol{x}:|\boldsymbol{x}|=G}\binom{G}{\boldsymbol{x}}\boldsymbol{y}^{\boldsymbol{x}}\cdot\sum_{i=2}^{r+1}ix_i\cdot\mathbb{1}\left\{\sum_{i=1}^{r+1}ix_i \leq m-h\right\},\\ m_{\text{r}}^{\text{BS}} &= m(1-p) - m_{\text{r,l}}^{\text{D2D}} - m_{\text{r,g}}^{\text{D2D}}, \end{aligned}$$

$p = e^{-\mu\Delta}$ and $\mathbb{1}\{\cdot\}$ is an indicator function.

Proof: See Appendix D. ∎

It is easy to verify that Corollary 2 holds also for LRCs.

D. Lowest Overall Communication Cost for Instantaneous Repair

For instantaneous repair, the minimum overall communication cost is given in the following lemma.

[4] The design of linear, exact-repair MSR codes with $r < 2(h-1)$ has been proven impossible [23].

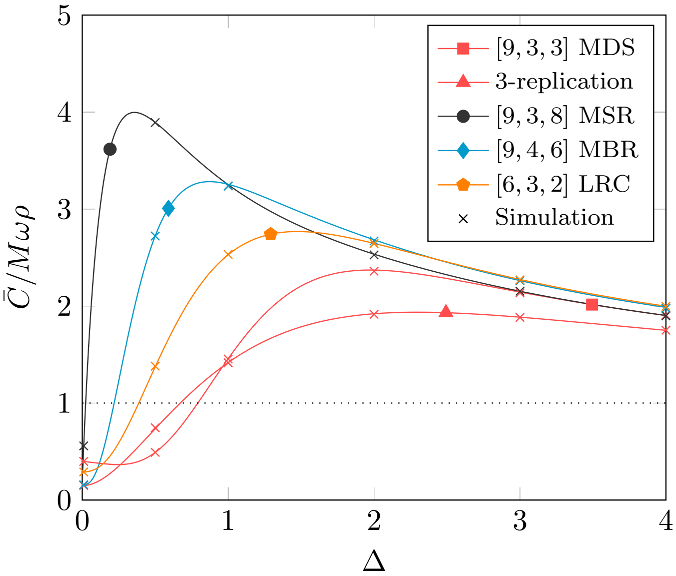

Figure 3. Normalized overall communication cost $\bar{C}/M\omega\rho$ versus the repair interval Δ for a selection of MDS codes, regenerating codes and LRCs with $R = 1/3$, compared to the normalized BS download cost (dotted line).

Lemma 1. *For $\Delta = 0$ (instantaneous repair), the lowest possible overall communication cost for any $[m, h, r]$ linear code with $m = n$, regenerating codes and LRCs is*

$$\bar{C}_{\min}(\Delta = 0) \triangleq \min_{m,h,r} \lim_{\Delta\to 0} \bar{C} = \rho_{\mathrm{D2D}}(2\mu + M\omega),$$

where $\lim_{\Delta\to 0} \bar{C}$ is given in (14) in Corollary 2. The minimum is achieved by 2-replication.

Proof: See Appendix E. ■

This is in agreement with the result in [9], where 2-replication was shown to be optimal.

VII. Numerical results

In this section, we evaluate the overall communication cost $\bar{C}$ (computed using (9) and (11)) for the erasure correcting codes discussed in the previous section. For the results, we consider a network with $M = 30$ nodes, where the number of storage nodes is $m \leq 10$. This gives a probability smaller than $7.2 \cdot 10^{-6}$ of having less than m nodes in the cell (see (4)), which is considered negligible. Without loss of generality, we set the departure rate $\mu = 1$ and $\rho_{\mathrm{D2D}} = 1$, i.e., $\rho = \rho_{\mathrm{BS}}$. Figs. 3–9 refer to a system with no incoming process, i.e., $\lambda_{\mathrm{c}} = 0$, while Figs. 10 and 11 consider the presence of an incoming process, $\lambda_{\mathrm{c}} \geq 0$.

Fig. 3 shows $\bar{C}$ normalized to the cost of downloading from the BS, $M\omega\rho$, i.e., $\bar{C}/M\omega\rho$, as a function of the normalized repair interval, $\mu\Delta = \Delta$, for a selection of MDS codes, regenerating codes and LRCs with $R = 1/3$. The ratio between the request rate and departure rate is $\omega/\mu = 0.02$, i.e., the average request rate in the cell is $M\omega = 0.6$ requests per t.u., and $\rho = 40$. The meaning of $\omega/\mu = 0.02$ is that each node places in average 0.02 requests per node life time. Also, in the figure $\Delta = 1$ means that the repair interval is equal to one average node lifetime. Simulation results[5] are also included in the figure (markers). Note that since we normalize $\bar{C}$ to the BS download cost, values below ordinate 1 correspond to the case where DS is beneficial. For relatively high repair frequencies, all codes yield lower $\bar{C}$ than BS download. However, $\bar{C}/M\omega\rho$ exceeds 1, i.e., BS download is less costly than the DS communication cost, for values of the repair interval larger than a threshold, which we define as

$$\Delta_{\max} \triangleq \sup\left\{\Delta : \bar{C} < M\omega\rho\right\}. \tag{26}$$

For $\Delta > \Delta_{\max}$, retrieving the file from the BS is always less costly, therefore storing data in the nodes is useless. $\Delta_{\max}$ depends on the network parameters M, ω, μ and ρ as well as the code parameters m, h and r.

We see from Fig. 3 that the value of Δ that minimizes $\bar{C}$, denoted by Δ_{opt}, depends on the code used for storage. In particular, $\Delta_{\mathrm{opt}} = 0$ for the $[9, 3, 8]$ MSR code, i.e., instantaneous repair is optimal. Performing an exhaustive search for $m \leq 10$, it is readily verified that the same is true for any of the codes in Section VI with $r = m - 1$. It is reasonable to assume that this will be the case also for $m > 10$. On the other hand, $\Delta_{\mathrm{opt}} > 0$ for the $[9, 3, 3]$ MDS code. Δ_{opt} depends on the network and code parameters. In particular, the tolerance to storage node departures in a repair interval affects Δ_{opt}. In Section VII-A, we investigate how the network parameters affect $\bar{C}$ and $\Delta_{\max}$. In Section VII-B, we explore how the code parameters affect $\bar{C}$.

A. Effect of Varying Network Parameters

Fig. 4 shows how $\Delta_{\max}$ increases with ρ for the same codes as in Fig. 3 and $\omega/\mu = 0.05$. For $\rho < 5$, approximately, $\Delta_{\max} = -\infty$ for all considered codes, i.e., it is never beneficial to use the devices for storage and the file should always be downloaded from the BS. It is worth noticing that, for moderate-to-large ρ, the $[9, 3, 8]$ MSR code requires in the order of 10 repairs per average node lifetime while the $[9, 3, 3]$ MDS code requires only around 0.66 repairs per node lifetime for DS to be beneficial over BS download. The main difference between the $[9, 3, 3]$ MDS code and the $[9, 3, 8]$ MSR code is the number of storage node departures in a repair interval that the code can tolerate such that D2D repair is still possible, i.e., $m - r$. The $[9, 3, 3]$ MDS code can handle the departure of up to 6 storage nodes while the $[9, 3, 8]$ MSR code can tolerate a single departure only. This explains the higher repair frequency required by the MSR code.

For the $[6, 3, 2]$ LRC and $\rho = 20$, Fig. 5 shows how $\bar{C}/M\omega\rho$ and $\Delta_{\max}$ are affected by the ratio ω/μ. We see that increasing ω/μ reduces $\bar{C}/M\omega\rho$ for all Δ and that $\Delta_{\max}$ increases with ω/μ. The same behavior is observed using any of the codes in Section VI, which can be verified by the following manipulations of the equations in Section III. The case $\omega/\mu \to \infty$ corresponds to $\bar{C}/M\omega\rho \to \bar{C}_{\mathrm{d}}/M\omega\rho$, which can be readily seen by taking the limit $\omega \to \infty$ in (13), using (9) and (11), for fixed and finite μ. This shows that the overall communication cost is essentially the download cost for a sufficiently high ω/μ. Since $\bar{C}_{\mathrm{d}}/M\omega\rho$ is monotonically increasing in Δ (Corollary 1) and $\bar{C}/M\omega\rho \to 1$ as $\Delta \to \infty$ (Corollary 2), we also have that $\Delta_{\max} \to \infty$ for $\omega/\mu \to \infty$.

[5] When simulating the wireless DS system, the repair process is not executed if the number of nodes in the cell is less than m at the particular repair instant.

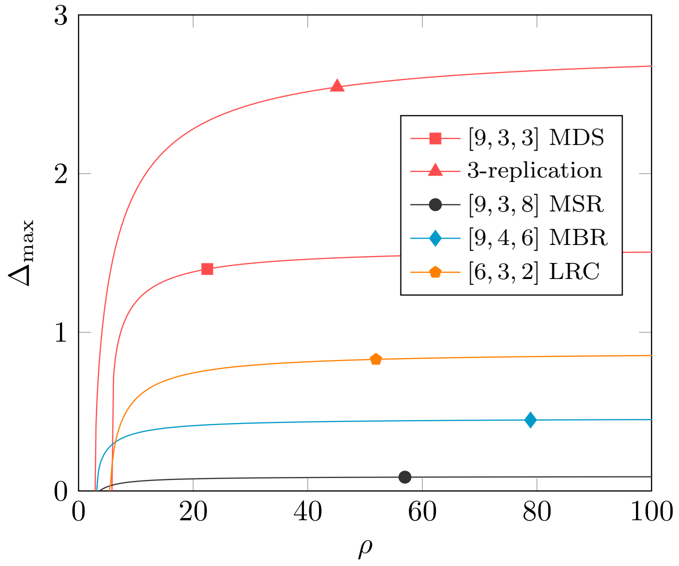

Figure 4. The maximum repair interval Δ_{max} versus the transmission cost ratio ρ.

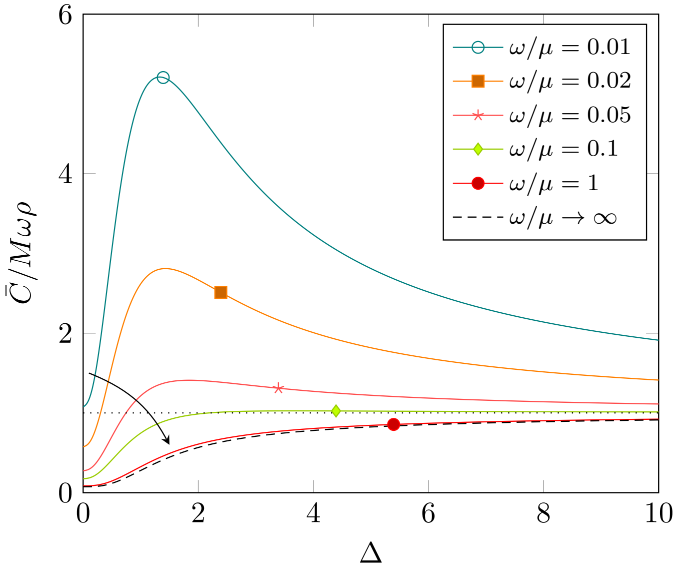

Figure 5. Normalized overall cost $\bar{C}/M\omega\rho$ versus the repair interval Δ for the $[6,3,2]$ LRC for different values of the ratio ω/μ, as compared with the normalized BS download cost (straight dotted line). The arrow points in the direction of increasing ω/μ.

Hence, DS always leads to a lower overall communication cost, as compared to the BS download cost, for sufficiently large ω/μ.

B. Results of Changing Code Parameters

We investigate how the repair locality r affects $\bar{C}$. Fig. 6 shows $\bar{C}/M\omega\rho$ versus Δ for the $[9,3,r]$ MSR code for $\rho = 40$ and $\omega/\mu = 0.02$. We observe that for $\Delta = 0$ the lowest $\bar{C}$ is achieved for $r = 8$, i.e., the highest possible repair locality. This is due to the fact that for regenerating codes γ_{D2D} is minimized for $r = m-1$ (see [10] and Section VI-B). However, increasing Δ requires decreasing r to yield the lowest $\bar{C}$. This is due to the improved tolerance to storage node departures as r decreases. The result is interesting, because it means that in wireless DS, if repairs cannot be accomplished very frequently, repair locality is a more important parameter than repair bandwidth. On the other hand, if repairs can be performed very frequently, repair bandwidth becomes more important than repair locality, because tolerance to storage node departures is not critical. In general, there is a tradeoff between the repair bandwidth and the tolerance to storage node departures (directly related to the repair locality), which holds true for any of the codes in Section VI. How to set the the parameter r depends on how frequently we can repair the DS system.

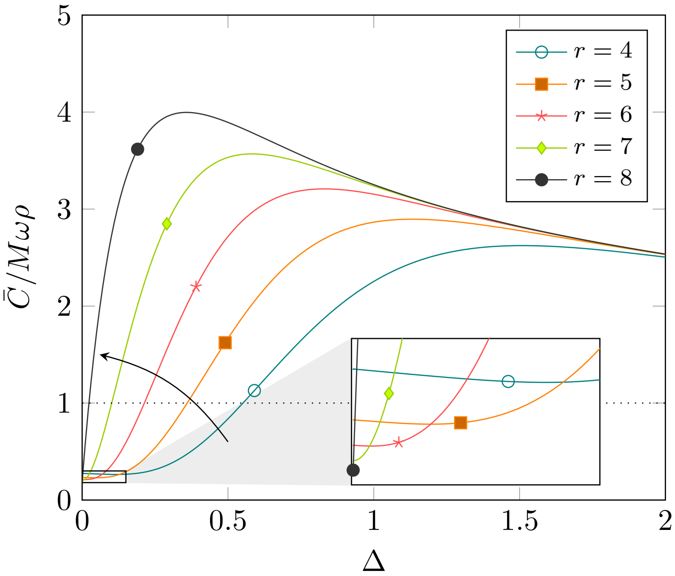

Figure 6. Normalized overall cost $\bar{C}/M\omega\rho$ versus the repair interval Δ for the $[9,3,r]$ MSR code compared with the normalized BS download cost (dotted line). The arrow shows the direction of increasing r.

C. Improved Communication Cost Using the Hybrid Scheme

We return to the hybrid repair and download scheme presented in Section IV to investigate the gains in overall communication cost as compared to the cost when using the conventional scheme. We remark that the hybrid scheme does not improve $\bar{C}$ for all codes in Section VI. In particular, for finite ρ, $\bar{C}_{\text{r}}$ is only reduced if $\beta < \alpha$ (Theorem 3) and $\bar{C}_{\text{d}}$ is only improved if $\alpha < F$ (Theorem 4). Fig. 7 shows $\bar{C}/M\omega\rho$ versus Δ for all codes in Fig. 3 that achieve lower $\bar{C}$ when using the hybrid scheme. We set $\omega/\mu = 0.1$ and $\rho = 10$ and include simulation results in the figure (markers). Dashed curves correspond to the conventional scheme, and solid curves to the hybrid scheme. We see from the figure that regenerating codes achieve a large cost reduction, especially for small Δ, when using the hybrid scheme. This is since both $\bar{C}_{\text{r}}$ and $\bar{C}_{\text{d}}$ are reduced. A smaller cost reduction is observed for MDS codes and LRCs.

D. Codes Achieving Minimum Cost for Given Δ

The analytical expressions for the overall communication cost derived in Sections III and IV can be used to find, for a given repair interval, the code achieving the lowest $\bar{C}$. We have performed an exhaustive search for all MDS codes (including replication), regenerating codes and LRCs, with $m \leq 10$, to

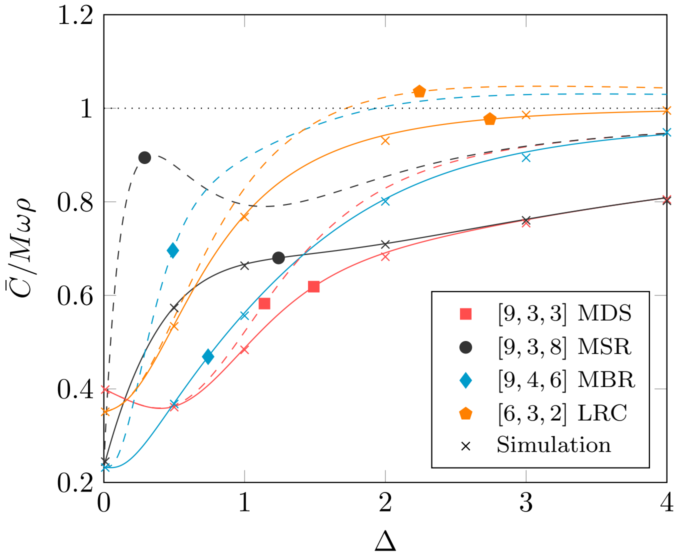

Figure 7. The normalized overall cost $\bar{C}/M\omega\rho$ versus the repair interval Δ when using the conventional scheme (dashed curves) and hybrid scheme (solid curves).

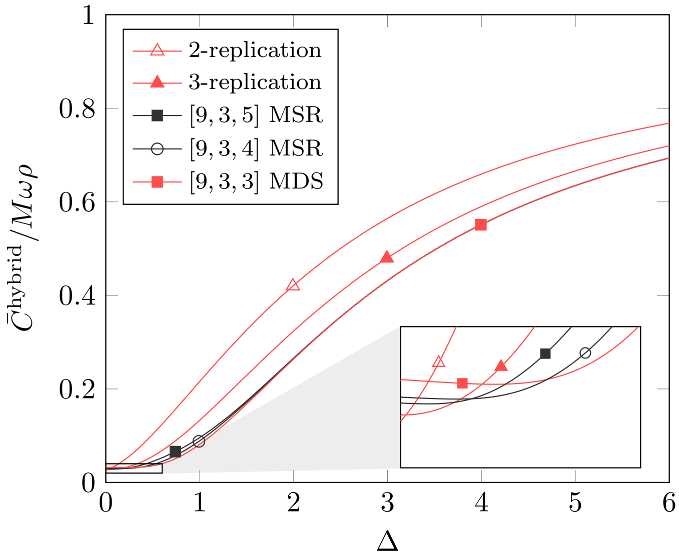

Figure 9. Codes achieving minimum $\bar{C}^{\text{hybrid}}$ with the hybrid repair and download scheme for some Δ when $\omega/\mu = 1$, $\rho = 40$, and $\Gamma = 3$.

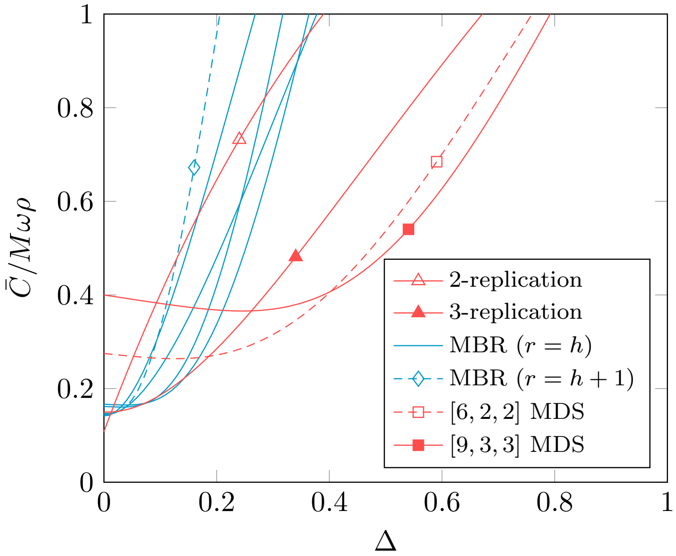

Figure 8. Codes achieving minimum $\bar{C}$ for some Δ for $\omega/\mu = 0.02$, $\rho = 40$, and $\Gamma = 3$.

find the code achieving the lowest $\bar{C}$ for each Δ. Like [15], we also introduce an overall storage budget constraint of Γ files (ΓF bits) across the nodes in the cell, i.e., $m\alpha \leq \Gamma F$. In particular, we set $\Gamma = 3$, meaning that the code rate is $R \geq 1/3$.

Fig. 8 shows $\bar{C}/M\omega\rho$ for all codes that entail the lowest $\bar{C}$ for some value of Δ for $\omega/\mu = 0.02$ and $\rho = 40$. For $\Delta = 0$ (instantaneous repair) 2-replication is optimal (see Lemma 1). However, 2-replication remains optimal only if repair can be accomplished at least around 80 times per average node lifetime. For slightly larger Δ, MBR codes achieve the lowest cost. It is worth stressing that the MBR codes achieving the lowest $\bar{C}$ for some Δ are characterized by a low repair locality ($r = h$ and $r = h + 1$), i.e., fault tolerance to storage node departures to allow D2D repair is more important than the repair bandwidth. Somewhat surprisingly, MDS codes offer the best performance for higher Δ, despite the large γ_{D2D}. We remark that LRCs are not optimal for any Δ due to the poor tolerance to storage node departures in local D2D repair and a larger α than MDS codes for a given global tolerance to storage node departures. $\Delta_{\text{max}} \approx 0.8$ is the largest Δ such that DS is beneficial over BS download, using any of the codes in Section VI.

Fig. 9 shows the codes that achieve the lowest overall cost $\bar{C}^{\text{hybrid}} = \bar{C}_{\text{r}}^{\text{hybrid}} + \bar{C}_{\text{d}}^{\text{hybrid}}$ for some values of Δ for the hybrid scheme with $\omega/\mu = 1$ and $\rho = 40$. Increasing ω/μ, $\bar{C}_{\text{d}}$ is the main contribution to $\bar{C}$ (see Section VII-A). Since α has significant impact on $\bar{C}_{\text{d}}$, we expect codes with small α to achieve the minimum cost. Indeed, we note that MDS codes and MSR codes, which have minimum α, achieve the lowest $\bar{C}$ for a region of values of Δ. As expected, 2-replication is optimal for instantaneous repair.

E. Scenario with an Incoming Process

In Fig. 10 we plot the analytical curves and simulation results for the [9, 3, 3] MDS code for the scenario with an incoming process and several values of λ_{c} when $\omega/\mu = 0.02$ and $\rho = 40$. The analytical curves for $\bar{C}_{\text{r}}$ (not shown here) match perfectly with the simulation results. However, as mentioned in Section V-B, to compute $\bar{C}_{\text{d}}$ we make the assumption that the pdf of the random variable representing the time instant at which the last of the storage nodes in a given class leaves the cell is exponentially distributed. This translates into some slight discrepancies for $\bar{C}_{\text{d}}$, which obviously show also for $\bar{C}$. However, Fig. 10 reveals a very good agreement between the analytical results and the simulation results, which justifies the assumption made. As expected, increasing λ_{c} decreases the overall communication cost, since the average lifetime of a storage node class increases. Note that $\lambda_{\text{c}} = 0$ corresponds to the case with no incoming process. For $\lambda_{\text{c}} = 0.5$ and $\lambda_{\text{c}} = 1$, where the latter corresponds to the realistic scenario where the arrival rate and departure rate of storage nodes is equal, wireless DS is beneficial for any Δ.

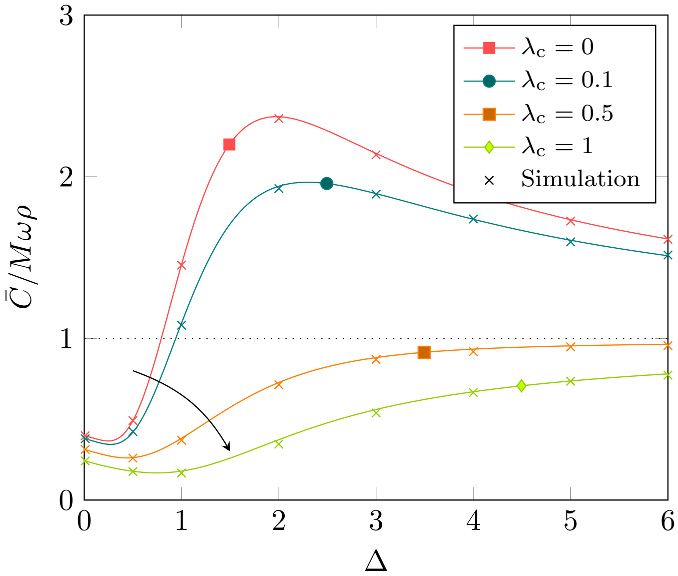

Figure 10. Normalized overall communication cost $\bar{C}/M\omega\rho$ versus the repair interval Δ for the $[9, 3, 3]$ MDS code for the scenario with an incoming process with different values of λ_c for $\mu = 1$, $\omega/\mu = 0.02$, and $\rho = 40$. The arrow shows the direction of increasing λ_c.

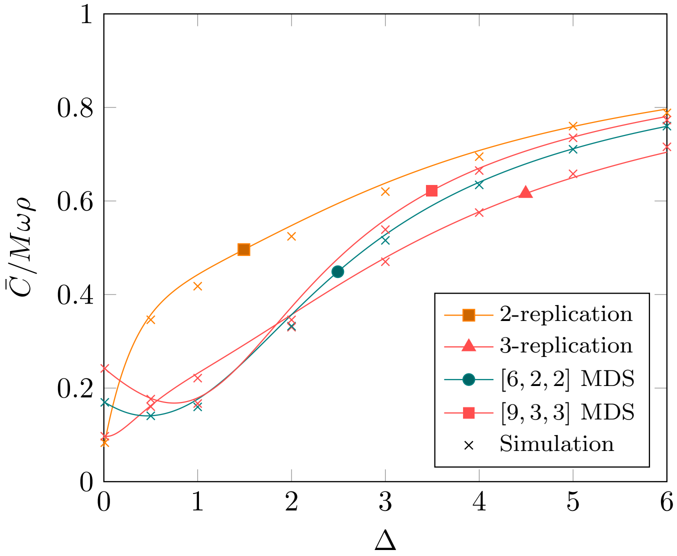

Figure 11. Codes achieving minimum $\bar{C}$ for some Δ when $\lambda_c = \mu = 1$, $\omega/\mu = 0.02$, $\rho = 40$, and $\Gamma = 3$.

Fig. 11 shows the codes that achieve lowest $\bar{C}$ for some values of Δ for the scenario with an incoming process, where $\lambda_c = \mu = 1$, $\omega/\mu = 0.02$, $\rho = 40$, and $\Gamma = 3$. DS is always beneficial, with replication and MDS codes performing the best for some Δ, while regenerating and LRC codes perform poorer. Note that the discrepancies between the analytical and simulation results, in particular for 2-replication, are due to the assumption in the computation of $\bar{C}_d$. However, the match is still very good.

VIII. Conclusions

We investigated the use of distributed storage in the mobile devices in a wireless network to reduce the communication cost of content delivery to the users. We introduced a repair scheduling where the repair of the data lost due to device departures is performed periodically. For this scenario, we derived analytical expressions for the overall communication cost, due to data download and repair, as a function of the repair interval. Using these expressions, we then investigated the performance of MDS codes, regenerating codes and LRCs.

We showed that wireless DS can reduce the overall communication cost with respect to the scenario where content is downloaded solely from the BS. However, depending on the network parameters, there may exist a maximum value of the repair interval after which retrieving the file from the BS is always less costly. Therefore, in such cases DS is useful if repairs can be performed frequently enough. The required repair frequency depends on the network parameters and the code used for storage. In the case of an incoming process of nodes storing content, the communication cost using DS can be further reduced. In this scenario, for large enough arrival rate of nodes bringing content, the use of wireless DS with D2D communication always entails a lower communication cost than downloading content only from the BS. Interestingly, MDS codes yield better performance than codes specifically designed for DS, such as regenerating codes and LRCs, if repair cannot be performed very frequently. The reason is that in this case a large tolerance to node failures and low repair locality is required.

Our analysis shows that the use of erasure correcting codes to store (cache) content in the mobile devices is promising to reduce the communication cost of content delivery in a wireless network.

One interesting extension of this work is to consider the location of the mobile devices. In this case, the communication cost can be modeled as being dependent on the transmission distance. Another interesting extension of the analysis is to consider a library of files of varying popularity. For this scenario, one may analyze the use of different erasure correcting codes for files with different popularity, and exploiting multicast opportunities [8].

Appendix A
Proof of Theorem 2

To derive p_{D2D} we first have to find the distribution of file requests within a repair interval Δ. Let W_ℓ be the time instant of the ℓth request and let $\tilde{W}_\ell \triangleq W_\ell \mod \Delta$ be the time of the ℓth request in relation to a repair interval. The pdf of $\tilde{W}_\ell$ is given by the following lemma.

Lemma 2. *The distribution of $\tilde{W}_\ell$ for $t \in [0, \Delta)$ is*

$$f_{\tilde{W}_\ell}(t) = \frac{\omega^\ell e^{-\omega t}}{(\ell-1)!} \sum_{i=0}^{\infty} (t + i\Delta)^{\ell-1} e^{-\omega\Delta i}. \tag{27}$$

Proof: W_ℓ is computed as the sum of ℓ inter-request times with pdf given by (6). Thus, W_ℓ is an Erlang distributed random variable with pdf [14, Sec. 3.4.5]

$$f_{W_\ell}(t) = \frac{\omega^\ell t^{\ell-1} e^{-\omega t}}{(\ell-1)!}, \quad t \geq 0. \tag{28}$$

The transformation $g : W_\ell \to \tilde{W}_\ell$ is given by $t = g(x)$, where

$$g(x) = x - i\Delta, \quad x \in [i\Delta, (i+1)\Delta), \; i \geq 0. \tag{29}$$

Note that $g'(x) = 1$ for $x \in (i\Delta, (i+1)\Delta)$. Moreover, $\lim_{x\to i\Delta_-} g'(x) = \lim_{x\to i\Delta_+} g'(x) = 1$ and $g'(x)$ is continuous and well defined. Let x_i be the roots of (29),

$$x_i = g^{-1}(t) = t + i\Delta, \quad t \in [0, \Delta).$$

Then, [14, Th. 4.2]

$$f_{\tilde{W}_\ell}(t) = \sum_{x_i} f_{W_\ell}(x_i) \left| \frac{1}{g'(x_i)} \right| = \sum_{i=0}^{\infty} f_{W_\ell}(t + i\Delta),$$

and (27) is obtained using (28). ■

Define $\tilde{W}_\infty \triangleq \lim_{\ell\to\infty} \tilde{W}_\ell$. We have the following result.

Lemma 3. *The distribution of $\tilde{W}_\infty$ for $t \in [0, \Delta)$ is*

$$f_{\tilde{W}_\infty}(t) = \frac{1}{\Delta},$$

and the limit is achieved exponentially fast in ℓ.

Proof: Using Lerch's transcendent [24, Sec. 25.14]

$$\Phi\left(e^{-\omega\Delta}, 1-\ell, \frac{t}{\Delta}\right) \triangleq \sum_{i=0}^{\infty} \left(\frac{t}{\Delta} + i\right)^{\ell-1} e^{-\omega\Delta i}, \quad \ell > 1,$$

the pdf of $\tilde{W}_\ell$ (Lemma 2) can be rewritten as

$$f_{\tilde{W}_\ell}(t) = \frac{(\omega\Delta)^\ell e^{-\omega t}}{\Delta \cdot (\ell-1)!} \Phi\left(e^{-\omega\Delta}, 1-\ell, \frac{t}{\Delta}\right).$$

According to [25, Cor. 4],

$$\lim_{\ell\to\infty} \frac{(\omega\Delta)^\ell}{(\ell-1)!} \Phi\left(e^{-\omega\Delta}, 1-\ell, \frac{t}{\Delta}\right) = e^{\omega t}.$$

Hence, for an infinite number of requests

$$\lim_{\ell\to\infty} f_{\tilde{W}_\ell}(t) = \frac{e^{-\omega t}}{\Delta} \lim_{\ell\to\infty} \frac{(\omega\Delta)^\ell}{(\ell-1)!} \Phi\left(e^{-\omega\Delta}, 1-\ell, \frac{t}{\Delta}\right) = \frac{1}{\Delta}.$$

Furthermore, using [25, Th. 3], as $\ell \to \infty$,

$$f_{\tilde{W}_\ell}(t) \leq \frac{1}{\Delta} + O\left(\left(\frac{\sqrt{4\pi^2 + (\omega\Delta)^2}}{\omega\Delta} \right)^{-\ell} \right), \tag{30}$$

where $\frac{\sqrt{4\pi^2+(\omega\Delta)^2}}{\omega\Delta} \geq 1$. Therefore, the convergence is exponentially fast in ℓ. ■

We proceed with the second step of the proof. Within a repair interval, the number of storage nodes $m(t)$ in the cell is described by a Poisson death process [14, Sec. 8.6]. Denote by T_i the time interval for which $m(t) = i$, $i = h, \dots, m$ (see Fig. 2 for an illustration). Note that T_i is exponentially distributed with rate $\mu_i = i\mu$, since there are i storage nodes in the cell and the departure rate per node is μ (see Section II). Denote by S_h the time instant at which $m(t)$ changes from h to $h-1$, i.e., the time after which D2D download is no longer possible. S_h can be written as

$$S_h = \sum_{i=h}^{m} T_i.$$

The pdf of S_h is given by [19, Sec. 1.3.1]

$$f_{S_h}(t) = \sum_{i=h}^{m} \frac{\mu_m \mu_{m-1} \cdots \mu_h}{\prod_{\substack{j=h \\ j\neq i}}^{m} (\mu_j - \mu_i)} e^{-\mu_i t}, \quad t \geq 0. \tag{31}$$

Note that $\mathbb{P}(S_h \geq \Delta) > 0$ for finite Δ, which implies that, with some probability, $m(t) \geq h$ for the duration of the repair interval. In this case, $p_{\text{D2D}} = 1$.

We now have all the prerequisites to derive p_{D2D}. D2D download is possible if at least h storage nodes are available in the cell. Thus,

$$p_{\text{D2D}} = \lim_{L\to\infty} \frac{1}{L} \sum_{\ell=1}^{L} \mathbb{P}\left(\tilde{W}_\ell < S_h\right).$$

From the convergence result of Lemma 3, it follows that

$$\begin{aligned} p_{\text{D2D}} = \mathbb{P}\left(\tilde{W}_\infty < S_h\right) &= \mathbb{P}\left(\tilde{W}_\infty - S_h < 0\right) \\ &= \int_{-\infty}^{0} f_{\tilde{W}_\infty - S_h}(t)\, dt, \end{aligned}$$

where [14]

$$f_{\tilde{W}_\infty - S_h}(t) = \int_{-\infty}^{\infty} f_{\tilde{W}_\infty}(t+s) f_{S_h}(s)\, ds.$$

Using the results of Lemma 3 and (31), we get after some calculation

$$\begin{aligned} p_{\text{D2D}} &= \frac{1}{\Delta} \sum_{i=h}^{m} \int_{-\infty}^{0} e^{\mu_i t} dt \left(1 - e^{-\mu_i\Delta}\right) \prod_{\substack{j=h \\ j\neq i}}^{m} \frac{j}{j-i} \\ &= \frac{1}{\Delta} \sum_{i=h}^{m} \frac{1-p_i}{\mu_i} \prod_{\substack{j=h \\ j\neq i}}^{m} \frac{j}{j-i}. \end{aligned} \tag{32}$$

By inserting (32) into (10) and using $p_{\text{D2D}} + p_{\text{BS}} = 1$, we obtain (11).

Appendix B
Proof of Corollary 2

Consider the case $\Delta \to 0$. For the repair cost (Theorem 1),

$$\begin{aligned} \lim_{\Delta\to 0} \bar{C}_{\text{r}} = \frac{1}{F} \Bigg(&\rho_{\text{BS}} \gamma_{\text{BS}} \sum_{i=0}^{r-1} (m-i) \lim_{\Delta\to 0} \frac{b_i(m,p)}{\Delta} \\ &+ \rho_{\text{D2D}} \gamma_{\text{D2D}} \sum_{i=r}^{m} (m-i) \lim_{\Delta\to 0} \frac{b_i(m,p)}{\Delta} \Bigg), \end{aligned}$$

where $b_i(m,p)$ is given in (7) and $p = e^{-\mu\Delta}$. Note that

$$\begin{aligned} &\lim_{\Delta\to 0} \frac{b_i(m,p)}{\Delta} \\ &= \binom{m}{i} \lim_{\Delta\to 0} \frac{e^{-\mu\Delta i}(1-e^{-\mu\Delta})^{m-i}}{\Delta} \\ &\overset{(a)}{=} \mu \binom{m}{i} \lim_{\Delta\to 0} e^{-\mu\Delta i} \left(1 - e^{-\mu\Delta}\right)^{m-i-1} \left(m e^{-\mu\Delta} - i\right) \\ &= \begin{cases} m\mu, & \text{if } i = m-1, \\ 0, & \text{otherwise.} \end{cases}, \end{aligned}$$

where in (a) we used l'Hôpital's rule. Hence,

$$\sum_{i=0}^{r-1} (m-i) \lim_{\Delta\to 0} \frac{b_i(m,p)}{\Delta} = 0,$$

and

$$\sum_{i=r}^{m}(m-i)\lim_{\Delta\to 0}\frac{b_i(m,p)}{\Delta} = (m-(m-1))m\mu = m\mu.$$

This implies

$$\lim_{\Delta\to 0}\bar{C}_{\text{r}} = \rho_{\text{D2D}}\gamma_{\text{D2D}}m\mu. \tag{33}$$

For the download cost (Theorem 2),

$$\begin{aligned}\lim_{\Delta\to 0}\bar{C}_{\text{d}} &= M\omega\Bigg(\rho_{\text{BS}}\\ &+\left(\rho_{\text{D2D}}\frac{h\alpha}{F}-\rho_{\text{BS}}\right)\sum_{i=h}^{m}\frac{1}{\mu_i}\lim_{\Delta\to 0}\frac{1-p_i}{\Delta}\prod_{\substack{j=h\\j\neq i}}^{m}\frac{j}{j-i}\Bigg)\\ &= M\omega\left(\rho_{\text{BS}}+\left(\rho_{\text{D2D}}\frac{h\alpha}{F}-\rho_{\text{BS}}\right)\sum_{i=h}^{m}\prod_{\substack{j=h\\j\neq i}}^{m}\frac{j}{j-i}\right). \end{aligned}\tag{34}$$

To simplify the expression, consider the function

$$f(x) = \frac{1}{\prod_{i=h}^{m}(i-x)}, \tag{35}$$

which can be expanded as the sum of partial fractions as [26, Ch. 6]

$$f(x) = \sum_{i=h}^{m}\frac{1}{(i-x)\prod_{\substack{j=h\\j\neq i}}^{m}(j-i)}. \tag{36}$$

Now, note that the sum in (34) can be expressed as

$$\sum_{i=h}^{m}\prod_{\substack{j=h\\j\neq i}}^{m}\frac{j}{j-i} = \sum_{i=h}^{m}\frac{\prod_{j=h}^{m}j}{i\prod_{\substack{j=h\\j\neq i}}^{m}(j-i)} \overset{(a)}{=} f(0)\prod_{j=h}^{m}j \overset{(b)}{=} 1,$$

where in (a) we used (36), and in (b) we used (35). Using this in (34) we obtain

$$\lim_{\Delta\to 0}\bar{C}_{\text{d}} = M\omega\rho_{\text{D2D}}\frac{h\alpha}{F}. \tag{37}$$

Finally, the expression (14) is obtained by using

$$\lim_{\Delta\to 0}\bar{C} = \lim_{\Delta\to 0}\bar{C}_{\text{r}} + \lim_{\Delta\to 0}\bar{C}_{\text{d}}.$$

Now, assume $\Delta\to\infty$. For the average repair cost (Theorem 1)

$$\begin{aligned}\lim_{\Delta\to\infty}\bar{C}_{\text{r}} = \frac{1}{F}\Bigg(&\rho_{\text{BS}}\gamma_{\text{BS}}\sum_{i=0}^{r-1}(m-i)\lim_{\Delta\to\infty}\frac{b_i(m,p)}{\Delta}\\ &+\rho_{\text{D2D}}\gamma_{\text{D2D}}\sum_{i=r}^{m}(m-i)\lim_{\Delta\to\infty}\frac{b_i(m,p)}{\Delta}\Bigg).\end{aligned}$$

Now,

$$\lim_{\Delta\to\infty}\frac{b_i(m,p)}{\Delta} = \binom{m}{i}\lim_{\Delta\to\infty}\frac{e^{-\mu\Delta i}(1-e^{-\mu\Delta})^{m-i}}{\Delta} = 0,$$

which implies $\lim_{\Delta\to\infty}\bar{C}_{\text{r}} = 0$.

For the average download cost (Theorem 2),

$$\begin{aligned}\lim_{\Delta\to\infty}\bar{C}_{\text{d}} = M\omega\Bigg[&\rho_{\text{BS}}+\left(\rho_{\text{D2D}}\frac{h\alpha}{F}-\rho_{\text{BS}}\right)\\ &\times\sum_{i=h}^{m}\frac{1}{\mu_i}\lim_{\Delta\to\infty}\frac{1-p_i}{\Delta}\prod_{\substack{j=h\\j\neq i}}^{m}\frac{j}{j-i}\Bigg],\end{aligned}$$

where $\mu_i = i\mu$, $p_i = e^{-\mu_i\Delta}$. As $\lim_{\Delta\to\infty}\frac{1-p_i}{\Delta} = 0\ \forall\ i$, then

$$\lim_{\Delta\to\infty}\bar{C}_{\text{d}} = M\omega\rho_{\text{BS}},$$

and (15) follows.

Appendix C
Proof of Theorem 4

Following the proof of Theorem 2 (Appendix A), the probability that there are $m(t) = i$ storage nodes available at the time of a request is

$$\begin{aligned}\mathsf{c}_i &\triangleq \mathbb{P}(S_{i+1} < \tilde{W}_\infty < S_i)\\ &= \mathbb{P}(\tilde{W}_\infty - S_i < 0) - \mathbb{P}(\tilde{W}_\infty - S_{i+1} < 0).\end{aligned}\tag{38}$$

The two probabilities in (38) can be obtained by replacing h with i and $i+1$ in (32),

$$\mathbb{P}(\tilde{W}_\infty - S_i < 0) = \frac{1}{\Delta}\sum_{i'=i}^{m}\frac{1-p_{i'}}{\mu_{i'}}\prod_{\substack{j=i\\j\neq i'}}^{m}\frac{j}{j-i'}$$

$$\mathbb{P}(\tilde{W}_\infty - S_{i+1} < 0) = \frac{1}{\Delta}\sum_{i'=i+1}^{m}\frac{1-p_{i'}}{\mu_{i'}}\prod_{\substack{j=i+1\\j\neq i'}}^{m}\frac{j}{j-i'}.$$

If no storage nodes are available, we always have to rely on BS download. By replacing h with 1 in (32), we get that this occurs with probability

$$p_{\text{BS}} = 1 - \frac{1}{\Delta}\sum_{i=1}^{m}\frac{1-p_i}{\mu_i}\prod_{\substack{j=1\\j\neq i}}^{m}\frac{j}{j-i}. \tag{39}$$

If $m(t)\geq h$, D2D download is performed. This occurs with probability p_{D2D}, derived in Theorem 2.

For $m(t) = i, 1\leq i\leq h-1$, the hybrid scheme will achieve a lower download cost if $\rho_{\text{BS}}F > (\rho_{\text{BS}}(h-i)+i\rho_{\text{D2D}})\alpha$, i.e., if

$$i > \frac{\rho_{\text{BS}}}{\rho_{\text{BS}}-\rho_{\text{D2D}}}\left(h-\frac{F}{\alpha}\right)\triangleq \mathsf{d}.$$

Let

$$a\triangleq\min\left\{\lfloor\mathsf{d}\rfloor, h-1\right\}.$$

For $1\leq i\leq a$, downloading F bits from the BS will give the lowest possible cost. For $a+1\leq i\leq h-1$, downloading $i\alpha$ bits through D2D communication and $(h-i)\alpha$ bits from the BS will give the lowest possible cost. The average download cost in the hybrid regime is hence

$$\begin{aligned}\bar{C}_{\text{d}}^{\text{hybrid}} = \frac{M\omega}{F}\Bigg(&\rho_{\text{BS}}Fp_{\text{BS}}+\rho_{\text{BS}}F\sum_{i=1}^{a}\mathsf{c}_i\\ &+\sum_{i=a+1}^{h-1}(\rho_{\text{BS}}(h-i)+i\rho_{\text{D2D}})\alpha\mathsf{c}_i+\rho_{\text{D2D}}h\alpha p_{\text{D2D}}\Bigg).\end{aligned}\tag{40}$$

Finally, (16) is obtained by using (32) and (39) in (40).

Appendix D
Proof of Theorem 7

Recall that a storage node can be repaired *locally* or *globally* in D2D communication. Only single node departures (within a repair group) can be repaired locally. Using (7), the average number of local D2D repairs in a repair group is

$$b_r(r+1,p) = (r+1)p^r(1-p),$$

where $p = e^{-\mu\Delta}$. Since there are $G = \frac{m}{r+1}$ disjoint repair groups, the average number of local D2D repairs per m storage nodes is

$$m_{\text{r,l}}^{\text{D2D}} = G(r+1)p^r(1-p) = mp^r(1-p).$$

This entails a cost $\rho_{\text{D2D}}\gamma_{\text{D2D}}m_{\text{r,l}}^{\text{D2D}}$ [c.u.].

We now compute the average number of global D2D repairs $m_{\text{r,g}}^{\text{D2D}}$. Let $\boldsymbol{X} = (X_0, X_1, \dots, X_{r+1})$, where $X_i \in \{0,1,\dots,G\}$, $\sum_i X_i = G$, is the random variable giving the number of repair groups with i storage node departures in a repair interval Δ. The number of global repairs is given by $\sum_{i=2}^{r+1} iX_i$, under the constraint that there are at least h storage nodes available at the time of a repair, i.e., if $\sum_{i=1}^{r+1} iX_i \le m-h$. Therefore, by averaging over all possible realizations $\boldsymbol{x} = (x_0, x_1, \dots, x_{r+1})$ of $\boldsymbol{X}$, we obtain

$$m_{\text{r,g}}^{\text{D2D}} = \sum_{\boldsymbol{x}:|\boldsymbol{x}|=G} \binom{G}{\boldsymbol{x}} \boldsymbol{y}^{\boldsymbol{x}} \cdot \sum_{i=2}^{r+1} ix_i \cdot \mathbb{1}\left\{\sum_{i=1}^{r+1} ix_i \le m-h\right\},$$

where $|\boldsymbol{x}| \triangleq \sum_i x_i$, $\binom{G}{\boldsymbol{x}} \triangleq \frac{G!}{x_0!x_1!\cdots x_{r+1}!}$, and $\boldsymbol{y}^{\boldsymbol{x}} \triangleq \prod_i y_i^{x_i}$. The communication cost associated to global D2D repairs is $\rho_{\text{D2D}}h\alpha_{\text{LRC}}m_{\text{r,g}}^{\text{D2D}}$ [c.u.].

Finally, using (7), the average total number of storage node departures in a repair interval is

$$\sum_{i=0}^{m}(m-i)b_i(m,p) = m(1-p).$$

All storage nodes that are not repaired in D2D are repaired by the BS. Therefore,

$$m_{\text{r}}^{\text{BS}} = m(1-p) - m_{\text{r,l}}^{\text{D2D}} - m_{\text{r,g}}^{\text{D2D}},$$

with communication cost $\rho_{\text{BS}}\gamma_{\text{BS}}m_{\text{r}}^{\text{BS}}$ [c.u.].

Finally, adding the three contributions $\rho_{\text{D2D}}\gamma_{\text{D2D}}m_{\text{r,l}}^{\text{D2D}}$, $\rho_{\text{D2D}}h\alpha_{\text{LRC}}m_{\text{r,g}}^{\text{D2D}}$ and $\rho_{\text{BS}}\gamma_{\text{BS}}m_{\text{r}}^{\text{BS}}$, and dividing by Δ and normalizing by F, we obtain (25).

Appendix E
Proof of Lemma 1

The overall communication cost for $\Delta = 0$ is (Corollary 2)

$$\lim_{\Delta\to 0} \bar{C} = \frac{\rho_{\text{D2D}}}{F}(\gamma_{\text{D2D}}m\mu + M\omega h\alpha). \tag{41}$$

Consider an $[m,h,r]$ linear code with $m = n$ and minimum Hamming distance $d \ge 2$. It follows that $\alpha = \frac{F}{k}$, $\beta = \alpha$, and $h \ge k$, where the equality is achieved for MDS codes. Furthermore, note that $d = m - h + 1$. Also, from [27],

$$d \le n - k - \left\lceil \frac{k}{r} \right\rceil + 2. \tag{42}$$

Using $m = n$ and the fact that $d \ge 2$ in (42), we can write

$$m \ge k + \left\lceil \frac{k}{r} \right\rceil \ge k + \frac{k}{r}.$$

Now, using this, $\gamma_{\text{D2D}} = r\beta = r\alpha$, and $\alpha = \frac{F}{k}$ in (41) we obtain

$$\begin{aligned}
\lim_{\Delta\to 0} \bar{C} &= \frac{\rho_{\text{D2D}}}{F}(\gamma_{\text{D2D}}m\mu + M\omega h\alpha) \\
&= \rho_{\text{D2D}}\left(\frac{r}{k}m\mu + M\omega\frac{h}{k}\right) \\
&\ge \rho_{\text{D2D}}\left((r+1)\mu + M\omega\frac{h}{k}\right) \\
&\ge \rho_{\text{D2D}}\left(2\mu + M\omega\right), \qquad (43)
\end{aligned}$$

where in the last inequality we used $r \ge 1$ and $h \ge k$. It is easy to verify that the lower bound in (43) is achieved by 2-replication.

Now, consider LRCs. We get

$$m\gamma_{\text{D2D}} = F\frac{m}{h}(r+1) > 2F,$$

since $h < m$ and $r \ge 1$. Also,

$$h\alpha_{\text{LRC}} = F\frac{r+1}{r} > F.$$

Inserting this into (41) gives that LRCs yield a higher overall communication cost than (43).

Consider now MBR codes. We would like to minimize $m\gamma_{\text{D2D}}$ under the constraints $m \ge 2$, $h \ge 1$, and $h < m$, for $r = m-1$. For $h = m-1$, $m\gamma_{\text{D2D}} = 2F$. For $h < m-1$, relaxing the integer constraints on m and h,

$$\frac{\partial}{\partial m}\ m\gamma_{\text{D2D}} = 4\frac{F}{h}\frac{m^2 - m(h+1) + 1}{(2m-h-1)^2} > 0.$$

Consequently, $m\gamma_{\text{D2D}}$ is minimized for $h = m-1$ and the minimum is equal to $2F$. We proceed to minimize $h\alpha_{\text{MBR}}$ for $r = m-1$ under the same constraints. For $h = 1$, we have $h\alpha_{\text{MBR}} = F$. Also, for $h > 1$,

$$\frac{\partial}{\partial h}\ h\alpha_{\text{MBR}} = 2F\frac{m-1}{(2m-h-1)^2} > 0.$$

As a result, $m\gamma_{\text{D2D}}$ and $h\alpha_{\text{MBR}}$ are jointly minimized for $m = 2$ and $h = 1$. Thus, the MBR code, which is indeed 2-replication, achieves the lower bound in (43).

We proceed to investigate the overall communication cost when $\Delta = 0$ for MSR codes. By setting $r = m-1$ we minimize γ_{D2D} with respect to r. We relax the integer constraints on m and h. By differentiating $m\gamma_{\text{D2D}}$ with respect to h and setting the derivative equal to zero, we find

$$\arg\min_h\ m\gamma_{\text{D2D}} = \frac{m}{2}.$$

Under the constraints $m \ge 2$, $h \ge 1$ and $h < m$, we have

$$\frac{\partial}{\partial m}\ m\gamma_{\text{D2D}}\Big|_{m=2h} = \frac{F}{h^2} > 0.$$

This implies that $m\gamma_{\text{D2D}}$ is minimized for $m = 2$ and $h = 1$ and that the minimum is equal to $2F$. Since $h\alpha_{\text{MSR}} = F$, $m\gamma_{\text{D2D}}$ and $h\alpha_{\text{MSR}}$ are jointly minimized for $m = 2$ and $h = 1$. Therefore, the $[2,1,1]$ MSR code, which corresponds to 2-replication, achieves the lower bound in (43). This concludes the proof.

References

[1] Cisco, "Cisco visual networking index: Global mobile data traffic forecast update, 2015-2020," White Paper, Cisco, Feb. 2016.

[2] K. Shanmugam, N. Golrezaei, A. G. Dimakis, A. F. Molisch, and G. Caire, "Femtocaching: Wireless content delivery through distributed caching helpers," *IEEE Trans. Inf. Theory*, vol. 59, no. 12, pp. 8402–8413, Dec. 2013.

[3] V. Bioglio, F. Gabry, and I. Land, "Optimizing MDS codes for caching at the edge," in *Proc. IEEE Global Commun. Conf. (GLOBECOM)*, San Diego, CA, 2015.

[4] N. Golrezaei, P. Mansourifard, A. F. Molisch, and A. G. Dimakis, "Base-station assisted device-to-device communications for high-throughput wireless video networks," *IEEE Trans. Wireless Commun.*, vol. 13, no. 7, pp. 3665–3676, Jul. 2014.

[5] M. Maddah-Ali and U. Niesen, "Fundamental limits of caching," *IEEE Trans. Inf. Theory*, vol. 60, no. 5, pp. 2856–2867, May 2014.

[6] N. Golrezaei, A. Dimakis, and A. Molisch, "Scaling behavior for device-to-device communications with distributed caching," *IEEE Trans. Inf. Theory*, vol. 60, no. 7, pp. 4286–4298, Jul. 2014.

[7] C. Yang, Z. Chen, Y. Yao, and B. Xia, "Performance analysis of wireless heterogeneous networks with pushing and caching," in *Proc. IEEE Int. Conf. Commun. (ICC)*, London, UK, Jun. 2015, pp. 2190–2195.

[8] M. Ji, G. Caire, and A. Molisch, "Fundamental limits of caching in wireless D2D networks," *IEEE Trans. Inf. Theory*, vol. 62, no. 2, pp. 849–869, Feb. 2016.

[9] J. Pääkkönen, C. Hollanti, and O. Tirkkonen, "Device-to-device data storage for mobile cellular systems," in *Proc. IEEE Globecom Workshops (GC Wkshps)*, Atlanta, GA, Dec. 2013, pp. 671–676.

[10] 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.

[11] D. Papailiopoulos and A. Dimakis, "Locally repairable codes," *IEEE Trans. Inf. Theory*, vol. 60, no. 10, pp. 5843–5855, Oct. 2014.

[12] F. Giroire, J. Monteiro, and S. Pérennes, "Peer-to-peer storage systems: A practical guideline to be lazy," in *Proc. IEEE Global Commun. Conf. (GLOBECOM)*, Miami, FL, Dec. 2010.

[13] M. Silberstein, L. Ganesh, Y. Wang, L. Alvisi, and M. Dahlin, "Lazy means smart: Reducing repair bandwidth costs in erasure-coded distributed storage," in *Proc. Int. Conf. Syst. Storage (SYSTOR)*, Haifa, Israel, 2014.

[14] S. L. Miller and D. Childers, *Probability and Random Processes*. Elsevier, 2004.

[15] D. Leong, A. G. Dimakis, and T. Ho, "Distributed storage allocations," *IEEE Trans. Inf. Theory*, vol. 58, no. 7, pp. 4733–4752, Jul. 2012.

[16] N. Golrezaei, A. Molisch, A. Dimakis, and G. Caire, "Femtocaching and device-to-device collaboration: A new architecture for wireless video distribution," *IEEE Commun. Mag.*, vol. 51, no. 4, pp. 142–149, Apr. 2013.

[17] W. J. Stewart, *Probability, Markov chains, queues, and simulation : the mathematical basis of performance modeling*. Princeton (N.J.), Oxford: Princeton University Press, 2009.

[18] A. H. Al-Mohy and N. J. Higham, "A new scaling and squaring algorithm for the matrix exponential," *SIAM J. Matrix Anal. Appl.*, vol. 31, no. 3, pp. 970–989, Aug. 2010.

[19] G. Bolch, S. Greiner, H. de Meer, and K. S. Trivedi, *Queueing Networks and Markov Chains: Modeling and Performance Evaluation with Computer Science Applications*. Wiley-Interscience, 2006.

[20] W. E. Ryan and S. Lin, *Channel Codes: Classical and Modern*. Cambridge University Press, 2009.

[21] A. 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.

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

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

[24] F. W. J. Olver, D. W. Lozier, R. F. Boisvert, and C. W. Clark, *NIST Handbook of Mathematical Functions*. Cambridge University Press, 2010.

[25] L. M. Navas, F. J. Ruiz, and J. L. Varona, "Asymptotic behavior of the lerch transcendent function," *J. Approx. Theory*, vol. 170, pp. 21–31, Jun. 2013.

[26] R. A. Adams and C. Essex, *Calculus: A Complete Course*. Pearson Addison Wesley, 2010.

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