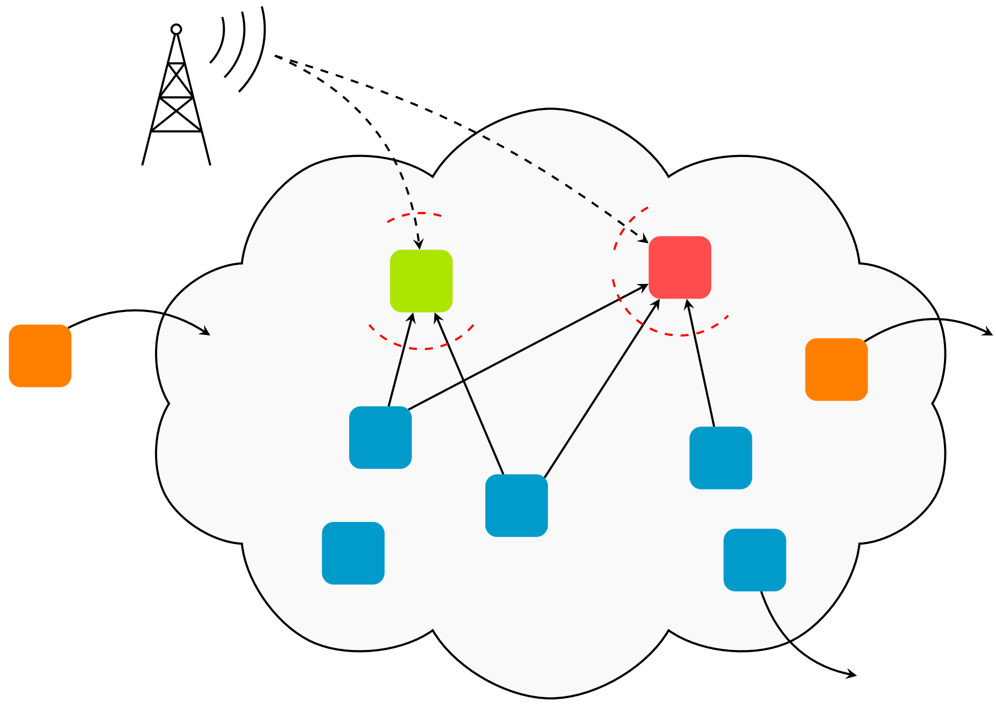
CHALMERS



Repair Scheduling in Wireless Distributed Storage with D2D Communication

JESPER PEDERSEN

Department of Signals and Systems
Chalmers University of Technology
Gothenburg, Sweden 2015

Master's thesis EX035/2015

Repair Scheduling in Wireless Distributed Storage with D2D Communication

JESPER PEDERSEN

CHALMERS

Department of Signals and Systems
Division of Communication Systems
Chalmers University of Technology
Gothenburg, Sweden 2015

Repair Scheduling in Wireless Distributed Storage with D2D Communication
JESPER PEDERSEN

Supervisor and Examiner: Alexandre Graell i Amat, Department of Signals and Systems
Co-supervisors: Iryna Andriyanova and Fredrik Brännström

Master's Thesis EX035/2015
Department of Signals and Systems
Division of Communication Systems
Chalmers University of Technology
412 96 Gothenburg
Telephone +46 31 772 1000

Cover: The mobile devices in a cell within the cellular network. The devices arrive, depart and request a file according to random processes, as described in detail in Section 2.1.

Typeset in LaTeX
Gothenburg, Sweden 2015

Abstract

We are witnessing an unprecedented increase in mobile data traffic driven by video on demand applications. The data rates achieved in the wireless link are approaching the theoretical limits and the asynchronous requests for content mostly renders traditional broadcasting useless. The methods of increasing the capacity of wireless links include increasing the spectral efficiency, increasing the spatial reuse and finding new frequency spectrum, which will all be addressed by the new 5G standard. It is questionable, however, whether these solutions alone will be enough to cope with the demand.

Storage has become an inexpensive and readily available resource. Recently, it has been proposed to exploit this fact to store content closer to the end users – a technique known as *caching*. The proposed solutions include storing content at access points or to use available storage on wireless devices such as smartphones, tablets and laptops to store content on the actual wireless devices. In both cases, content is typically stored in a distributed fashion over a number of storage devices using an erasure correcting code (ECC). This bears great similarities with distributed storage (DS) for wired systems such as data centers or peer-to-peer networks. Therefore, we will use the term wireless DS. The main difference with respect to wired DS is that content is stored to decrease the strain in the base station (BS) wireless link and increase throughput while in wired DS the main goal is to store data and maintain its availability over long periods of time.

In this work we consider a DS for a wireless network, where mobile devices arrive and depart according to a Poisson random process. Content is stored in a number of mobile devices, using an ECC. When requesting a piece of content, a user retrieves the content from the mobile devices using device-to-device communication or, if not possible, from the BS, at the expense of a higher communication cost. The repair problem when a device that stores data leaves the network is investigated. In particular, we introduce a repair scheduling, where repair is performed (from storage devices or the BS) periodically. Analytical expressions for the overall communication cost of repair and download as a function of the repair interval are derived. The analysis is illustrated by giving results for maximum distance separable codes and regenerating codes. The results indicate that DS can reduce the overall communication cost with respect to the case where content is only downloaded from the BS. The required repair frequency depends on the code used for storage and the network parameters. In particular, regenerating codes relying on many storage devices for repair require very frequent repairs. It is also shown that instantaneous repair is not the optimal repair scheduling for some code families.

Acknowledgements

I would first like to thank my supervisor Alexandre Graell i Amat for all the guidance in the beginning of my research career, encouragement when the work has been tough and forthright discussions throughout the duration of my master thesis. I would also like to thank my co-supervisors Fredrik Brännström and Iryna Andriyanova for their invaluable feedback.

I would furthermore like to express my gratitude for all the support given by my family, especially my beloved wife Lisa, without whom none of this would have been possible. Last but certainly not least, many thanks also to my friends in the master thesis room, as well as to all the PhD students in the Communications division. You help establish a very creative environment.

Jesper Pedersen, Göteborg, June 2015

Acronyms

BS	base station
CDF	cumulative distribution function
CDN	content delivery network
c.u.	cost unit
D2D	device-to-device
DS	distributed storage
ECC	erasure correcting code
i.i.d.	independent, identically distributed
MBR	minimum bandwidth regenerating
MDS	maximum distance separable
MIMO	multiple input multiple output
MSR	minimum storage regenerating
OFDM	orthogonal frequency division multiplexing
P2P	peer-to-peer
PDF	probability density function
PMF	probability mass function
t.u.	time unit

Symbols

α	storage per node [bits]
β	data transmitted from storage node during repair [bits]
Γ	storage budget [files]
γ_{BS}	repair bandwidth in BS communication [bits]
γ_{D2D}	repair bandwidth in D2D communication [bits]
Δ	repair interval [t.u.]
λ	arrival rate [t.u.$^{-1}$]
μ	departure rate [t.u.$^{-1}$]
ρ_{BS}	transmission penalty in BS communication [c.u./bit]
ρ_{D2D}	transmission penalty in D2D communication [c.u./bit]
ρ	transmission penalty ratio
ω	request rate [t.u.$^{-1}$]
C	total cost [c.u./(bit×t.u.)]
C_{d}	download cost [c.u./(bit×t.u.)]
C_{r}	repair cost [c.u./(bit×t.u.)]
d_{min}	minimum Hamming distance
k	download access
k_{c}	number of information symbols
M	file size [bits]
N	average number of nodes
n	number of coded symbols (storage nodes)
r	repair access
S_k	stopping time [t.u.]
T_{a}	node inter-arrival time [t.u.]
T_{l}	node lifetime [t.u.]
T_{r}	inter-request time [t.u.]
W_l	time of lth request [t.u.]
$\tilde{W}_l$	time of lth request in a repair interval [t.u.]

Contents

Chapter 1

Introduction

1.1 Background and Motivation

There is an ever increasing demand for high quality video content on mobile devices like smartphones, tablets and laptops. It is predicted that the global mobile data traffic will reach 24.3 exabytes per month by 2019, nearly a tenfold increase compared to the traffic in 2014 [1]. The transmission of video is responsible for approximately 55% of all mobile data traffic, where the video segment is predicted to increase to 75% by 2019. Video traffic is driven by video on demand applications like Youtube and Netflix where user requests are asynchronous, i.e., users wish to download content at different times. The implication is a shift from traditional video broadcasting to transmitting video files to each user separately. This threatens to completely congest the already burdened wireless links as current 4G systems are operating close to the theoretical limits [2]. To emphasize the seriousness of the situation, the European Network and Information Security Agency (ENISA) listed congestion of wireless links as a top ten smartphone risk, already in 2010 [3].

The methods to increase the capacity of existing wireless links include increasing the spectral efficiency, increasing the spatial reuse and to find new available frequency spectrum. For the present 4G standard, there are already technologies in place, such as multiple input multiple output (MIMO) and orthogonal frequency division multiplexing (OFDM), that achieve a spectral efficiency close to the theoretical limits [2]. For 5G, massive MIMO (where the number of antennas is scaled up by several orders of magnitude) is proposed to increase the spectral efficiency [4]. Increasing spatial reuse means using the same carrier frequency at another spatial location. The way that the new 5G standard will most likely address this is by introducing much smaller cells, i.e., a densification of base stations (BSs) [4]. A lot of research is also invested in tapping the millimeter-wave frequency spectrum, with frequencies ranging from 3 to 300 GHz [2]. It is likely, however, that these methods alone will be too costly or insufficient to cope with the demand.

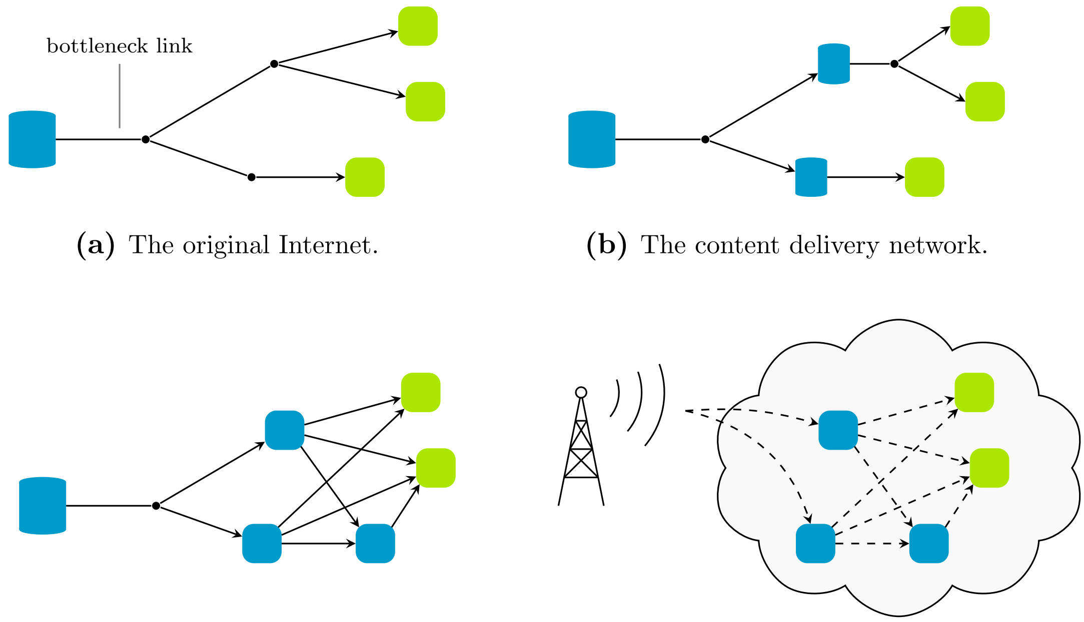
Recently, another solution to alleviate the problem of the wireless bottleneck link has been proposed. Facilitated by the availability of inexpensive storage and the knowledge that intense video traffic is caused by a few very popular video files [5], it has been suggested to store content closer to the end users – a technique known as *caching* [5–10]. The authors of [6–8] consider a central server transmitting content to a number of users, with individual storage capabilities, over a shared link. It is assumed that the link experiences periods of low and high traffic. The main idea is that during intervals of low network congestion popular content is stored by the users and it is shown that less data has to be transmitted over the shared link during periods of high traffic.

The idea presented in [9] is to deploy a number of access points (called helpers) within the cellular network and to store data across them. The helpers have a large storage capacity but low-rate wireless backhaul links to the BS. Users download content from the helpers or, if not possible, from the BS. The wireless links between helpers and users, as well as BS to users, experience delays and the BS to user links have the highest delay. Users can connect to multiple helpers and both uncoded and coded (using ideal fountain codes [11]) content placement is considered. The optimum way of assigning content to helpers to achieve the shortest total download time is analyzed in [9] and the storage of coded content is shown to outperform the case where there are no helpers, i.e., all content is downloaded from the BS. Furthermore, the coded scheme achieves a lower download delay than the uncoded scheme.

The concept in [9] was pushed further in [10], where it was suggested to store content directly in the mobile devices, taking advantage of the high storage capacity of modern smart phones and tablets. Hence, no additional infrastructure is required. [10] assumes that video files available for download in the wireless network have different popularities, as shown for wired systems in [5], which is assumed to follow a Zipf distribution. The position of mobile devices within a cell is a uniformly distributed random variable and devices that are geographically close to each other can communicate using device-to-device (D2D) communication. Traffic to the BS is alleviated by optimizing the cell size, equivalently maximizing the number of times a requested file can be retrieved from the mobile devices storing content.

The results in [6–10] were derived using information theoretical arguments. [12] presents a more practical scheme where mobile devices roam in and out of a cell within a cellular network according to a random process and request a file at random times. The mobile devices, assumed to have infinite storage capacity, are used to store copies of the file. The file can be downloaded through D2D communication as long as a device storing a copy of the file remains in the network. Since a device storing data will eventually depart from the cell, the problem of repairing the lost data is investigated for the case of instantaneous repair. Since repair is instantaneous, this system will only face single device departures and simply replicating the file and storing the copies on two mobile devices is sufficient to maintain the file in the network.

Storing content closer to the end users in a wireless network bears some similarities with the concept of content delivery networks (CDNs). Because the Internet was designed for end-to-end transmission from a single server to multiple users at the edge of the network, shown in Fig. 1.1(a), the capacity of the backhaul links could not support the required data rates. The solution was the CDN, depicted in Fig. 1.1(b), where popular content was stored at storage nodes closer to the users. This evolved into peer-to-peer (P2P) networks, visualized in Fig. 1.1(c), where content was stored at, and sent between, the actual users (peers), a solution where



(a) The original Internet.

(b) The content delivery network.

(c) The peer-to-peer network.

(d) Wireless DS with D2D communication.

Figure 1.1: The evolution of content delivery. Blue nodes store data and green nodes are users downloading content. Solid and dashed arrows indicate wired and wireless connections respectively.

both the demand and the storage capacity scales with the number of clients. An example of a P2P network is the BitTorrent technology where the storage of popular files is distributed in the sense that a user downloads content from multiple storage nodes. One of the important conditions that facilitated the transition from CDN to P2P networks was the availability of sizable memory at the user side, which is now exactly the case for wireless devices such as access points, and mobile devices. A wireless distributed storage (DS) system with D2D communication, investigated in [9, 10, 12] and visualized in Fig. 1.1(d), is a natural evolution of P2P networks.

1.2 Distributed Storage

There is a demand to store huge amounts of data. Using a single piece of hardware to store data reliably would be very costly, if not impossible, due to the sheer size of the data to be stored. The solution adopted in data centers is to store data in a distributed fashion over several inexpensive devices, called storage nodes, which form a DS system. Because the devices are prone to failures, one must provide resilience to device failures, referenced to as fault tolerance, by introducing redundancy. The simplest way, and still the most prevailing one, is to replicate data over several storage nodes. For example, consider replicating a file three times and storing a copy of the file at three storage nodes. If two nodes fail, the data can still be recovered from the last functioning node. We say that 3-replication has a fault tolerance of two failures.

A problem with replication is the large storage overhead for a given fault tolerance. This has motivated the use of erasure correcting codes (ECCs) to achieve a better fault tolerance/storage overhead tradeoff. In coding theory, an (n, k_c) ECC transform k_c information symbols into a codeword of n symbols by adding $n-k_\mathrm{c}$ parity symbols. In DS, each code symbol is typically stored in a different storage node. The storage overhead is defined as $\frac{n}{k_\mathrm{c}}$. From a coding perspective, a node failure corresponds to a symbol erasure. The ECCs providing the best tradeoff between fault tolerance and storage overhead are the maximum distance separable (MDS) codes [13]. The minimum Hamming distance, d_min, of MDS codes achieve the Singleton bound [13]

$$d_\mathrm{min} \leq n - k_\mathrm{c} + 1,$$

and it follows that an (n, k_c) MDS codeword can be decoded from any k_c coded symbols. This is referred to as the MDS property. For example, the (n, k_c) Reed-Solomon code [13] is an MDS code, widely used for providing redundancy in data transmission, that has also been applied in DS.

Facing storage node failures in DS, the initial state of reliability has to be restored by populating one or more nodes with reconstructed data, which is commonly referred to as repair. The *repair access* is the number of nodes involved in the repair of a failed node and is a critical parameter especially for data centers. The reason is that a storage node that is accessed during the repair process is not accessible for data download. If the repair access is high, many storage nodes become unavailable for download. The *repair bandwidth* is the amount of data (in bits) transmitted to repair one failed node and is one of the most significant parameters for the wireless systems introduced in Section 1.2.1. Other important parameters for DS include *download bandwidth*, which is the amount of information that needs to be downloaded to decode the file, as well as encoding and decoding complexity, defined as the number of bit level operations required to encode and decode data for download and repair.

The new requirements imposed on ECCs for DS calls for new code designs. For example, the high repair bandwidth [14] and repair complexity [15] is a known disadvantage of MDS codes. Proposed coding schemes include pyramid codes [16], local reconstruction codes [17] and locally repairable codes [18], mainly focusing on minimizing the repair access. Regenerating codes [14], zigzag codes [19] and the piggybacking framework [20] are designed to minimize the repair bandwidth.

1.2.1 Wireless Distributed Storage

Modern mobile devices like smartphones and tablets have substantial storage capacities and can be used as storage nodes. The devices are assumed to be in the cellular network, served by a BS, and can communicate through D2D communication. We refer to this setup as a wireless DS system. The fact that the wireless devices share the same transmission environment means that D2D connections can be set up with almost no investments in new infrastructure which is definitely advantageous compared to other solutions, e.g., installing storage in access points.

Like the storage nodes in a data center, the nodes in a wireless DS system are also unreliable since there is always a risk of temporary or permanent data loss due to outages or device departures. There is, however, one significant difference between the DS in data centers and the wireless DS. In data centers the reliability is crucial. If the data cannot be maintained, it is permanently lost and the coding scheme is designed to secure this reliability. In the case of wireless DS, the data can always be downloaded from the BS, although possibly at a higher transmission cost because the used bandwidth in the BS to device link is a scarce resource. The coding applied to the wireless DS is designed to increase the availability of content within the cell to offload the wireless bottleneck link.

Another important difference between the wired and wireless DS is that, in the latter scenario, devices communicate over a shared channel. Besides the importance of minimizing the used bandwidth in the BS-to-device link, minimizing the amount of data transmitted between the devices through D2D communication is equally important, hence codes focusing on minimizing the repair bandwidth [14, 19, 20] are best suited for wireless DS.

In practice, a wireless DS network storing a single file works as follows. The file, always available at the BS, is partitioned into packets and encoded onto storage nodes using an ECC. Users within the cell request and download content from storage nodes through D2D communication. If the data cannot be retrieved from the storage nodes remaining in the cell, the users download file from the BS. Because of the occurring permanent device failures, lost content must be repaired to restore the initial state of data availability within the cell. The repair can be carried out either by transmitting data from storage nodes, or if not enough data remain within the wireless DS, repair is carried out by the BS.

1.3 Aim and Outline

In this work, we consider a wireless DS scenario, similar to the one in [12]. Mobile devices roam in and out of a cell within the cellular network 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 ECC. When a user requests a piece of content, it attempts to download it from the mobile devices using D2D communication. If not possible, the content is downloaded from the BS, at the expense of a higher communication cost. The main focus is on the repair problem when a device that stores data leaves the network. In particular, we introduce a repair scheduling, where lost content is repaired (from storage devices sojourning in the cell or from the BS) at periodic times. We derive analytical expressions for the total communication cost of repair and download as a function of the repair interval. Furthermore, we analyze several ECCs, namely MDS, and regenerating codes. It is shown that DS can reduce the overall communication cost as compared to the classical scenario where content is only downloaded from the BS. Somewhat surprisingly, instantaneous repair is not the optimal repair scheduling for some code families. The work in this thesis has so far resulted in a submitted conference paper [21] and an invited talk at the IC1104 COST meeting in Novi Sad, Serbia.

The remainder of the thesis is organized as follows. We explain the arrival, departure and request processes of the wireless network, as well as the storage, delivery and repair policies, define and analyze the average repair and delivery costs and present the coding schemes considered for DS in Chapter 2. The numerical results are shown in Chapter 3 with conclusions in Chapter 4.

Chapter 2

Wireless Distributed Storage

2.1 System Model

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 process. The average number of nodes in the network is N. Nodes wish to download content from the network. For simplicity, assume that there is a single object (file), of size M bits, stored at the BS. Further assume that nodes can store data and communicate error free between them using D2D communication, that the nodes can communicate error free with the BS and that the state of the network is known by all nodes and the BS at all times where the communication overhead to get this information is neglected. The scenario is depicted in Fig. 2.1.

Arrival-departure model. There are N nodes, each arriving according to a Poisson process with exponential independent, identically distributed (i.i.d.) random inter-

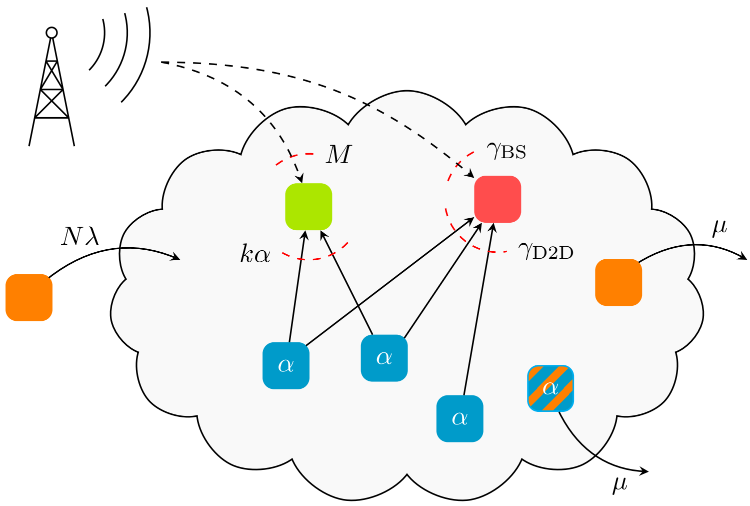


Figure 2.1: A wireless network with data storage in the mobile devices (nodes). A new node arrives to the network at rate $N\lambda$. The departure rate per node is λ. Arriving and departing nodes are marked orange. Blue nodes store exactly α bits each. The green node requests the file and downloads it from the storage nodes (solid arrows), or from the BS (dashed arrow). The repair of 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).

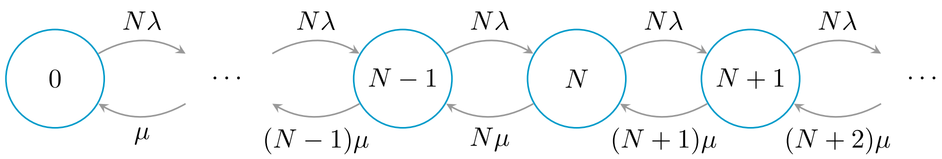


Figure 2.2: Markov chain for the M/M/∞ model used to describe the arrival-departure process.

arrival times with probability density function (PDF)

$$\lambda e^{-\lambda t}, \quad t \geq 0,$$

hence a new node arrives in the network at inter-arrival times T_{a} with PDF

$$f_{T_{\mathrm{a}}}(t) = N\lambda e^{-N\lambda t}, \quad t \geq 0,$$

where $N\lambda$ is the expected arrival rate of a node and $t \in \mathbb{R}$ is time, measured in time units (t.u.).

A node stays in the cell for an i.i.d. exponential random inter-departure time T_{l} with PDF

$$f_{T_{\mathrm{l}}}(t) = \mu e^{-\mu t}, \quad t \geq 0, \tag{2.1}$$

where μ is the expected departure rate of a node. The number of nodes in the cell can be described by an M/M/∞ queuing model and the Markov chain for such a model is shown in Fig. 2.2.

Assume that $\mu = \lambda$, i.e., the average flow of nodes in and out of the cell is the same and the average number of nodes in the cell stays constant (equal to N). The probability that there are i nodes in the cell is [22]

$$\pi(i) = \frac{N^i}{i!} e^{-N}.$$

Data storage. The file is partitioned into k_{c} packets and encoded using an (n, k_{c}) ECC of rate $R_{\mathrm{c}} = k_{\mathrm{c}}/n$. The encoded data is stored in n nodes, referred to as *storage nodes*. For simplicity, we assume $n \ll N$, hence the probability that the number of nodes in the cell is smaller than n is negligibly small, i.e.,

$$\sum_{i=0}^{n-1} \pi(i) \ll 1. \tag{2.2}$$

Therefore, the file can always be stored in the network. In particular, each storage node stores exactly α bits, i.e., we consider a symmetric allocation [23]. Hence,

$$\alpha = \frac{M}{k_{\mathrm{c}}}. \tag{2.3}$$

Like [23], we impose an overall storage budget constraint of $\Gamma M \geq 0$ bits across the nodes in the cell, i.e., $n\alpha \leq \Gamma M$. Note that to satisfy the storage budget constraint, $R_{\mathrm{c}} \geq 1/\Gamma$.

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

$$f_{T_\mathrm{r}}(t) = \omega e^{-\omega t}, \quad t \geq 0, \tag{2.4}$$

where ω is the expected request rate per node, i.e., the total expected request rate in the cell is $N\omega$. Whenever possible, the file is downloaded from the storage nodes using D2D communication, referred to as D2D download. In particular, it is assumed that data can be downloaded from any subset of k storage nodes, where $k \in \{1, \ldots, n-1\}$ and we may refer to k as the download access. In other words, D2D download is possible if k or more storage nodes remain in the cell. In this case, the amount of downloaded data, the *download bandwidth*, is $k\alpha$ bits. The parameter k depends on the properties of the ECC used for storage, and will be discussed in Section 2.3. In the case where there are less than k storage nodes in the cell, the file is downloaded from the BS, referred to as BS download. In this case, M bits are downloaded. To simplify the analysis in Section 2.2, the download bandwidth is assumed to be the same irrespective of whether the request comes from a storage node or not. This is a reasonable approximation, since $n \ll N$.

Transmission cost. It is assumed that transmission from the BS and from a storage node (in D2D communication) have different costs. 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 node, respectively, and by $\rho = \rho_\mathrm{BS}/\rho_\mathrm{D2D}$ its ratio.

2.1.1 Repair Process

When a storage node leaves the network, its stored data is lost (see blue node with orange stripes in Fig. 2.1). Therefore, another node needs to be populated with data to maintain the initial state of reliability of the DS network, i.e., n 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. In particular, we introduce a scheduled repair scheme where the repair process is launched periodically. 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 [12].

Similarly 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, γ. In particular, assume that D2D repair can be performed from any subset of r storage nodes by retrieving β 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. r is usually referred to as the repair access in the literature. In this case $\gamma_\mathrm{D2D} = r\beta$, where the subindex indicates that repair is performed from the storage nodes. If there are less than r storage nodes in the network at the moment of repair, then the repair is carried out by the BS. In this case $\gamma_\mathrm{BS} = \alpha$. It is assumed that repair always succeeds. Furthermore, for both repair and download, error-free transmission is assumed.

2.2 Analysis of Repair and Delivery Cost

In this Section, we derive analytical expressions for the repair cost, download cost and total cost as a function of the repair interval, Δ. We denote by $\mathbb{E}(\cdot)$ the expectation, or average, and by $\mathbb{E}(C_{\mathrm{r}})$, $\mathbb{E}(C_{\mathrm{d}})$ and $\mathbb{E}(C) = \mathbb{E}(C_{\mathrm{r}}) + \mathbb{E}(C_{\mathrm{d}})$ the average repair, download and total cost respectively. The cost is defined in cost units per bit and time unit [c.u./(bit×t.u.)].

2.2.1 Average Repair Cost

We begin with deriving the expression for the average repair cost. Denote by $n_{\mathrm{r}}^{\mathrm{D2D}}$ and $n_{\mathrm{r}}^{\mathrm{BS}}$ the average number of nodes repaired from the storage nodes and from the BS, respectively, in one repair interval. Also, let

$$b_i(n,p) = \binom{n}{i} p^i (1-p)^{n-i}, \quad i \in \{0, \dots, n\} \tag{2.5}$$

denote the probability mass function (PMF) of the binomial distribution with parameters n and p. We have the following lemma, describing how to calculate the average number of nodes repaired by storage nodes or by the BS.

Lemma 1.

$$n_{\mathrm{r}}^{\mathrm{D2D}} = \sum_{i=r}^{n} (n-i) b_i(n,p), \tag{2.6}$$

$$n_{\mathrm{r}}^{\mathrm{BS}} = \sum_{i=0}^{r-1} (n-i) b_i(n,p), \tag{2.7}$$

where $p = e^{-\mu\Delta}$.

Proof. The cumulative distribution function (CDF) of the node lifetime (2.1) is

$$F_{T_{\mathrm{l}}}(t) = \int_0^t f_{T_{\mathrm{l}}}(s)\, ds = -e^{-\mu s}\big|_{s=0}^{t} = 1 - e^{-\mu t}, \quad t \geq 0.$$

The probability that a storage node has not left the network during a time Δ and is accessible for repair is

$$p = \Pr(T_{\mathrm{l}} > \Delta) = 1 - F_{T_{\mathrm{l}}}(\Delta) = e^{-\mu\Delta}.$$

Hence, the probability that i storage nodes are accessible is $b_i(n,p)$ from (2.5). If i storage nodes remain in the network, then $n-i$ repairs need to be performed. D2D repair is performed if $i \geq r$; BS repair is performed otherwise. Therefore, (2.6) and (2.7) hold. ■

We are now ready to state the full expression for the average repair cost, which is given by the following theorem.

Theorem 1. *Consider the* DS network in Section 2.1 with parameters M, Δ, ρ_{BS}, γ_{BS}, ρ_{D2D}, γ_{D2D}, μ, n and r. The average repair cost is

$$\mathbb{E}\left(C_{\mathrm{r}}\right) = \frac{1}{M\Delta}\left(\rho_{\mathrm{BS}}\gamma_{\mathrm{BS}}n_{\mathrm{r}}^{\mathrm{BS}} + \rho_{\mathrm{D2D}}\gamma_{\mathrm{D2D}}n_{\mathrm{r}}^{\mathrm{D2D}}\right) \tag{2.8}$$

$$= \frac{1}{M\Delta}\left(\rho_{\mathrm{BS}}\gamma_{\mathrm{BS}}\sum_{i=0}^{r-1}(n-i)b_i(n,p) + \rho_{\mathrm{D2D}}\gamma_{\mathrm{D2D}}\sum_{i=r}^{n}(n-i)b_i(n,p)\right), \tag{2.9}$$

where $p = e^{-\mu\Delta}$.

Proof. From the system model, it follows that the cost of repairing a single storage node from the BS is $\rho_{\mathrm{BS}}\gamma_{\mathrm{BS}}$ c.u. Similarly, the cost of D2D repair of a single node is $\rho_{\mathrm{D2D}}\gamma_{\mathrm{D2D}}$ c.u. Normalizing by the file size (M bits) and the duration of the repair interval Δ, we obtain (2.8) in [c.u./(bit×t.u.)]. Finally, using Lemma 1, we obtain (2.9). ■

2.2.2 Average Download Cost

We now shift the attention to the derivation of the average download cost in the following theorem.

Theorem 2. *Consider the* DS network in Section 2.1 with parameters N, ω, M, ρ_{BS}, ρ_{D2D}, n, k, α, μ and Δ. Let $\mu_i = i\mu$, for $i \in \{k,\dots,n\}$, and $p_i = e^{-\mu_i\Delta}$. Then,

$$\mathbb{E}\left(C_{\mathrm{d}}\right) = \frac{N\omega}{M}\left(\rho_{\mathrm{BS}}M\,\Pr\{\text{BS download}\} + \rho_{\mathrm{D2D}}k\alpha\,\Pr\{\text{D2D download}\}\right) \tag{2.10}$$

$$= N\omega\left[\rho_{\mathrm{BS}} + \left(\rho_{\mathrm{D2D}}\frac{k\alpha}{M} - \rho_{\mathrm{BS}}\right)\frac{1}{\Delta}\sum_{i=k}^{n}\frac{1-p_i}{\mu_i}\prod_{\substack{j=k\\ j\neq i}}^{n}\frac{\mu_j}{(\mu_j-\mu_i)}\right], \tag{2.11}$$

where $\Pr\{\text{BS download}\} + \Pr\{\text{D2D download}\} = 1$.

A file request entails a cost $\rho_{\mathrm{D2D}}k\alpha$ with probability $\Pr\{\text{D2D download}\}$, and a cost $\rho_{\mathrm{BS}}M$ with probability $\Pr\{\text{BS download}\}$. The overall request rate per t.u. is $N\omega$. Normalizing by the file size M gives (2.10). We assume that $\Pr\{\text{BS download}\}+\Pr\{\text{D2D download}\} = 1$, implying that download always succeeds, either from storage nodes or from the BS and we can focus on obtaining $\Pr\{\text{D2D download}\}$. The derivation of $\Pr\{\text{D2D download}\}$ requires the derivation of two distributions: The distribution of the time within a repair interval such that less than k storage nodes are remaining in the cell, and the distribution of requests within a repair interval.

First we concentrate on the distribution of the time within a repair interval when the number of storage nodes goes from k to $k-1$. Within a repair interval, the number of storage nodes $n(t)$ in the cell is described by a Poisson death process [22]. Denote by T_i the time interval for which $n(t) = i$, $i \in \{k,\dots,n\}$ (see Fig. 2.3 for illustration). From (2.1), T_i is exponentially distributed with rate $\mu_i = i\mu$. This means that we expect to see increasing inter-departure times of storage nodes in

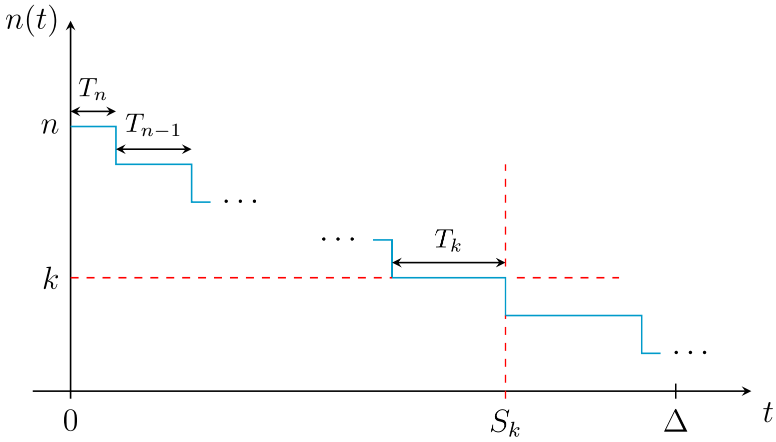


Figure 2.3: The number of available storage nodes vs. time t, within the repair interval Δ. At $t = 0$, there are n nodes available. During the intervals T_i, there are i nodes. Hence, during the time interval $t \in [0, S_k)$ there are at least k nodes available for D2D download.

average. Denote by S_k the time instant when $n(t)$ changes from k to $k-1$. We will refer to S_k as the *stopping time* of the random process. We calculate

$$S_k = \sum_{i=k}^{n} T_i.$$

It can be shown that S_k follows the hypoexponential distribution with PDF [24]

$$f_{S_k}(t) = \sum_{i=k}^{n} \frac{\mu_n \mu_{n-1} \dots \mu_k}{\prod_{\substack{j=k \\ j \neq i}}^{n} (\mu_j - \mu_i)} e^{-\mu_i t}, \quad t \geq 0. \tag{2.12}$$

This concludes the first part of finding $\Pr\{\text{D2D download}\}$ and we shift our attention to finding the distribution of the time of a request within a repair interval Δ.

Let W_l be the absolute time instant of the lth request. W_l is computed as the sum of l inter-request times with PDF given by (2.4). Thus, W_l is an Erlang-distributed random variable with PDF [22]

$$f_{W_l}(t) = \frac{\omega^l t^{l-1} e^{-\omega t}}{(l-1)!}, \quad t \geq 0. \tag{2.13}$$

The time of the lth request related to a repair interval is given by the following definition.

Definition 1. *The absolute time of the lth request in relation to a repair interval Δ is defined*

$$\tilde{W}_l \triangleq W_l \mod \Delta.$$

We can calculate the distribution of $\tilde{W}_l$ by using the following lemma.

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

$$f_{\tilde{W}_l}(t) = \sum_{i=0}^{\infty} \frac{\omega^l (t + i\Delta)^{l-1} e^{-\omega(t+i\Delta)}}{(l-1)!}. \tag{2.14}$$

Proof. See appendix A.1. ■

The result of Lemma 2 is difficult to analyze directly. If we assume that we have already seen an infinite number of requests we have the following definition.

Definition 2. *The absolute time of the* l*th request in relation to a repair interval as* $l \to \infty$ *is defined*

$$\tilde{W}_\infty \triangleq \lim_{l \to \infty} \tilde{W}_l.$$

The distribution of $\tilde{W}_\infty$ is calculated using the subsequent lemma.

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

$$f_{\tilde{W}_\infty}(t) = \frac{1}{\Delta}. \tag{2.15}$$

Proof. See appendix A.2. ■

As it turns out, the absolute request time is uniformly distributed over a repair interval if we wait an infinite amount of time. It can be verified numerically that the convergence of $\tilde{W}_l$ to the uniform distribution is fast, i.e., $\tilde{W}_l$ is uniformly distributed already for small values of l. Approximating (2.14)

$$f_{\tilde{W}_l}(t) \approx \sum_{i=0}^{10^5} \frac{\omega^l (t + i\Delta)^{l-1} e^{-\omega(t+i\Delta)}}{(l-1)!},$$

we define the maximum error of the PDF of $\tilde{W}_l$ from the PDF of the uniform distribution

$$\epsilon_{\max}(l) = \max_{t \in [0,\Delta)} \left| f_{\tilde{W}_l}(t) - \frac{1}{\Delta} \right| \approx \max_{t \in [0,\Delta)} \left| \sum_{i=0}^{10^5} \frac{\omega^l (t + i\Delta)^{l-1} e^{-\omega(t+i\Delta)}}{(l-1)!} - \frac{1}{\Delta} \right|$$

As an example, Fig. 2.4 shows the maximum error when $\mu = 50$, $\omega = 0.5$ and $\mu\Delta = 1$. We observe that the error decreases fast for increasing l. Consequently, we will approximate

$$f_{\tilde{W}_l}(t) \approx \frac{1}{\Delta} = f_{\tilde{W}_\infty}(t), \quad t \in [0, \Delta). \tag{2.16}$$

Now we know the PDFs of the stopping time (2.12) and the request time within a repair interval (2.16). D2D download is possible if at least k storage nodes are available in the network, i.e., a request comes in the time interval $t \in [0, S_k)$. Given the sequence of random variables $\{\tilde{W}_1, \tilde{W}_2, \ldots\}$,

$$\Pr\{\text{D2D download}\} = \lim_{L \to \infty} \frac{1}{L} \sum_{l=1}^{L} \Pr(\tilde{W}_l < S_k) \approx \Pr(\tilde{W}_\infty < S_k) \tag{2.17}$$

under the approximation (2.16). Using the PDFs of the stopping time (2.12) and the distribution of requests within a repair interval (2.16), the PDF of the transformation

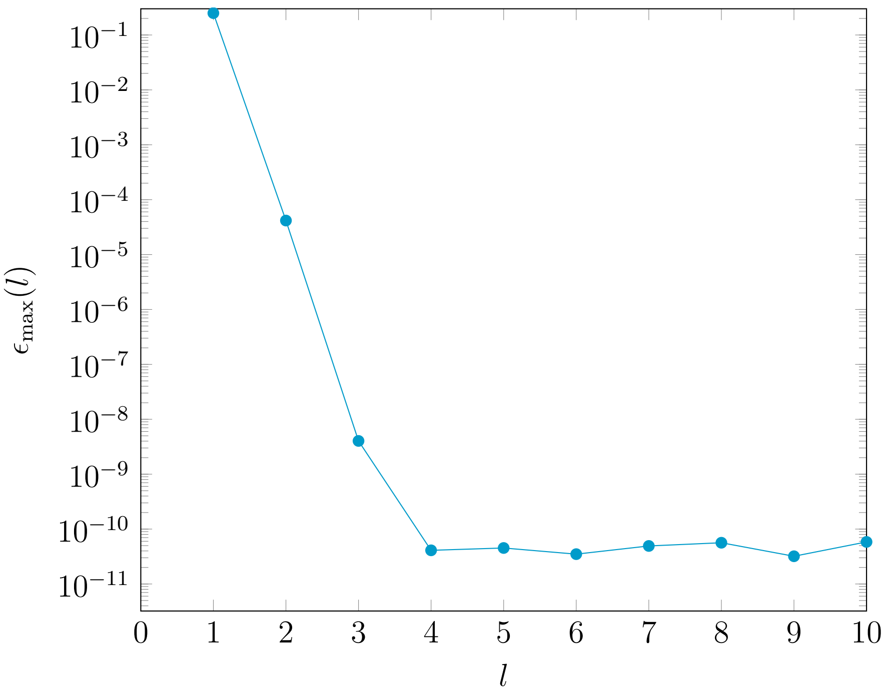


Figure 2.4: The maximum error as a function of l, assuming that the time of the lth request related to the repair interval is uniformly distributed when $\mu = 50$, $\omega = 0.5$ and $\mu\Delta = 1$.

$\tilde{W}_\infty - S_k$ is calculated [22]

$$
\begin{aligned}
f_{\tilde{W}_\infty - S_k}(t) &= \int_{-\infty}^{\infty} f_{\tilde{W}_\infty}(t+s) f_{S_k}(s)\, ds = \frac{1}{\Delta}\int_{\max(-t,0)}^{\Delta - t} \sum_{i=k}^{n} \frac{\mu_n \mu_{n-1}\cdots\mu_k}{\prod_{\substack{j=k\\ j\neq i}}^{n}(\mu_j - \mu_i)} e^{-\mu_i s}\, ds \\
&= \frac{1}{\Delta}\sum_{i=k}^{n} \frac{\mu_n \mu_{n-1}\cdots\mu_k}{\prod_{\substack{j=k\\ j\neq i}}^{n}(\mu_j - \mu_i)} \int_{\max(-t,0)}^{\Delta - t} e^{-\mu_i s}\, ds \\
&= \frac{1}{\Delta}\sum_{i=k}^{n} \left(e^{-\mu_i \max(-t,0)} - e^{-\mu_i(\Delta - t)}\right) \prod_{\substack{j=k\\ j\neq i}}^{n} \frac{\mu_j}{(\mu_j - \mu_i)}.
\end{aligned}
$$

Hence, we calculate

$$
\begin{aligned}
\Pr(\tilde{W}_\infty < S_k) = \Pr\left(\tilde{W}_\infty - S_k < 0\right) &= \int_{-\infty}^{0} f_{\tilde{W}_\infty - S_k}(t)\, dt \\
&= \frac{1}{\Delta}\sum_{i=k}^{n} \int_{-\infty}^{0} e^{\mu_i t} dt \left(1 - e^{-\mu_i \Delta}\right) \prod_{\substack{j=k\\ j\neq i}}^{n} \frac{\mu_j}{(\mu_j - \mu_i)} \\
&= \frac{1}{\Delta}\sum_{i=k}^{n} \frac{1 - p_i}{\mu_i} \prod_{\substack{j=k\\ j\neq i}}^{n} \frac{\mu_j}{(\mu_j - \mu_i)}. \qquad (2.18)
\end{aligned}
$$

Finally, using (2.18), (2.17) and $\Pr\{\text{BS download}\} = 1 - \Pr\{\text{D2D download}\}$ we obtain (2.11). Similar to the average repair cost, the average download cost is a complex function that is difficult to analyze. Despite this fact we can derive the following result.

Corollary 1. *For $\mu > 0$, $\mathbb{E}\left(C_{\mathrm{d}}\right)$ is strictly increasing if $\rho > \frac{k\alpha}{M}$, strictly decreasing if $\rho < \frac{k\alpha}{M}$ and constant otherwise.*

Proof. See appendix A.3. ∎

2.2.3 Average Total Cost

The average total cost is obtained by combining Theorems 1 and 2,

$$\begin{aligned}\mathbb{E}(C) &= \mathbb{E}\left(C_{\mathrm{r}}\right) + \mathbb{E}\left(C_{\mathrm{d}}\right)\\ &= \frac{1}{M\Delta}\left(\rho_{\mathrm{BS}}\gamma_{\mathrm{BS}}\sum_{i=0}^{r-1}(n-i)b_i(n,p) + \rho_{\mathrm{D2D}}\gamma_{\mathrm{D2D}}\sum_{i=r}^{n}(n-i)b_i(n,p)\right)\\ &\quad + N\omega\left[\rho_{\mathrm{BS}} + \left(\rho_{\mathrm{D2D}}\frac{k\alpha}{M} - \rho_{\mathrm{BS}}\right)\frac{1}{\Delta}\sum_{i=k}^{n}\frac{1-p_i}{\mu_i}\prod_{\substack{j=k\\ j\neq i}}^{n}\frac{\mu_j}{(\mu_j-\mu_i)}\right]\end{aligned} \tag{2.19}$$

where $p = e^{-\mu\Delta}$ and $p_i = e^{-i\mu\Delta}$.

The average total cost when repair is instantaneous ($\Delta \to 0$) and when lost data is never repaired ($\Delta \to \infty$) is given by the following corollary.

Corollary 2. *For instantaneous repairs*

$$\lim_{\Delta\to 0}\mathbb{E}(C) = \frac{\rho_{\mathrm{D2D}}}{M}(n\mu\gamma_{\mathrm{D2D}} + N\omega k\alpha), \tag{2.20}$$

Moreover, for $\mu > 0$

$$\lim_{\Delta\to\infty}\mathbb{E}(C) = N\omega\rho_{\mathrm{BS}}. \tag{2.21}$$

Proof. See appendix A.4. ∎

For instantaneous repair ($\Delta \to 0$), both repair and download are always performed from the storage nodes if $n > 1$. If $n = 1$ there are no storage nodes remaining in the cell to repair from since the sole storage node departs. The two terms in (2.20) correspond to the repair and download costs in the D2D regime. For $\Delta \to \infty$, data is never repaired (hence, $\mathbb{E}\left(C_{\mathrm{r}}\right) \to 0$) and, for $\mu > 0$, the number of storage nodes in the cell will become smaller than k at some point, and D2D download is not possible. Therefore, the average download cost is the average BS download cost. If $\mu = 0$, there are no departures of storage nodes from the cell and D2D repair and D2D download will always succeed.

2.3 Coding Schemes

From Section 2.2 it can be seen that the total cost, $\mathbb{E}(C)$, depends on the DS system parameters n, k, r, $\gamma_{\text{D2D}} = r\beta$, and $\gamma_{\text{BS}} = \alpha$ (among others). This section describes how, in turn, these parameters depend on the (n, k_{c}) ECCs used for storage. There is a one-to-one mapping

$$(n, k_{\text{c}}) \longleftrightarrow [n, k, r]$$

and we will use this notation interchangeably. Note that n refers to the number of coded symbols in (n, k_{c}) and the number of storage nodes in $[n, k, r]$ and here they are equal by construction. We consider as examples MDS codes [13] and regenerating codes [14].

2.3.1 Maximum Distance Separable Codes

Assume an (n, k_{c}) MDS code [13] together with a DS system with $k = k_{\text{c}}$. Then, each of the n storage nodes stores $\alpha_{\text{MDS}} = \frac{M}{k}$ bits. Due to the MDS property, D2D repair and D2D download require to contact $r = k$ storage nodes. Moreover, $\beta_{\text{MDS}} = \alpha_{\text{MDS}} = \frac{M}{k}$, which means that $\gamma_{\text{D2D}} = M$. 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 [14].

The simplest MDS code, and also the most widely used for DS in data centers, is the $(n, 1)$ MDS code, referred to as n-replication. In this case, each storage node stores the entire file, i.e., $\alpha_{\text{rep}} = M$. For replication, $r = k = 1$ and $\beta_{\text{rep}} = M$.

2.3.2 Regenerating Codes

A lower repair bandwidth γ_{D2D} (as compared to MDS codes) can be obtained by using regenerating codes, formally defined as codes achieving the optimal storage per node and repair bandwidth tradeoff [14]. The lower repair bandwidth achieved by such codes comes at the expense of increasing r [14]. Two main classes of regenerating codes are covered here, minimum storage regenerating (MSR) codes and minimum bandwidth regenerating (MBR) codes. For given n and k, MSR codes yield the best storage efficiency, i.e., α_{MSR} is minimum, while MBR codes achieve minimum D2D repair bandwidth, i.e., $\gamma_{\text{D2D}} = r\beta_{\text{MBR}}$ is minimum.

Minimum storage regenerating codes. For an (n, k_{c}) MSR code in a DS system with $k = k_{\text{c}}$, $\alpha_{\text{MSR}} = \alpha_{\text{MDS}} = \frac{M}{k}$. Hence, the download cost $\mathbb{E}(C_{\text{d}})$ for an (n, k_{c}) MSR code is equal to that of an (n, k_{c}) MDS code. Moreover, $r \in \{k, \ldots, n-1\}$ storage nodes are contacted during the D2D repair process. However, $\beta_{\text{MSR}} = \frac{M}{k}\frac{1}{r-k+1} \leq \beta_{\text{MDS}}$ [14]. $\gamma_{\text{D2D}} = r\beta_{\text{MSR}}$ is minimized for $r = n - 1$. Note that an $[n, k, r]$ MSR code with $r = k$ has the exact same performance as an (n, k_{c}) MDS code.

Recall that the file is partitioned into k_{c} packets. In a practical MSR code design, each of the k_{c} packets has to be further divided into at least 2 subpackets. As an example, consider the design of a (4,2) MSR code, with $\gamma_{\text{D2D}} = 0.75M$. The file is partitioned into $k_{\text{c}} = 2$ packets, namely $\{a, b\}$, and the two packets are further divided into sub-packets $\{a_1, a_2, b_1, b_2\}$. The repair for the example $[4, 2, 3]$ MSR code is shown in Fig. 2.5.

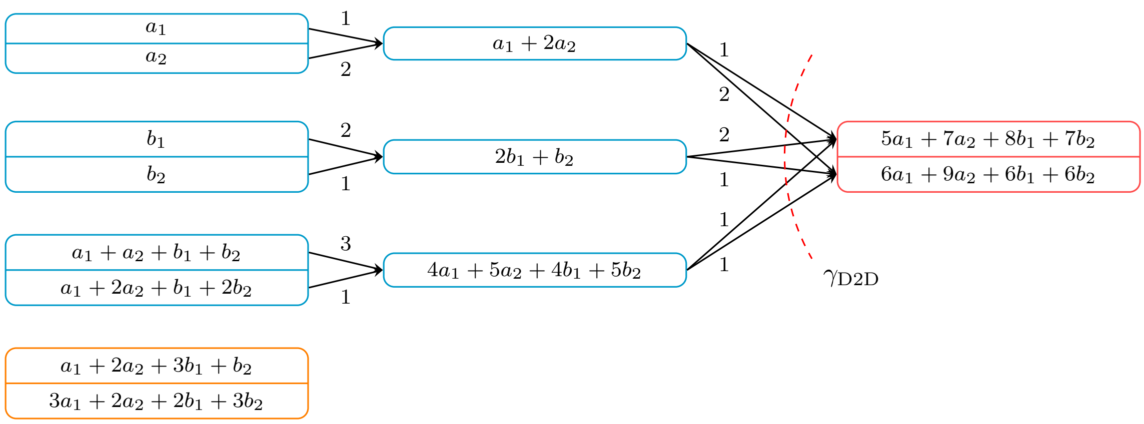


Figure 2.5: Repair for an example $[4, 2, 3]$ MSR code, achieving a repair bandwidth of $0.75M$. Linear combinations of packets are stored on blue nodes. The weights are chosen to be the same as in [14]. The orange storage node has failed. The remaining nodes create linear combinations of their stored packets and transmit to an idle node (red).

Minimum bandwidth regenerating codes. As described in [14], an (n, k_{c}) MBR code in a DS system has $r \in \{k, \ldots, n-1\}$, hence $\gamma_{\text{D2D}} = r\beta_{\text{MBR}} = \frac{M}{k}\frac{2r}{2r-k+1}$ which is minimized for $r = n-1$. Furthermore, $\gamma_{\text{D2D}} = \alpha_{\text{MBR}} = \frac{M}{k_{\text{c}}}$ [14], where the last equality comes from (2.3). The relationship between k_{c}, k and r is therefore

$$k_{\text{c}} = \frac{k(2r-k+1)}{2r},$$

with constraint $k > k_{\text{c}}$. The relation between k_{c}, k, r, α and γ_{D2D} for the MDS code as well as for the regenerating codes is presented in Table 2.1. The optimal storage and repair bandwidth tradeoff achieved by regenerating codes, derived in [14], is shown in Fig. 2.6 for $[n, k, r] = [10, 5, r]$. The $[10, 5, 5]$ MDS code is included for reference.

Table 2.1: The relation between code parameter k_{c} and network parameter k, repair access r, storage per node α and repair bandwidth γ_{D2D} for MDS and regenerating codes.

	k_{c}	r	α	γ_{D2D}
MDS	k	k	$\frac{M}{k}$	M
MSR	k	$\{k, \ldots, n-1\}$	$\frac{M}{k}$	$\frac{M}{k}\frac{r}{r-k+1}$
MBR	$k\frac{2r-k+1}{2r}$	$\{k, \ldots, n-1\}$	$\frac{M}{k}\frac{2r}{2r-k+1}$	$\frac{M}{k}\frac{2r}{2r-k+1}$

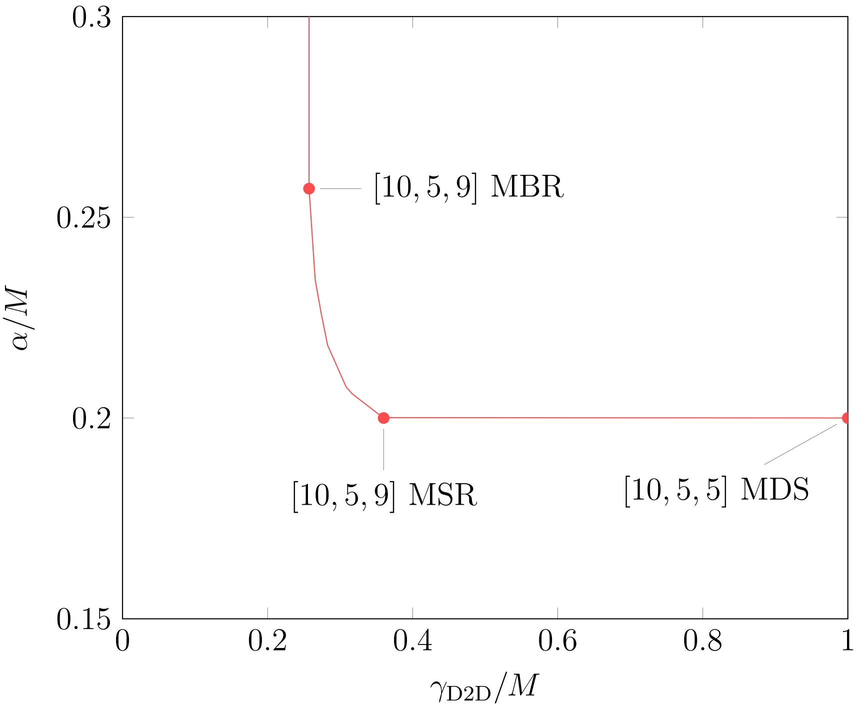


Figure 2.6: The optimal storage per node and repair bandwidth tradeoff for $[n, k, r] = [10, 5, r]$ regenerating codes compared with the $(10, 5)$ MDS code. The storage per node and the repair bandwidth is normalized by the file size M.

Chapter 3

Numerical Results

In this chapter, we evaluate the total cost $\mathbb{E}(C)$ for MDS and regenerating codes. For all results, we consider a network with $N = 100$ nodes on average with $\mu = 50$. We also assume that no more than 10 nodes can be used for storage to not violate (2.2). Without loss of generality we set $\rho_{\text{D2D}} = 1$ c.u./bit, i.e., $\rho = \rho_{\text{BS}}$.

We initially set a request rate $\omega = 0.5$ and a penalty cost ratio $\rho = 200$ to resemble a network where most nodes depart the cell without ever requesting the file. Fig. 3.1 shows the normalized cost $\mathbb{E}(C)/N\omega\rho$ versus the normalized repair interval $\mu\Delta$, for four codes: $[10, 2, 2]$ MDS code; $[10, 2, 9]$ MSR code; $[10, 2, 9]$ MBR code; and 5-replication. The code rate for all codes is $R_{\text{c}} = 1/5$, except for the MBR code that has $R_{\text{c}} \approx 0.19$. This is a modest code rate but recall that a low code rate corresponds to more data stored on storage nodes, i.e., $n\alpha \propto R_{\text{c}}^{-1}$ using

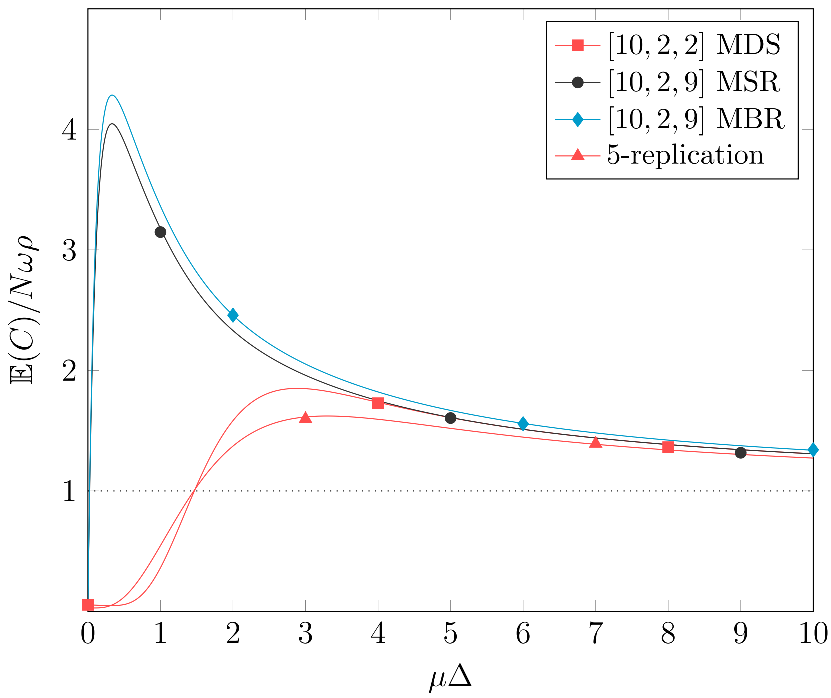


Figure 3.1: Normalized total cost $\mathbb{E}(C)/N\omega\rho$ versus the normalized repair interval $\mu\Delta$. The dotted line shows the cost of BS download. The curves correspond to the analytical expression for the total cost and the markers are simulated values.

(2.3). The term $N\omega\rho$ is the cost of BS download (2.21). In the figure, $\mu\Delta = 1$ means that the repair interval is equal to one average node lifetime. The curves correspond to the analytical expression for the total cost (2.19) and all markers are simulated values. We have simulated 10^3 requests, or 10^3 repairs, whichever value is the largest. Note that since α, β (and hence γ_{D2D}) and γ_{BS} are proportional to the file size M as specified in Section 2.3, the repair and download cost in (2.8) and (2.11), respectively, are independent of the file size M.

The following key observations can be made from Fig. 3.1: 1) For some repair interval Δ the total cost exceeds the cost of BS download; 2) the cost of the regenerating codes with high repair access, i.e., $r = n - 1$, appears to be minimized for $\Delta \to 0$, i.e., instantaneous repair; 3) for the MDS codes, the cost seem to be decreasing for small repair intervals; and 4) the $[10, 2, 2]$ MDS code is pertaining the lowest cost for some repair intervals.

3.1 Exceeding the Cost of BS Download

We begin by addressing the first observation. From Corollary 2, $\mathbb{E}(C)/N\omega\rho \to 1$ (the cost of always downloading content from the BS) when $\Delta \to \infty$. We observe from Fig. 3.1 that this is indeed the case. It is interesting to point out that the normalized total cost exceeds 1 for values of the repair interval larger than a threshold $\Delta_{\max}$. We define the maximum repair interval as

$$\Delta_{\max} \triangleq \sup\left\{\Delta : \mathbb{E}(C) < \lim_{\Delta\to\infty} \mathbb{E}(C)\right\}. \tag{3.1}$$

For $\Delta > \Delta_{\max}$, retrieving the file from the BS is always less costly, therefore

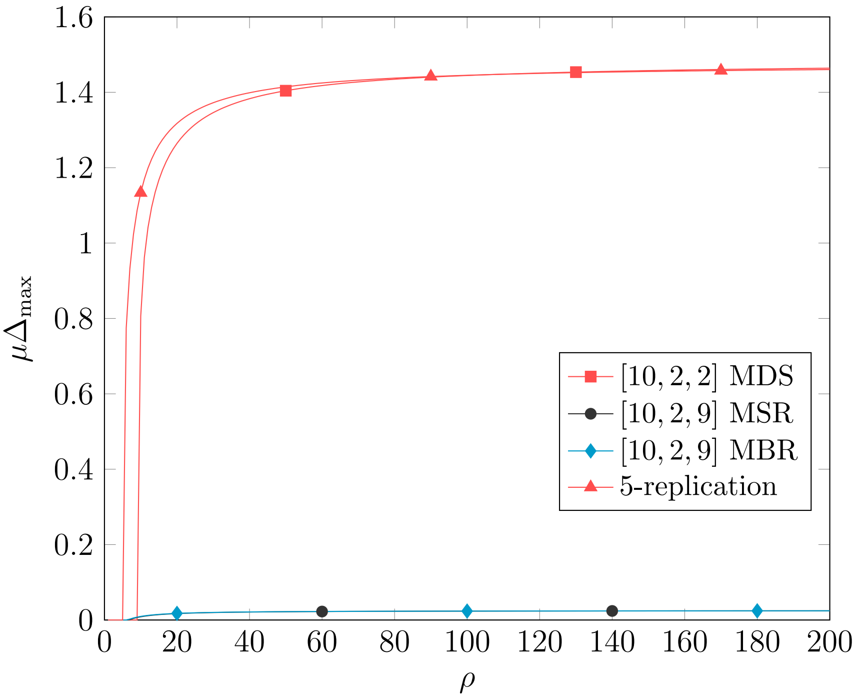


Figure 3.2: $\mu\Delta_{\max}$ as a function of the cost ratio ρ.

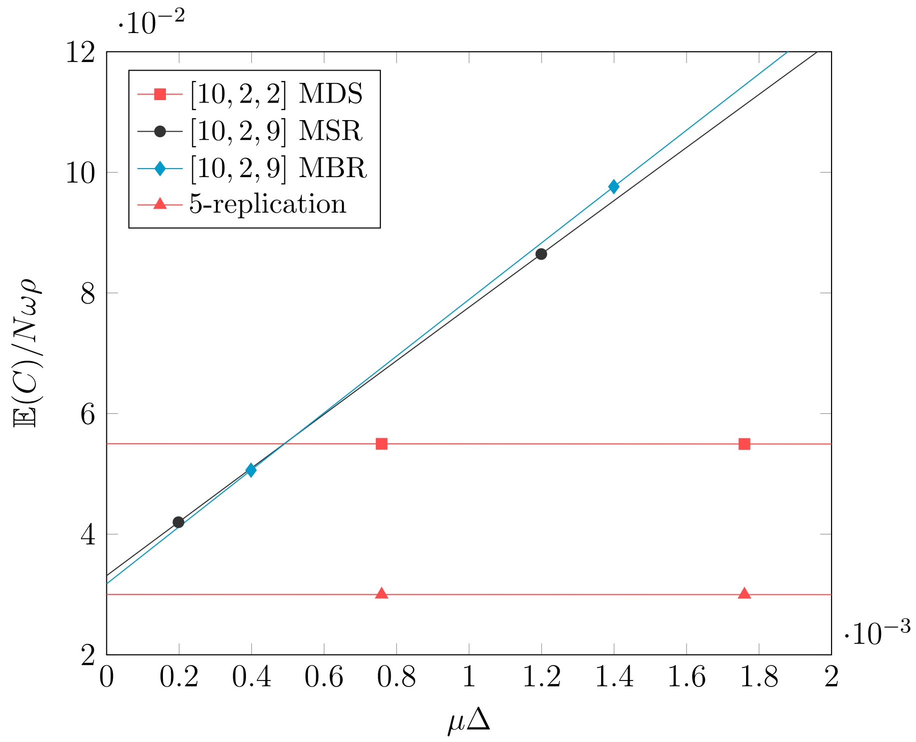


Figure 3.3: Normalized total cost $\mathbb{E}(C)/N\omega\rho$ for the same codes as in Fig. 3.1, but for much smaller normalized repair intervals $\mu\Delta$.

storing data in the nodes is useless. Clearly, $\Delta_{\max}$ is a function of the cost ratio ρ. Fig. 3.2 shows $\mu\Delta_{\max}$ as a function of $\rho \in [1, 200]$, for all codes in Fig. 3.1 when $\omega = 0.5$. We observe that if $\rho < 5$, approximately, it is never beneficial to use the considered codes for storage, i.e., the file should instead be downloaded from the BS. As ρ increases, using the considered codes for storage is beneficial, if repair is performed with $\Delta \leq \Delta_{\max}$, although the MBR code and the MSR code require very frequent repairs. The MDS codes require less frequent repairs; for large ρ, the repair interval must be at most around 1.5 average node lifetimes.

3.2 Investigating the Impact of the Code Rate

We will now focus on to the next key observation. For the same parameters and codes used in Fig. 3.1, Fig. 3.3 shows the normalized total cost for very short repair intervals. We observe that instantaneous repair is optimal for the high repair access $(r = n-1)$ regenerating codes. This highlights the fact that we should optimize the high repair access MSR and MBR codes for very short repair intervals, i.e., when $\Delta \to 0$.

As can be seen from Fig. 3.1 and Fig. 3.3, for regenerating codes with high repair access, instantaneous repair entails the lowest total cost for $\omega = 0.5$ and $\rho = 200$. Using the same network parameters we calculate the average total cost when $\Delta \to 0$ using (2.20). From Section 2.3 we already know that the repair bandwidth γ_{D2D} for regenerating codes is minimized for a high repair access, i.e., we cannot achieve a lower total cost when repair is instantaneous by decreasing the repair access. Fig. 3.4

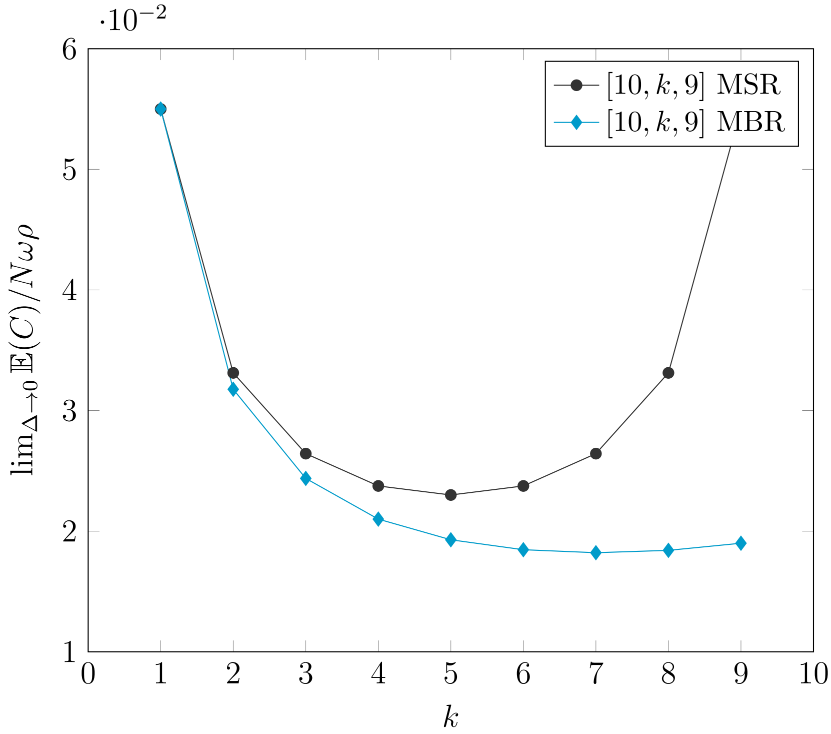


Figure 3.4: Average total cost versus k for MSR and MBR codes with high repair access ($r = n - 1$) when $n = 10$ and $\Delta \to 0$.

shows how the total cost varies with k when $\Delta \to 0$ for the MSR and MBR codes with $n = 10$.

The optimal regenerating codes for instantaneous repair appears to be the $[10, 5, 9]$ MSR code and the $[10, 7, 9]$ MBR code. It can be readily seen from (2.20) and Section 2.3 that 2-replication is the MDS code achieving the lowest total cost when $\Delta \to 0$. A comparison of the aforementioned codes is shown in Fig. 3.5 for very frequent repairs.

The 2-replication scheme is achieving a lower total cost than the considered regenerating codes for small repair intervals. Fig. 3.6 gives the complete picture, showing the average total cost for three codes: $[10, 5, 9]$ MSR code; $[10, 7, 9]$ MBR code; and 2-replication as a function of the normalized repair interval $\mu\Delta$. The $[10, 5, 5]$ MDS code, with same code rate $R_c = 1/2$ as 2-replication and the $[10, 5, 9]$ MSR code is included for reference. The $[10, 7, 9]$ MBR code has rate $R_c \approx 0.47$. It turns out that 2-replication is achieving a lower cost than the considered high repair access regenerating codes for all repair intervals.

3.3 Varying the Repair Access

Concentrating on the last two key observations, the normalized total cost $\mathbb{E}(C)/N\omega\rho$ for the MDS codes from Fig. 3.1 for moderate normalized repair intervals $\mu\Delta$ is shown in Fig. 3.7. Indeed, the average total cost for the MDS codes is minimized for some $\Delta > 0$. The reason is that the particular codes can handle multiple storage node departures. The $[10, 2, 2]$ code can recover from up to 8 storage node departures

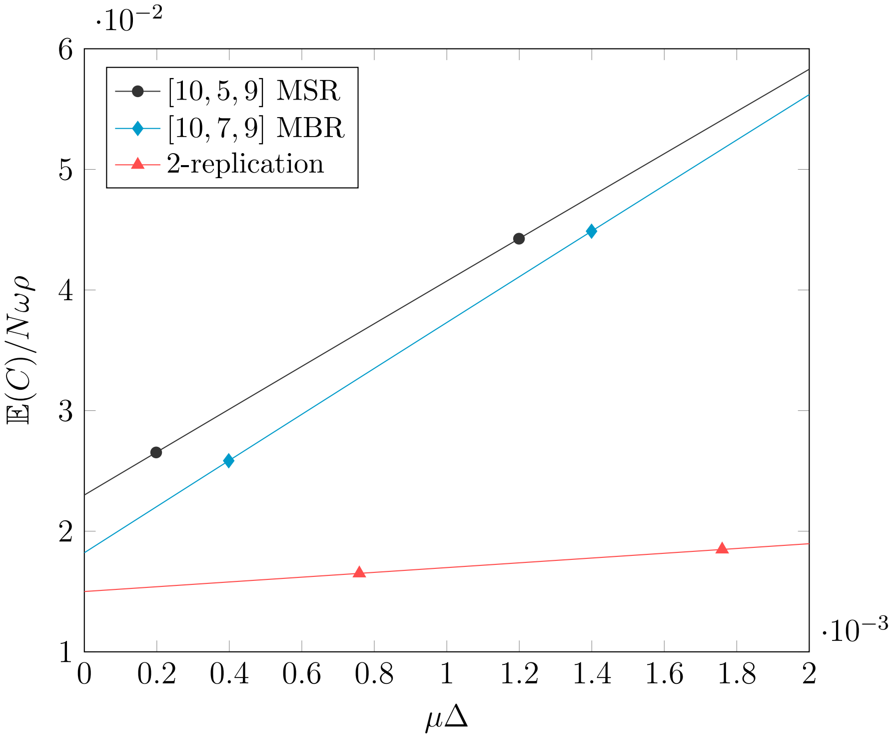


Figure 3.5: Normalized total cost $\mathbb{E}(C)/N\omega\rho$ for very short normalized repair intervals $\mu\Delta$.

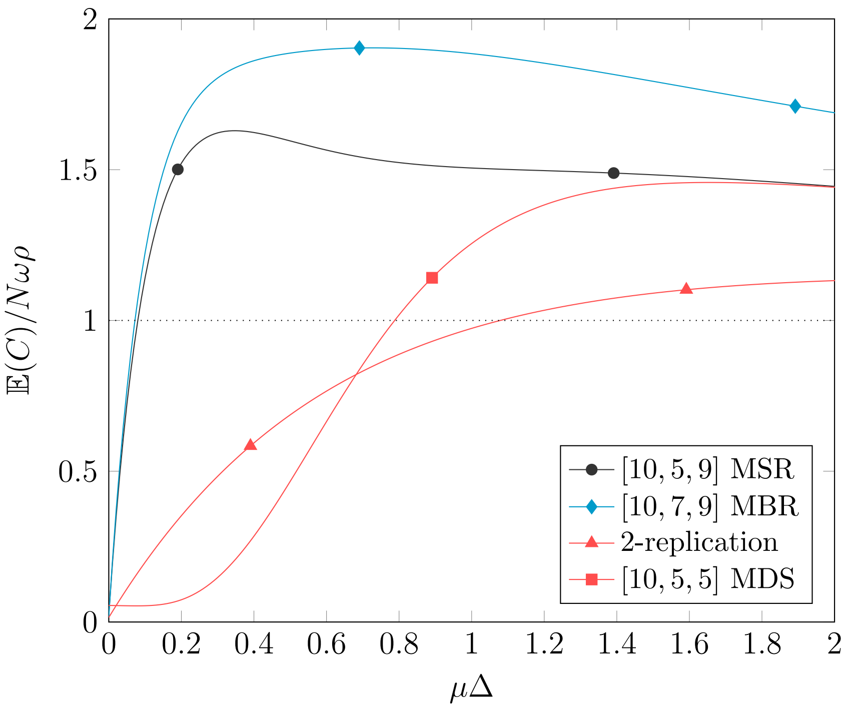


Figure 3.6: Normalized total cost $\mathbb{E}(C)/N\omega\rho$ versus the normalized repair interval $\mu\Delta$. The dotted line shows the cost of BS download.

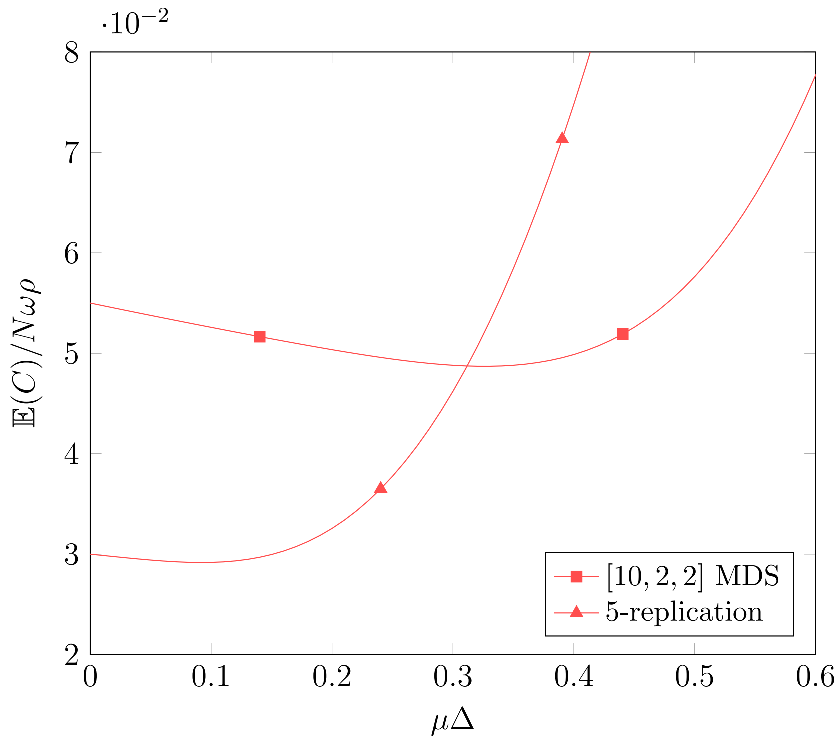


Figure 3.7: Normalized total cost $\mathbb{E}(C)/N\omega\rho$ for the same codes as in Fig. 3.1, but for shorter normalized repair intervals $\mu\Delta$.

before D2D repair and D2D download is no longer possible. By waiting to see on average 3 storage node departures, for example, the risk of BS repair and download is still reasonably low so the average total cost is roughly the same. We do however gain a little bit of node lifetime, i.e., the probability is higher that we repair onto a node that departs at a later time than what a node that we performed instantaneous repair onto would have.

The most important difference between the MDS and regenerating codes considered so far is the repair access r. Increasing the repair access lowers the repair bandwidth γ_{D2D}, as previously stated in Section 2.3, but also decreases the failure tolerance during repair. This motivates us to investigate regenerating codes with a lower repair access. We will compare with the $[10, 5, 9]$ MSR code analyzed in the previous section. As explained in Section 2.3, an $[n, k, r]$ MSR code with $r = k$ has the exact same performance as an $[n, k, r]$ MDS code. Fig. 3.8 show the total normalized cost $\mathbb{E}(C)/N\omega\rho$ for the $[10, 5, r]$ MSR code with $r \in \{5, \ldots, 9\}$. We carry the 2-replication scheme, which was shown to achieve the lowest total cost for instantaneous repair, from the preceding section for reference. The network parameters are $\omega = 0.5$ and $\rho = 200$.

We see in Fig. 3.8 that increasing the repair access results in a higher total cost for normalized repair intervals $\mu\Delta \geq 0.12$, approximately. Also seen from the figure is that there are repair intervals $\mu\Delta \in (0, 0.12)$ when increasing the repair access produces a lower cost, e.g., for $\mu\Delta = 5 \cdot 10^{-2}$ the $[10, 5, 6]$ MSR code performs better than the $[10, 5, 5]$ MSR code. We know from Section 3.2 that 2-replication attains a lower cost for very short repair intervals, i.e., as $\Delta \to 0$, and for the current network

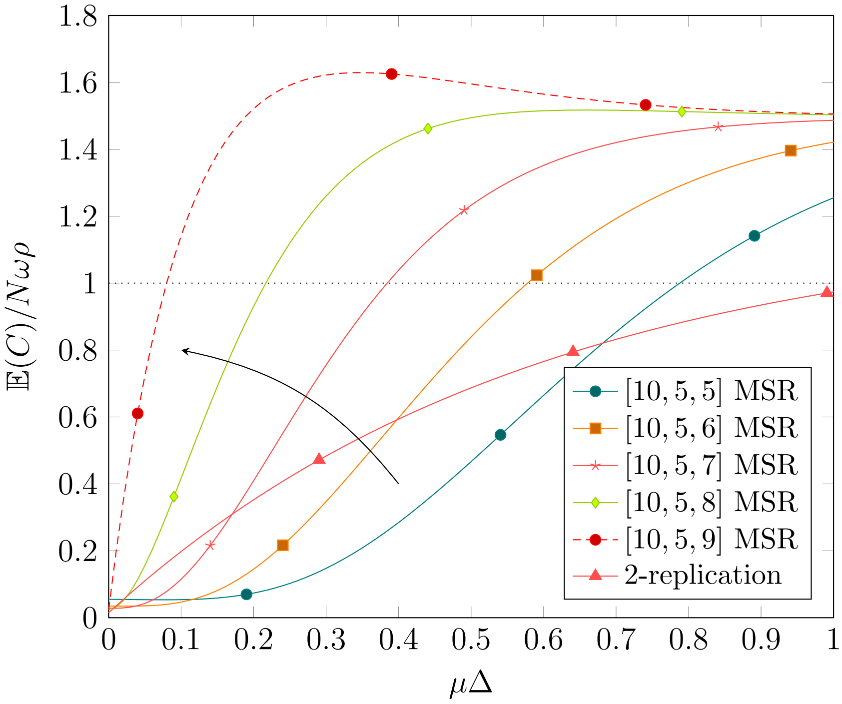


Figure 3.8: Normalized total cost $\mathbb{E}(C)/N\omega\rho$ versus the normalized repair interval $\mu\Delta$ for $[10, 5, r]$ MSR codes when the repair access $r \in \{5, \ldots, 9\}$. The arrow points in the direction of increasing r and the dotted line shows the cost of BS download.

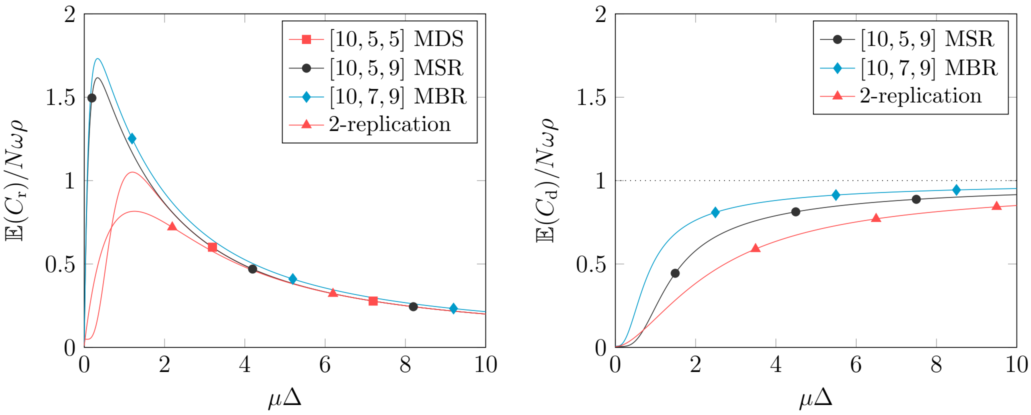
parameters. This means that a high repair access ($r = n - 1$) is never beneficial for these parameters despite the reduced repair bandwidth.

3.4 Changing the Request Rate

As can be seen from the equation describing the average total cost (2.19), increasing the request rate ω puts more emphasis on the download cost (2.10). For the same codes and network parameters as in Fig. 3.6, Fig. 3.9 shows the normalized repair cost $\mathbb{E}(C_\mathrm{r})/N\omega\rho$ and the normalized download cost $\mathbb{E}(C_\mathrm{d})/N\omega\rho$ versus the normalized repair interval $\mu\Delta$. Note that the $[10, 5, 5]$ MDS code is not included in Fig. 3.9(b) because the download cost is exactly the same as the $[10, 5, 9]$ MSR code since $\alpha_{\mathrm{MDS}} = \alpha_{\mathrm{MSR}}$ as explained in Section 2.3.

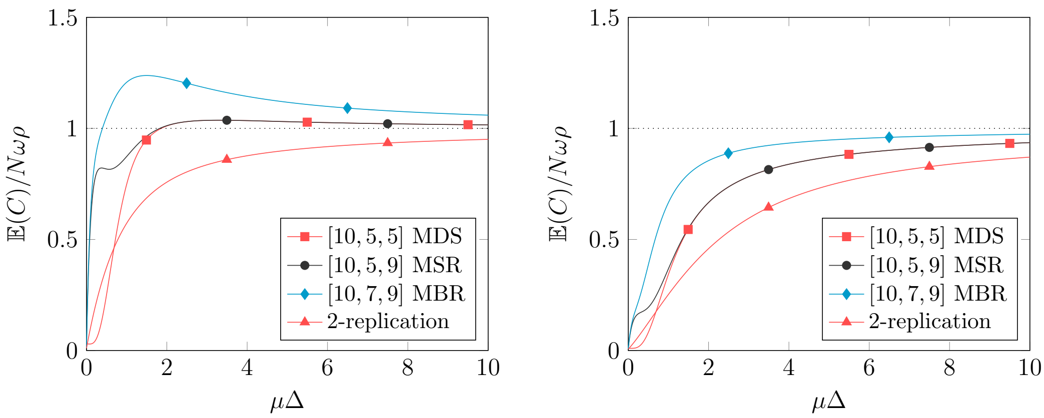
First, we see that the download cost in Fig. 3.9(b) is indeed a monotonically increasing function of Δ, as predicted by Corollary 1. Comparing Fig. 3.9(a) and Fig. 3.6 we see that the repair cost have a substantial impact on the total cost, shown in Fig. 3.6, when the request rate is set to $\omega = 0.5$. The ratio $\omega/\mu = 10^{-2}$ reflects the fact that it is more likely that a node departs the cell without ever requesting the file. Maintaining $\rho = 200$, we plot the total cost for the ratio $\omega/\mu \in \{2{\cdot}10^{-2}, 10^{-1}, 1\}$ in Fig. 3.10.

The first observation from Fig. 3.10 is that the maximum repair interval $\Delta_{\max}$ (3.1) vanishes with increasing request rate ω, i.e., with a high request rate it is always useful to store data on the nodes in the cell. The second remark is that none of the high repair access ($r = n - 1$) regenerating codes perform better than 2-replication

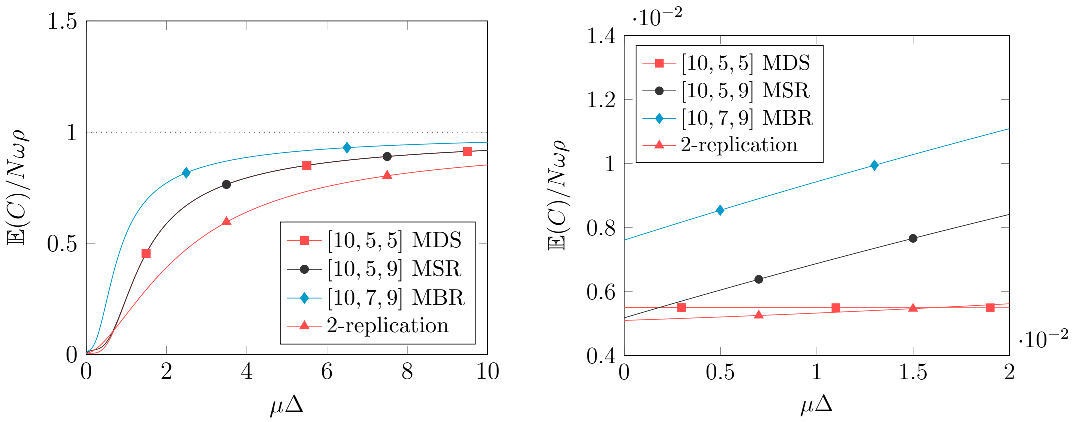


(a) Normalized repair cost. **(b)** Normalized download cost.

Figure 3.9: Normalized cost versus the normalized repair interval $\mu\Delta$. The dotted line shows the expected cost of BS download.



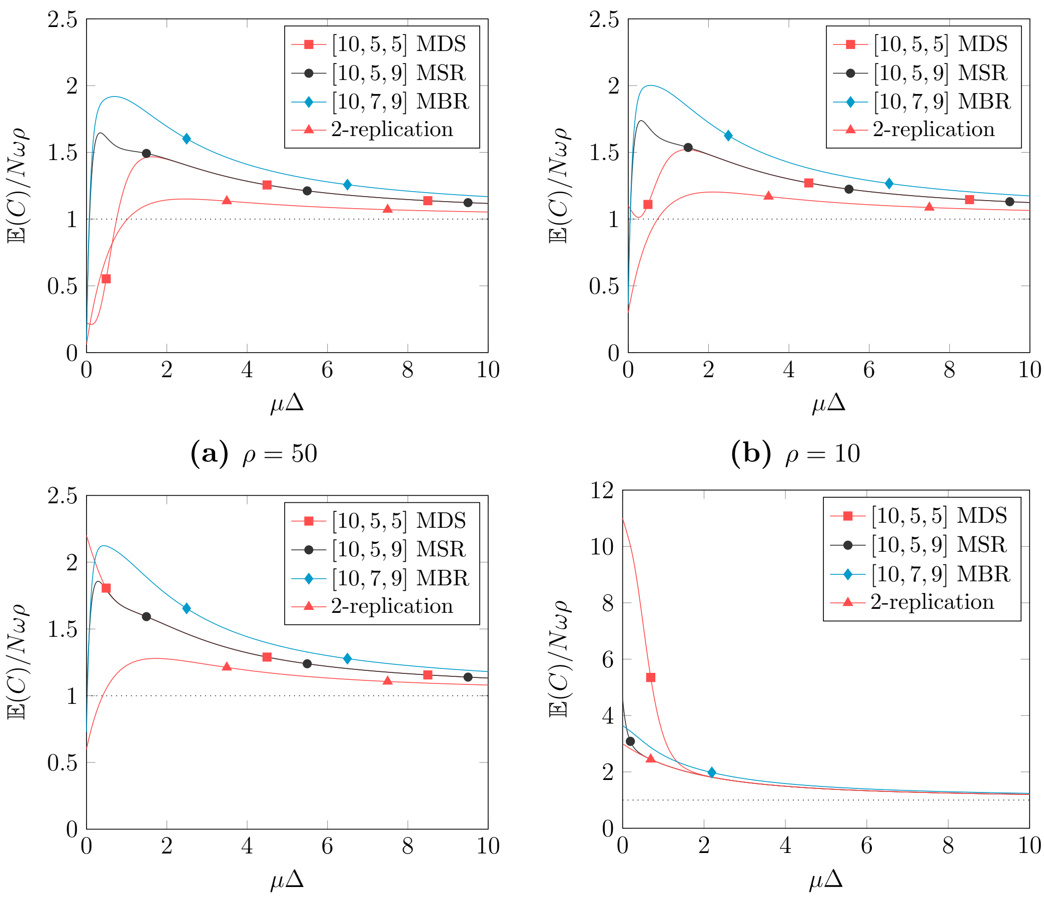
(a) $\omega/\mu = 2 \cdot 10^{-2}$ **(b)** $\omega/\mu = 10^{-1}$



(c) $\omega/\mu = 1$ **(d)** $\omega/\mu = 1$ for very short repair intervals.

Figure 3.10: Normalized total cost $\mathbb{E}(C)/N\omega\rho$ versus the normalized repair interval $\mu\Delta$ varying the ratio ω/μ. The dotted line shows the expected cost of BS download.

for any request/departure rate ratio ω/μ considered here. The performance of the $[10, 5, 9]$ MSR code will approach that of the $[10, 5, 5]$ MDS code as ω becomes



(a) $\rho = 50$

(b) $\rho = 10$

(c) $\rho = 5$

(d) $\rho = 1$

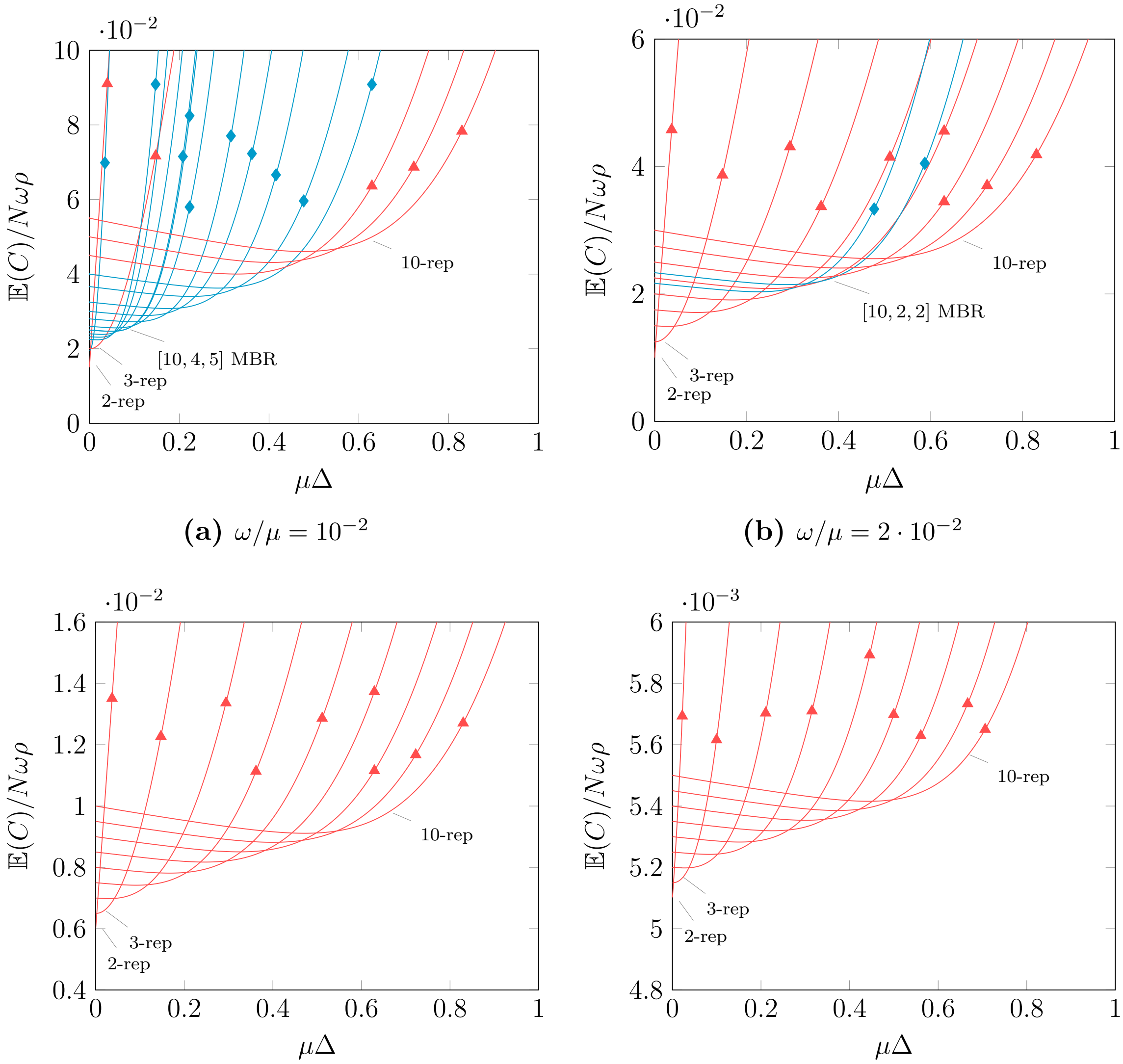
Figure 3.11: Normalized total cost $\mathbb{E}(C)/N\omega\rho$ versus the normalized repair interval $\mu\Delta$ when decreasing the transmission penalty ratio ρ and fixing $\omega = 0.5$. The dotted line shows the expected cost of BS download.

large since $\mathbb{E}(C) \to \mathbb{E}(C_\mathrm{d})$ as $\omega \to \infty$ using (2.19) and following the description in Section 2.3.

3.5 Reducing the Transmission Penalty Ratio

For all the previous figures (except Fig. 3.2), we have assumed a high penalty cost ratio ρ. In this section, we reduce this penalty ratio. We plot the total normalized cost $\mathbb{E}(C)/N\omega\rho$ versus the normalized repair interval $\mu\Delta$ for $\omega = 0.5$ and transmission penalty ratio $\rho \in \{1, 5, 10, 50\}$ in Fig. 3.11.

The maximum repair interval $\Delta_{\max}$, explaining when storing data on the nodes is advantageous, decreases with the penalty ratio in agreement with Fig. 3.2. Also, the $[10, 5, 5]$ MDS code is rendered useless for $\rho < 10$ and the current network parameters. As a final remark, when $\rho = 1$, $\mathbb{E}(C)/N\omega\rho > 1$ for all considered codes, i.e., we should rely on the BS for file downloads. The reason for this is that $\rho \leq k\alpha/M$ and the download cost $\mathbb{E}(C_\mathrm{d})$ is decreasing or constant in Δ depending on the code, as explained by Corollary 1. Since $\mathbb{E}(C_\mathrm{r}) \geq 0$, with equality when $\Delta \to \infty$, the total cost is minimized if we never attempt repair.



(a) $\omega/\mu = 10^{-2}$

(b) $\omega/\mu = 2 \cdot 10^{-2}$

(c) $\omega/\mu = 10^{-1}$

(d) $\omega/\mu = 1$

Figure 3.12: Normalized total cost $\mathbb{E}(C)/N\omega\rho$ for codes achieving the minimum total cost for some repair intervals versus the normalized repair interval $\mu\Delta$ when $\rho = 200$ and varying the ratio ω/μ.

3.6 Codes Minimizing the Total Cost

Now, when we know how the code and network parameters affect the average total cost, we are ready to summarize the performance of MDS codes and regenerating codes. For $\rho = 200$, we compare all MDS and regenerating codes with $n \in \{3, \dots, 10\}$, i-replication, $i \in \{2, \dots, n\}$ and the uncoded case where one node stores the file. Note that for MDS and regenerating codes with $n \in \{1, 2\}$, we get the uncoded and replication scheme respectively since $k < n$. Recall that we use a maximum number of 10 storage nodes to not violate (2.2). Fig. 3.12 shows normalized total cost $\mathbb{E}(C)/N\omega\rho$ as a function of the normalized repair interval $\mu\Delta \in [0, 1]$ for all codes that achieve the lowest total cost for some repair intervals when we vary the request/departure rate ratio $\omega/\mu \in \{10^{-2}, 2 \cdot 10^{-2}, 10^{-1}, 1\}$. Blue curves with

♦ markers correspond to MBR codes. Red curves with ▲ markers represent replication schemes. The same colors and markers are used in the subsequent figures. Particularly interesting codes are highlighted in the figure.

In Fig. 3.12 we see that for repair intervals $\Delta \in [\Delta_i, \Delta_{i+1})$, $i \in \{0, 1, \dots\}$, a particular code is attaining the minimum total cost. We refer to the shortest time, Δ_i, as the repair interval threshold Δ_{th} for that code. Another code is achieving the minimum cost in repair intervals $\Delta \in [\Delta_{i+1}, \Delta_{i+2})$. We refer to the periodic repair time Δ_{i+1} as the repair interval threshold for the second code, and so on. Table 3.1 summarizes the normalized repair interval thresholds $\mu\Delta_{\text{th}}$ for the codes presented in Fig. 3.12 and others for the extended normalized repair intervals $\mu\Delta \in [0, 10]$. Note that if BS download pertains the minimum total cost we write "BS" in the column "optimal code". For the same codes and $\omega/\mu = 10^{-2}$, Fig. 3.13 shows the normalized total cost $\mathbb{E}(C)/N\omega\rho$ versus the normalized repair interval $\mu\Delta$ for $\rho \in \{5, 10, 50\}$. Table 3.2 lists the normalized repair interval thresholds for the codes presented in Fig. 3.13 as well as other codes performing well in extended normalized repair intervals $\mu\Delta \in [0, 10]$.

If repairs can be carried out very frequently, the 2- and 3-replication schemes yield the minimum total cost for all considered scenarios. We note that the MSR codes are never achieving the minimum total cost and neither are the MDS codes for which $k \neq 1$. For some repair intervals the MBR codes are exhibiting good performance but with an increasing request rate ω, or a decreasing transmission penalty ratio, replication is the best scheme to use for wireless DS.

We impose the storage budget constraint $n\alpha \leq \Gamma M$, introduced in Section 2.1. If we let $\Gamma = 5$, some of the codes presented in the previous figures and tables are disqualified for using too much storage. For example, the i-replication schemes, $i \in \{\lfloor\Gamma\rfloor + 1, \dots, 10\}$ are no longer applicable. We again consider MDS and regenerating codes for which $n \in \{3, \dots, 10\}$ but only the codes conforming to the storage budget constraint. The uncoded scheme is included in the comparison. Fig. 3.14 shows the normalized total cost $\mathbb{E}(C)/N\omega\rho$ versus the normalized repair interval $\mu\Delta \in [0, 1]$ for codes that achieve the minimum total cost for some repair intervals. We investigate a selection of request/departure rate ratios ω/μ and transmission penalty ratios ρ. Red curves with ■ markers correspond to MDS codes and black curves with ● markers represent MSR codes. Red curves with ▲ markers represent replication schemes as before. The normalized repair interval thresholds $\mu\Delta_{\text{th}}$ for the codes in Fig. 3.14 are listed in Table 3.3 in addition to other codes achieving the minimum total cost for some repair intervals in the extended normalized repair intervals $\mu\Delta \in [0, 10]$.

When imposing a storage budget constraint, we see that there exists repair intervals for which the $[9, 2, 2]$ MDS code, the $[10, 2, 2]$ MDS code and the $[10, 2, 3]$ MSR code yield the lowest total cost among all the considered codes. They are replacing the less storage efficient replication schemes that are no longer applicable due to the storage budget constraint. None of the high repair access ($r = n - 1$) regenerating codes are performing well. The fact that the $[10, 2, 3]$ MSR code is achieving the minimum cost for some repair intervals means that it can be beneficial to increase the repair access slightly, which confirms the results in Section 3.3.

Table 3.1: Codes achieving the lowest average total cost for different request rate ratios ω/μ when $\rho = 200$. The normalized repair interval threshold $\mu\Delta_{\text{th}}$ is the shortest normalized repair interval such that the code achieves the minimum total cost which is valid until the next repair interval threshold.

(a) $\omega/\mu = 10^{-2}$

$\mu\Delta_{\text{th}}$	optimal code	
0	2-replication	▲
$2.4 \cdot 10^{-3}$	$[8, 5, 6]$ MBR	◆
$3.3 \cdot 10^{-3}$	3-replication	▲
$3.1 \cdot 10^{-2}$	$[10, 5, 6]$ MBR	◆
$5.2 \cdot 10^{-2}$	$[9, 4, 5]$ MBR	◆
$5.7 \cdot 10^{-2}$	$[8, 4, 4]$ MBR	◆
$6.3 \cdot 10^{-2}$	$[10, 4, 5]$ MBR	◆
$7.7 \cdot 10^{-2}$	$[10, 5, 5]$ MBR	◆
$9.7 \cdot 10^{-2}$	$[9, 4, 4]$ MBR	◆
0.12	$[10, 4, 4]$ MBR	◆
0.17	$[9, 3, 3]$ MBR	◆
0.20	$[10, 3, 3]$ MBR	◆
0.29	$[9, 2, 2]$ MBR	◆
0.33	$[10, 2, 2]$ MBR	◆
0.44	8-replication	▲
0.48	9-replication	▲
0.56	10-replication	▲
1.8	BS	

(b) $\omega/\mu = 2 \cdot 10^{-2}$

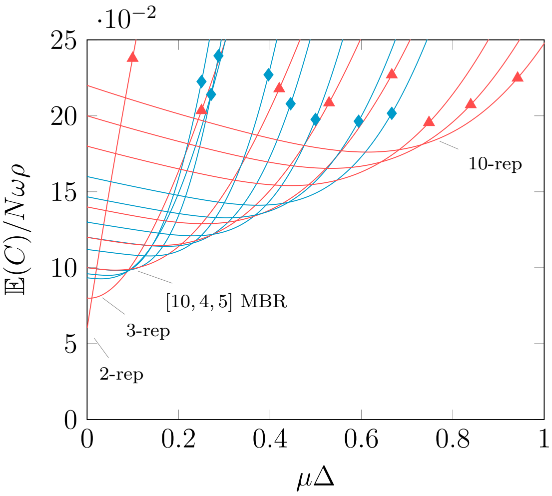
$\mu\Delta_{\text{th}}$	optimal code	
0	2-replication	▲
$2.5 \cdot 10^{-3}$	3-replication	▲
$4.3 \cdot 10^{-2}$	4-replication	▲
0.12	5-replication	▲
0.21	6-replication	▲
0.29	$[9, 2, 2]$ MBR	◆
0.32	7-replication	▲
0.36	$[10, 2, 2]$ MBR	◆
0.40	8-replication	▲
0.48	9-replication	▲
0.56	10-replication	▲
2.21	2-replication	▲
5.4	uncoded	▲

(c) $\omega/\mu = 10^{-1}$

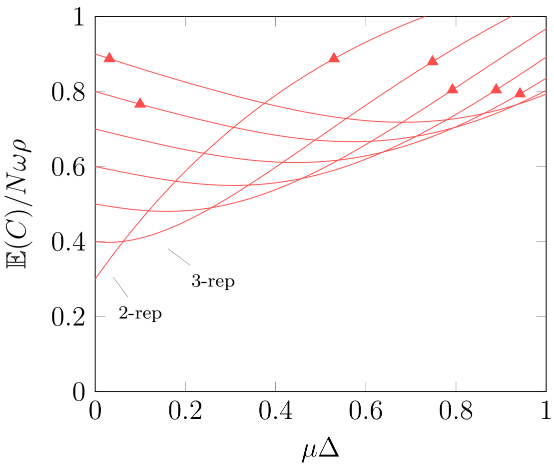
$\mu\Delta_{\text{th}}$	optimal code	
0	2-replication	▲
$2.5 \cdot 10^{-3}$	3-replication	▲
$4.2 \cdot 10^{-2}$	4-replication	▲
0.12	5-replication	▲
0.21	6-replication	▲
0.30	7-replication	▲
0.39	8-replication	▲
0.47	9-replication	▲
0.55	10-replication	▲
7.5	9-replication	▲

(d) $\omega/\mu = 1$

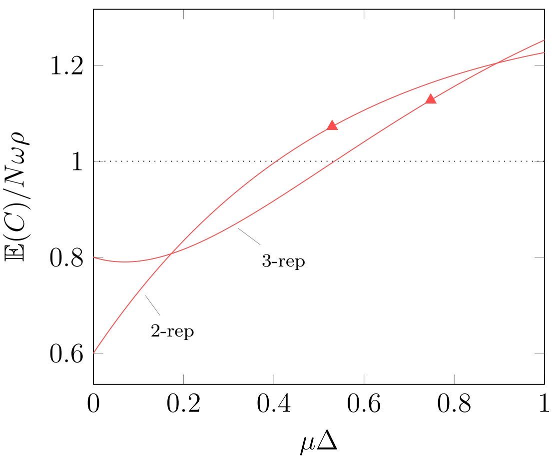
$\mu\Delta_{\text{th}}$	optimal code	
0	2-replication	▲
$2.5 \cdot 10^{-3}$	3-replication	▲
$3.8 \cdot 10^{-2}$	4-replication	▲
0.10	5-replication	▲
0.18	6-replication	▲
0.27	7-replication	▲
0.35	8-replication	▲
0.43	9-replication	▲
0.50	10-replication	▲



(a) $\rho = 50$



(b) $\rho = 10$



(c) $\rho = 5$. The dotted line shows the cost of BS download.

Figure 3.13: Normalized total cost $\mathbb{E}(C)/N\omega\rho$ for codes achieving the minimum total cost for some repair intervals versus the normalized repair interval $\mu\Delta$ when $\omega/\mu = 10^{-2}$ and varying the transmission penalty ratio ρ.

Table 3.2: Codes achieving the lowest average total cost for different transmission penalty costs ρ when $\omega/\mu = 10^{-2}$. The normalized repair interval threshold $\mu\Delta_{\text{th}}$ is the shortest normalized repair interval such that the code achieves the minimum total cost which is valid until the next repair interval threshold.

(a) $\rho = 50$

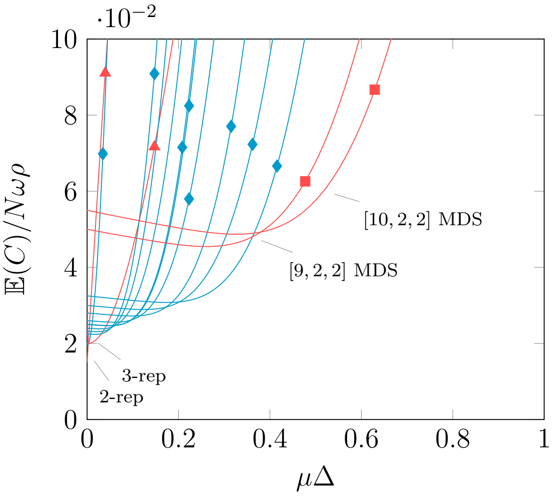
$\mu\Delta_{\text{th}}$	optimal code
0	2-replication
$1.1 \cdot 10^{-2}$	3-replication
$8.9 \cdot 10^{-2}$	$[5, 2, 2]$ MBR
$9.0 \cdot 10^{-2}$	$[8, 4, 4]$ MBR
$9.4 \cdot 10^{-2}$	$[10, 4, 5]$ MBR
0.12	4-replication
0.18	$[10, 4, 4]$ MBR
0.23	$[9, 3, 3]$ MBR
0.27	5-replication
0.28	$[10, 3, 3]$ MBR
0.37	6-replication
0.38	$[9, 2, 2]$ MBR
0.45	$[10, 2, 2]$ MBR
0.55	8-replication
0.65	9-replication
0.74	10-replication
1.8	BS

(b) $\rho = 10$

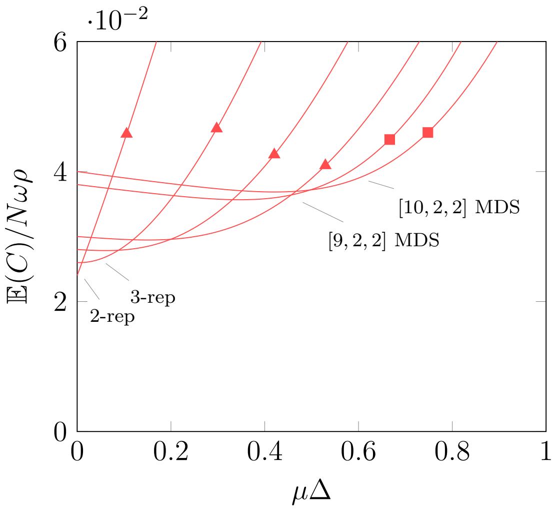
$\mu\Delta_{\text{th}}$	optimal code
0	2-replication
$6.3 \cdot 10^{-2}$	3-replication
0.26	4-replication
0.46	5-replication
0.64	6-replication
0.80	7-replication
0.94	8-replication
1.1	9-replication
1.2	10-replication
1.4	BS

(c) $\rho = 5$

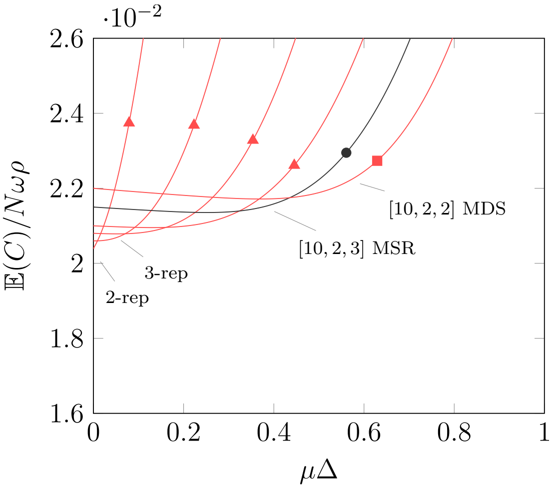
$\mu\Delta_{\text{th}}$	optimal code
0	2-replication
0.17	3-replication
0.54	BS



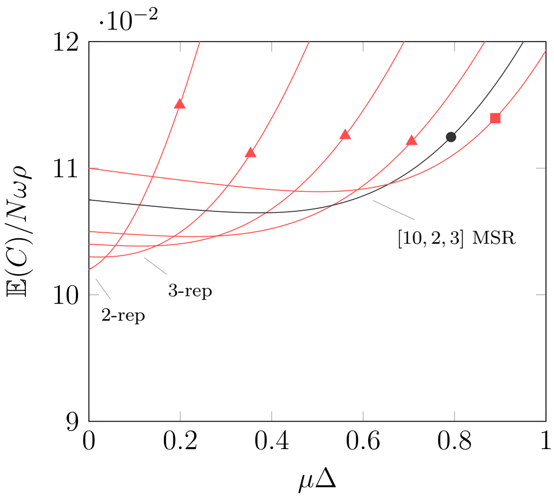
(a) $\omega/\mu = 10^{-2}$ and $\rho = 200$



(b) $\omega/\mu = 10^{-1}$ and $\rho = 50$



(c) $\omega/\mu = 1$ and $\rho = 50$



(d) $\omega/\mu = 1$ and $\rho = 10$

Figure 3.14: Normalized total cost $\mathbb{E}(C)/N\omega\rho$ for codes achieving the minimum total cost for some repair intervals versus the normalized repair interval $\mu\Delta$ when $\Gamma = 5$.

Table 3.3: Codes achieving the lowest average total cost for a selection of request rate ratios ω/μ and transmission penalty cost ratios ρ when $\Gamma = 5$. The normalized repair interval threshold $\mu\Delta_{\text{th}}$ is the shortest normalized repair interval such that the code achieves the minimum total cost which is valid until the next repair interval threshold.

(a) $\omega/\mu = 10^{-2}$ and $\rho = 200$

$\mu\Delta_{\text{th}}$	optimal code	
0	2-replication	▲
$2.4 \cdot 10^{-3}$	$[8, 5, 6]$ MBR	♦
$3.3 \cdot 10^{-3}$	3-replication	▲
$3.1 \cdot 10^{-2}$	$[10, 5, 6]$ MBR	♦
$5.1 \cdot 10^{-2}$	$[9, 4, 5]$ MBR	♦
$5.7 \cdot 10^{-2}$	$[8, 4, 4]$ MBR	♦
$6.2 \cdot 10^{-2}$	$[10, 4, 5]$ MBR	♦
$7.7 \cdot 10^{-2}$	$[10, 5, 5]$ MBR	♦
$9.7 \cdot 10^{-2}$	$[9, 4, 4]$ MBR	♦
0.12	$[10, 4, 4]$ MBR	♦
0.17	$[9, 3, 3]$ MBR	♦
0.20	$[10, 3, 3]$ MBR	♦
0.37	$[9, 2, 2]$ MDS	■
0.38	$[10, 2, 2]$ MDS	■
1.5	BS	

(b) $\omega/\mu = 10^{-1}$ and $\rho = 50$

$\mu\Delta_{\text{th}}$	optimal code	
0	2-replication	▲
$1.0 \cdot 10^{-2}$	3-replication	▲
$8.9 \cdot 10^{-2}$	4-replication	▲
$0.20 \cdot 10^{-2}$	5-replication	▲
0.46	$[9, 2, 2]$ MDS	■
0.50	$[10, 2, 2]$ MDS	■
1.6	5-replication	▲

(c) $\omega/\mu = 1$ and $\rho = 50$

$\mu\Delta_{\text{th}}$	optimal code	
0	2-replication	▲
$9.1 \cdot 10^{-3}$	3-replication	▲
$7.0 \cdot 10^{-2}$	4-replication	▲
0.16	5-replication	▲
0.33	$[10, 2, 3]$ MSR	●
0.44	$[10, 2, 2]$ MDS	■
1.8	5-replication	▲

(d) $\omega/\mu = 1$ and $\rho = 10$

$\mu\Delta_{\text{th}}$	optimal code	
0	2-replication	▲
$3.7 \cdot 10^{-2}$	3-replication	▲
0.15	4-replication	▲
0.28	5-replication	▲
0.54	$[10, 2, 3]$ MSR	●
0.66	$[10, 2, 2]$ MDS	■
1.8	5-replication	▲

Chapter 4

Conclusions and Future Work

We considered DS for a wireless network where data is stored in a distributed manner across mobile devices. We introduced a repair scheduling where the repair of the data lost due to device departures is performed periodically. We derived analytical expressions for the total communication cost, due to repair and download, as a function of the repair interval.

If the request rate is low, we showed that there exists a maximum value of the repair interval after which retrieving the file from the BS is always less costly provided that the transmission penalty ratio is high. Therefore, DS in wireless networks is useful if the repair can be performed frequently enough. The maximum repair interval decreases with decreasing values of the transmission penalty ratio. For sufficiently low penalty the maximum repair interval is no longer defined since it is always less costly to download the file from the BS. An increasing request rate has the opposite effect on the maximum repair interval. With a high request rate, DS in wireless networks is always beneficial.

The optimal repair interval that minimizes the total communication cost depends on the code used for storage. For high repair access regenerating codes, instantaneous repair is indeed optimal but for MDS codes using more than two storage nodes, this is not the case. Varying the repair access, there is a tradeoff between the reduction in repair bandwidth and the tolerance for storage node departures and we showed that it is never beneficial to set the repair access too high. For a high transmission penalty ratio and a low request/departure rate ratio, i.e., a network dealing mainly with repairs and where it is very costly to use the BS, the MBR codes with moderate repair access obtain the minimum total cost for some repair intervals.

We also demonstrated that if we are unlimited in storage capacity, replication is the scheme attaining the lowest total cost if the request rate is high or the transmission penalty ratio is low. Imposing a storage budget constraint, the more storage efficient MDS and MSR codes proved to have good performance for some repair intervals.

This work may be extended in numerous ways. We have to further analyze the expressions for the average total cost to be able to predict what ECC to use for wireless DS given a transmission penalty ratio and an arrival, departure and request rate. Also, closed form expressions for the repair interval minimizing the total cost have to be derived. It would furthermore be interesting to investigate the statistics of a real wireless network to find out if our assumptions that node arrivals and departures follow a Poisson process is valid and, in that case, what are reasonable arrival, departure and request rates. Connecting this knowledge to the complexity of a particular code would allow us to say whether it is at all reasonable to achieve a certain repair frequency.

There are many codes that could be analyzed using our system model where the most interesting include locally repairable codes and rateless codes. The latter codes might be more suitable if we consider the time it takes to perform repair, i.e., storage nodes might depart in the middle of the repair process. It would also be interesting to vary the D2D transmission penalty according to the geographical distance of nodes within the cell, i.e., it requires more battery power to transmit over longer distances. Finally, we have not yet considered the communication overhead required to maintain the wireless DS network.

Appendix A

Proof of Theorems

A.1 Proof of Lemma 2

Let the function

$$g(x) = x - i\Delta = t, \quad x \in [i\Delta, (i+1)\Delta),\ i \in \{0, 1, \ldots\}, \tag{A.1}$$

describe the transformation $g : W_l \to \tilde{W}_l$. The derivative of g is not defined in the points $x = i\Delta$ but since $g'(x) = 1,\ x \in (i\Delta, (i+1)\Delta)$ and $\lim_{x \to i\Delta} g'(x) = 1$, $g'(x) \equiv 1$ in the domain of g. We have that

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

are the roots of (A.1) and the PDF of $\tilde{W}_l$ is calculated as [22]

$$f_{\tilde{W}_l}(t) = \sum_{x_i} f_{W_l}(x_i) \left| \frac{1}{g'(x_i)} \right| = \sum_{i=0}^{\infty} f_{W_l}(t + i\Delta) = \sum_{i=0}^{\infty} \frac{\omega^l (t + i\Delta)^{l-1} e^{-\omega(t+i\Delta)}}{(l-1)!}.$$

where $f_{W_l}(t)$ is given by (2.13).

A.2 Proof of Lemma 3

Using the gamma function

$$\Gamma(l) = (l-1)!, \quad l \in \{1, 2, \ldots\},$$

and the Lerch's transcendent [25]

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

(2.14) can be rewritten as

$$f_{\tilde{W}_l}(t) = \frac{\omega^l e^{-\omega t}}{\Gamma(l)} \sum_{i=0}^{\infty} \Delta^{l-1} \left(\frac{t}{\Delta} + i\right)^{l-1} e^{-\omega\Delta i} = \frac{(\omega\Delta)^l e^{-\omega t}}{\Delta\Gamma(l)} \Phi\left(e^{-\omega\Delta}, 1 - l, \frac{t}{\Delta}\right)$$

According to [26]

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

and finally we obtain (2.15) as

$$\lim_{l\to\infty}f_{\tilde{W}_l}(t)=\frac{e^{-\omega t}}{\Delta}e^{\omega t}=\frac{1}{\Delta}.$$

A.3 Proof of Corollary 1

Let

$$l_i(x)=\prod_{\substack{j=k\\ j\neq i}}^{n}\frac{x-j}{i-j},$$

then for a function $f(x)$, the Lagrange interpolation polynomial [25]

$$L(x)=\sum_{i=k}^{n}f(i)l_i(x),$$

represents the function $f(x)$ at the points $x=i'\in\{k,\dots,n\}$, i.e.,

$$L(i')=\sum_{i=k}^{n}f(i)l_i(i')=f(i').$$

Now, using (2.17)

$$\begin{aligned}\frac{\partial}{\partial\Delta}\Pr\{\text{D2D download}\}&=\sum_{i=k}^{n}\frac{\partial}{\partial\Delta}\frac{1-p_i}{\Delta\mu_i}\prod_{\substack{j=k\\ j\neq i}}^{n}\frac{\mu_j}{(\mu_j-\mu_i)}\\&=\sum_{i=k}^{n}\frac{1}{\Delta^2}\left(\Delta p_i+\frac{p_i-1}{\mu_i}\right)\prod_{\substack{j=k\\ j\neq i}}^{n}\frac{j}{(j-i)},\end{aligned}$$

and if we let

$$f(x)=\frac{1}{\Delta^2}\left(\Delta e^{-x\mu\Delta}+\frac{e^{-x\mu\Delta}-1}{x\mu}\right),$$

then

$$\frac{\partial}{\partial\Delta}\Pr\{\text{D2D download}\}=L(0).$$

Since

$$x\mu\Delta<e^{x\mu\Delta}-1,\quad\forall\ x,\mu,\Delta>0,$$

we have

$$f(x)<0,\quad\forall\ x,\mu,\Delta>0,$$

Seeing that $L(x)$ interpolates the function $f(x)$ in the points $i \in \{k, \ldots, n\}$ gives

$$\frac{\partial}{\partial \Delta} \Pr\{\text{D2D download}\} < 0, \quad \forall\, \Delta > 0.$$

From (2.10)

$$\frac{\partial}{\partial \Delta} \mathbb{E}\left(C_{\mathrm{d}}\right) = N\omega \left(\rho_{\mathrm{D2D}} \frac{k\alpha}{M} - \rho_{\mathrm{BS}}\right) \frac{\partial}{\partial \Delta} \Pr\{\text{D2D download}\},$$

and the sign of $\frac{\partial}{\partial \Delta} \mathbb{E}\left(C_{\mathrm{d}}\right)$ only depends on the sign of

$$\rho_{\mathrm{D2D}} \frac{k\alpha}{M} - \rho_{\mathrm{BS}} \propto \frac{k\alpha}{M} - \rho,$$

which completes the proof.

A.4 Proof of Corollary 2

From Theorem 1

$$\lim_{\Delta \to 0} \mathbb{E}\left(C_{\mathrm{r}}\right) = \frac{1}{M} \left(\rho_{\mathrm{BS}} \gamma_{\mathrm{BS}} \sum_{i=0}^{r-1} (n-i) \lim_{\Delta \to 0} \frac{b_i(n,p)}{\Delta} + \rho_{\mathrm{D2D}} \gamma_{\mathrm{D2D}} \sum_{i=r}^{n} (n-i) \lim_{\Delta \to 0} \frac{b_i(n,p)}{\Delta}\right)$$

By l'Hôpitals rule

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

We have that

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

and

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

giving the result

$$\lim_{\Delta \to 0} \mathbb{E}\left(C_{\mathrm{r}}\right) = \rho_{\mathrm{D2D}} \gamma_{\mathrm{D2D}} n\mu.$$

Using Theorem 2 and l'Hôpitals rule

$$\lim_{\Delta\to 0}\mathbb{E}\left(C_{\rm d}\right) = N\omega\left[\rho_{\rm BS} + \left(\rho_{\rm D2D}\frac{k\alpha}{M} - \rho_{\rm BS}\right)\sum_{i=k}^{n}\frac{1}{\mu_i}\lim_{\Delta\to 0}\frac{1-p_i}{\Delta}\prod_{\substack{j=k\\ j\neq i}}^{n}\frac{\mu_j}{(\mu_j-\mu_i)}\right]$$

$$= N\omega\left[\rho_{\rm BS} + \left(\rho_{\rm D2D}\frac{k\alpha}{M} - \rho_{\rm BS}\right)\sum_{i=k}^{n}\prod_{\substack{j=k\\ j\neq i}}^{n}\frac{\mu_j}{(\mu_j-\mu_i)}\right] \tag{A.2}$$

Consider the function

$$F(x) = \frac{1}{\prod_{j=k}^{n}(\mu_j - x)},$$

which can be expanded as the sum of partial fractions as

$$F(x) = \frac{c_k}{\mu_k - x} + \frac{c_{k+1}}{\mu_{k+1}-x} + \ldots + \frac{c_n}{\mu_n - x} = \sum_{i=k}^{n}\frac{c_i}{\mu_i - x},$$

where the constants $c_i,\ i\in\{k,k+1,\ldots,n\}$, are given by evaluating

$$1 = \prod_{j=k}^{n}(\mu_j - x)F(x) = c_k\prod_{\substack{j=k\\ j\neq k}}^{n}(\mu_j - x) + c_{k+1}\prod_{\substack{j=1\\ j\neq k+1}}^{n}(\mu_j - x) + \ldots + c_n\prod_{\substack{j=1\\ j\neq n}}^{n}(\mu_j - x)$$

at the points $x\in\{\mu_k,\mu_{k+1},\ldots,\mu_n\}$, thus producing the result

$$c_i = \frac{1}{\prod_{\substack{j=k\\ j\neq i}}^{n}(\mu_j-\mu_i)}.$$

Therefore

$$F(x) = \sum_{i=1}^{n}\frac{1}{(\mu_i - x)\prod_{j=1}^{n}(\mu_j-\mu_i)},$$

which evaluated at $x=0$ gives

$$F(0) = \frac{1}{\prod_{j=k}^{n}\mu_j} = \sum_{i=k}^{n}\frac{1}{\mu_i\prod_{\substack{j=k\\ j\neq i}}^{n}(\mu_j-\mu_i)}.$$

We may apply this observation to (A.2) and calculate

$$\lim_{\Delta\to 0}\mathbb{E}\left(C_{\rm d}\right) = N\omega\left[\rho_{\rm BS} + \left(\rho_{\rm D2D}\frac{k\alpha}{M} - \rho_{\rm BS}\right)\sum_{i=k}^{n}\prod_{\substack{j=k\\ j\neq i}}^{n}\frac{\mu_j}{(\mu_j-\mu_i)}\right]$$

$$= N\omega\left[\rho_{\rm BS} + \left(\rho_{\rm D2D}\frac{k\alpha}{M} - \rho_{\rm BS}\right)\sum_{i=k}^{n}\frac{\prod_{j=k}^{n}\mu_j}{\mu_i\prod_{\substack{j=k\\ j\neq i}}^{n}(\mu_j-\mu_i)}\right]$$

$$= N\omega\left[\rho_{\rm BS} + \left(\rho_{\rm D2D}\frac{k\alpha}{M} - \rho_{\rm BS}\right)F(0)\prod_{j=k}^{n}\mu_j\right] = N\omega\rho_{\rm D2D}\frac{k\alpha}{M},$$

which, using (2.19), gives (2.20).

For the second part of the Corollary, again using Theorems 1 and 2, we have for the average repair cost

$$\lim_{\Delta\to\infty}\mathbb{E}(C_r)=\frac{1}{M}\left(\sum_{i=0}^{r-1}(n-i)\lim_{\Delta\to\infty}\frac{b_i(n,p)}{\Delta}+\sum_{i=r}^{n}(n-i)\lim_{\Delta\to\infty}\frac{b_i(n,p)}{\Delta}\right),$$

where

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

and hence

$$\lim_{\Delta\to\infty}\mathbb{E}\left(C_{\mathrm{r}}\right)=0.$$

For the average delivery cost

$$\lim_{\Delta\to\infty}\mathbb{E}\left(C_{\mathrm{d}}\right)=N\omega\left[\rho_{\mathrm{BS}}+\left(\rho_{\mathrm{D2D}}\frac{k\alpha}{M}-\rho_{\mathrm{BS}}\right)\sum_{i=k}^{n}\frac{1}{\mu_i}\lim_{\Delta\to\infty}\frac{1-p_i}{\Delta}\prod_{\substack{j=k\\ j\neq i}}^{n}\frac{\mu_j}{(\mu_j-\mu_i)}\right],$$

where

$$\lim_{\Delta\to\infty}\frac{1-p_i}{\Delta}=0,\quad \forall\, i.$$

Now

$$\lim_{\Delta\to\infty}\mathbb{E}\left(C_{\mathrm{d}}\right)=N\omega\rho_{\mathrm{BS}},$$

and (2.21) follows.

Bibliography

[1] Cisco, "Cisco visual networking index: Global mobile data traffic forecast update, 2014-2019," Cisco, Tech. Rep., 2015.

[2] Z. Pi and F. Khan, "An introduction to millimeter-wave mobile broadband systems," *IEEE Communications Magazine*, vol. 49, no. 6, pp. 101–107, June 2011.

[3] G. Hogben and M. Dekker, "Smartphones: Information security risks, opportunities and recommendations for users," ENISA, Tech. Rep. December, 2010.

[4] J. Andrews, S. Buzzi, W. Choi, S. Hanly, A. Lozano, A. Soong, and J. Zhang, "What will 5G be?" *IEEE Journal on Selected Areas in Communications*, vol. 32, no. 6, pp. 1065–1082, June 2014.

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

[6] M. Maddah-Ali and U. Niesen, "Fundamental limits of caching," in *IEEE International Symposium on Information Theory Proceedings*, July 2013, pp. 1077–1081.

[7] J. Llorca, A. M. Tulino, K. Guan, and D. C. Kilper, "Network-coded caching-aided multicast for efficient content delivery," in *IEEE International Conference on Communications*, June 2013, pp. 3557–3562.

[8] J. Hachem, N. Karamchandani, and S. Diggavi, "Multi-level coded caching," in *Information Theory (ISIT), 2014 IEEE International Symposium on*, June 2014, pp. 56–60.

[9] 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.

[10] 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 Transactions on Wireless Communications*, vol. 13, no. 7, pp. 3665–3676, Jul. 2014.

[11] A. Shokrollahi, "Raptor codes," *IEEE Transactions on Information Theory*, vol. 52, no. 6, pp. 2551–2567, June 2006.

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

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

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

[15] S. Jiekak, A.-M. Kermarrec, N. Le Scouarnec, G. Straub, and A. Van Kempen, "Regenerating codes: A system perspective," in *IEEE 31st Symposium on Reliable Distributed Systems (SRDS)*, Oct 2012, pp. 436–441.

[16] C. Huang, M. Chen, and J. Li, "Pyramid codes: Flexible schemes to trade space for access efficiency in reliable data storage systems," in *Sixth IEEE International Symposium on Network Computing and Applications*, July 2007, pp. 79–86.

[17] C. Huang, H. Simitci, Y. Xu, A. Ogus, B. Calder, P. Gopalan, J. Li, and S. Yekhanin, "Erasure coding in windows azure storage," in *USENIX Annual Technical Conference (USENIX ATC 12)*, 2012, pp. 15–26.

[18] D. Papailiopoulos and A. Dimakis, "Locally repairable codes," *IEEE Transactions on Information Theory*, vol. 60, no. 10, pp. 5843–5855, Oct 2014.

[19] I. Tamo, Z. Wang, and J. Bruck, "Zigzag codes: Mds array codes with optimal rebuilding," *IEEE Transactions on Information Theory*, vol. 59, no. 3, pp. 1597–1616, March 2013.

[20] K. Rashmi, N. Shah, and K. Ramchandran, "A piggybacking design framework for read-and download-efficient distributed storage codes," in *IEEE International Symposium on Information Theory Proceedings (ISIT)*, July 2013, pp. 331–335.

[21] J. Pedersen, A. Graell i Amat, I. Andriyanova, and F. Brännström, "Repair scheduling in wireless distributed storage with D2D communication," in *ArXiv*, April 2015. [Online]. Available: http://arxiv.org/abs/1504.06231

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

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

[24] 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.

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

[26] 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.