Copyright Warning & Restrictions

The copyright law of the United States (Title 17, United States Code) governs the making of photocopies or other reproductions of copyrighted material.

Under certain conditions specified in the law, libraries and archives are authorized to furnish a photocopy or other reproduction. One of these specified conditions is that the photocopy or reproduction is not to be "used for any purpose other than private study, scholarship, or research." If a, user makes a request for, or later uses, a photocopy or reproduction for purposes in excess of "fair use" that user may be liable for copyright infringement,

This institution reserves the right to refuse to accept a copying order if, in its judgment, fulfillment of the order would involve violation of copyright law.

Please Note: The author retains the copyright while the New Jersey Institute of Technology reserves the right to distribute this thesis or dissertation

Printing note: If you do not wish to print this page, then select "Pages from: first page # to: last page #" on the print dialog screen

ABSTRACT

ONLINE EDGE CACHING AND WIRELESS DELIVERY IN FOG-AIDED NETWORKS

by
Seyyed Mohammadreza Azimi

Multimedia content is the significant fraction of transferred data over the wireless medium in the modern cellular and wireless communication networks. To improve the quality of experience perceived by users, one promising solution is to push the most popular contents as close as to users, also known as the "edge" of network. Storing content at the edge nodes (ENs) or base stations (BSs) is called "caching". In Fog Radio Access Network (F-RAN), each EN is equipped with a cache as well as a "fronthaul" connection to the content server. Among the new design problems raised by the outlined scenarios, two key issues are addressed in this dissertation: 1) How to utilize cache and fronthaul resources while taking into account the wireless channel impairments; 2) How to incorporate the time-variability of popular set in the performance evaluation of F-RAN. These aspects are investigated by using information-theoretic models, obtaining fundamental insights that have been corroborated by various illustrative examples. To address point 1), two scenarios are investigated. First, a single-cell scenario with two transmitters is considered. A fog-aided small-cell BS as one of the transmitters and a cloud-aided macro-cell BS as the second transmitter collaborate with each other to send the requested content over a partially connected wireless channel. The intended and interference channels are modeled by erasure channels. Assuming a static set of popular contents, *offline caching* maps the library of files to cached contents stored at small-cell BS such that the cache capacity requirement is met. The delivery time per bit (DTB) is adopted as a measure of the coding latency, that is, the duration of the transmission block, required for reliable delivery. It is proved that optimal DTB is a linear decreasing

function of cache capacity as well as inversely proportional with capacity of fronthaul link. In the second scenario, the same single-cell model is used with the only caveat that the set of popular files is time-varying. In this case, *online caching* maps the library of files to cached contents at small-cell BS. Thanks to availability of popular set at macro-BS, the DTB is finite and has upper and lower bounds which are functions of system resources i.e., cache and fronthaul link capacities. As for point 2), the model is comprised of an arbitrary number of ENs and users connected through an interference-limited wireless channel at high-SNR regime. All equally important ENs are benefited from cache capacity as well as fronthaul connection to the content server. The time-variability of popular set necessitates online caching to enable ENs keep track of changes in the popular set. The analysis is centered on the characterization of the long-term Normalized Delivery Time (NDT), which captures the temporal dependence of the coding latencies accrued across multiple time slots in the high-SNR regime. Online edge caching and delivery schemes based on reactive and proactive caching principles are investigated for both serial and pipelined transmission modes across fronthaul and edge segments. The outcome of analytical results provides a controversial view of contemporary research on the edge caching. It is proved that with a time-varying set of popular files, the capacity of fronthaul link between ENs and content server set a fundamental limit on the system performance. This is due to the fact that the original information source is content server and the only way to retrieve information is via fronthaul links. While edge caching can provide some gains in term of reduced latency, the gain diminishes as a result of the fact that the cached content is prone to be outdated with time-varying popularity.

ONLINE EDGE CACHING AND WIRELESS DELIVERY IN FOG-AIDED NETWORKS

by
Seyyed Mohammadreza Azimi

A Dissertation
Submitted to the Faculty of
New Jersey Institute of Technology
in Partial Fulfillment of the Requirements for the Degree of
Doctor of Philosophy in Electrical Engineering

Helen and John C. Hartmann Department of
Electrical and Computer Engineering

May 2018

APPROVAL PAGE

ONLINE EDGE CACHING AND WIRELESS DELIVERY IN FOG-AIDED NETWORKS

Seyyed Mohammadreza Azimi

Prof. Osvaldo Simeone, Dissertation Advisor — Date
Professor of Electrical and Computer Engineering, NJIT

Prof. Alexander Haimovich, Committee Member — Date
Distinguished Professor of Electrical and Computer Engineering, NJIT

Prof. Ali Abdi, Committee Member — Date
Professor of Electrical and Computer Engineering, NJIT

Dr. Joerg Kliewer, Committee Member — Date
Associate Professor of Electrical and Computer Engineering, NJIT

Dr. Ravi Tandon, Committee Member — Date
Assistant Professor of Electrical and Computer Engineering,
The University of Arizona

BIOGRAPHICAL SKETCH

Author: Seyyed Mohammadreza Azimi

Degree: Doctor of Philosophy

Date: May 2018

Date of Birth:

Place of Birth:

Undergraduate and Graduate Education:

- Doctor of Philosophy in Electrical Engineering,
  New Jersey Institute of Technology, Newark, NJ, 2018.
- Master of Science in Electrical Engineering,
  Isfahan University of Technology, Isfahan, Iran, 2010
- Bachelor of Science in Electrical Engineering,
  University of Guilan, Rasht, Iran, 2007

Major: Electrical Engineering

Presentations and Publications:

S. M. Azimi, O. Simeone, O. Sahin, P. Popovski, "Ultra-Reliable Cloud Mobile Computing with Service Composition and Superposition Coding," in Proc. *Annual Conference on Information Science and Systems (CISS)*, Princeton, NJ, March, 2016.

S. M. Azimi, O. Simeone, R. Tandon, "Fundamental Limits on Latency in Small-Cell Caching Systems: An Information-Theoretic Analysis," in Proc. *IEEE Global Communications Conference (GLOBECOM)*, Washington. D.C., December, 2016.

S. M. Azimi, O. Simeone, A. Sengupta, R. Tandon, "Online Edge Caching in Fog-Aided Wireless Network," in Proc. *IEEE International Symposium on Information Theory (ISIT)*, Barcelona, Spain, July, 2017.

S. M. Azimi, O. Simeone, R. Tandon, "Content Delivery in Fog-Aided Small-Cell Systems with Offline and Online Caching: An Information-Theoretic Analysis," in *Entropy*, vol. 19, no. 7, pp. 1-23, July 2017.

S. M. Azimi, O. Simeone, A. Sengupta, R. Tandon, "Online Edge Caching and Wireless Delivery in Fog-Aided Networks with Dynamic Content Popularity," in *IEEE Journal on Selected Areas in Communications*, vol. 36, no. 7, pp. 1-14, July 2018.

Dedicated to my inspiring parents and sisters, whose love and support cannot be described by words. Without their help, I could have not gone this far and none of this would be possible without their kind and useful guidance.

ACKNOWLEDGMENT

First, I would like to sincerely thank my advisor, Prof. Osvaldo Simeone, for all his help and support throughout my PhD studies. Prof. Simeone is extremely committed to the professional and academic development of his students. He always emphasized the importance of tackling hard problems in a systematic way, and I am sure learning his approach of tackling challenging problems will be very valuable to me later on in my career. In addition, he helped me a lot on my journey to become an independent researcher. His passion for research is always contagious to people who work with him; and his ambition for tackling challenging research problems and breaking the barriers has always been inspiring.

Specifically, I want to thank Prof. Ravi Tandon from University of Arizona for his guidelines, advice on my research, and I appreciate his willingness to work on a tight schedule.

I would also like to thank my committee members and my sincere gratitude to Prof. Alexander Haimovich, Prof. Ali Abdi, and Prof. Joerg Kliewer; and I appreciate their time as well as their encouraging and constructive comments, feedbacks and guides on my dissertation.

Ms. Kathleen Bosco and Ms. Angela Retino deserve a very special acknowledgment from all of students at CWiP. They were always ready to help us and they have made everything easy.

Special thanks go to Hashimoto Fellowship fund and for the financial support during my doctoral studies.

Further thanks go to Ms. Clarisa Gonzalez-Lenahan, the staff of the Graduate Studies office of NJIT and the staff of the Office of Global Initiatives and faculty for their advice, help and support with administrative matters during my PhD studies.

At the end, I want to thank my parents, Tahereh Chamani and Seyyed Esmaeil Azimi, whom I owe all of my achievement to and they will forever be my teachers. I would like to express my deepest gratitude and appreciation to my lovely sisters, Tayyebeh and Elham who always giving me hope no matter how dire the situation. Our bond is stronger than blood.

TABLE OF CONTENTS

TABLE OF CONTENTS
(Continued)

Chapter **Page**

TABLE OF CONTENTS
(Continued)

Chapter **Page**

LIST OF FIGURES

Figure **Page**

LIST OF FIGURES
(Continued)

Figure **Page**

CHAPTER 1

MOTIVATION AND OVERVIEW

The current trend in wireless communication traffic suggests that video will represent 82% of the total mobile data traffic volume by 2021 [1]. Moreover, beside handling the additional traffic, the next wireless standard, i.e., 5G, should be capable of supporting low-latency communication and massive number of devices. The current consensus is that this can be achieved by means of an architectural transformation of wireless network to comply with content data networks such as edge caching and fog-radio access networks (F-RANs).

The aim of this thesis is to address two important questions that arise in the design of the F-RAN: 1) How to optimally utilize F-RAN resources such as cache storage and fronthaul capacity while taking into account the wireless channel impairments? 2) How to evaluate the performance of F-RAN for a realistic set-up with time-varying set of popular files? These issues are addressed from an information-theoretic point of view. To this end, different models are investigated that exemplify various key scenarios of interest.

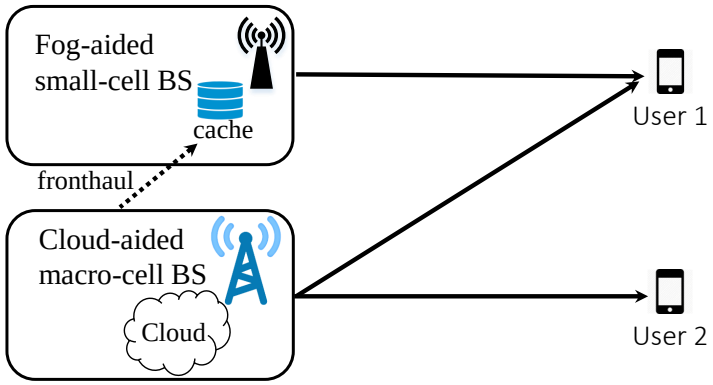

Figure 1.1 Coexistence of fog-aided small-cell BS and cloud-aided macro-BS with one-sided interference channel.

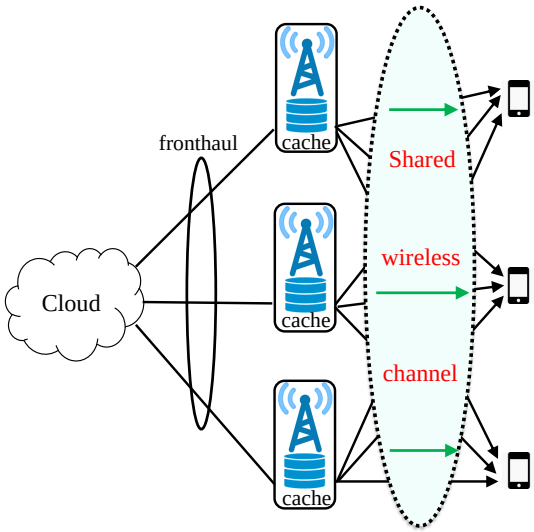

Figure 1.2 F-RAN architecture.

To address the first issue, a single-cell model comprised of one fog-aided small-cell BS and a cloud-aided macro-BS is considered. This model is shown in Figure 1.1. The topology of wireless channel reflects the limited transmission power of small-cell BS as well as high transmission power of macro-BS. Each channel is modeled by an erasure channel. For both time-invariant and time-variant set of popular files the delivery latency of the model is characterized in terms of delivery time per bit (DTB) defined as the duration of the transmission block for reliable delivery.

The second aforementioned issue is tackled by focusing on *online caching*, first introduced by Pedarsani et. al, [3], with the aim of modeling the time-variability of set of popular files. Considering a F-RAN architecture shown in Figure 1.2 in which edge nodes (ENs) has access to the time-varying set of popular file using locally cached popular files as well as central processing at the cloud via fronthaul connection, the delivery latency of the system is evaluated for time-varying set of popular files.

1.1 Organization and Contributions

In this section, the main contributions and organization of the thesis are outlined.

Chapter 2: This Chapter investigates the problem of content delivery for the system model shown in Figure 1.1. The key assumption is that the set of popular files is time-invariant. The performance measure is defined as delivery time per bit (DTB) which measures the number channel uses normalized by the required file size for vanishing probability of error. First, an achieveable scheme is introduced that exploits interference management among edge nodes i.e., fog-aided small-cell and cloud-aided macro-BSs. The resulting upper bound on the DTB is a function of fog system resources, namely cache capacity at small-cell BS as well as the capacity of fronthaul link connecting the cloud to the small-cell BS. Next, a lower bound on the DTB of the system under study is obtained using information theoretic inequalities. The achievable DTB is optimal due to the fact that lower and upper bounds match with each other. The material in this chapter has been reported in the document:

- S. M. Azimi, O. Simeone, R. Tandon, "Fundamental limits on latency in small-cell caching systems: An information-theoretic analysis," in *Proc. IEEE Global Communication Conference (GLOBECOM)*, pp. 1-6, Washington D.C., USA, Dec. 2016.

Chapter 3: In this Chapter, a time-variant set of popular files is considered for the system model shown in Figure 1.1. The performance measure is revised as *long-term DTB* which is the temporal average of DTB. Upper and lower bounds on the long-term DTB are obtained as a function fog system resources as well as the rate of change in the popularity. The key observation is that upper bound on the DTB is finite due to the fact that cloud-aided macro-BS can deliver the requested content thanks to local access to the library of contents. The material in this chapter has been reported in the document:

- S. M. Azimi, O. Simeone, R. Tandon, "Content Delivery in Fog-Aided Small-Cell Systems with Offline and Online Caching: An Information-Theoretic Analysis," *Entropy*, vol. 19, no. 7, pp. 1-23, Jul. 2017.

Chapter 4: In this Chapter, as shown in Figure 1.2, an interference channel at high-SNR regime that connects arbitrary number of ENs to users is considered.

Normalized delivery time (NDT) is introduced as a performance measure that relates the latency of a given scheme normalized by the latency of an interference-free system with unlimited resources. To account for the time-variability of popular set, *long-term NDT* is defined as the temporal average of NDT. For achieveability, different online caching schemes such as C-RAN, reactive, proactive and combination of them as well as the adaptive caching are introduced. A genie-aided argument is used to obtain a lower bound on the long-term NDT of F-RAN system under study. By comparing upper and lower bounds on the long-term NDT, it is shown that the capacity of fronthaul link set a fundamental performance limit on the long-term NDT. The material in this chapter has been presented in the document:

- S. M. Azimi, O. Simeone, A. Sengupta, R. Tandon, "Online edge caching in fog-aided wireless network," in *Proc. IEEE International Symposium on Information Theory* (ISIT), pp. 1217-1221, Aachen, Germany, Jun. 2017.

CHAPTER 2

FUNDAMENTAL LIMITS ON LATENCY IN FOG-AIDED SMALL-CELL SYSTEMS WITH OFFLINE CACHING

Caching of popular multimedia content at small-cell base stations (BSs) is a promising solution to reduce the traffic load of macro-BSs without relying on a high-speed fronthaul architecture. While most prior work analyzed the effect of small-cell caching, or femto-caching, under the assumption of negligible interference between macro-BS and small-cell BS, this chapter contributes to a more recent line of work in which the benefits of caching are reconsidered in the presence of interference on the downlink channel. The key assumption through this chapter is that the set of popular files is time-invariant during delivery phase. This results in offline caching of popular content. Chapter 3 provides a generalization to time-varying set of popular files. Interference channel is modeled by a binary fading one-sided channel in which the small-cell BS, whose transmission is interfered by the macro-BS, has a limited-capacity cache. An information-theoretic metric that captures the delivery latency is defined and fully characterized through information-theoretic achievability and converse arguments as a function of the cache capacity, as well as of the capacity of the fronthaul link connecting cloud and small-cell BS.

2.1 Introduction

Edge or *femto-caching* relies on the storage of popular multimedia content at small-cell base stations (BSs) of a cellular system. This approach has been widely studied in recent years as a means to deliver video files with reduced latency and limited overhead on fronthaul connections to the "cloud" [6, 7]. Caching at the edge can be seen

as an instance of fog networking, whereby storage, computing and communication capabilities are moved closer to the end users [7]. Edge caching has been initially studied for wireless channel models in which small-cell BSs and macro-BSs cannot coordinate their transmissions and hence cannot cooperatively manage their mutual interference (see [6, 7] and references therein). In contrast, recent work in [8, 9] addresses the possibility of interference management among edge nodes, such as small-cell and macro-BSs, based on the respective cached contents.

State of the Art: The papers [8,9] proposed caching and transmission schemes that enables coordination and cooperation at the BSs based on the cached contents for a system with three BSs and three users. The performance of these schemes was evaluated in terms of the information-theoretic high signal-to-noise ratio (SNR) metric of the degrees of freedom (DoF), or, more precisely, of its inverse, as a function of the cache capacity of the BSs. More recent research in [10] provided an operational meaning for the inverse of the degrees of freedom metric used in [8, 9] in terms of delivery latency, and derived a lower bound on the resulting metric, known as Normalized Delivery Time (NDT), for a general system with any number of BSs and users. The delivery coding latency, henceforth delivery latency, measures the duration of the transmission block. A scenario in which both BSs and users have cache storage is considered in [11, 12] under one-shot linear transmission and in [13] under several transmission schemes for both centralized and decentralized caching strategies. It is proved that both BSs and users' caches have the same quantitative contribution to the achievable sum-DoF. Naderializadeh et al. [14] proposed a universal scheme for content placement and delivery which is independent of underlying communication networks and is order-optimal in the high-SNR regime. In [15], upper and lower bounds on the NDT of cache-aided MIMO interference channels are provided.

In [16, 17] the analysis in [8–10] was generalized to study a system in which a cloud server is connected to the BSs via finite-capacity fronthaul links and can

compensate for partial caching of the library of files at the BSs. This system was referred to as *Fog-Radio Access Networks* (F-RAN). The minimum NDT latency metric was characterized within a multiplicative factor of 2 in [17] as a function of the cache and fronthaul capacity by developing achievability and converse arguments. Other works on NDT characterization include [18–21]. In [18], a scenario with a multicast fronthaul is studied. In [19], decentralized content placement and file delivery are considered for a F-RAN system with caching at both BSs and users. Reference [20] studies the achievable NDT region to account for heterogenous requirements on the delivery of different files. Kakar et al. [21] considered the set-up in [2] under linear deterministic channel model to provide upper and lower bounds on the NDT. The optimization of linear processing and often signal processing aspects of F-RAN systems are considered in [22–26].

Main Contributions: In this chapter, the F-RAN model in Figure 2.1 is considered, which includes a small-cell BS and a macro-BS, represented by Encoder 1 and Encoder 2, respectively. The small-cell BS (Encoder 1) is equipped with a cache of finite capacity and can serve a small-cell mobile user, represented by Decoder 1. The macro-BS (Encoder 2) can serve a macro-cell user, namely Decoder 2, as well as, possibly, also Decoder 1. The transmission from the macro-BS (Encoder 2) to Decoder 2 interferes with Decoder 1. It is assumed that the small-cell BS transmits with sufficiently small power so as not to create interference at Decoder 2, which is modeled here as a partially connected wireless channel.

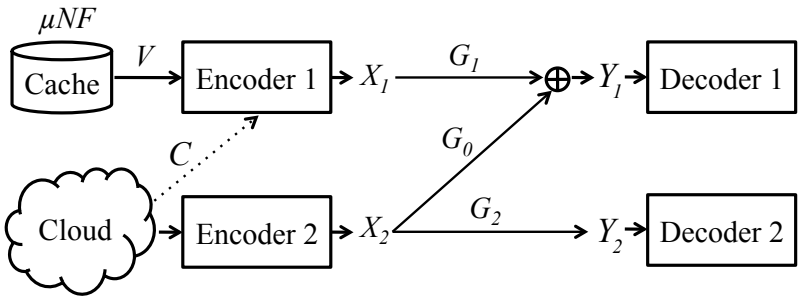

Figure 2.1 Cloud and edge-aided data delivery over binary fading interference channels.

The main contributions of this chapter are as follows:

- An information-theoretic formulation for the analyses of the system in Figure 2.1 is presented that centers on the characterization of the delivery coding latency measured in terms of the Delivery Time per Bit (DTB), for offline caching. The system model is based on a one-sided interference channel.
- Assuming a fixed set of popular contents, the minimum DTB for the system in Figure 2.1 is obtained as a function of the cache capacity at Encoder 1 and the capacity of the fronthaul link that connects the cloud to Encoder 1 in the offline setting.

Notation: Throughout this chapter, given $a > 0$, a set is denoted by $[a] = \{1, 2, ..., \lceil a \rceil\}$. For any probability p, the complementary probability is defined as $\bar{p} = 1 - p$.

2.2 System Model for Offline Caching

In this section, the fog-aided system depicted in Figure 2.1 is studied . Let $\mathcal{L} = \{W_1, ..., W_N\}$ be a static library of N files. Each file is independent and identically distributed according to uniform distribution, so that $W_i \sim \mathcal{U}([2^F])$, for $i \in [N]$, where F is the file size in bits. Encoder 1, which models a small-cell BS, has a local cache and is able to store μNF bits. The parameter μ, with $0 \leq \mu \leq 1$, is hence the fractional cache size and represents the portion of library that can be stored at the cache. Encoder 2, which models a macro-BS, can access the entire library $\mathcal{L}$ thanks

to its direct connection to the cloud. Encoder 1 is also connected to the cloud but only through a rate-limited link of capacity C bits per channel use. First, the scenario of edge-aided offline caching with $C = 0$ is considered. Hence, Encoder 1 does not have access to the cloud. Then, the analysis is extend to cloud and edge-aided offline caching, i.e., when $C \geq 0$.

It is assumed that encoders and decoders are connected by a binary fading interference channel, previously studied in [27, 28]. This model represents a special case of the deterministic linear model of [29] as generalized to account for random fading (see [30]). As illustrated in Figure 2.1, the signal received at Decoder 1 and Decoder 2 at time t can be written as

$$\begin{aligned} Y_1(t) &= G_1(t)X_1(t) \oplus G_0(t)X_2(t) \\ Y_2(t) &= G_2(t)X_2(t), \end{aligned} \tag{2.1}$$

where $\mathbf{G}(t) = (G_0(t), G_1(t), G_2(t)) \in \{0, 1\}^3$ is the vector of binary channel coefficients at time t, and $X_1(t)$ and $X_2(t)$ are the binary transmitted signals from Encoder 1 and Encoder 2, respectively. In (2.1), all operations are in the binary field. The channel gains are distributed as $G_1(t) \sim$ Bernoulli(ϵ_1) and $G_0(t), G_2(t) \sim$ Bernoulli(ϵ_2), are mutually independent and change independently over time. The parameters ϵ_1 and ϵ_2 describes the average quality of the communication links originating at Encoder 1 and Encoder 2, respectively, and are hence in practice related to the transmission powers of Encoder 1 and Encoder 2. It should be noted that a more general model with different erasure probabilities for the links $G_0(t)$ and $G_2(t)$ could also be considered but at the expense of a more cumbersome notation and analysis, which is not further pursued here.

Each user, or decoder, k requests a file W_{d_k} from the library $\mathcal{L}$ at every transmission interval for $k = 1, 2$. The demand vector is defined as $\mathbf{d} = (d_1, d_2) \in [N]^2$. In the next two subsections, first the edge-aided scenario is described and then it is generalized to the cloud and edge-aided system.

2.2.1 Edge-Aided Offline Caching

The edge-aided small-cell system corresponds to the case with $C = 0$ in Figure 2.1. The system operates according to the following two phases.

(1) *Placement phase*: The placement phase is defined by functions $\phi_i(\cdot)$, at Encoder 1, which maps each file $W_i \in \mathcal{L}$ to its cached version V_i

$$V_i = \phi_i(W_i) \quad \forall i \in \{1, ..., N\}. \tag{2.2}$$

To satisfy cache storage constraint, it is required that

$$H(V_i) \leq \mu F. \tag{2.3}$$

The total cache content at encoder 1 is given by

$$V = (V_1, ..., V_N). \tag{2.4}$$

Note that, as in [10, 16], the focus is on the caching strategies that allow for arbitrary intra-file coding but not for inter-file coding as per (2.2). Furthermore, the caching policy is kept fixed over multiple transmission intervals and is thus

independent of the receivers' requests and of the channel realizations in the transmission intervals.

(2) *Delivery phase*: The delivery phase is in charge of satisfying the given request vector $\mathbf{d}$ in each transmission interval given the current channel realization. For simplicity of exposition, it is assumed that full Channel State Information (CSI) is available throughout the transmission block, although this is not required by achievable schemes that will be proven to be optimal (see Remark 2.1). Note that in practice non-causal CSI for the coding block can be justified for multi-carrier transmission schemes, such as OFDM, in which index t runs over the subcarriers. It is defined by the following two functions.

Encoding: Encoder 1 uses the encoding function

$$\psi_1 : [2^{\mu NF}] \times [N]^2 \times \{0,1\}^{3T} \to \{0,1\}^T \tag{2.5}$$

which maps the cached content V, the demand vector $\mathbf{d}$ and the CSI sequence $\mathbf{G}^T = (\mathbf{G}(1), ..., \mathbf{G}(T))$ to the transmitted codeword $X_1^T = (X_1[1], ..., X_1[T]) = \psi_1(V, \mathbf{d}, \mathbf{G}^T)$. Note that T represents the duration of transmission in channel uses. Encoder 2 uses the following encoding function

$$\psi_2 : [2^{NF}] \times [N]^2 \times \{0,1\}^{3T} \to \{0,1\}^T \tag{2.6}$$

which maps the library $\mathcal{L}$ of all files, the demand vector $\mathbf{d}$, and the CSI vector $\mathbf{G}^T$ to the transmitted codeword $X_2^T = (X_2[1], ..., X_2[T]) = \psi_2(\mathcal{L}, \mathbf{d}, \mathbf{G}^T)$.

Decoding: Each decoder $j \in \{1, 2\}$ is defined by the following mapping

$$\eta_j : \{0,1\}^T \times [N]^2 \times \{0,1\}^{3T} \to [2^F] \tag{2.7}$$

which outputs the detected message $\hat{W}_{d_j} = \eta_j(Y_j^T, \mathbf{d}, \mathbf{G}^T)$ where $Y_j^T = (Y_j(1), ..., Y_j(T))$ is the received signal (2.1) at receiver j.

A selection of caching, encoding, and decoding functions in (2.5)–(2.7) is referred as a policy. The probability of error is evaluated with respect to the worst-case demand vector and decoder as

$$P_e^F = \max_{\mathbf{d} \in [N]^2} \max_{j \in \{1,2\}} \Pr(\hat{W}_{d_j} \neq W_{d_j}). \tag{2.8}$$

The delivery time per bit (DTB) of a code is defined as T/F and is measured in channel symbols per bit. A DTB δ_{off} is said to be *achievable* if there exists a sequence of policies indexed by the file size F for which the limits

$$\lim_{F \to \infty} \frac{T}{F} = \delta_{\text{off}}(\mu) \tag{2.9}$$

and $P_e^F \to 0$ as $F \to \infty$ hold. The subscript "off" represents the fact that DTB is defined for offline caching. The *minimum DTB* $\delta_{\text{off}}^*(\mu)$ is the infimum of all achievable DTB when the fractional cache capacity at encoder 1 is equal to μ.

2.2.2 Cloud and Edge-Aided Offline Caching

In this section, the model described above is generalized to the case in which there is a link with capacity $C \geq 0$ between Cloud and Encoder 1. The content placement

phase is the same as Section 2.2.1. In the delivery phase, the Cloud implements an encoding function

$$\psi_C : [2^{NF}] \times [N]^2 \times \{0,1\}^{3T} \to [2^{T_C C}], \tag{2.10}$$

which maps the library $\mathcal{L}$ of all files, the demand vector $\mathbf{d}$ and the CSI vector $\mathbf{G}^T$ to the signal $U^{T_C} = (U_1, ..., U_{T_C}) = \psi_C(\mathcal{L}, \mathbf{d}, \mathbf{G}^T)$ to be delivered to Encoder 1. Here, parameter T_C represents the duration of the transmission from Cloud to Encoder 1 in terms of number of channel uses of the fading channel from encoders to decoders. The inequality $H(U_i) \leq C$ for $i \in [T_C]$ represents the capacity limitations on the Cloud-to-Encoder 1 link. Furthermore, Encoder 1 uses the encoding function

$$\psi_1 : [2^{\mu NF}] \times [2^{T_C C}] \times [N]^2 \times \{0,1\}^{3T} \to \{0,1\}^T, \tag{2.11}$$

which maps the cached content V, the received signal U^{T_C}, the demand vector $\mathbf{d}$ and the CSI sequence $\mathbf{G}^T = (\mathbf{G}(1), ..., \mathbf{G}(T))$ to the transmitted codeword $X_1^T = (X_1[1], ..., X_1[T]) = \psi_1(V, U^{T_C}, \mathbf{d}, \mathbf{G}^T)$. Note that, as for the edge-aided case, it is assumed that non-causal CSI is available at both cloud and edge. As discussed, this is a sensible assumption for multi-carrier modulation schemes. However, as indicated in Remark 2.2, it will be proven that the optimal strategy requires only causal CSI at the encoders and no CSI at the cloud. As above, T represents the duration of transmission on the binary fading channel in channel uses.

Decoding and probability of error are defined as in Section 2.2.1. Instead, a DTB δ_{off} is said to be achievable if there exists a sequence of policies, defined by (2.2), (2.6), (2.7), (2.10) and (2.11) and indexed by F, such that the limits:

$$\lim_{F\to\infty} \frac{T + T_C}{F} = \delta_{\text{off}}(\mu, C) \tag{2.12}$$

and $P_e^F \to 0$ as $F \to \infty$ hold. The *minimum DTB* $\delta^*_{\text{off}}(\mu, C)$ is the infimum of all achievable DTBs when the fractional cache size at Encoder 1 is equal to μ and the Cloud-to-Encoder 1 capacity is equal to C.

2.3 Minimum DTB under Offline Caching

In this section, first the minimum DTB for edge-aided offline caching scenario is characterized. Then, the minimum DTB for the cloud and edge-aided system is derived.

2.3.1 Edge-Aided System ($C = 0$)

In this subsection, the minimum DTB $\delta^*_{\text{off}}(\mu)$ for the system in Figure 2.1 with $C = 0$ is derived.

Proposition 2.1. *The minimum DTB for the fog-aided system in Figure 2.1 with* $C = 0$ *is*

$$\delta^*_{\text{off}}(\mu) = \begin{cases} \frac{2-\mu}{1-\epsilon_2^2} & \text{if } \mu \leq \mu_0 \\ \delta_0 & \text{if } \mu \geq \mu_0, \end{cases} \tag{2.13}$$

where μ_0 *and* δ_0 *are given by*

$$\mu_0 = \begin{cases} 1 - \epsilon_2 & \text{if } \bar{\epsilon}_1\epsilon_2 > \bar{\epsilon}_2^2\epsilon_1 \\ \frac{2(1-\epsilon_1)(\epsilon_2^2-\epsilon_2+1)}{2-\epsilon_1-\epsilon_2+\epsilon_1\epsilon_2-\epsilon_1\epsilon_2^2} & \text{if } \bar{\epsilon}_1\epsilon_2 \leq \bar{\epsilon}_2^2\epsilon_1 \end{cases} \tag{2.14}$$

and

$$\delta_0 = \max\Big(\frac{1}{1-\epsilon_2}, \frac{2}{2-\epsilon_1-\epsilon_2+\epsilon_1\epsilon_2-\epsilon_1\epsilon_2^2}\Big). \tag{2.15}$$

Proof. The converse is presented in Appendix A.1, and the achievable scheme is presented next. □

To provide some insights obtained from the result in Proposition 2.1, consider first the set-up in which Encoder 1 has no caching capabilities, i.e., $\mu = 0$. In this case, Encoder 2 needs to deliver the requested files to both decoders on a binary erasure broadcast channel. Considering the worst-case in which two different files are requested by two decoders, the minimum average time to serve both users is $T = 2F/(1-\epsilon_2^2)$, since with probability $(1-\epsilon_2^2)$ a bit can be delivered to either Decoder 1 or Decoder 2 by Encoder 2, yielding a minimum DTB of $\delta^*_{\text{off}}(0) = 2/(1-\epsilon_2^2)$. In contrast, when the entire library is available at Encoder 1, i.e., $\mu = 1$, depending on the relative values of ϵ_1 and ϵ_2, two different cases should be distinguished. Roughly speaking, if the channel between Encoder 2 and the Decoders is weaker on average than the channel between Encoder 1 and Decoder 1, or more precisely if $\bar{\epsilon}_1 \geq \bar{\epsilon}_2$, then the minimum DTB is limited by transmission delay to Decoder 2 and the minimum DTB is $\delta^*_{\text{off}}(1) = 1/(1-\epsilon_2)$. Instead, when the channel between Encoder 1 and Decoder 1 is weaker on average than the channel between Encoder 2 and both decoders, or $\bar{\epsilon}_1 \leq \bar{\epsilon}_2$, the resulting minimum DTB depends on both ϵ_1 and ϵ_2. In both cases, Encoder 2 serves a fraction $(1-\mu_0)$ of the requested file to Decoder 1, so that Encoder 1 only needs to deliver a fraction μ_0 of the requested file by Decoder 1.

As will be detailed below, a key element of the transmission policies is that, in the channel state in which all three links are active, the presence of the cache at Encoder 1 allows the latter to coordinate its transmission with Encoder 2 and cancel the interference caused by Encoder 2 to Decoder 1. Furthermore, from the discussion above, a fractional cache size $\mu \geq \mu_0$ is sufficient to achieve the same DTB δ_0 as with full caching. Figure 3.1 shows the value μ_0 as a function of ϵ_1 for different values of ϵ_2. It is observed that, for fixed ϵ_2, the fraction μ_0 decreases with ϵ_1, showing

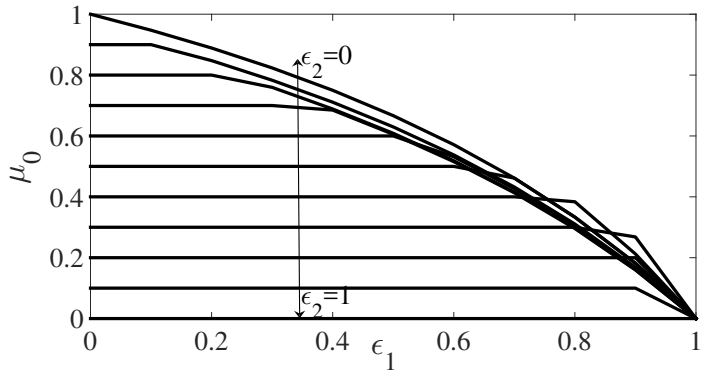

Figure 2.2 Optimum fractional cache size μ_0 as a function of ϵ_1 for different values of ϵ_2, which ranges from 0 to 1 with step size 0.1.

that an Encoder 1 with a low channel quality cannot benefit from a large cache size. Furthermore, as the channel from Encoder 2 becomes more reliable, i.e., for small ϵ_2, a larger cache at Encoder 1 enables the latter to coordinate more effectively with Encoder 2, hence improving the DTB.

Remark 2.1. The achievable schemes proposed above only require the encoders to know the current state of the CSI, i.e., at each time t, only the CSI $\mathbf{G}(t)$ is needed. As a result, even if the encoders know only the current CSI, as well as the CSI statistics, the optimal performance is the same as for the case in which the entire sequence $\mathbf{G}^T$ is known as per definition (2.5)–(2.6).

Proof of Achievability Here, details on the policies that achieve the minimum DTB identified in Proposition 2.1 is provided. First, it is proved that the minimum DTB $\delta^*_{\text{off}}(\mu)$ is a convex function of μ. The proof leverages the splitting of files into subfiles delivered using different strategies via time sharing.

Lemma 2.1. *The minimum DTB $\delta^*_{\text{off}}(\mu)$ is a convex function of $\mu \in [0, 1]$.*

Proof. Consider two policies that require fractional cache sizes μ_1 and μ_2 and achieve DTBs δ_1 and δ_2, respectively. Given a fractional cache size $\mu = \alpha\mu_1 + (1-\alpha)\mu_2$ for any $\alpha \in [0, 1]$, the system can operate by splitting each file into two parts, one of size αF and the other of size $(1-\alpha)F$, while satisfying the cache constraints. The first

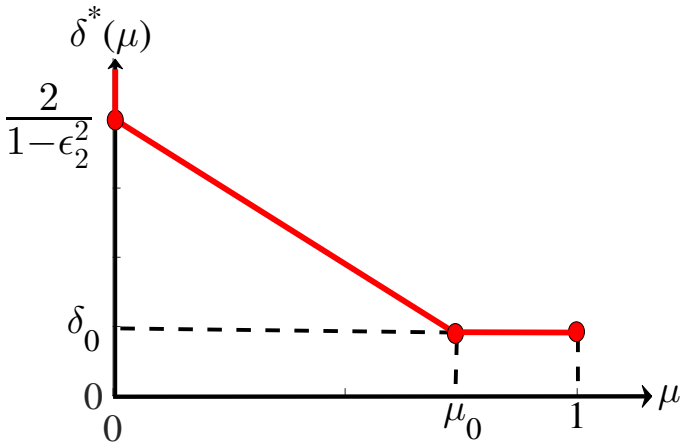

Figure 2.3 Minimum Delivery Time per Bit (DTB) $\delta^*_{\text{off}}(\mu)$ for the system in Figure 2.1 with $C = 0$.

fraction of the files is delivered following the first policy, while the second fraction is delivered using the second policy. Since the delivery time is additive over the two file fractions, the DTB $\delta = \alpha\delta_1 + (1-\alpha)\delta_2$ is achieved. □

By the convexity of $\delta^*_{\text{off}}(\mu)$ proved in Lemma 2.1, it suffices to prove that the corner points $(\mu = 0,\ \delta^*_{\text{off}}(0) = 2/(1-\epsilon_2^2))$ and $(\mu = \mu_0, \delta_0)$ are achievable. In fact, the minimum DTB $\delta^*_{\text{off}}(\mu)$ can then be achieved, following the proof of Lemma 2.1, by file splitting and time sharing between the optimal policies for $\mu = 0$ and $\mu = \mu_0$ in the interval $0 \le \mu \le \mu_0$ and by using the optimal policy for $\mu = \mu_0$ in the interval $\mu_0 \le \mu \le 1$ (see Figure 2.3).

In the following, the notation $(g_0, g_1, g_2) \in \{0,1\}^3$ identifies the channel realization $(G_0 = g_0, G_1 = g_1, G_2 = g_2)$. For instance, $(0,1,1)$ represents the channel realization in which $Y_1 = X_1$ and $Y_2 = X_2$, and $(1,0,1)$ that in which $Y_1 = X_2$ and $Y_2 = X_2$.

No Caching ($\mu = 0$): First, the corner point $(\mu = 0, \delta^*_{\text{off}}(0) = 2/(1-\epsilon_2^2))$ is considered. In this setting, in which Encoder 1 has no caching capabilities, the model reduces to a broadcast erasure channel from Encoder 2 to both decoders. The worst-case demand vector is any one in which the decoders request different files. In fact, if the same file is requested, it can always be treated as two distinct

files achieving the same latency as for a scenario with distinct files. Focusing on this worst-case scenario, the following delivery policy is adopted.

Encoder 1 always transmits $X_1 = 0$. Encoder 2 transmits 1 bit of information to Decoder 1 in the states $(1, 0, 0)$ and $(1, 1, 0)$, in which the channel from Encoder 2 to Decoder 1 is on while the channel to Decoder 2 is off. It transmits 1 bit of information to Decoder 2 in the states $(0, 0, 1)$ and $(0, 1, 1)$, in which the channel to Decoder 2 is on while the channel to decoder 1 is off. Instead, in states $(1, 0, 1)$ and $(1, 1, 1)$, in which both channels to Decoder 1 and Decoder 2 are on, Encoder 2 transmits 1 bit of information to Decoder 1 or to Decoder 2 with equal probability.

Consider now the time T_1 required for Decoder 1 to decode successfully F bits. This random variable can be written as

$$T_1 = \sum_{k=1}^{F} T_{1,k}, \tag{2.16}$$

where $T_{1,k}$ denotes the number of channel uses required to transmit the kth bit. Given the discussion above, the variables $T_{1,k}$ are independent for $k \in [F]$ and have a Geometric distribution with mean $(\Pr[\mathbf{G} = (1, 0, 0)] + \Pr[\mathbf{G} = (1, 1, 0)] + 1/2\Pr[\mathbf{G} = (1, 0, 1)] + 1/2 \Pr[\mathbf{G} = (1, 1, 1)])^{-1} = 2/(1 - \epsilon_2^2)$. By the strong law of large numbers we now have the limit

$$\lim_{F\to\infty} \frac{T_1}{F} = E[T_1] = \frac{2}{1 - \epsilon_2^2} \tag{2.17}$$

with probability 1. In a similar manner, the resulting delivery time for Decoder 2 for any given bit has a Geometric distribution with mean $(\Pr[\mathbf{G} = (0, 0, 1)] + \Pr[\mathbf{G} = (0, 1, 1)] + \frac{1}{2}\Pr[\mathbf{G} = (1, 0, 1)] + \frac{1}{2} \Pr[\mathbf{G} = (1, 1, 1)])^{-1} = 2/(1 - \epsilon_2^2)$; and, by the strong law of large numbers, we obtain that the time T_2 needed to transmit F bits to Decoder 2 satisfies the limit $\lim_{F\to\infty} \frac{T_2}{F} = E[T_2] = \frac{2}{1-\epsilon_2^2}$ almost surely. Using this

limit along with (2.17) allows to conclude that there exists a sequence of policies with $T/F \to 2/(1-\epsilon_2^2)$ for any arbitrarily small probability of error.

Partial Caching ($\boldsymbol{\mu = \mu_0}$) with $\boldsymbol{\bar{\epsilon}_1\epsilon_2 \geq \epsilon_1\bar{\epsilon}_2^2}$: Next, we consider the corner point (μ_0, δ_0) under the condition $\bar{\epsilon}_1\epsilon_2 \geq \epsilon_1\bar{\epsilon}_2^2$. In this case, in which Encoder 1 has a better channel than Decoder 2 in the average sense discussed above, our findings show that Encoder 2 should communicate to Decoder 1 only in the channel states in which the channel to Decoder 2 is off. Using these states, Encoder 2 sends $(1-\mu_0)F$ bits to Decoder 1. Encoder 1 cache a fraction μ_0 of each file in the library and delivers $\mu_0 F$ bits of the requested file to Decoder 1. For this purpose, coordination between Encoder 1 and Encoder 2 is needed to manage interference in the state $(1,1,1)$ in which all links are on.

A detailed description of the transmission strategy is provided below as a function of the channel state $\mathbf{G}$.

(1) $\mathbf{G} = (0,0,1)$: Only the channel between Encoder 2 and Decoder 2 is active, and Encoder 2 transmits 1 bit of information to Decoder 2.

(2) $\mathbf{G} = (0,1,0)$: The only active channel is between Encoder 1 and Decoder 1, and Encoder 1 transmits 1 information bit to Decoder 1.

(3) $\mathbf{G} = (0,1,1)$: The cross channel is off, and each encoder transmits 1 bit of information to its decoder.

(4) $\mathbf{G} = (1,0,0)$: Only the channel between Encoder 2 and Decoder 1 is active, and Encoder 2 transmits 1 bit of information to Decoder 1.

(5) $\mathbf{G} = (1,0,1)$: The direct channel between Encoder 1 and Decoder 1 is off, while two other channels are on. Encoder 2 transmits 1 bit of information to Decoder 2.

(6) $\mathbf{G} = (1, 1, 0)$: Both channels from Encoder 1 and Encoder 2 to Decoder 1 are on. Encoder 1 transmits $X_1 = 0$ and Encoder 2 transmits 1 bit of information to Decoder 1.

(7) $\mathbf{G} = (1, 1, 1)$: Encoder 2 transmits 1 bit X_2 of information to Decoder 2. Encoder 1 transmits $X_1 = \tilde{X}_1 \oplus X_2$, where $\tilde{X}_1$ is an information bit for Decoder 1. This form of coordination is enabled by the fact that Encoder 1 knows the bit X_2, since it is part of the $\mu_0 F$ cached bits from the file requested by Decoder 2. In this way, interference from Encoder 2 is cancelled at Decoder 1.

From the previous discussion, Encoder 2 transmits 1 bit of information to Decoder 2 in the states (1), (3), (5) and (7). For large F, the normalized transmission delay for transmitting the requested file to Decoder 2 is then equal to

$$\begin{aligned}\delta_{22} = &\Big(\Pr[\mathbf{G} = (0, 0, 1) + \Pr[\mathbf{G} = (0, 1, 1)] \\ &+ \Pr[\mathbf{G} = (1, 0, 1)] + \Pr[\mathbf{G} = (1, 1, 1)]\Big)^{-1} \\ = &\frac{1}{\bar{\epsilon}_2}. \end{aligned} \tag{2.18}$$

Furthermore, Encoder 2 transmits $(1 - \mu_0)F$ bits to decoder 1 in the states at (4) and (6). The required normalized time for large F is hence

$$\delta_{21} = \frac{1 - \mu_0}{\epsilon_2 \bar{\epsilon}_2} \tag{2.19}$$

Finally, Encoder 1 transmits $\mu_0 F$ bits to Decoder 1 in the states at (2), (3) and (7). The required time is thus

$$\delta_{11} = \frac{\mu_0}{\bar{\epsilon}_1 \bar{\epsilon}_2 + \bar{\epsilon}_1 \epsilon_2^2} \tag{2.20}$$

It can be shown that $\delta_{11} \leq \delta_{21} = \delta_{22} = \delta_0$ under the given condition $\bar{\epsilon}_1\epsilon_2 \geq \epsilon_1\bar{\epsilon}_2^2$, and hence the DTB is given by $\max(\delta_{11}, \delta_{21}, \delta_{22}) = \delta_0$.

Partial Caching ($\boldsymbol{\mu = \mu_0}$) with $\boldsymbol{\bar{\epsilon}_1\epsilon_2 \leq \epsilon_1\bar{\epsilon}_2^2}$: Finally, we consider the corner point (μ_0, δ_0) under the complementary condition $\bar{\epsilon}_1\epsilon_2 \leq \epsilon_1\bar{\epsilon}_2^2$, in which Encoder 2 has better channels to the decoders. In this case, as above, Encoder 1 caches a fraction μ_0 of all files. Transmission take place as described in the previous case except for state (5) which is modified as follows: (5) $\mathbf{G} = (1, 0, 1)$: Encoder 2 transmits 1 bit of information to either Decoder 1 or Decoder 2 with probabilities $\alpha = (1 - \bar{\epsilon}_1\epsilon_2/\epsilon_1\bar{\epsilon}_2^2)/2$ and $1 - \alpha$, respectively.

Encoder 2 hence transmits 1 bit of information to Decoder 2 in the states at cases (1), (3) and (7) and also with probability $1 - \alpha$ in case (5). For large F, the normalized transmission delay for transmitting the requested file to Decoder 2 tends to

$$\begin{aligned}\delta_{22} = \Big(&\Pr[\mathbf{G} = (0,0,1) + \Pr[\mathbf{G} = (0,1,1)] + \Pr[\mathbf{G} = (1,1,1)] \\ &+ (1-\alpha)\Pr[\mathbf{G} = (1,0,1)]\Big)^{-1} = \frac{2}{2 - \epsilon_1 - \epsilon_2 + \epsilon_1\epsilon_2 - \epsilon_1\epsilon_2^2}.\end{aligned} \tag{2.21}$$

In addition, Encoder 2 transmits 1 bit to Decoder 1 in cases (4) and (6) as well as in case (5) with probability α. The required time to transmit $(1 - \mu_0)F$ bits from Encoder 2 to Decoder 1 is hence

$$\delta_{21} = \frac{1 - \mu_0}{\epsilon_2\bar{\epsilon}_2 + \frac{1}{2}(\epsilon_1\bar{\epsilon}_2^2 - \bar{\epsilon}_1\epsilon_2)}. \tag{2.22}$$

It can be shown that $\delta_{11} = \delta_{21} = \delta_{22} = \delta_0$, where δ_{11} is given in (2.20) under the given condition $\bar{\epsilon}_1\epsilon_2 \leq \epsilon_1\bar{\epsilon}_2^2$, yielding the DTB $\max(\delta_{11}, \delta_{21}, \delta_{22}) = \delta_0$. This concludes the proof of achievability.

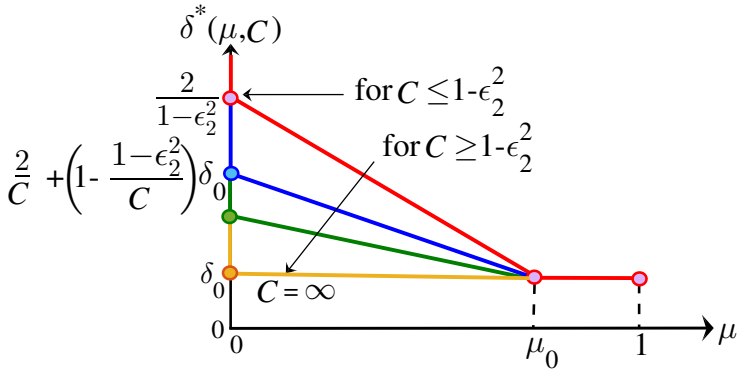

Figure 2.4 Minimum Delivery Time per Bit (DTB) $\delta^*_{\text{off}}(\mu, C)$ for the system in Figure 2.1.

2.3.2 Cloud and Edge-Aided System ($C \geq 0$)

In the following proposition, we derive the minimum DTB $\delta^*_{\text{off}}(\mu, C)$ for the system in Figure 2.1 with $C \geq 0$.

Proposition 2.2. *The minimum DTB for the fog-aided system in Figure 2.1 is:*

$$\delta^*_{\text{off}}(\mu, C) = \delta^*_{\text{off}}(\mu), \tag{2.23}$$

if $C \leq 1 - \epsilon_2^2$. *Otherwise, it is given by:*

$$\delta^*_{\text{off}}(\mu, C) = \begin{cases} \frac{2-\mu}{C} + \left(1 - \frac{1-\epsilon_2^2}{C}\right)\delta_0 & \text{if } \mu \leq \mu_0 \\ \delta_0 & \text{if } \mu \geq \mu_0, \end{cases} \tag{2.24}$$

where $\delta^*_{off}(\mu)$, μ_0 *and* δ_0 *are defined in* (2.13), (2.14) *and* (2.15), *respectively.*

Proof. See below and Appendix A.2. □

Figure 2.4 shows the minimum DTB as a function of μ and C. To elaborate on the results in Proposition 2.2, we focus first on the setting in which Encoder 1 has no

caching capability, i.e., $\mu = 0$. In this case, unlike the scenario studied in the previous section, Encoder 1 can deliver part of the file requested by Decoder 1 through the connection to the Cloud. Nevertheless, if $C \leq 1 - \epsilon_2^2$, that is, if the average delay for transmission of 1 bit from cloud to Encoder 1, namely $1/C$, is larger than the corresponding delay between Encoder 2 and both decoders, namely $1/(1 - \epsilon_2^2)$, then it is optimal to neglect Encoder 1 and operate as discussed in Section 2.3.1. Instead, if $C \geq 1 - \epsilon_2^2$, it is optimal for Encoder 1 to transmit parts of the requested files, or functions thereof, which are received from the cloud. In fact, as discussed below, it is necessary for the cloud to transmit a coded signal obtained from both the files requested by the users in order to obtain the DTB in Proposition 2.2. Moreover, if the fractional cache size satisfies the inequality $\mu \geq \mu_0$, then the cache size at Encoder 1 is sufficient to achieve the DTB δ_0 corresponding to full caching and the Cloud-to-Encoder 1 link can be neglected with no loss of optimality.

Proof of Achievability In this section, we detail the policies that achieve the minimum DTB described in Proposition 2.2. We start by noting that for $C \leq 1 - \epsilon_2^2$, the achievability of the DTB follows from Proposition 2.1, and hence we can concentrate on the case $C \geq 1 - \epsilon_2^2$. We first note that the minimum DTB $\delta_{\text{off}}^*(\mu, C)$ is a convex function of μ for any value of C. The proof follows as in Lemma 2.1 by file splitting and time sharing and is hence omitted.

Lemma 2.2. *The minimum DTB $\delta_{\text{off}}^*(\mu, C)$ is a convex function of $\mu \in [0, 1]$ for any given value of $C \geq 0$.*

By the convexity of $\delta_{\text{off}}^*(\mu, C)$ in Lemma 2.2, and by the achievability of the DTB in Proposition 2.1 with $C = 0$, and hence also for $C \geq 0$, it suffices to prove that the corner point $\delta_{\text{off}}^*(0, C) = 2/C + (1 - (1 - \epsilon_2^2)/C)\delta_0$ is achievable for $C \geq 1 - \epsilon_2^2$. To this end, we consider the worst case in which each decoder requests a different file, and we adopt the following policy.

The Cloud-to-Encoder 1 link is used for a normalized time $\delta_C = T_C/F = (2 - \delta_0(1-\epsilon_2^2))/C$ to transmit ρF bits from the file requested by Encoder 1, with

$$\rho = 2 - \delta_0(1-\epsilon_2^2). \tag{2.25}$$

Of these bits, $\rho F\bar{\epsilon}_1\epsilon_2/(\bar{\epsilon}_1\epsilon_2 + \bar{\epsilon}_1\bar{\epsilon}_2^2)$ bits are sent to Encoder 1 by the Cloud in an uncoded form. Instead, the remaining $\rho F\bar{\epsilon}_1\bar{\epsilon}_2^2/(\bar{\epsilon}_1\epsilon_2 + \bar{\epsilon}_1\bar{\epsilon}_2^2)$ bits are transmitted by XORing each bit of the file with the corresponding bit of the file requested by Decoder 2. The mentioned ρF bits are sent to Decoder 1 by Encoder 1, while the remaining $(1-\rho)F$ bits are sent by Encoder 2 to Decoder 1, as discussed next.

The transmission strategy follows the approach described in Section 2.3.1. As for (2.20) the transmission of uncoded bits from Encoder 1 to Decoder 1 requires a normalized time on the channel

$$\delta_{11}^u = \frac{\rho}{\bar{\epsilon}_1\epsilon_2 + \bar{\epsilon}_1\bar{\epsilon}_2^2}. \tag{2.26}$$

while the transmission of coded bits requires time

$$\delta_{11}^c = \frac{\rho}{\bar{\epsilon}_1\epsilon_2 + \bar{\epsilon}_1\bar{\epsilon}_2^2}. \tag{2.27}$$

Similar to (2.19) and (2.22), the time required for Encoder 2 to transmit to Decoder 1 is

$$\delta_{21} = \begin{cases} \frac{1-\rho}{\epsilon_2\bar{\epsilon}_2} & \text{if } \bar{\epsilon}_1\epsilon_2 > \bar{\epsilon}_2^2\epsilon_1 \\ \frac{1-\rho}{\epsilon_2\bar{\epsilon}_2 + \frac{1}{2}(\epsilon_1\bar{\epsilon}_2^2 - \bar{\epsilon}_1\epsilon_2)} & \text{if } \bar{\epsilon}_1\epsilon_2 \leq \bar{\epsilon}_2^2\epsilon_1 \end{cases} \tag{2.28}$$

while $\delta_{22} = \delta_0$ is sufficient to communicate to Decoder 2. Under the channel conditions $\bar{\epsilon}_1\epsilon_2 > \bar{\epsilon}_2^2\epsilon_1$, from (2.25), (2.26) and (2.28), it can be shown that $\delta_{11}^u = \delta_{11}^c \leq \delta_{21} = \delta_{22} = \delta_0$. Therefore, the normalized time required on the edge channel is δ_E=max$(\delta_{11}^u, \delta_{11}^c, \delta_{21}, \delta_{22}) = \delta_0$. Instead, under the condition $\bar{\epsilon}_1\epsilon_2 \leq \bar{\epsilon}_2^2\epsilon_1$, using the same equations, it can be seen that $\delta_{11}^c = \delta_{11}^u = \delta_{21} = \delta_{22} = \delta_0$. It follows that δ_E=max$(\delta_{21}, \delta_{11}^c, \delta_{11}^u, \delta_{22}) = \delta_0$. We can conclude that DTB is $\delta_C + \delta_E = \delta_0 + (2 - \delta_0(1 - \epsilon_2^2))/C$, which is equal to $\delta_{\text{off}}^*(0, C)$ in (2.24).

Remark 2.2. In a manner similar to the edge-aided case, the optimal scheme described above requires only causal CSI at the encoders, and, furthermore, it requires no CSI at the Cloud (but only knowledge of the channel statistics.) This shows that the assumption of non-causal CSI is not needed to obtain optimal performance.

2.4 Concluding Remarks

In this chapter, the potential of interference management as a function of the caching and fronthaul capacity limitations is studied. Assuming a static set of popular files, a one-sided interference scenario modeling a macro-BS coexisting with a fog-aided small-cell BS is analytically evaluated. Using an original information-theoretic framework that centers on the evaluation of a minimum delivery latency metric, the trade-off between latency and system resources has been studied, and a full characterization has been provided under a simplified binary fading interference channel and in the presence of full CSI. Interesting extensions include the analysis of the impact of imperfect CSI as well as of a more general channel model.

CHAPTER 3

FUNDAMENTAL LIMITS ON LATENCY IN FOG-AIDED SMALL-CELL SYSTEMS WITH ONLINE CACHING

Chapter 2 focused on an offline caching scenario in which there is a fixed set of popular contents and the operation of the system is divided between a placement phase and a delivery phase. In this chapter, instead, an online caching set-up is considered in which the set of popular files varies from one time slot to the next. As a result, both content delivery and cache update should be generally performed in every time slot, where the latter is needed to ensure the timeliness of the cached content.

3.1 Introduction

Edge caching as an instance of content distribution networks (CDNs) relies on the storing popular content at base stations. In most of the prior works, the underlying assumption is that there exists a fixed library of popular files out of which users make arbitrary request. The caching phase consists of filling up the caches with functions of the files whose entropy is constrained to be not larger than the corresponding cache capacity. After this set-up phase, the network is used for an arbitrary long time, referred to as the delivery phase. At each request round, users request subsets of the files in the library and the network must coordinate transmissions such that these requests are satisfied, i.e., at the end of each round all destinations must decode the requested set of files. The proposed performance metric in Chapter 2 is DTB which is the number of channel uses necessary to satisfy all the demands normalized by the size of files. A more realistic assumption is that the set of popular files evolves over time. This necessitates online caching which is first introduced in [3]. The key

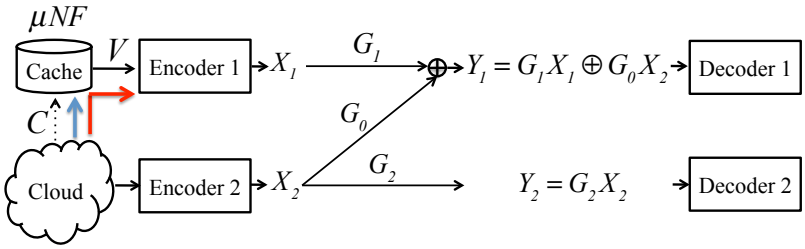

Figure 3.1 Cloud and edge-aided data delivery over binary fading interference channels with online caching. Blue arrow represents the cache update while red arrow represents the delivery.

difference with offline caching is that content placement and delivery should be done simultaneously to account for time-variability of popular files.

Main Contributions: In this chapter, the F-RAN model in Figure 3.1 is considered, which includes a small-cell BS and a macro-BS, represented by Encoder 1 and Encoder 2, respectively. The small-cell BS (Encoder 1) is equipped with a cache of finite capacity and can serve a small-cell mobile user, represented by Decoder 1. The macro-BS (Encoder 2) can serve a macro-cell user, namely Decoder 2, as well as, possibly, also Decoder 1. The transmission from the macro-BS (Encoder 2) to Decoder 2 interferes with Decoder 1. It is assumed that the small-cell BS transmits with sufficiently small power so as not to create interference at Decoder 2, which is modeled here as a partially connected wireless channel. The main contributions of this chapter are as follows:

- An information-theoretic formulation for the analyses of the system in Figure 3.1 is introduced as long-term Delivery Time per Bit (DTB), for online caching. The system model is based on a one-sided interference channel.
- Online caching and delivery schemes based on both reactive and proactive caching principles (see, e.g., [7]) are proposed in the presence of a time-varying set of popular files, and bounds on the corresponding achievable long-term DTBs are derived.
- A lower bound on the achievable long-term DTB is obtained, which is a function of the time-variability of the set of popular files. The lower bound is then utilized to compare the achievable DTBs under offline and online caching.
- Numerical results are provided in which the DTB performance of reactive and proactive online caching schemes is compared with offline caching. In addition,

different eviction mechanisms, such as random eviction, Least Recently Used (LRU) and First In First Out (FIFO) (see, e.g., [31]), are evaluated.

3.2 System Model

Let $\mathcal{L}_t$ be the set of N popular files at time slot t. As in [3], we assume that with probability $1-p$, the popular set is unchanged and we have $\mathcal{L}_t = \mathcal{L}_{t-1}$; while, with probability p, the set $\mathcal{L}_t$ is constructed by randomly and uniformly selecting one of the files in the set $\mathcal{L}_{t-1}$ and replacing it by a new popular file. At each time slot t, users request files $\mathbf{d}_t$, which are drawn uniformly at random from the set $\mathcal{L}_t$ without replacement. We consider two cases, namely:

Known popular set: The Cloud is informed about the set $\mathcal{L}_t$ at time t, e.g., by leveraging data analytics tools.

Unknown popular set: The set $\mathcal{L}_t$ may only be inferred at the Cloud via the observation of the users' requests. This assumption is typically made in the networking literature [31].

Define as $T_{C,t}$ the duration of the transmission from Cloud to Encoder 1 and as T_t the duration of the transmission from both encoders to decoders at time slot t. As in the offline setup, durations are measured in terms of number of channel uses of the binary fading channel. Since the set of popular files $\mathcal{L}_t$ is time-varying, both cache update and file delivery are generally performed at each time slot t. To this end, at time slot t, the Cloud encodes via the function:

$$\psi_C : [2^{NF}] \times [N]^2 \times \{0,1\}^{3T_t} \to [2^{T_{C,t}C}], \tag{3.1}$$

which maps the library $\mathcal{L}_t$ of all files, the demand vector $\mathbf{d}_t$ and the CSI vector $\mathbf{G}^{T_t}$ to the signal $U^{T_{C,t}} = (U_1, ..., U_{T_{C,t}}) = \psi_C(\mathcal{L}_t, \mathbf{d}_t, \mathbf{G}^{T_t})$ to be delivered to Encoder 1. The inequality $H(U^{T_{C,t}}) \leq T_{C,t}C$ presents capacity constraint on the Cloud-to-Encoder 1 link. Moreover, Encoder 1 uses the encoding function

$$\psi_1 : [2^{\mu NF}] \times [2^{T_{C,t}C}] \times [N]^2 \times \{0,1\}^{3T_t} \to \{0,1\}^{T_t}, \tag{3.2}$$

which maps the cached content V_t, the received signal $U^{T_{C,t}}$, the demand vector $\mathbf{d}_t$ and the CSI sequence $\mathbf{G}^{T_t} = (\mathbf{G}(1), ..., \mathbf{G}(T_t))$ to the transmitted codeword $X_1^{T_t} = (X_1[1], ..., X_1[T_t]) = \psi_1(V_t, U^{T_{C,t}}, \mathbf{d}_t, \mathbf{G}^{T_t})$.

The probability of error is defined as

$$\mathrm{P}_{e,t}^{F} = \max_{j \in \{1,2\}} \Pr(\hat{W}_{d_{j,t}} \neq W_{d_{j,t}}), \tag{3.3}$$

where $d_{j,t}$ is the index of the requested file by jth user at time slot t so that we have $\mathbf{d}_t = (d_{1,t}, d_{2,t})$. The probability of error in (3.3) is evaluated with respect to the distribution of the popular set $\mathcal{L}_t$ and of the request vector $\mathbf{d}_t$. A sequence of policies indexed by t is said to be feasible if $\mathrm{P}_{e,t}^{F} \to 0$ as $F \to \infty$ for all t. In a manner similar to the offline case, the DTB at time slot t is defined as

$$\delta_t(\mu, C) = \lim_{F \to \infty} \frac{\mathrm{E}[T_t + T_{C,t}]}{F}, \tag{3.4}$$

where the average is taken over the distribution of the popular set $\mathcal{L}_t$ and of the request vector $\mathbf{d}_t$. To measure the performance of online caching, the long-term DTB

is defined as

$$\bar{\delta}_{\text{on}}(\mu, C) = \limsup_{T\to\infty} \frac{1}{T} \sum_{t=1}^{T} \delta_t(\mu, C). \tag{3.5}$$

The minimum long-term DTB over all feasible policies under the known popular set assumption is denoted by $\bar{\delta}^*_{\text{on,k}}(\mu, C)$, while $\bar{\delta}^*_{\text{on,u}}(\mu, C)$ denotes the minimum long-term DTB under the unknown popular set assumption. By definition, the inequality $\bar{\delta}^*_{\text{on,k}}(\mu, C) \leq \bar{\delta}^*_{\text{on,u}}(\mu, C)$ holds. Furthermore, both DTBs $\bar{\delta}^*_{\text{on,k}}(\mu, C)$ and $\bar{\delta}^*_{\text{on,u}}(\mu, C)$ are not smaller than the offline DTB $\delta^*_{\text{off}}(\mu, C)$, given that in the offline set-up caching takes place in a separate phase with no overhead on the Cloud-to-Encoder 1 link. In the rest of this chapter, the performance of two proposed online caching schemes is evaluated and then a lower bound on the the minimum long-term DTB is provided.

3.3 Proactive Online Caching

If the popular set $\mathcal{L}_t$ is known, the cloud can proactively cache any new content at the small-cell BS by replacing the outdated file. Specifically, a μ-fraction of the new popular file is transferred from the Cloud to Encoder 1 in order to update the cache content at the small-cell BS. Since, after this update, the cache configuration with respect to the current set $\mathcal{L}_t$ of popular files is the same as in the offline case with respect to $\mathcal{L}$, delivery can then be performed by following the offline delivery policy detailed in Chapter 2. The following proposition presents the resulting achievable long-term DTB of proactive online caching.

Proposition 3.1. *The proposed proactive online caching for the fog-aided system in Figure 3.1 achieves the long-term DTB*

$$\bar{\delta}_{\text{on,pro}}(\mu, C) = \delta^*_{\text{off}}(\mu, C) + \frac{p\mu}{C}, \tag{3.6}$$

with $\delta^*_{\text{off}}(\mu, C)$ *is given by* (2.23) *and* (2.24). *We hence have the upper bound* $\bar{\delta}^*_{\text{on,k}}(\mu, C) \leq \bar{\delta}_{\text{on,pro}}(\mu, C)$.

Proof. With probability p, there is a new file in the popular set $\mathcal{L}_t$ and hence a μ-fraction of the new content is sent on the cloud-to-Encoder 1 link resulting in a latency of $T_{C,t} = \mu F/C$. The achievable scheme in Section 2.3.2 is then used to deliver both requested files. As a result, the DTB at time slot t is $\delta_t = p(\delta^*_{\text{off}}(\mu, C) + \mu/C) + (1-p)\delta^*_{\text{off}}(\mu, C)$. Using (4.6), the long-term DTB is given by (3.6). □

3.4 Reactive Online Caching

When the popular set is highly time-varying, the proactive scheme sends a large number of new contents on the Cloud-to-Encoder 1 link to update the cache content at small-cell BS. However, only a subset of these files will generally be requested before becoming outdated. To potentially solve this problem, the Cloud can update the small-cell BS's cache by means of a reactive scheme. Accordingly, the Cloud updates the cache only if the files requested by Decoder 1 and/or Decoder 2 are not (partially) cached at the small-cell BS.

The reactive strategy, unlike the proactive one, can operate under the unknown popular set assumption. It is also possible to define a reactive strategy that leverages knowledge of the set of popular files to outperform proactive caching. This will be addressed in Chapter 4.

To elaborate, in a manner similar to [3], in each time slot t, small-cell BS stores a (μ/α)-fraction of $N' = \alpha N$ files for some $\alpha > 1$. Note that the set of $N' > N$ cached files in the cached contents of small-cell BS generally contains files that are no longer in the set $\mathcal{L}_t$ of N popular files. Caching $N' > N$ files is instrumental in keeping the intersection between the set of cached files and $\mathcal{L}_t$ from vanishing [3].

To update the cache content, a (μ/α)-fraction of the requested and uncached files is sent on the Cloud-to-Encoder 1 link and is cached at the small-cell BS by randomly and uniformly evicting the same number of cached files. The following proposition presents an achievable long-term DTB for the proposed reactive online caching policy.

Proposition 3.2. *The proposed reactive online caching for the fog-aided system in Figure 3.1 achieves a long-term DTB that is upper bounded as*

$$\bar{\delta}_{\text{on,react}}(\mu, C) \leq \delta^*_{\text{off}}\Big(\frac{\mu}{\alpha}, C\Big) + \frac{p\mu}{C(1-p/N)(\alpha-1)}, \tag{3.7}$$

for any $\alpha > 1$. This yields the upper bound $\bar{\delta}^*_{\text{on,u}}(\mu, C) \leq \bar{\delta}_{\text{on,react}}(\mu, C)$.

Proof. Denoting $Y_t \in \{0, 1, 2\}$ the number of requested and uncached files at time slot t, the cloud send a (μ/α)-fraction of the Y_t requested and uncached files to the small-cell BS. Hence, the achievable DTB at each time slot t is

$$\delta_t(\mu, C) = \delta^*_{\text{off}}\Big(\frac{\mu}{\alpha}, C\Big) + \frac{\mu \mathrm{E}(Y_t)}{\alpha C}. \tag{3.8}$$

By plugging (3.8) into the definition of long-term DTB (4.6), we have

$$\bar{\delta}_{\text{on,react}}(\mu, C) = \delta^*_{\text{off}}\Big(\frac{\mu}{\alpha}, C\Big) + \Big(\frac{\mu}{\alpha C}\Big) \limsup_{T\to\infty} \frac{1}{T} \sum_{t=1}^{T} \mathrm{E}[Y_t]. \tag{3.9}$$

Noting the fact that content placement and random eviction are the same as [3], the result of ([3] Lemma 3) can be invoked to obtain the upper bound

$$\limsup_{T\to\infty} \frac{1}{T} \sum_{t=1}^{T} \mathrm{E}[Y_t] \leq \frac{p}{(1-p/N)(1-1/\alpha)}. \tag{3.10}$$

Plugging (3.10) into (3.9) completes the proof. □

3.5 Lower Bound on the Minimum Long-Term DTB

The following proposition provides a lower bound on the the minimum long-term DTB

Proposition 3.3. *(Lower bound on the Long-Term DTB of Online Caching). For the fog-aided system in Figure 3.1 with $N \geq 2$, the long-term DTB is lower bounded as*

$$\bar{\delta}^*_{\text{on,u}}(\mu, C) \geq \bar{\delta}^*_{\text{on,k}}(\mu, C) \geq \left(1 - \frac{2p}{N}\right)\delta^*_{\text{off}}(\mu, C) + \left(\frac{2p}{N}\right)\delta^*_{\text{off}}(0, C) \tag{3.11}$$

with $\delta^*_{\text{off}}(\mu, C)$ given in (2.23) *and* (2.24)*.*

Proof. See Appendix B.1. □

The lower bound (3.11) will be leveraged in the next section to relate the performance of offline and online caching.

3.6 Comparison between Online and Offline Caching

In this section, we compare the performance of the offline caching system studied in Chapter 2 and of the online caching system introduced in this chapter. The following proposition presents that the minimum long-term DTB can be upper and lower bounded in terms of the minimum DTB of offline caching.

Proposition 3.4. *For the fog-aided system in Figure 3.1 with $N \geq 2$, the long-term DTB satisfies the inequalities*

$$\Big(1 - \frac{2p}{N}\Big)\delta^*_{\text{off}}(\mu, C) + \frac{2p}{N}\frac{2}{1-\epsilon_2^2} \leq \bar{\delta}^*_{\text{on,k}}(\mu, C) \leq \bar{\delta}^*_{\text{on,u}}(\mu, C) \leq 2\delta^*_{\text{off}}(\mu, C) \tag{3.12}$$

if $C \leq 1 - \epsilon_2^2$, and

$$\Big(1-\frac{2p}{N}\Big)\delta^*_{\text{off}}(\mu, C)+\frac{2p}{N}\Big(\frac{2-(1-\epsilon_2^2)\delta_0}{C}+\delta_0\Big) \leq \bar{\delta}^*_{\text{on,k}}(\mu, C) \leq \bar{\delta}^*_{\text{on,u}}(\mu, C) \leq \delta^*_{\text{off}}(\mu, C)+\frac{4}{C} \tag{3.13}$$

if $C \geq 1 - \epsilon_2^2$.

Proof. The upper bound is obtained by comparing the performance (3.6) of the proposed reactive scheme with the minimum offline DTB in Proposition 2.2, while the lower bound is from Proposition 3.3. Details are provided in Appendix B.2. □

Proposition 3.4 shows that the long-term DTB with online caching is no larger than twice the minimum offline DTB in the regime of low capacity C. Instead for larger values of C, the minimum online DTB is proportional to minimum offline DTB with an additive gap that decreases as $1/C$. Informally, these results demonstrate that the additive loss of online caching decreases as $1/C$ for sufficiently large C, while, for lower values of C, the performance gap is bounded. This stands in contrast to [2], in which the performance gap between offline and online caching increases as the inverse of the capacity of the link between Cloud and BSs when the latter becomes smaller. The key distinction here is that the macro-BS has direct access to the set of popular files and can directly serve the users, while in [2] the Cloud can only access the users through the finite-capacity links.

3.7 Numerical Results

In this section, we evaluate the performance of the proposed online caching schemes numerically. We specifically consider the long-term DTB achievable by the proposed proactive scheme (3.6) and the proposed reactive scheme (3.7). For the latter, we evaluate the expectation in (3.8) via Monte Carlo simulations by averaging over a large number of realizations, i.e., 10,000, of the random process Y_t. It is assumed that the small-cell cache is empty at the start of simulation, i.e., at time $t = 1$.

The impact of the cloud-to-Encoder 1 capacity C is first considered in Figure 3.2. As a reference, we also plot the minimum DTB for offline caching in (2.23) and (2.24) and the performance with no caching, that is, $\delta^*_{\text{off}}(0, C)$ in (2.24). For reactive caching, we assume random eviction for reactive caching. Parameters are set as $\mu = 0.5$, $p = 0.5$, $\epsilon_1 = \epsilon_2 = 0.5$ and $N = 10$. It is seen that both proactive and reactive caching can significantly improve over the no caching scheme by updating the content stored at the small-cell BS. However, as the capacity of Cloud-to-Encoder 1 link decreases, it is deleterious in terms of delivery latency to use the link in order to update the cache content. As a result, if C is small enough, the performance of reactive and proactive caching coincides with the no caching system. When C is large enough, instead, the latency of cache update is negligible and both proactive and reactive schemes achieve the same DTB, which tends to the minimum offline DTB.

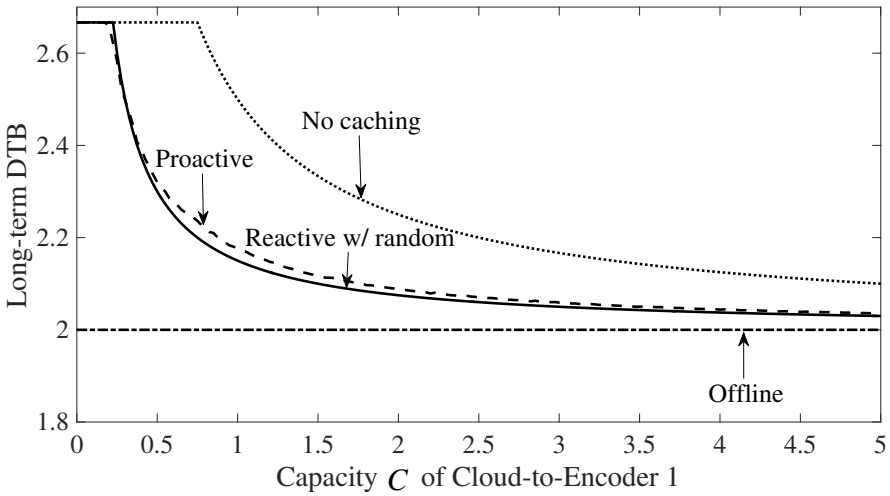

Figure 3.2 Achievable long-term DTB versus the capacity C of the Cloud-to-Encoder 1 for proactive scheme (3.6) and reactive caching with random eviction (3.7). For reference, the DTB with no caching, namely $\delta^*_{\text{off}}(0, C)$, and the offline minimum DTB (2.23) and (2.24) are also shown ($p = 0.5$, $\mu = 0.5$, $\epsilon_1 = \epsilon_2 = 0.5$, $N = 10$).

Next, we compare the performance of reactive and proactive online caching schemes as a function of the probability p of new content. As shown in Figure 3.3 for $\mu = 0.5$, $C = 0.5$, $\epsilon_1 = \epsilon_2 = 0.5$ and $N = 10$, when p is small, proactive caching outperforms reactive caching, since it uses the Cloud-to-Encoder 1 connection only with rare event that there is a new popular file. On the other hand, when p is large, as explained in the previous section, the reactive approach yields a smaller latency than the proactive scheme. It is also seen that the LRU eviction strategy, whereby the replaced file is the one that has been least recently requested by any user, and FIFO eviction strategy, whereby the file that has been in the caches for the longest time is replaced, are both able to improve over randomized eviction.

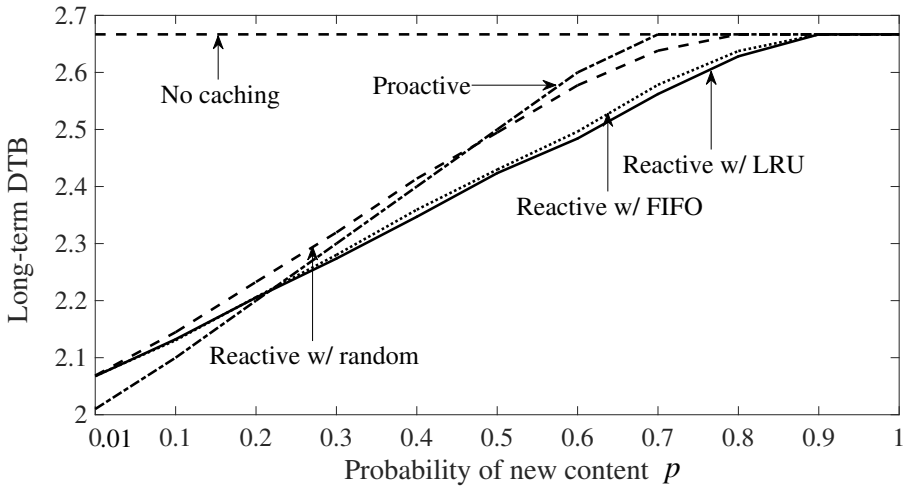

Figure 3.3 Achievable long-term DTB versus probability p of new content for the proactive scheme (3.6) and reactive caching scheme with random, LRU or FIFO eviction (3.7). For reference, the DTB with no caching, namely $\delta^*_{\text{off}}(0, C)$, and the offline minimum DTB (2.23) and (2.24) are also shown ($C = 0.5$, $\mu = 0.5$, $\epsilon_1 = \epsilon_2 = 0.5$, $N = 10$).

3.8 Concluding Remarks

Motivated by recent advances in fog-aided wireless network architectures, this chapter considered a fog-assisted system for content delivery. The system model includes a macro-BS that coexists with a cache and cloud-aided small-cell BS whose user can also be served by the macro-BS. Using the minimum delivery latency as performance measure, the trade-off between latency and system resources has been studied. A characterization of this optimal trade-off has been derived under a binary fading interference channel and in the presence of full CSI when the set of popular contents is time-varying. The average DTB within a long time horizon is shown to be at most two times larger than for the offline scenario case when the capacity of the link used to update the cache content is small and to have otherwise a gap inversely proportional to this capacity.

CHAPTER 4

ONLINE EDGE CACHING IN FOG-AIDED NETWORKS WITH DYNAMIC CONTENT POPULARITY AND INTERFERENCE LIMITED WIRELESS CHANNELS

Having investigated the performance limits of Fog Radio Access Network (F-RAN) architectures comprised of one small-cell BS and one macro-cell BS with binary fading interference channel in previous chapters, in this chapter a generalization to arbitrary number of BSs and end-users as well as a general wireless channel is investigated. Specifically, this chapter assumes given number of BSs with identical cache size and fronthaul capacities. Assuming high signal-to-noise (SNR) regime, there is a fully connected wireless channel between BSs and users. Hence, interference from concurrent transmissions is the only drawback of wireless channel. The set of popular files evolve over time and appropriate cache update and delivery scheme is required to keep track of changes in popular content.

The analysis is centered on the characterization of the long-term Normalized Delivery Time (NDT), which captures the temporal dependence of the coding latencies accrued across multiple time slots in the high signal-to-noise ratio regime. Online edge caching and delivery schemes based on reactive and proactive caching principles are investigated for both serial and pipelined transmission modes across fronthaul and edge segments. Analytical results demonstrate that, in the presence of a time-varying content popularity, the rate of fronthaul links sets a fundamental limit to the long-term NDT of F-RAN system. Analytical results are further verified by numerical simulation, yielding important design insights.

4.1 Introduction

Delivery of wireless multimedia content poses one of the main challenges in the definition of enhanced mobile broadband services in 5G (see, e.g., [32]). In-network caching, including *edge caching*, is a key technology for the deployment of information-centric networking with reduced bandwidth [33], latency [34], and energy [35]. Specifically, edge caching stores popular content at the edge nodes (ENs) of a wireless system, thereby reducing latency and backhaul usage when the requested contents are cached [6].

While edge caching moves network intelligence closer to the end users, *Cloud Radio Access Network* (C-RAN) leverages computing resources at a central cloud processing unit, and infrastructure communication resources in the form of fronthaul links connecting cloud to ENs [36, 37]. The cloud processor is typically part of the transport network that extends to the ENs, and is also known as "edge cloud" [38]. The C-RAN architecture can be used for content delivery as long as the cloud has access to the content library [17, 34].

Through the use of Network Function Virtualization (NFV) [39], 5G networks will enable network functions to be flexibly allocated between edge and cloud elements, hence breaking away from the purely edge- and cloud-based solutions provided by edge caching and C-RAN, respectively. To study the optimal operation of networks that allow for both edge and cloud processing, references [17, 34] investigated a *Fog Radio Access Network (F-RAN)* architecture, in which the ENs are equipped with limited-capacity caches and fronthaul links. These works addressed the optimal use of the communication resources on the wireless channel and fronthaul links and storage resources at the ENs, under the assumption of offline caching. With offline caching, caches are replenished periodically, say every night, and the cached content is kept

fixed for a relatively long period of time, e.g., throughout the day, during which the set of popular contents is also assumed to be invariant.

Related work: The information theroretic analysis of offline edge caching in the presence of a static set of popular contents was first considered in [9]. In this seminal work, an achievable number of degrees of freedom (DoF), or more precisely its inverse, is determined as a function of cache storage capacity for a system with three ENs and three users. In [11, 13, 24, 40, 41], generalization to the scenario with cache-aided transmitters as well as receivers is considered. In particular, in [11, 40], it is proved that, under the assumption of one-shot linear precoding, the maximum achievable sum-DoF of a wireless system with cache-aided transmitters and receivers scales linearly with the aggregate cache capacity across the nodes with both fully connected and partially connected topologies. In [41], separation of network and physical layers is proposed, which is proved to be approximately optimal. Reference [13] extended the works in [11, 24, 40, 41] to include decentralized content placement at receivers.

In contrast to abovementioned works, references [10, 17, 18, 20, 21] investigated the full F-RAN scenario with both edge caching and cloud processing in the presence of offline caching. Specifically, reference [17] derives upper and lower bounds on a high-SNR coding latency metric defined as Normalized Delivery Time (NDT), which generalizes the inverse-of-DoF metric studied in [9] and in most of the works reviewed above. The minimum NDT is characterized within a multiplicative factor of two. While in [17] it is assumed that there are point-to-point fronthaul links with dedicated capacities between cloud and ENs, a wireless broadcast channel is considered between cloud and ENs in [18]. The scenario with heterogeneous cache requirements is considered in [20]. Offline caching for a small-cell system with limited capacity fronthaul connection between small-cell BS and cloud-processor and partial wireless connectivity is invistigated in [21] and [4] under different channel models.

Main contributions: In this chapter, an online caching set-up with arbitrary number of BSs and users and a general wireless channel is considered, in which the set of popular files is time-varying, making it generally necessary to perform cache replenishment as the system delivers contents to the end users. The main contributions are as follows.

- The performance metric of the long-term NDT, which captures the temporal dependence of the high-SNR coding latencies accrued in different slots, is introduced;
- Online edge caching and delivery schemes based on reactive online caching are proposed for a set-up with serial fronthaul-edge transmission, and bounds on the corresponding achievable long-term NDTs are derived by considering both fixed and adaptive caching;
- Reactive and proactive online caching schemes are proposed for a pipelined fronthaul-edge transmission mode, and bounds on the corresponding achievable long-term NDTs are derived in Section 4.5;
- The performance loss caused by the time-varying content popularity in terms of delivery latency is quantified by comparing the NDTs achievable under offline and online edge caching in Section 4.6;
- Numerical results are provided in Section 4.7 that offer insights into the comparsion of reactive online edge caching schemes with different eviction mechanisms, such as random, Least Recently Used (LRU) and First In First Out (FIFO) (see, e.g., [31]), proactive online edge caching schemes, under both serial and pipelined transmission.

Notation: Given random variable X, the corresponding entropy is denoted by $\mathrm{H}(X)$. The equality $f(x) = O(g(x))$ indicates the relationship $\lim_{x\to\infty} |f(x)/g(x)| < \infty$.

4.2 System Model

Let consider an $M \times K$ F-RAN with online edge caching shown in Figure 4.1, in which M ENs serve a number of users through a shared downlink wireless channel.

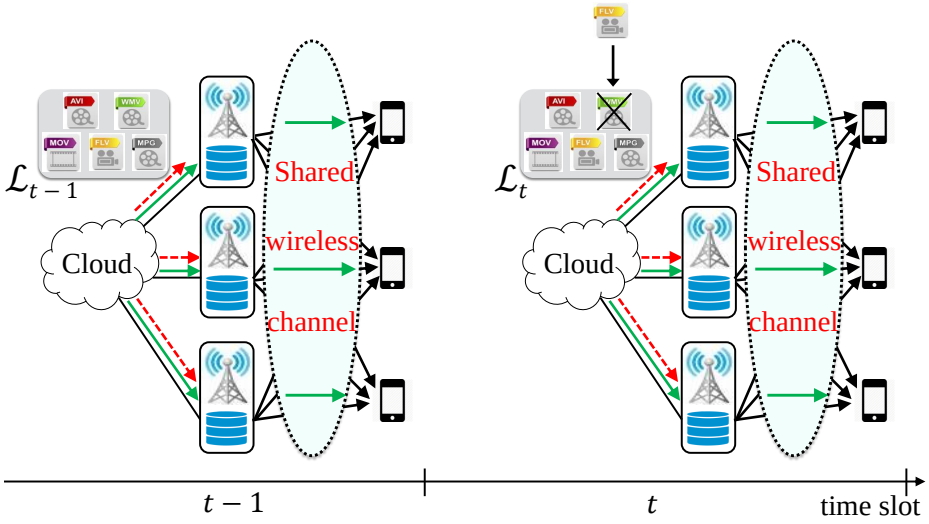

Figure 4.1 With online edge caching, the set of popular files $\mathcal{L}_t$ is time-varying, and online cache update (red dashed arrows) generally can be done at each time slot along with content delivery (green arrows).

Time is organized into time slots, and K users are active in each time slot. Each EN is connected to the cloud via a dedicated fronthaul link with capacity C_F bits per symbol period, where the latter henceforth refers to the symbol period for the wireless channel. Each time slot contains a number of symbols. In each time slot t, K users make requests from the current set $\mathcal{L}_t$ of N popular files. All files in the popular set are assumed to have the same size of F bits. The cache capacity of each EN is μNF bits, where μ, with $0 \leq \mu \leq 1$, is defined as the *fractional cache capacity* since NF is the dimension of set $\mathcal{L}_t$ in bits.

Time-varying popular set: At each time slot t, each of the K active users requests a file from the time-varying set $\mathcal{L}_t$ of N popular files, with $t \in \{1, 2, ...\}$. The indices of the files requested by the K users are denoted by the vector $d_t = (d_{1,t}, ..., d_{K,t})$, where $d_{k,t}$ represents the file requested by user k. As in [3], indices are chosen uniformly without replacement in the interval $[1 : N]$ following an arbitrary order. The set of popular files $\mathcal{L}_t$ evolves according to the Markov model considered in [3]. Accordingly, given the popular set $\mathcal{L}_{t-1}$ at time slot $t-1$, with probability $1-p$, no new popular content is generated and we have $\mathcal{L}_t = \mathcal{L}_{t-1}$; while, with probability p, a new popular

file is added to the set $\mathcal{L}_t$ by replacing a file selected uniformly at random from $\mathcal{L}_{t-1}$. In a similar manner to Chapter 3, two cases are considered, namely:

Known popular set: The cloud is informed about the set $\mathcal{L}_t$ at time t, e.g., by leveraging data analytics tools

Unknown popular set: The set $\mathcal{L}_t$ may only be inferred via the observation of the users' requests. This assumption is typically made in the networking literature (see, e.g., [31]).

Edge channel: The signal received by the kth user in any symbol of the time slot t is

$$Y_{k,t} = \sum_{m=1}^{M} H_{k,m,t} X_{m,t} + Z_{k,t}, \tag{4.1}$$

where $H_{k,m,t}$ is the channel gain between mth EN and kth user at time slot t; $X_{m,t}$ is the signal transmitted by the mth EN; and $Z_{k,t}$ is additive noise at kth user. The channel coefficients are assumed to be independent and identically distributed (i.i.d.) according to a continuous distribution and to be time-invariant within each slot. Also, the additive noise $Z_{k,t} \sim \mathcal{CN}(0,1)$ is i.i.d. across time and users. At each time slot t, all the ENs, cloud and users have access to the global CSI about the wireless channels $H_t = \{\{H_{k,m,t}\}_{k=1}^{K}\}_{m=1}^{M}$.

System operation: The system operates according to a combined fronthaul, caching, edge transmission and decoding policy, which is defined as follows.

Fronthaul policy: The cloud transmits a message $U_{m,t}$ to each EN m in any time slot t as a function of the current demand vector d_t, ENs' cache contents, and CSI H_t, as well as, in the case of known popular set, the set $\mathcal{L}_t$. The fronthaul capacity limitations impose the condition $\mathrm{H}(U_{m,t}) \leq T_{F,t} C_F$, where $T_{F,t}$ is the duration (in symbols) of the fronthaul transmission $U_{m,t}$ in time slot t for all ENs $m = 1, ..., M$.

Caching policy: After fronthaul transmission, in each time slot t, any EN m updates its cached content $S_{m,t-1}$ in the previous slot based on the fronthaul message $U_{m,t}$, producing the updated cache content $S_{m,t}$. Due to cache capacity constraints, we have the inequality $\mathrm{H}(S_{m,t}) \leq \mu NF$, for all slots t and ENs m. More specifically, as in [17], we allow only for intra-file coding. Therefore, the cache content $S_{m,t}$ can be partitioned into independent subcontents $S_{m,t}^{l}$, each obtained as a function of a single file $l \in \mathcal{L}_t$, with the condition $\mathrm{H}(S_{m,t}^{l}) \leq \mu F$. We also assume that, at time $t = 1$, all the caches are empty.

Edge transmission policy: Upon updating the caches, the edge transmission policy at each EN m transmits the codeword $X_{m,t}$, of duration $T_{E,t}$ on the wireless channel as a function of the current demand vector d_t, CSI H_t, cache contents $S_{m,t}$ and fronthaul messages $U_{m,t}$. We assume a per-slot power constraint P for each EN.

Decoding policy: Each user k maps its received signal $Y_{k,t}$ in (4.1) over a number $T_{E,t}$ of channel uses to an estimate $\hat{W}_{d_{t,k}}$ of the demanded file $W_{d_{t,k}}$.

The probability of error of a policy Π at slot t is defined as the worst-case probability

$$\mathrm{P}_{e,t} = \max_{k \in \{1,\ldots,K\}} \Pr(\hat{W}_{d_{t,k}} \neq W_{d_{t,k}}), \tag{4.2}$$

which is evaluated over the distributions of the popular set $\mathcal{L}_t$, of the request vector d_t and of the CSI H_t. A sequence of policies Π indexed by the file size F is said to be *feasible* if, for all t, we have $\mathrm{P}_{e,t} \to 0$ when $F \to \infty$.

4.2.1 Long-term Normalized Delivery Time (NDT)

For given parameters (M, K, N, μ, C_F, P), the average delivery time per bit in slot t achieved by a feasible policy under serial fronthaul-edge transmission is defined as

the sum of fronthaul and edge contributions

$$\Delta_t(\mu, C_F, P) = \Delta_{F,t}(\mu, C_F, P) + \Delta_{E,t}(\mu, C_F, P), \tag{4.3}$$

where the fronthaul and edge latencies per bit are given as

$$\Delta_{F,t}(\mu, C_F, P) = \lim_{F\to\infty} \frac{1}{F}\mathrm{E}[T_{F,t}] \text{ and } \Delta_{E,t}(\mu, C_F, P) = \lim_{F\to\infty} \frac{1}{F}\mathrm{E}[T_{E,t}]. \tag{4.4}$$

In (4.4), the average is taken with respect to the distributions of $\mathcal{L}_t$, d_t and H_t, and we have made explicit only the dependence on the system resource parameters (μ, C_F, P).

As in [17], in order to evaluate the impact of a finite fronthaul capacity in the high-SNR regime, we let the fronthaul capacity scale with the SNR parameter P as $C_F = r\log(P)$, where $r \geq 0$ measures the ratio between fronthaul and wireless capacities at high SNR. Furthermore, we study the scaling of the latency with respect to a reference system in which each user can be served with no interference at the Shannon capacity $\log(P)+o(\log P)$. Accordingly, for any achievable delivery time per bit (4.4), the Normalized Delivery Times (NDTs) for fronthaul and edge transmissions in time slot t [17] are defined as

$$\delta_{F,t}(\mu, r) = \lim_{P\to\infty} \frac{\Delta_{F,t}(\mu, r\log(P), P)}{1/\log(P)} \text{ and } \delta_{E,t}(\mu, r) = \lim_{P\to\infty} \frac{\Delta_{E,t}(\mu, r\log(P), P)}{1/\log(P)}, \tag{4.5}$$

respectively. In (4.5), the delivery time(s) per bit in (4.4) are normalized by the term $1/\log(P)$, which measures the delivery time per bit at high SNR of the mentioned

reference system [17]. The NDT in time slot t is defined as the sum $\delta_t(\mu, r) = \delta_{E,t}(\mu, r) + \delta_{F,t}(\mu, r)$.

In order to capture the memory entailed by online edge caching policies on the system performance, we introduce the *long-term NDT* metric. This is defined as the time average:

$$\bar{\delta}_{\text{on}}(\mu, r) = \limsup_{T\to\infty} \frac{1}{T} \sum_{t=1}^{T} \delta_t(\mu, r). \tag{4.6}$$

We denote the minimum long-term NDT over all feasible policies under the known popular set assumption as $\bar{\delta}^*_{\text{on,k}}(\mu, r)$, while $\bar{\delta}^*_{\text{on,u}}(\mu, r)$ denotes the minimum long-term NDT under the unknown popular set assumption. As a benchmark, we also consider the minimum NDT for offline edge caching $\delta^*_{\text{off}}(\mu, r)$ as studied in [17]. By definition, we have the inequalities $\delta^*_{\text{off}}(\mu, r) \leq \bar{\delta}^*_{\text{on,k}}(\mu, r) \leq \bar{\delta}^*_{\text{on,u}}(\mu, r)$.

4.3 Preliminaries: Offline Caching

In this section, first, some key results on offline caching in F-RAN from [17] are summarized. With offline caching, the set of popular files $\mathcal{L}_t = \mathcal{L}$ is time invariant and caching takes place in a separate placement phase. Reference [17] identified offline caching and delivery policies that are optimal within a multiplicative factor of 2 in terms of NDT achieved in each time slot. The policies are based on fractional caching, whereby an uncoded fraction μ of each file is cached at the ENs, and on three different delivery approaches, namely EN cooperation, EN coordination, and C-RAN transmission.

EN cooperation: This approach is used when all ENs cache all files in the library. In this case, joint Zero-Forcing (ZF) precoding can be carried out at the ENs so as to null interference at the users. This can be shown to require an edge and

fronthaul-NDTs in (4.5) equal to [17]

$$\delta_{\text{E,Coop}} = \frac{K}{\min\{M, K\}} \text{ and } \delta_{\text{F,Coop}} = 0 \tag{4.7}$$

in order to communicate reliably the requested files to all users. Note that, when the number of ENs is larger than the number of active users, i.e., $M \geq K$, we have $\delta_{\text{E,Coop}} = 1$, since the performance becomes equivalent to that of the considered interference-free reference system (see Section 4.2.1). For reference, we also write the fronthaul-NDT $\delta_{\text{F,Coop}} = 0$ since this scheme does not use fronthaul resources.

EN coordination: This approach is instead possible when the ENs store non-overlapping fractions of the requested files. Specifically, if each EN caches a different fraction of the popular files, *Interference Alignment (IA)* can be used on the resulting so-called X-channel[1]. This yields the edge and fronthaul-NDTs [17]

$$\delta_{\text{E,Coor}} = \frac{M + K - 1}{M}, \text{ and } \delta_{\text{F,Coor}} = 0. \tag{4.8}$$

C-RAN transmission: While EN coordination and cooperation are solely based on edge caching, C-RAN transmission uses cloud and fronthaul resources. Specially, C-RAN performs ZF precoding at the cloud, quantizes the resulting signals and sends them on the fronthaul links to the ENs. The ENs act as relays that transmit the received fronthaul messages on the wireless channel. The resulting edge and fronthaul-NDTs are equal to [17]

$$\delta_{\text{E,C-RAN}}(r) = \delta_{\text{E,Coop}} = \frac{K}{\min\{M, K\}} \text{ and } \delta_{\text{F,C-RAN}}(r) = \frac{K}{Mr}. \tag{4.9}$$

[1]In an X-channel, each transmitter has an independent message for each receiver [43].

Note that the edge-NDT is the same as for EN cooperation due to ZF precoding at the cloud, while the fronthaul NDT is inversely proportional to the fronthaul rate r.

4.3.1 Offline Caching Policy

In the placement phase, the offline caching strategy operates differently depending on the values of the fronthaul rate r.

Low fronthaul regime: If the fronthaul rate r is smaller than a threshold r_{th}, the scheme attempts to maximize the use of the ENs' caches by distributing the maximum fraction of each popular file among all the ENs. When $\mu \leq 1/M$, this is done by storing non-overlapping fractions of each popular file at different ENs, leaving a fraction uncached (see top-left part of Figure 4.2). When $\mu \geq 1/M$, this approach yields a fraction $(\mu M - 1)/(M - 1)$ of each file that is shared by all ENs with no uncached parts (see top-right part of Figure 4.2). The threshold is identified in [17] as $r_{th} = K(M-1)/(M(\min\{M, K\} - 1))$.

High fronthaul: If $r \geq r_{th}$, a common μ-fraction of each file is placed at all ENs, as illustrated in the bottom part of Figure 4.2, in order to maximize the opportunities for EN cooperation, hence always leaving a fraction uncached unless $\mu = 1$.

4.3.2 Offline Delivery Policy

In the delivery phase, the policy operates as follows. With reference to Figure 4.2, fractions of each requested file stored at different ENs are delivered using EN coordination; the uncached fractions are delivered using C-RAN transmission; and fractions shared by all ENs are delivered using EN cooperation. Time sharing between pairs of such strategies is used in order to transmit different fractions of the requested files. For instance, when $r \geq r_{th}$, the approach time-shares between EN cooperation and C-RAN transmission (see bottom of Figure 4.2).

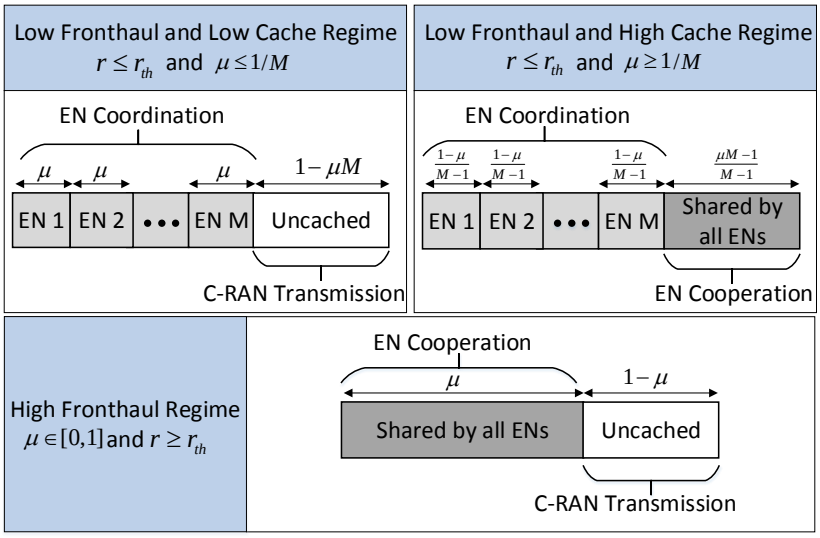

Figure 4.2 Illustration of the offline caching policy proposed in [17]. Note that each EN caches a fraction μ of each file.

4.3.3 Achievable NDT

The achievable NDT in each time slot of the outlined offline caching and delivery policy is denoted by $\delta_{\text{off,ach}}(\mu, r)$. Analytical expression for $\delta_{\text{off,ach}}(\mu, r)$ can be obtained from (4.7)-(4.9) using time sharing. In particular if a fraction λ of the requested file is delivered using a policy with NDT δ' and the remaining fraction using a policy with NDT δ'', the overall NDT is given by $\lambda\delta' + (1-\lambda)\delta''$ Following proposition presents the resulting achievable offline NDT of the described scheme.

Lemma 4.1. *(Achievable Offline NDT [17, Propositions 4]). For an $M \times K$ F-RAN with $N \ge M \ge K \ge 2$, the achievable NDT of offline caching and delivery policy is given as*

$$\delta_{\text{off,ach}}(\mu, r) \triangleq (M+K-1)\mu + (1-\mu M)\Big[\frac{K}{\min\{M,K\}} + \frac{K}{Mr}\Big] \tag{4.10}$$

for $\mu \in [0, 1/M]$, *and* $r \leq r_{th}$ *or*

$$\delta_{\text{off,ach}}(\mu, r) \triangleq \frac{K}{\min\{M, K\}}\left(\frac{\mu M - 1}{M - 1}\right) + (1 - \mu)\frac{M + K - 1}{M - 1} \tag{4.11}$$

for $\mu \in [1/M, 1]$ *and* $r \leq r_{th}$ *or*

$$\delta_{\text{off,ach}}(\mu, r) \triangleq \frac{K}{\min\{M, K\}} + \frac{(1 - \mu)K}{Mr}. \tag{4.12}$$

for $r \geq r_{th}$.

Letting $\delta^*_{\text{off}}(\mu, r)$ be the minimum offline NDT, the achievable NDT of the offline caching and delivery policy was proved to be within a factor of 2 of optimality in the sense that we have the inequality [17, Proposition 8]

$$\frac{\delta_{\text{off,ach}}(\mu, r)}{\delta^*_{\text{off}}(\mu, r)} \leq 2. \tag{4.13}$$

4.4 Achievable Long-Term NDT

This section proposes online edge caching with fronthaul-edge transmission policies operating under known and unknown popular set assumptions, and evaluates the performance for serial fronthaul-edge transmission. Lower bounds on the minimum long-term NDT will be presented in Section 4.5.

4.4.1 C-RAN Delivery

C-RAN delivery neglects the cached contents and uses C-RAN transmission in each time slot. This achieves a long-term NDT that coincides with the offline NDT (4.9) obtained in each time slot, i.e.,

$$\bar{\delta}_{\text{C-RAN}}(r) = \delta_{\text{E,C-RAN}}(r) + \delta_{\text{F,C-RAN}}(r) = \frac{K}{\min\{M, K\}} + \frac{K}{Mr}. \tag{4.14}$$

4.4.2 Reactive Online Caching with Known Popular Set

This section considers online caching schemes with reactive strategies that update the ENs' caches every time an uncached file is requested by any user. Discussion on proactive strategies is postponed to Section 4.5. First, the simple case of known popular set is considered , and then the unknown popular set case is studied in the next subsection.

If, at time slot t, Y_t requested files, with $0 \leq Y_t \leq K$, are not cached at the ENs, a fraction of each requested and uncached file is reactively sent on the fronthaul link to each EN at the beginning of the time slot. To select this fraction, we follow the offline caching policy summarized in Section 4.3 and Figure 4.2. Therefore, we cache a fraction μ of the file at all ENs, where the fraction is selected depending on the values of r and μ as in Figure 4.2. An improved selection of the size of cached fraction will be discussed later in this section. In order to make space for new files, the ENs evict files that are no longer popular as instructed by the cloud. Note that this eviction mechanism is feasible since the cloud knows the set $\mathcal{L}_t$. Furthermore, it is guaranteed to satisfy the ENs' cache capacity of μNF bits in each time slot. As a result of the cache update, each requested file is cached as required by the offline delivery strategy summarized in Section 4.3 and Figure 4.2, which is adopted for delivery.

The overall NDT is hence the sum of the NDT $\delta_{\text{off,ach}}(\mu, r)$ achievable by the offline delivery policy described in Section 4.3 and of the NDT due to the fronthaul transfer of the μ-fraction of each requested and uncached file on the fronthaul link. By (4.5), the latter equals $(\mu/C_F) \times \log P = \mu/r$, and hence the achievable NDT at each time slot t is

$$\delta_t(\mu, r) = \delta_{\text{off,ach}}(\mu, r) + \frac{\mu \mathrm{E}[Y_t]}{r}. \tag{4.15}$$

The following proposition presents the resulting achievable long-term NDT of reactive online caching with known popular set.

Proposition 4.1. *For an $M \times K$ F-RAN with $N \geq K$, in the known popular set case, online reactive caching achieves the long-term NDT*

$$\bar{\delta}_{\text{react,k}}(\mu, r) = \delta_{\text{off,ach}}(\mu, r) + \frac{\mu}{r}\Big(\frac{Kp}{K(1-p/N)+p}\Big), \tag{4.16}$$

where $\delta_{\text{off,ach}}(\mu, r)$ is the offline achievable NDT.

Proof. The result follows by analyzing the Markov chain that describes the number of popular cached files which in turn contributes to the second term in (4.15). Details can be found in Appendix C.1. □

We now propose an improvement that is based on an *adaptive* choice of the file fraction to be cached, as a function of the probability p of new file, as well as the fractional cache size μ and the fronthaul rate r. The main insight here is that, if the probability p is large, it is preferable to cache a fraction smaller than μ when the resulting fronthaul overhead offsets the gain accrued by means of caching. It is emphasized that caching a fraction smaller than μ entails that the ENs' cache capacity is partially unused.

Proposition 4.2. *For an $M \times K$ F-RAN with $N \geq K$, in the known popular set case, online reactive caching with adaptive fractional caching achieves the following long-term NDT*

$$\bar{\delta}_{\text{react,adapt,k}}(\mu, r) = \begin{cases} \bar{\delta}_{\text{react,k}}(\mu, r) & \text{if } p \leq p_0(\mu, r) \\ \bar{\delta}_{\text{react,k}}(1/M, r) & \text{if } p_0(\mu, r) \leq p \leq p_1(\mu, r) \\ \bar{\delta}_{\text{C-RAN}}(r) & \text{if } p_1(\mu, r) \leq p \leq 1 \end{cases} \tag{4.17}$$

where probabilities $p_0(\mu, r)$ and $p_1(\mu, r)$ satisfy $p_0(\mu, r) \leq p_1(\mu, r)$ and the full expressions are given in Appendix C.2. Furthermore, we have the inequalities $\bar{\delta}_{\text{react,adapt,k}}(\mu, r) \leq \bar{\delta}_{\text{react,k}}(\mu, r)$ with equality if and only if $p \leq p_0(\mu, r)$.

Proof. See Appendix C.2. □

As anticipated, Proposition 4.2 is based on the observation that, when the probability of a new file is larger than a threshold, identified as $p_1(\mu, r)$ in (4.17), it is preferable not to update the caches and simply use C-RAN delivery. Instead, with moderate probability p, i.e., $p_0(\mu, r) \leq p \leq p_1(\mu, r)$, the performance of the reactive scheme in Proposition 4.1 is improved by caching a smaller fraction than μ, namely $1/M \leq \mu$. Finally, when $p \leq p_0(\mu, r)$, no gain can be accrued by caching a fraction smaller than μ. This observation is illustrated in Figure 4.3, which shows the NDT of reactive caching in the known popular set case, without adaptive caching (Proposition 4.1) and with adaptive caching (Proposition 4.2) in the top figure and the corresponding cached fraction in the bottom figure for $M = 10$, $K = N = 5$, $r = 1.1$ and $\mu = 0.5$.

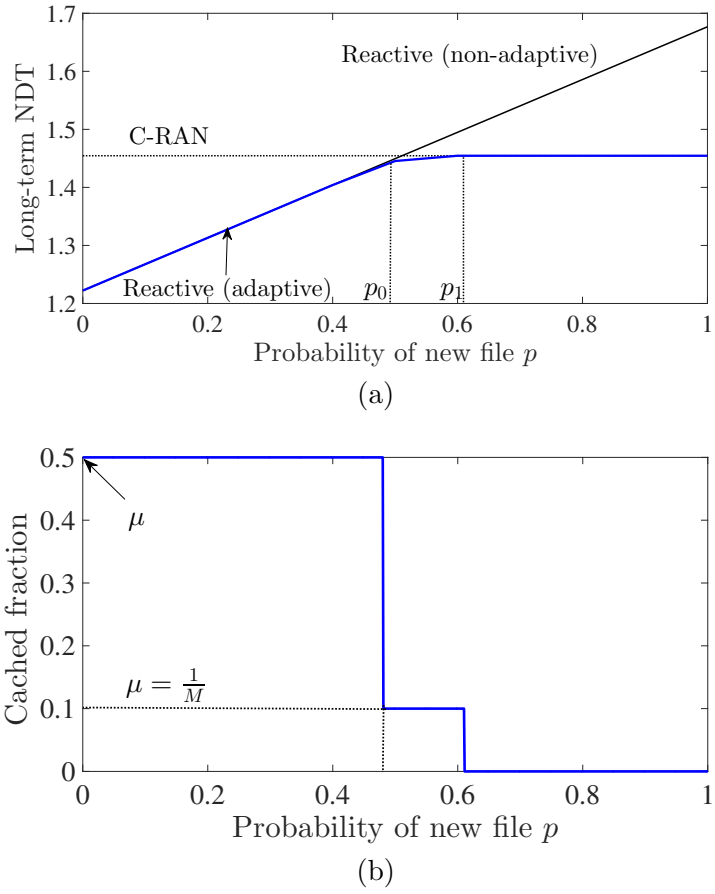

Figure 4.3 Reactive online caching with known popular set under non-adaptive caching (Proposition 4.1) and adaptive caching (Proposition 4.2) with $M = 10$, $K = N = 5$, $r = 1.1$ and $\mu = 0.5$: (a) NDT; (b) fraction cached by the adaptive caching scheme.

4.4.3 Reactive Online Caching with Unknown Popular Set

In the absence of knowledge about the popular set $\mathcal{L}_t$, the cloud cannot instruct the ENs about which files to evict while guaranteeing that no popular files will be removed from the caches. To account for this constraint, we now consider a reactive caching scheme whereby the ENs evict from the caches a randomly selected file. Random eviction can be improved by other eviction strategies, which are more difficult to analyze, as discussed in Section 4.7.

To elaborate, as pointed out in [3], in order to control the probability of evicting a popular file, it is useful for the ENs to cache a number $N^{'} = \alpha N$ files for some $\alpha > 1$. Note that in general the set of $N^{'} > N$ cached files in the cached contents $S_{m,t}$ of all ENs m generally contains files that are no longer in the set $\mathcal{L}_t$ of N popular files.

If Y_t requested files, with $0 \leq Y_t \leq K$, are not cached at the ENs, we propose to transfer a μ/α-fraction of each requested and uncached file on the fronthaul link to each EN by following the offline caching policy in Figure 4.2 with μ/α in lieu of μ. Caching a fraction μ/α is necessary in order to satisfy the cache constraint given the larger number N' of cached files. Delivery then takes place via the achievable offline delivery strategy reviewed in Figure 4.2, with the only caveat that μ/α should replace μ.

The overall NDT is hence the sum of the NDT $\delta_{\text{off,ach}}(\mu/\alpha, r)$ achievable by the offline delivery policy when the fractional cache size is μ/α and of the NDT due to the fronthaul transfer of the μ/α-fraction of each requested and uncached file on the fronthaul link. By (4.5), the latter equals $((\mu/\alpha)/C_F) \times \log P = \mu/(\alpha r)$, and hence the overall achievable NDT at each time slot t is

$$\delta_t(\mu, r) = \delta_{\text{off,ach}}\Big(\frac{\mu}{\alpha}, r\Big) + \frac{\mu}{\alpha}\Big(\frac{\mathrm{E}[Y_t]}{r}\Big). \tag{4.18}$$

The following proposition presents an achievable long-term NDT for the proposed reactive online caching policy.

Proposition 4.3. *For an $M \times K$ F-RAN with $N \geq K$, in the unknown popular set case, the online reactive caching scheme achieves the long-term NDT that is upper bounded as*

$$\bar{\delta}_{\text{react,u}}(\mu, r) \leq \delta_{\text{off,ach}}\Big(\frac{\mu}{\alpha}, r\Big) + \frac{p\mu}{r(1-p/N)(\alpha-1)}, \tag{4.19}$$

where $\delta_{\text{off,ach}}(\mu, r)$ is offline achievable NDT and $\alpha > 1$ is an arbitrary parameter.

Proof. Plugging the achievable NDT (4.18) into the definition of long-term NDT in (4.6), we have

$$\bar{\delta}_{\text{react,u}}(\mu, r) = \delta_{\text{off,ach}}\Big(\frac{\mu}{\alpha}, r\Big) + \Big(\frac{\mu}{\alpha r}\Big)\limsup_{T\to\infty}\frac{1}{T}\sum_{t=1}^{T}\mathrm{E}[Y_t]. \tag{4.20}$$

Furthermore, since the users' demand distribution, caching and random eviction policies are the same as in [3], we can leverage [3, Lemma 3] to obtain the following upper bound on the long-term average number of requested but not cached files as

$$\limsup_{T\to\infty}\frac{1}{T}\sum_{t=1}^{T}\mathrm{E}[Y_t] \leq \frac{p}{(1-p/N)(1-1/\alpha)}. \tag{4.21}$$

Plugging (4.21) into (4.20) completes the proof. □

We emphasize that the right-hand side of (4.20) is an upper bound on an achievable NDT and hence it is also achievable.

In the same way as in the known popular set case, the achievable long-term NDT in the unknown popular set case can be improved by adaptive caching, as elaborated by the following proposition.

Proposition 4.4. *For an $M \times K$ F-RAN with $N \geq K$, in the unknown popular set case, online reactive caching with adaptive fractional caching achieves a long-term NDT that is upper bounded as*

$$\bar{\delta}_{\text{react,adapt,u}}(\mu, r) \leq \begin{cases} \bar{\delta}_{\text{react,u}}(\mu, r) & \text{if } p \leq p_0(\mu, r) \\ \bar{\delta}_{\text{react,u}}(\alpha/M, r) & \text{if } p_0(\mu, r) \leq p \leq p_1(\mu, r) \\ \bar{\delta}_{\text{C-RAN}}(r) & \text{if } p_1(\mu, r) \leq p \leq 1 \end{cases} \tag{4.22}$$

where probabilities $p_0(\mu, r)$ and $p_1(\mu, r)$ satisfy $p_0(\mu, r) \leq p_1(\mu, r)$ and are defined in Appendix C.3.

Proof. See Appendix C.3. □

The right-hand side of (4.22) is always less than the right-hand side of (4.19), except for $p \leq p_0(\mu, r)$, where equality holds. In a similar manner to Proposition 4.2, depending on the probability p of new file, the adaptive caching scheme chooses among cloud-only delivery and online caching with different file fractions delivered on the fronthaul. Specifically, the fraction is either selected as α/M or as μ, where, as discussed, parameter α controls the eviction probability of popular files. The performance is akin to that illustrated in Figure 4.3 for the known popular set case.

4.5 Pipelined Fronthaul-Edge Transmission

As an alternative to the serial delivery model discussed in Section 4.4, in the pipelined fronthaul-edge transmission model, the ENs can transmit on the wireless channel

while receiving messages on the fronthaul link. Intuitively, pipelining can make caching more useful in reducing the transmission latency with respect to serial delivery. In fact, with pipelining, while the ENs transmit the cached files, they can receive the uncached information on the fronthaul links at no additional cost in terms of latency. Pipelined delivery was studied in [17,42] under offline caching. With online caching, as we will discuss here, pipelined fronthaul-edge transmission creates new opportunities that can be leveraged by means of proactive, rather than reactive, caching. We recall that proactive caching entails the storage of as-of-yet unrequested files at the ENs.

In the following, first the system model for pipelined transmission is described, then results for offline caching from [17, 42] are reviewed, and finally reactive and proactive online caching policies are proposed.

4.5.1 System Model

The system model for pipelined fronthaul-edge transmission follows in Section 4.2 with the following differences. First, as discussed, each EN can transmit on the edge channel and receive on the fronthaul link at the same time. Transmission on the wireless channel can hence start at the beginning of the transmission interval, and the ENs use the information received on the fronthaul links in a causal manner. Accordingly, at any time instant l within a time slot t, the edge transmission policy of EN m maps the demand vector d_t, the global CSI H_t, the local cache content $S_{m,t}$ and the fronthaul messages $U_{m,t,l'}$ received at previous instants $l' \leq l-1$, to the transmitted signal $X_{m,t,l}$ at time l.

Second, the NDT performance metric needs to be adapted. To this end, we denote the overall transmission time in symbols within slot t as T_t^{pl}, where the superscript "pl" indicates pipelined transmission. For a given sequence of feasible

policies, the average achievable delivery time per bit in slot t is defined as

$$\Delta_t^{pl}(\mu, C_F, P) = \lim_{F\to\infty} \frac{1}{F}\mathrm{E}[T_t^{pl}], \tag{4.23}$$

where the average is taken with respect to the distributions of the random variables $\mathcal{L}_t$, d_t and H_t. The corresponding NDT achieved at time slot t is

$$\delta_t^{pl}(\mu, r) = \lim_{P\to\infty} \frac{\Delta_t^{pl}(\mu, r\log P, P)}{1/\log(P)} \tag{4.24}$$

and, the *long-term NDT* is defined as

$$\bar{\delta}_{\text{on}}^{pl}(\mu, r) = \limsup_{T\to\infty} \frac{1}{T}\sum_{t=1}^{T} \delta_t^{pl}(\mu, r). \tag{4.25}$$

We denote the minimum long-term NDT over all feasible policies under the known popular set assumption as $\bar{\delta}_{\text{on,k}}^{pl^*}(\mu, r)$, while $\bar{\delta}_{\text{on,u}}^{pl^*}(\mu, r)$ indicates the minimum long-term NDT under the unknown popular set assumption. As a benchmark, we also consider the minimum NDT for offline edge caching $\delta_{\text{off}}^{pl^*}(\mu, r)$ as studied in [17]. By construction, we have the inequalities $\delta_{\text{off}}^{pl^*}(\mu, r) \leq \bar{\delta}_{\text{on,k}}^{pl^*}(\mu, r) \leq \bar{\delta}_{\text{on,u}}^{pl^*}(\mu, r)$. Furthermore, while pipelining generally improves the NDT performance as compared to serial delivery, the following result demonstrates that the NDT can be reduced by at most a factor of 2.

Lemma 4.2. *For an $M \times K$ F-RAN with $N \geq K$, the minimum long-term NDT under online caching with pipelined fronthaul-edge transmission satisfies*

$$\bar{\delta}_{\text{on}}^{pl^*}(\mu, r) \geq \frac{1}{2}\bar{\delta}_{\text{on}}^{*}(\mu, r), \tag{4.26}$$

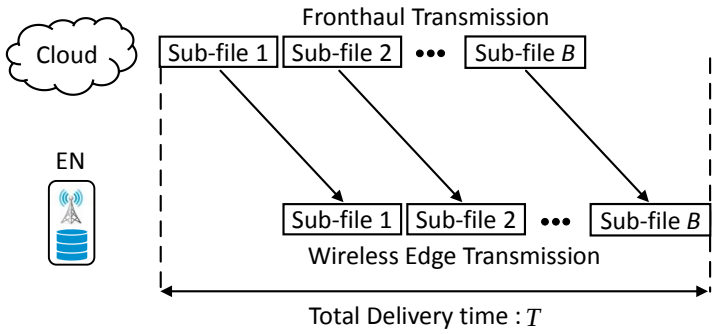

Figure 4.4 Block-Markov encoding converts a serial fronthaul-edge transmission policy into a pipelined transmission policy.

where $\bar{\delta}^*_{\text{on}}(\mu, r)$ *is the minimum long-term NDT of online caching under serial fronthaul-edge transmission.*

Proof. Consider an optimal policy for pipelined transmission. We show that it can be turned into a serial policy with a long-term NDT which is at most double the long-term NDT for the pipelined scheme. To this end, it is sufficient for the ENs to start transmission on the edge after they receive messages on the fronthaul links. The resulting NDT in each time slot is hence at most twice the optimal long-term NDT of pipelined transmission, since in the pipelined scheme both fronthaul and edge transmission times are bounded by the overall latency. □

4.5.2 Preliminaries

As done for serial transmission, we first review the existing results for offline caching with pipelined transmission. The achievable scheme for offline caching proposed in [17] utilizes *block-Markov encoding*, which is illustrated in Figure 4.4. Block-Markov encoding converts a serial fronthaul-edge strategy into a pipelined scheme. To this end, each file of F bits is divided into B sub-files, each of F/B bits. The transmission interval is accordingly divided into $B + 1$ sub-frames. The first sub-file is sent in the first sub-frame on the fronthaul, and then on the wireless link in the second sub-frame.

During the transmission of first sub-file on the wireless link, the second sub-file is sent on the fronthaul link in the second sub-frame. As illustrated in Figure 4.4, the same transmission scheme is used for the remaining sub-blocks.

If the original serial transmission scheme has edge-NDT δ_E and fronthaul-NDT δ_F, then the NDT of pipelined scheme when $B \to \infty$ can be proved to be [17]

$$\delta_{\text{off}}^{pl} = \max\left(\delta_E, \delta_F\right). \tag{4.27}$$

This is because the maximum of the latencies of the simulatneously occuring fronthaul and edge transmissions determines the transmisson time in each time slot. In [17], an achievable NDT $\delta_{\text{off,ach}}^{pl}(\mu, r)$ is obtained by using block Markov encoding along with file splitting and time sharing using the same constituent schemes considered in Section 4.3 for serial transmission. We refer for details to [17, Sec. VI-B]. We now discuss online caching strategies.

4.5.3 C-RAN Delivery

Similar to Section 4.4.1, with C-RAN delivery, the long-term NDT coincides with the NDT obtained in each time slot using in (4.27) the edge and fronthaul-NDTs in (4.9), yielding

$$\bar{\delta}_{\text{C-RAN}}^{pl}(\mu, r) = \max\left(\frac{K}{\min(M, K)}, \frac{K}{Mr}\right). \tag{4.28}$$

4.5.4 Reactive Online Caching

We now briefly discuss reactive caching policies that extend the approach in Section 4.4.2 and Section 4.4.3 to pipelined transmission. We recall that, with reactive

caching, the requested files that are not partially cached at ENs are reactively sent on the fronthaul. Furthermore, if the set of popular files is known, the ENs can evict the unpopular cached files to make space for newly received files. Otherwise, in the unknown popular set case, a randomly selected file can be evicted. As for serial transmission, we propose to choose the fraction to be cached and to perform delivery by following the offline caching strategy of [17]. Obtaining closed forms for the achievable long-term NDTs appears to be prohibitive. In Section 4.7, we provide Monte Carlo simulation to numerically illustrate the performance of the proposed scheme.

4.5.5 Proactive Online Caching

With pipelined transmission, by (4.27), when the rate of the fronthaul link is sufficiently large, the NDT in a time slot may be limited by the edge transmission latency. In this case, the fronthaul can be utilized to send information even if no uncached file has been requested without affecting the NDT. This suggests that proactive caching, whereby uncached and unrequested files are pushed to the ENs, potentially advantageous over reactive caching. Note that this is not the case for serial transmission, whereby any information sent on the fronthaul contributes equally to the overall NDT, irrespective of the edge latency, making it only useful to transmit requested files on the fronthaul links.

To investigate this point, here we study a simple proactive online caching scheme. Accordingly, every time there is a new file in the popular set, a μ-fraction of the file is proactively sent on the fronthaul links in order to update the ENs' cache content. This fraction is selected to be distinct across the ENs if $\mu \leq 1/M$, hence enabling EN coordination (recall the top-left part of Figure 4.2); while the same fraction μ is cached at all ENs otherwise, enabling delivery via EN cooperation (see bottom of Figure 4.2).

Proposition 4.5. *For an $M \times K$ F-RAN with $N \geq K$ and pipelined transmission, proactive online caching achieves the long-term NDT*

$$\begin{aligned}\bar{\delta}^{pl}_{\text{proact}}(\mu, r) = p \max \Big\{ &(\mu M)\delta_{\text{F,Coor}} + (1-\mu M)\delta_{\text{F,C-RAN}}(r) + \frac{\mu}{r}, \\ &(\mu M)\delta_{\text{E,Coor}} + (1-\mu M)\delta_{\text{E,C-RAN}}(r) \Big\} \\ + (1-p) \max \Big\{ &(\mu M)\delta_{\text{F,Coor}} + (1-\mu M)\delta_{\text{F,C-RAN}}(r), \\ &(\mu M)\delta_{\text{E,Coor}} + (1-\mu M)\delta_{\text{E,C-RAN}}(r) \Big\}, \end{aligned} \tag{4.29}$$

for $\mu \in [0, 1/M]$, and

$$\begin{aligned}\bar{\delta}^{pl}_{\text{proact}}(\mu, r) = p \max \Big\{ &\mu\delta_{\text{F,Coop}} + (1-\mu)\delta_{\text{F,C-RAN}}(r) + \frac{\mu}{r}, \\ &\mu\delta_{\text{E,Coop}} + (1-\mu)\delta_{\text{E,C-RAN}}(r) \Big\} \\ + (1-p) \max \Big\{ &\mu\delta_{\text{F,Coop}} + (1-\mu)\delta_{\text{F,C-RAN}}(r), \\ &\mu\delta_{\text{E,Coop}} + (1-\mu)\delta_{\text{E,C-RAN}}(r) \Big\}, \end{aligned} \tag{4.30}$$

for $\mu \in [1/M, 1]$, with the definitions given in (4.7)-(4.9).

Proof. With probability of $(1-p)$, the popular set remains unchanged and the NDT in the given slot is obtained by time sharing between C-RAN delivery, for the uncached $(1-\mu)$-fraction of the requested files, and either EN coordination and EN cooperation, depending on the value of μ as discussed above, for the cached μ-fraction. Instead, with probability p, there is a new file in the popular set, and a μ fraction of file is proactively sent on the fronthaul links, resulting in an additional term μ/r to the fronthaul-NDT of offline schemes. □

4.6 Impact of Time-Varying Popularity

This section compares the performance of offline caching in the presence of a static set of popular files with the performance of online caching under the considered dynamic popularity model. The analysis is intended to bring insight into the impact of a time-varying popular set on the achievable delivery latency. We focus here on the case in which the number of ENs is larger than the number of users, namely, $M \geq K$.

Proposition 4.6. *For an $M \times K$ F-RAN and $N > M \geq K \geq 2$ and $r > 0$, under both serial and pipelined delivery modes with known and unknown popular set, the minimum long-term NDT $\bar{\delta}^*_{\text{on}}(\mu, r)$ satisfies the condition*

$$\bar{\delta}^*_{\text{on}}(\mu, r) = c\delta^*_{\text{off}}(\mu, r) + O\Big(\frac{1}{r}\Big), \tag{4.31}$$

where $c \leq 4$ is a constant and $\delta^*_{\text{off}}(\mu, r)$ is the minimum NDT under offline caching.

Proof. See Appendix C.4. □

Proposition 4.6 shows that the long-term NDT with online caching is proportional to the minimum NDT for offline caching and static popular set, with an additive gap that is inversely proportional to the fronthaul rate r. To see intuitively why this result holds, note that, when $\mu \geq 1/M$ and hence the set of popular files can be fully stored across all the M EN's caches, offline caching enables the delivery of all possible users' requests with a finite delay even when $r = 0$. In contrast, with online caching, the time variability of the set $\mathcal{L}_t$ of popular files implies that, with non-zero probability, some of the requested files cannot be cached at ENs and hence should be delivered by leveraging fronthaul transmission. Therefore, the additive latency gap as a function of r is a fundamental consequence of the time-variability of the content set.

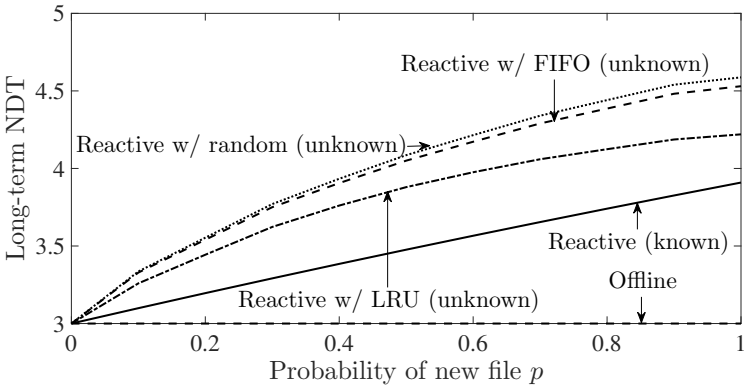

Figure 4.5 Long-term NDT of reactive online caching with known popular set, as well as reactive online caching with unknown popular set using different eviction policies ($M = 2$, $K = 5$, $N = 10$, $\mu = 0.5$ and $r = 0.5$).

4.7 Numerical Results

In this section, we complement the analysis of the previous sections with numerical experiments. We consider in turn serial transmission, as studied in Section 4.4, and pipelined transmission covered in Section 4.5. Unless stated otherwise, we set $M = 2$ and $K = 5$.

Serial delivery: For serial transmission, we consider the performance of reactive online caching with known popular set (eq. (4.16)) and unknown popular set (bound in (4.19)). For the latter, we evaluate the NDT via Monte Carlo simulations by averaging over a large number of realizations of the random process Y_t of requested but uncached files (see (4.20)), which is simulated starting from empty caches at time $t = 1$ and with $N = 10$. We also plot the NDT of offline scheme of [17] described in Section 4.3 in the presence of a time-invariant popular set. We first consider the impact of the rate of change of the popular content set for $\mu = 0.5$ and $r = 0.5$. To this end, the long-term NDT is plotted as a function of the probability p in Figure 4.5. We observe that variations in the set of popular files entail a performance loss of online caching with respect to offline caching with static popular set that increases with p. Furthermore, under random eviction, the lack of knowledge of the popular

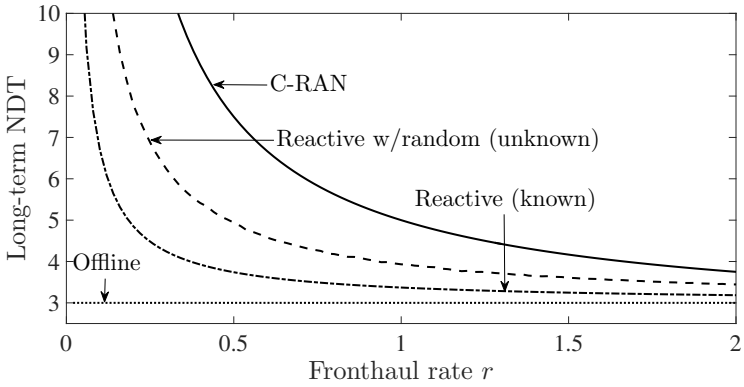

Figure 4.6 Long-term NDT of reactive online caching with known and unknown popular set, as well as C-RAN transmission and offline caching, under serial fronthaul-edge transmission ($M = 2$, $K = 5$, $N = 10$, $\mu = 0.5$ and $p = 0.8$).

set is seen to cause a significantly larger NDT than the scheme that can leverage knowledge of the popular set. This performance gap can be reduced by using better eviction strategies.

To investigate this point, we evaluate also the Monte Carlo performance of reactive online caching with unknown popular set under the following standard eviction strategies: Least Recently Used (LRU), whereby the replaced file is the one that has been least recently requested by any user; and First In First Out (FIFO), whereby the file that has been in the caches for the longest time is replaced. From Figure 4.5, LRU and FIFO are seen to be both able to improve over randomized eviction, with the former generally outperforming the latter, especially for large values of p. Finally, we note that for the long-term NDT of C-RAN (eq. (4.14)) is constant for all values of p and equal to 7.5 (not shown).

The impact of the fronthaul rate is studied next by means of Figure 4.6, in which the long-term NDT is plotted as a function of r for $\mu = 0.5$ and $p = 0.8$. The main observation is that, as r increases, delivering files from the cloud via fronthaul resources as in C-RAN yields decreasing latency losses, making edge caching less useful. In contrast, when r is small, an efficient use of edge resources via edge caching

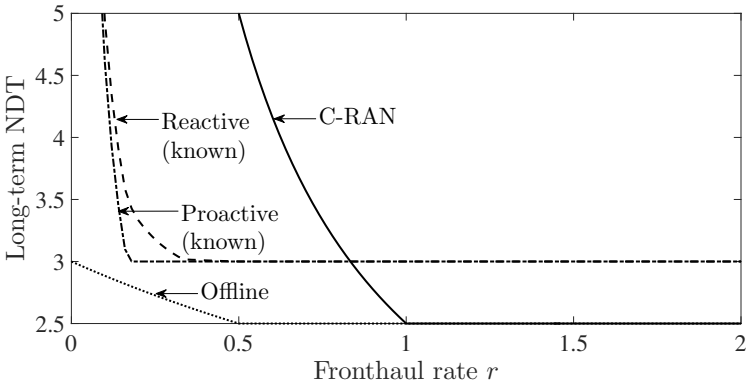

Figure 4.7 Long-term NDT of reactive and proactive online caching with known popular set, as well as C-RAN transmission and offline caching, under pipelined fronthaul-edge transmission ($M = 2$, $K = 5$, $N = 10$, $\mu = 0.5$ and $p = 0.8$).

becomes critical, and the achieved NDT depends strongly on the online cache update strategy.

Pipelined delivery: We now evaluate the long-term NDT performance of pipelined fronthaul-edge transmissions in Figure 4.7. The NDTs are computed using Monte Carlo simulations as explained above with reactive and proactive caching under known popular set. The plot shows the long-term NDT as a function of the fronthaul rate r for the same parameters considered in Figure 4.6. It is observed that, for all schemes, when the fronthaul rate is sufficiently large, the long-term NDT is limited by the edge-NDT (recall (4.27)) and hence increasing μ cannot reduce the NDT. In contrast, for smaller values of r, caching can decrease the long-term NDT. In this regime, proactively updating the ENs' cache content can yield a lower long-term NDT than reactive schemes, whereby the fronthaul links are underutilized, as discussed in Section 4.5.4. *Serial vs. pipelined delivery*: We now compare the long-term NDT performance of pipelined and serial fronthaul-edge transmission by means of Figure 4.8. The plot shows the long-term NDT of reactive online schemes under unknown popular set for serial transmission (eq. (4.20)) as well as pipelined transmission (see Section 4.5.4) as a function of the fractional cache size μ for $M = 2$, $K = 20$, $N = 30$, $r = 0.5$ and $p = 0.8$. For both cases, we evaluate the NDT via Monte Carlo

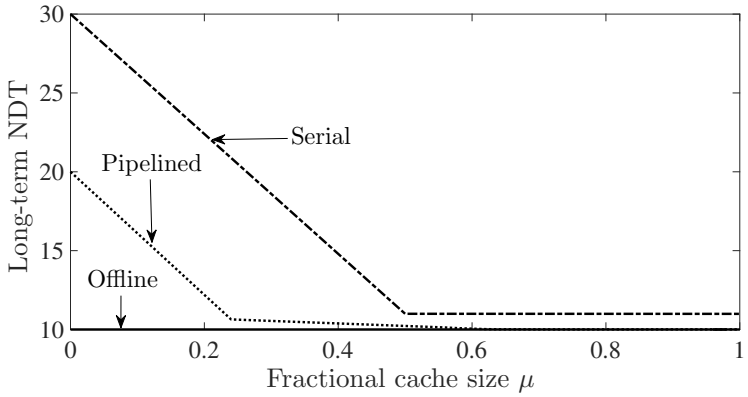

Figure 4.8 Long-term NDT of reactive online caching with known popular set for serial and pipelined transmission, as well as of offline caching ($M = 2$, $K = 20$, $N = 30$, $r = 0.5$ and $p = 0.8$).

simulations by averaging over a large number of realizations of the random process Y_t of requested but uncached files. As discussed in Section 4.5.5, pipelined transmission generally provides a smaller long-term NDT, and the gain in terms of NDT is limited to a factor of at most two. It is observed that, following the discussion in Section 4.5.5, edge caching enables larger gains in the presence of pipelined transmission owing to the capability of the ENs to transmit and receive at the same time. In particular, when μ is large enough, the achieved long-term NDT coincides with that of offline caching, indicating that, in this regime, the performance is dominated by the bottleneck set by wireless edge transmission.

4.8 Concluding Remarks

In this work, we considered the problem of content delivery in a fog architecture with time-varying content popularity. In this setting, online edge caching is instrumental in reducing content delivery latency. For the first time, an information-theoretic analysis is provided to obtain insights into the optimal use of fog resources, namely fronthaul link capacity, cache storage at edge, and wireless bandwidth. The analysis adopts a

high-SNR latency metric that captures the performance of online caching design. We first studied a serial transmission mode whereby the ENs start transmission on the wireless channel after completion of cloud-to-EN transmission. Then, we investigated a pipelined delivery mode that allows for simultaneous transmission and reception of information by the ENs. In both cases, the impact of knowledge of the set of popular files was considered.

One of the main insights of the analysis is that, regardless of the cache capacity at the edge and of prior knowledge at the cloud about the time-varying set of popular contents, the rate of the fronthaul links sets a fundamental limit to the achievable latency performance, since the only means of delivering new content is through the fronthaul links. Furthermore, unlike the serial mode, under the pipelined transmission mode, proactive online caching was found to provide potential gains as compared to reactive caching. This is due to the ability of proactive caching to opportunistically leverage unused fronthaul transmission capacity. Another interesting conclusion is that content eviction mechanisms such as LRU can bridge to some extent the gap between the performance under known and unknown popular set.

Among directions for future work, we mention here the analysis of the impact of imperfect CSI, of partial connectivity (see [44] and references therein), and of more realistic request models [31], as well as the investigation of online caching within large fog architectures including metro and core segments [45].

APPENDIX A

LOWER BOUNDS ON THE DELIVERY TIME PER BIT OF OFFLINE CACHING

In this appendix, we utilize information-theoretic approach to find fundamental lower bounds on the delivery time per bit of offline caching. First, a lower bound is derived for the scenario shown in Figure 2.1 with $C = 0$. i.e., the case that there is not a fronthaul link between cloud and small-cell BS. Then, a more general lower bound is derived for the case with $C > 0$.

A.1 Proof of Converse for Proposition 2.1

Consider any request vector $\mathbf{d}$ containing two arbitrary, different files W_1 and W_2, and any coding scheme satisfying $P_e^F \to 0$ as $F \to \infty$. The following set of inequalities is based on the fact that, under any such coding scheme, a hypothetical decoder provided with the CSI vector $\mathbf{G}^T$, with the cached contents V_1 and V_2 in (2.2) relative to files W_1 and W_2, and with the signal $\tilde{G}^T X_2^T$, to be described below, must be able to decode both messages W_1 and W_2. The signal $\tilde{G}^T X_2^T = (\tilde{G}(1)X_2(1), ..., \tilde{G}(T)X_2(T))$ is such that $\tilde{G}(t) = 0$ if $G_0(t) = G_2(t) = 0$ and $\tilde{G}(t) = 1$ otherwise. Note, therefore, that $\tilde{G}(t)X_2(t) = X_2(t)$ as long as either or both $G_0(t)$ and $G_2(t)$ are equal to one. The intuition here is that from $\tilde{G}^T X_2^T$ and G^T, the hypothetical decoder can recover Y_2^T and hence W_2; while from $\tilde{G}^T X_2^T$, G^T and V_1, the decoder can reconstruct Y_1^T

and hence decode W_1. Details are as follows

$$
\begin{aligned}
2F &= H(W_1, W_2) \\
&= I(W_1, W_2; \tilde{G}^T X_2^T, V_1, V_2, \mathbf{G}^T) \\
&\quad + \; H(W_1, W_2 | \tilde{G}^T X_2^T, V_1, V_2, \mathbf{G}^T) \\
&= I(W_1, W_2; \tilde{G}^T X_2^T, V_1, V_2, \mathbf{G}^T) \\
&\quad + \; H(W_1 | \tilde{G}^T X_2^T, V_1, V_2, \mathbf{G}^T) \\
&\quad + \; H(W_2 | \tilde{G}^T X_2^T, V_1, V_2, \mathbf{G}^T, W_1) \\
&\overset{(a)}{=} I(W_1, W_2; \tilde{G}^T X_2^T, V_1, V_2, \mathbf{G}^T) \\
&\quad + \; H(W_1 | \tilde{G}^T X_2^T, V_1, V_2, \mathbf{G}^T, Y_1^T) \\
&\quad + \; H(W_2 | \tilde{G}^T X_2^T, V_1, V_2, \mathbf{G}^T, W_1, Y_2^T) \\
&\overset{(b)}{\leq} I(W_1, W_2; \tilde{G}^T X_2^T, V_1, V_2, \mathbf{G}^T) + F\gamma_F \\
&= I(W_1, W_2; \tilde{G}^T X_2^T, V_1, V_2 | \mathbf{G}^T) + F\gamma_F \\
&\overset{(c)}{\leq} H(V_1) + H(\tilde{G}^T X_2^T | \mathbf{G}^T) + F\gamma_F \\
&\overset{(d)}{\leq} \mu F + T(1 - \epsilon_2^2) + F\gamma_F,
\end{aligned}
\tag{A.1}
$$

where γ_F indicates any function that satisfies $\gamma_F \to 0$ as $F \to \infty$. In above derivation, (a) follows from the facts that: (i) Y_1^T is a function of V_1, V_2, $\mathbf{G}^T$, and $\tilde{G}^T X_2^T$, since X_1^T can be assumed to depend on without loss of generality only on V_1 and V_2, and the vector $G_0^T X_2^T$ can be obtained from $\tilde{G}^T X_2^T$ and $\mathbf{G}^T$; (ii) Y_2^T is a function of $(\mathbf{G}^T, \tilde{G}^T X_2^T)$; (b) follows from Fano's inequality; (c) follows from

the fact that the messages are independent of channel realization and from Fano inequality $H(V_2|\tilde{G}^T X_2^T, \mathbf{G}^T) \leq F\gamma_F$; (d) hinges on the cache constraint (2.3) and by the following bounds

$$\begin{aligned} H(\tilde{G}^T X_2^T|\mathbf{G}^T) &\leq \sum_{t=1}^{T} H(\tilde{G}(t)X_2(t)|\mathbf{G}(t)) \\ &\leq T\sum_{\mathbf{g}\in\mathcal{G}} p(\mathbf{g}) \max_{p(X_2)} H(\tilde{G}X_2|\mathbf{G}=\mathbf{g}) \\ &\leq T(1-\epsilon_2^2), \end{aligned} \tag{A.2}$$

where $\mathcal{G}$ is the set of all channel states and the last inequality follows from the fact that the entropy in all states $\mathbf{G} = \mathbf{g}$ is maximized for $X_2 \sim$ Bernoulli(1/2). For $F \to \infty$, (A.1) yields the bound on the minimum DTB

$$\delta_{\text{off}}^*(\mu) \geq \frac{2-\mu}{1-\epsilon_2^2}. \tag{A.3}$$

Based on the fact that requested files should be retrieved from the received signals, another bound can be derived as follows:

$$\begin{aligned}
2F &= H(W_1, W_2) \\
&= I(W_1, W_2; Y_1^T, Y_2^T, \mathbf{G}^T) + H(W_1, W_2 | Y_1^T, Y_2^T, \mathbf{G}^T) \\
&\overset{(a)}{\leq} I(W_1, W_2; Y_1^T, Y_2^T, \mathbf{G}^T) + F\gamma_F \\
&\overset{(b)}{\leq} I(W_1, W_2; Y_1^T, Y_2^T | \mathbf{G}^T) + F\gamma_F \\
&= H(Y_1^T, Y_2^T | \mathbf{G}^T) + F\gamma_F \\
&\overset{(c)}{\leq} T \sum_{\mathbf{g} \in \mathcal{G}} p(\mathbf{g}) \max_{p(X_1, X_2)} H(Y_1, Y_2 | \mathbf{G} = \mathbf{g}) + F\gamma_F \\
&\overset{(d)}{=} T(2 - \epsilon_1 - \epsilon_2 + \epsilon_1\epsilon_2 - \epsilon_1\epsilon_2^2) + F\gamma_F,
\end{aligned} \tag{A.4}$$

where (a) follows from Fano's inequality; (b) follows from the fact that channel gains are independent from files; (c) follows in a manner similar to (A.2); and (d) is due to the fact that the entropy terms in the previous step are maximized by choosing X_1 and X_2 to be independent and identically distributed as Bernoulli(1/2). With $F \to \infty$, we obtain the bound

$$\delta_{\text{off}}^*(\mu) \geq \frac{2}{2 - \epsilon_1 - \epsilon_2 + \epsilon_1\epsilon_2 - \epsilon_1\epsilon_2^2}. \tag{A.5}$$

Considering decoder 2, the file W_2 should be decodable from Y_2^T, leading to the following bounds

$$\begin{aligned} F = H(W_2) &= I(W_2; Y_2^T, \mathbf{G}^T) + H(W_2|Y_2^T, \mathbf{G}^T) \\ &\overset{(a)}{\leq} I(W_2; Y_2^T|\mathbf{G}^T) + F\gamma_F \\ &\leq H(Y_2^T|\mathbf{G}^T) + F\gamma_F \\ &\overset{(b)}{\leq} T(1-\epsilon_2) + F\gamma_F, \end{aligned} \tag{A.6}$$

where (a) follows from Fano's inequality and (b) follows in a manner similar to (A.2) and the independence of channel gains from files. Therefore, based on (A.6) as $F \to \infty$, we obtain the bound

$$\delta^*_{\text{off}}(\mu) \geq \frac{1}{1-\epsilon_2}. \tag{A.7}$$

Combining (A.3), (A.5) and (A.7) yields the desired lower bound.

A.2 Proof of Converse for Proposition 2.2

Let us denote $\delta_C = T_C/F$ the normalized latency on the Cloud-to-Encoder 1 link and $\delta_E = T/F$ the normalized latency on the channel between encoders and decoders. We first observe that, following the same argument as in (A.4)–(A.7), we have the bound

$$\delta_E \geq \delta_0 \tag{A.8}$$

for any sequence of feasible policies. We now obtain a lower bound on both normalized delays δ_E and δ_C by observing that a hypothetical decoder provided with the CSI vector $\mathbf{G}^T$, with the cached content V_1 and V_2 in (2.2), with the cloud-aided message U^{T_C}, and with the signal $\tilde{G}^T X_2^T$ described in Appendix A.1 can decode both messages W_1 and W_2. Details are as follows

$$\begin{aligned} 2F &= H(W_1, W_2) \\ &= I(W_1, W_2; \tilde{G}^T X_2^T, V_1, V_2, U^{T_C}, \mathbf{G}^T) \\ &\quad + \; H(W_1, W_2 | \tilde{G}^T X_2^T, V_1, V_2, U^{T_C}, \mathbf{G}^T) \\ &\overset{(a)}{\leq} I(W_1, W_2; \tilde{G}^T X_2^T, V_1, V_2, U^{T_C}, \mathbf{G}^T) + F\gamma_F \\ &= I(W_1, W_2; \tilde{G}^T X_2^T, V_1, V_2, U^{T_C} | \mathbf{G}^T) + F\gamma_F \\ &\overset{(b)}{\leq} H(V_1) + H(U^{T_C}) + H(\tilde{G}^T X_2^T | \mathbf{G}^T) + F\gamma_F \\ &\overset{(c)}{\leq} \mu F + T_C C + T(1 - \epsilon_2^2) + F\gamma_F, \end{aligned} \tag{A.9}$$

where, as in Appendix A.1, γ_F indicates any function that satisfies $\gamma_F \to 0$ as $F \to \infty$. In above derivation, steps (a)–(b) follow as steps (a)–(b) in (A.1), where we note that the inequality $H(V_2 | \tilde{G}^T X_2^T, \mathbf{G}^T) \leq F\gamma_F$ by Fano inequality, while (c) hinges on the cache constraint (2.3) and the bound $H(U^{T_C}) \leq \sum_{i=1}^{T_C} H(U_i) \leq T_C C$ due to the capacity constraint on the cloud-to-encoder 1 link. As $F \to \infty$, the inequality (A.9)

yields the bound on the latency components δ_c and δ_E

$$\frac{1-\epsilon_2^2}{C}\delta_E + \delta_C \geq \frac{2-\mu}{C}. \tag{A.10}$$

To complete the proof, we combine bounds (A.8) and (A.10) as follows.

Low fronthaul capacity regime: For $C \leq 1-\epsilon_2^2$, the bound (A.10), directly yields

$$\delta^*_{\text{off}}(\mu, C) = \delta_E + \delta_C \geq \delta_E + \frac{C}{1-\epsilon_2^2}\delta_C \geq \frac{2-\mu}{1-\epsilon_2^2}. \tag{A.11}$$

High fronthaul capacity regime: For $C \geq 1-\epsilon_2^2$, two scenarios are possible. If $\mu \leq \mu_0$, multiplying (A.8) by the positive coefficient $1-(1-\epsilon_2^2)/C$ and summing the result with (A.10), provides the corresponding result in (2.24). Instead, if $\mu \geq \mu_0$, from (A.8), we directly obtain $\delta^*_{\text{off}}(\mu, C) \geq \delta_E \geq \delta_0$.

APPENDIX B

BOUNDS ON THE LONG-TERM DELIVERY TIME PER BIT OF ONLINE CACHING

In this appendix, we utilize information-theoretic approach to find upper and lower bounds on the long-term delivery time per bit of online caching. First, a lower bound is derived for the scenario shown in Figure 3.1. Then, an upper bound is derived for the scenario under study.

B.1 Proof of Proposition 3.3

To obtain a lower bound on the long-term DTB, following [2], we consider an enhanced system in which, at each time slot t, the small-cell BS is informed of the optimal cache content of an offline scheme tailored to the current popular set $\mathcal{L}_t$. In this system, at each time slot t, with probability of p there is a new file in the set of popular files, and hence the probability that an uncached file is requested by one of the users is $2p/N$. As a result, the DTB in time slot t for the genie-aided system can be lower bounded as

$$\delta_t \geq \Big(1 - \frac{2p}{N}\Big)\delta^*_{\text{off}}(\mu, C) + \Big(\frac{2p}{N}\Big)\delta_{\text{on,lb}}(C), \tag{B.1}$$

where $\delta^*_{\text{off}}(\mu, C)$ is the minimum DTB for the offline caching set-up in Proposition 2.2, while $\delta_{\text{on,lb}}(C)$ is a lower bound on the minimum DTB for offline caching in which all files but one can be cached.

To obtain the lower bound $\delta_{\text{on,lb}}(C)$, we start by noting that the set-up is equivalent to that for the proof in Appendix A.2 with the only difference is that one of the requested files by users cannot be cached at the small-cell BS. Since the probability of error (3.3) should be small for any request vector, in order to obtain a lower bound, we assume that the message W_1 requested by user 1 cannot be cached at the small-cell BS. Using the resulting condition $H(V_1) = 0$ in step (b) of (50) yields the inequality

$$2F \leq T_{C,t}C + T(1 - \epsilon_2^2) + F\gamma_F, \tag{B.2}$$

and hence, letting $\gamma_F \to 0$ as $F \to \infty$, we have the inequality

$$\frac{1 - \epsilon_2^2}{C}\delta_E + \delta_C \geq \frac{2}{C}. \tag{B.3}$$

To complete the proof, we combine bounds (A.8) and (B.3) as follows.

Low fronthaul capacity regime: For $C \leq 1 - \epsilon_2^2$, the bound (B.3), directly yields

$$\delta_{\text{on,lb}}(C) = \delta_E + \delta_C \geq \delta_E + \frac{C}{1 - \epsilon_2^2}\delta_C \geq \frac{2}{1 - \epsilon_2^2}. \tag{B.4}$$

High fronthaul capacity regime: For $C \geq 1-\epsilon_2^2$, multiplying (A.8) by the positive coefficient $1-(1-\epsilon_2^2)/C$ and summing the result with (B.3) yields the lower bound

$$\delta_{\text{on,lb}}(C) \geq \frac{2}{C} + \Big(1 - \frac{1-\epsilon_2^2}{C}\Big)\delta_0. \tag{B.5}$$

We note that comparing (B.4) and (B.5) with Propositions 2.1 and 2.2 reveals that when one of the requested files is not available at the small-cell BS, the system degrades to the case with zero caching at small-cell BS and hence we have

$$\delta_{\text{on,lb}}(C) \geq \delta^*_{\text{off}}(0, C). \tag{B.6}$$

Plugging (B.6) into (B.1) and then using (4.6) completes the proof.

B.2 Proof of Proposition 3.4

The lower bound follows directly from Proposition 3.3. To prove the upper bound, we leverage the following lemma.

Lemma B.1. *For any $\alpha > 1$, we have the following inequality*

$$\delta^*_{\text{off}}\Big(\frac{\mu}{\alpha}, C\Big) \leq \delta^*_{\text{off}}(\mu, C) + \max\left(\frac{2}{C}, \frac{\mu\left(1-\frac{1}{\alpha}\right)}{C}\right). \tag{B.7}$$

Proof. See Appendix B.3. □

Using Proposition 3.2 and Lemma B.1, an upper bound on the long-term DTB of the proposed reactive caching scheme is obtained as

$$\bar{\delta}_{\text{on,react}}(\mu, C) \leq \delta^*_{\text{off}}(\mu, C) + f(\alpha), \tag{B.8}$$

where

$$f(\alpha) = \frac{p\mu}{C(1-p/N)(\alpha-1)} + \max\left(\frac{2}{C}, \frac{\mu\left(1-\frac{1}{\alpha}\right)}{C}\right). \tag{B.9}$$

Since the additive gap (B.9) is a decreasing function of N and an increasing function of p and μ, it can be further upper bounded by setting $N = 2$, $p = 1$ and $\mu = 1$. By plugging $\alpha = 2$, we have

$$\bar{\delta}^*_{\text{on,k}}(\mu, C) \leq \bar{\delta}^*_{\text{on,u}}(\mu, C) \leq \bar{\delta}_{\text{on,react}}(\mu, C) \leq \min\left(\delta^*_{\text{off}}(0,0), \delta^*_{\text{off}}(\mu, C) + \frac{4}{C}\right). \tag{B.10}$$

The upper bound in (B.10) is obtained using the fact that the maximum delivery latency namely, $\delta^*_{\text{off}}(0,0)$ is achieved when both requested files are delivered by transmission from macro-BS. To complete the proof, we consider the following regimes

Low capacity regime ($C \leq 1 - \epsilon_2^2$): In this regime, using Propositions 2.2 and 3.3, the lower bound is

$$\left(1 - \frac{2p}{N}\right)\delta^*_{\text{off}}(\mu, C) + \frac{2p}{N}\frac{2}{1-\epsilon_2^2} \leq \bar{\delta}^*_{\text{on,k}}(\mu, C) \leq \bar{\delta}^*_{\text{on,u}}(\mu, C). \tag{B.11}$$

To prove the upper bound, we consider the following two sub-regimes

Low cache regime ($\mu \leq \mu_0$): In this case, using Proposition 2.2 and (B.10), we have

$$\bar{\delta}^*_{\text{on,k}}(\mu, C) \leq \bar{\delta}^*_{\text{on,u}}(\mu, C) \leq \min\Big(\delta^*_{\text{off}}(0,0), \delta^*_{\text{off}}(\mu, C) + \frac{4}{C}\Big) = \frac{2}{1-\epsilon_2^2}. \tag{B.12}$$

Using Proposition 2.2, the minimum offline DTB is $\delta^*_{\text{off}}(\mu, C) = (2-\mu)/(1-\epsilon_2^2)$ and therefore we have

$$\frac{\bar{\delta}^*_{\text{on,u}}(\mu, C)}{\delta^*_{\text{off}}(\mu, C)} \leq \frac{2}{2-\mu} \overset{(a)}{\leq} \frac{2}{2-\mu_0} \overset{(b)}{\leq} 2, \tag{B.13}$$

where (a) follows from $\mu \leq \mu_0$ and (b) follows from $0 \leq \mu_0 \leq 1$.

High cache regime ($\mu \geq \mu_0$): In this regime, using Proposition 2.2, the minimum offline DTB is $\delta^*_{\text{off}}(\mu, C) = \delta_0$ with δ_0 given by (2.15). Using (B.10), we have

$$\frac{\bar{\delta}^*_{\text{on,u}}(\mu, C)}{\delta^*_{\text{off}}(\mu, C)} \leq \frac{2}{\delta_0(1-\epsilon_2^2)} \overset{(a)}{\leq} \frac{2}{1+\epsilon_2} \overset{(b)}{\leq} 2, \tag{B.14}$$

where (a) follows from the definition of δ_0 in (2.15) and (b) follows from $0 \leq \epsilon_2 \leq 1$.

Combining (B.11), (B.13) and (B.14) results in (3.12).

High capacity regime ($C \geq 1-\epsilon_2^2$): In this regime, using Propositions 2.2 and 3.3, the lower bound is

$$\Big(1 - \frac{2p}{N}\Big)\delta^*_{\text{off}}(\mu, C) + \frac{2p}{N}\Big(\frac{2-(1-\epsilon_2^2)\delta_0}{C} + \delta_0\Big) \leq \bar{\delta}^*_{\text{on,k}}(\mu, C) \leq \bar{\delta}^*_{\text{on,u}}(\mu, C). \tag{B.15}$$

To prove the upper bound, using (B.10) and Proposition 2.2, we have

$$\bar{\delta}^*_{\text{on,u}}(\mu, C) \leq \delta^*_{\text{off}}(\mu, C) + \frac{4}{C}. \tag{B.16}$$

Combining (B.15) and (B.16) results in (3.13) and completes the proof.

B.3 Proof for Lemma B.1

To prove Lemma B.1, for any given $\alpha > 1$, we first define

$$\mu_1 = \min(1, \alpha\mu_0), \tag{B.17}$$

where μ_0 given by (2.14). Then, we consider separately small-cache regime with $\mu \in [0, \mu_0]$, medium-cache regime $\mu \in [\mu_0, \mu_1]$ and the high-cache regime with $\mu \in [\mu_1, 1]$.

Small-cache Regime ($\mu \in [0, \mu_0]$): Using (2.24), we have the following upper bound

$$\begin{aligned}
\delta^*_{\text{off}}\Big(\frac{\mu}{\alpha}, r\Big) &= \frac{2 - \frac{\mu}{\alpha}}{C} + \Big(1 - \frac{1 - \epsilon_2^2}{C}\Big)\delta_0 \\
&= \frac{2 - \mu}{C} + \Big(1 - \frac{1 - \epsilon_2^2}{C}\Big)\delta_0 + \frac{\mu\left(1 - \frac{1}{\alpha}\right)}{C} \\
&\overset{(a)}{=} \delta^*_{\text{off}}(\mu, C) + \frac{\mu\left(1 - \frac{1}{\alpha}\right)}{C},
\end{aligned} \tag{B.18}$$

where (a) follows from (2.24) in the regime of interest.

Medium-cache Regime ($\mu \in [\mu_0, \mu_1]$): Using (2.24), we have the following upper bound

$$\begin{aligned}\delta^*_{\text{off}}\Big(\frac{\mu}{\alpha}, r\Big) - \delta^*_{\text{off}}(\mu, C) &= \frac{2 - \frac{\mu}{\alpha}}{C} + \Big(1 - \frac{1 - \epsilon_2^2}{C}\Big)\delta_0 - \delta_0 \\ &\overset{(a)}{\leq} \frac{2}{C},\end{aligned} \tag{B.19}$$

where (a) is obtained by omitting the negative terms.

High-cache Regime ($\mu \in [\mu_1, 1]$): Using (2.24), we have

$$\delta^*_{\text{off}}\Big(\frac{\mu}{\alpha}, r\Big) = \delta^*_{\text{off}}(\mu, C) = \delta_0. \tag{B.20}$$

Finally, using (C.42), (B.19) and (B.20) concludes the proof.

APPENDIX C

BOUNDS ON THE LONG-TERM NORMALIZED DELIVERY TIME OF ONLINE EDGE CACHING IN FOG NETWORKS

In this appendix, we first utilize Markov analysis to find the expected number of files that should be sent on the fronthaul link for the system described in Chapter 4 with time varying set of popular files. Next, upper bounds on the long-term normalized delivery times of both reactive online caching with known and unknown popular set are derived. Finally, upper and lower bounds on the long-term normalized delivery time of a fog radio access network with online edge caching are derived.

C.1 Proof of Proposition 4.1

The proof follows closely the approach used in [3, Sec. V-A]. The goal is to compute the long-term NDT (4.6) using (4.15). We recall that $Y_t \in \{0, 1, ..., K\}$ is the number of new files requested by users at time slot t which are not available at ENs' caches. We further define as $X_t \in \{0, 1, ..., N\}$ the number of files in $\mathcal{L}_t$ which are available at ENs' caches at the beginning of time slot t, and as $V_t \in \{0, 1\}$ the number of files that were correctly cached at the end of time slot t but are no longer popular at time slot $t + 1$.

Using the above-mentioned random processes, the following update equation for the process X_t holds

$$X_{t+1} = X_t + Y_t - V_t. \tag{C.1}$$

As in [3], it can be shown that $\{X_t\}$ is a Markov process with a single ergodic recurrent class consisting the states $\{K-1, K, ..., N\}$ and transient states $\{0, 1, ..., K-2\}$. Hence, in the steady state where $\mathrm{E}[X_{t+1}] = \mathrm{E}[X_t]$, we have the equality

$$\mathrm{E}[Y_t] = \mathrm{E}[V_t]. \tag{C.2}$$

Furthermore, since each user requests a file in $\mathcal{L}_t$ according to uniform distribution without replacement, conditioned on X_t, the random variable Y_t has the expected value

$$\mathrm{E}[Y_t|X_t] = K\Big(1 - \frac{X_t}{N}\Big). \tag{C.3}$$

We will use (C.2) and (C.3) to compute the expectation $\mathrm{E}[Y_t]$ in steady state. Given the asymptotic stationarity of X_t, and hence of Y_t, by the standard Cesaro mean argument, the long-term NDT (4.15) is finally obtained by substituting in (4.6) the steady state mean $\mathrm{E}[Y_t]$.

To this end, denoting the number of cached popular files at the end of time slot t as X'_t, we have

$$X'_t = X_{t+1} + V_t, \tag{C.4}$$

since the number of cached popular files at the start of time slot $t+1$ is either the same as at the end of time slot t or to has one less file due to arrival of a new file in the popular set. Conditioning on X'_t, we have

$$\Pr(V_t = 1|X'_t) = p\frac{X'_t}{N}, \tag{C.5}$$

since with probability of p there is a new file in the popular set which replaces one of the cached popular files at ENs selected with probability of X'_t/N and these two events are independent of each other. Taking expectation with respect to X'_t in (C.5) and using the fact that $V_t \in \{0, 1\}$, we have

$$\mathrm{E}[V_t] = \mathrm{E}[\Pr(V_t = 1|X'_t)] = p\frac{\mathrm{E}[X'_t]}{N} \overset{(a)}{=} p\frac{\mathrm{E}[X_{t+1}] + \mathrm{E}[V_t]}{N} \overset{(b)}{=} p\frac{\mathrm{E}[X_t] + \mathrm{E}[V_t]}{N}, \tag{C.6}$$

where (a) is obtained using (C.4) and (b) is obtained for steady state where $\mathrm{E}[X_{t+1}] = \mathrm{E}[X_t]$. Solving for $E[V_t]$ yields

$$\mathrm{E}[V_t] = \frac{p}{1 - p/N}\frac{\mathrm{E}[X_t]}{N}. \tag{C.7}$$

Taking expectation respect to X_t from (C.3) and then using (C.7) and (C.2), we have

$$\mathrm{E}[Y_t] = K\Big(1 - \frac{\mathrm{E}[X_t]}{N}\Big) = \frac{Kp}{K(1 - p/N) + p}. \tag{C.8}$$

Plugging (C.8) into (4.15) completes the proof.

C.2 Proof of Proposition 4.2

The proposed adaptive caching scheme optimizes the choice of the cached fraction within the reactive scheme achieving $\bar{\delta}_{\text{react,k}}(\mu, r)$ in (4.16). To this end, it solves the problem $\bar{\delta}_{\text{react,adapt,k}}(\mu, r) = \min_{\mu' \leq \mu} \bar{\delta}_{\text{react,k}}(\mu', r)$ over the fraction μ'. With some algebra, this optimization yields

$$\begin{aligned} p_0(\mu, r) &= \min\left(\frac{Kr\big(\min(M,K)-1\big)}{K(M-1) + r\big(\min(M,K)-1\big)\big(\frac{K}{N}-1\big)}, 1 \right) \\ p_1(\mu, r) &= \min\left(\frac{K - r\big(\min(M,K)-1\big)}{1 + \Big(K - r\big(\min(M,K)-1\big)\Big)\big(1/N - 1/K\big)}, 1 \right) \end{aligned} \tag{C.9}$$

for $r \leq r_{th}$; and

$$p_0(\mu, r) = p_1(\mu, r) = \min\left(\frac{K}{M + K/N - 1}, 1 \right), \tag{C.10}$$

for $r \geq r_{th}$.

C.3 Proof of Proposition 4.4

We consider the following regimes:

Low fronthaul regime $r \leq r_{th}$: We consider two following regimes of cache capacity:

Low cache regime $\mu \leq \alpha/M$: Using Lemma 4.1 and Proposition 4.3, the NDT in the regime of interest is upper bounded as

$$\bar{\delta}_{\text{react,u}}(\mu, r) \leq \frac{\mu}{\alpha}\left(M + K - 1\right) + \left(1 - \frac{\mu M}{\alpha}\right)\left(\frac{K}{\min(M, K)} + \frac{K}{Mr}\right) + \frac{\mu}{r}\left(\frac{p}{(1 - p/N)(\alpha - 1)}\right). \tag{C.11}$$

The derivative of (C.11) with respect to μ is negative if

$$p \leq p_1(\mu, r) = \min\left(\frac{(\alpha - 1)\Big(K - r\big(\min(M, K) - 1\big)\Big)}{\alpha + \Big(\big(\alpha - 1\big)/N\Big)\Big(K - r\big(\min(M, K) - 1\big)\Big)}, 1\right), \tag{C.12}$$

while the derivative of (C.11) with respect to μ is positive if $p_1(\mu, r) \leq p \leq 1$. If derivative is positive, $\mu = 0$ minimizes (C.11), hence upper bound on the achievable NDT is

$$\bar{\delta}_{\text{react,adapt,u}}(\mu, r) \leq \bar{\delta}_{\text{react,u}}(0, r) = \bar{\delta}_{\text{C-RAN}}(r) = \frac{K}{\min\{M, K\}} + \frac{K}{Mr} \tag{C.13}$$

for $\mu \leq \alpha/M$, $r \leq r_{th}$ and $p_1(\mu, r) \leq p \leq 1$. Instead, if the derivative is negative the achievable NDT is obtained by (C.11) which is the extension of (4.19). As a result, upper bound o the achievable long-term NDT is

$$\bar{\delta}_{\text{react,adapt,u}}(\mu, r) \leq \bar{\delta}_{\text{react,u}}(\mu, r), \tag{C.14}$$

for $\mu \leq \alpha/M$, $r \leq r_{th}$ and $p \leq p_1(\mu, r)$.

High cache regime $\alpha/M \leq \mu \leq 1$: Using Lemma 4.1 and Proposition 4.3, the upper bound on the achievable NDT in the regime of interest is

$$\bar{\delta}_{\text{react,u}}(\mu, r) \leq \frac{K}{\min(M, K)} \left(\frac{\frac{\mu M}{\alpha} - 1}{M - 1} \right) + \left(1 - \frac{\mu}{\alpha}\right) \left(\frac{M + K - 1}{M - 1} \right) + \frac{\mu}{r} \left(\frac{p}{(1 - p/N)(\alpha - 1)} \right), \tag{C.15}$$

The derivative of (C.15) with respect to μ is negative if

$$p \leq p_0(\mu, r) = \min \left(\frac{r\big(\alpha - 1\big)\big(\min(M, K) - 1\big)}{\alpha(M - 1) + r\big(\big(\alpha - 1\big)/N\big)\big(\min(M, K) - 1\big)}, 1 \right), \tag{C.16}$$

Therefore, upper bound on the achievable long-term NDT is obtained by (C.15) which is the extension of (4.19), so we have

$$\bar{\delta}_{\text{react,adapt,u}}(\mu, r) \leq \bar{\delta}_{\text{react,u}}(\mu, r), \tag{C.17}$$

for $\alpha/M \leq \mu \leq 1$, $r \leq r_{th}$ and $p \leq p_0(\mu, r)$. The reverse of condition (C.16) in the regime of interest namely $p \geq p_0(\mu, r)$ means that the NDT (C.15) is an increasing function of μ and upper bound on the achievable long-term NDT is obtained by

plugging $\mu = \alpha/M$ into (C.15)

$$\bar{\delta}_{\text{react,adapt,u}}(\mu, r) \leq \bar{\delta}_{\text{react,u}}\Big(\frac{\alpha}{M}, r\Big) = \frac{M+K-1}{M} + \frac{p}{Mr}\left(\frac{\alpha}{(1-p/N)(\alpha-1)}\right). \tag{C.18}$$

as long as $\bar{\delta}_{\text{react,u}}(\alpha/M, r) \leq K/\min(M,K) + K/(Mr)$. Hence, (C.18) is the achievable long-term NDT for $\alpha/M \leq \mu \leq 1$, $r \leq r_{th}$ and $p_0(\mu, r) \leq p \leq p_1(\mu, r)$ with $p_0(\mu, r)$ and $p_1(\mu, r)$ are given in (C.16) and (C.12), respectively. Finally, upper bound on the achievable long-term NDT is

$$\bar{\delta}_{\text{react,adapt,u}}(\mu, r) \leq \bar{\delta}_{\text{react,u}}(0, r) = \bar{\delta}_{\text{C-RAN}}(r) = \frac{K}{\min\{M,K\}} + \frac{K}{Mr}. \tag{C.19}$$

for $\alpha/M \leq \mu \leq 1$, $r \leq r_{th}$ and $p_1(\mu, r) \leq p \leq 1$.

High fronthaul regime $r \geq r_{th}$**:** Using Lemma 4.1 and Proposition 4.3, upper bound on the achievable NDT in the regime of interest is

$$\bar{\delta}_{\text{react,u}}(\mu, r) \leq \frac{K}{\min(M,K)} + \Big(1 - \frac{\mu}{\alpha}\Big)\frac{K}{Mr} + \frac{\mu}{r}\left(\frac{p}{(1-p/N)(\alpha-1)}\right), \tag{C.20}$$

The derivative of (C.20) with respect to μ is negative if

$$p \leq p_1(\mu, r) = \min\left(\frac{(\alpha-1)K}{\alpha(M+K/N) - K/N}, 1\right), \tag{C.21}$$

while the derivative of (C.20) with respect to μ is positive if $p_1(\mu, r) \leq p \leq 1$. If derivative is positive, $\mu = 0$ minimizes (C.20), hence upper bound on the achievable long-term NDT is

$$\bar{\delta}_{\text{react,adapt,u}}(\mu, r) \leq \bar{\delta}_{\text{react,u}}(0, r) = \bar{\delta}_{\text{C-RAN}}(r) = \frac{K}{\min\{M, K\}} + \frac{K}{Mr}, \tag{C.22}$$

for $0 \leq \mu \leq 1$, $r \geq r_{th}$ and $p_1(\mu, r) \leq p \leq 1$. Instead, if the derivative is negative upper bound on the achievable NDT is obtained by (C.20) which is the extension of (4.19), so we have

$$\bar{\delta}_{\text{react,adapt,u}}(\mu, r) \leq \bar{\delta}_{\text{react,u}}(\mu, r), \tag{C.23}$$

for $0 \leq \mu \leq 1$, $r \geq r_{th}$ and $p \leq p_1(\mu, r)$.

(C.13), (C.14), (C.17), (C.18), (C.19), (C.22), (C.23) and their corresponding conditions complete the proof.

C.4 Proof of Proposition 4.6

To prove Proposition 4.6, we will show that for serial transmission the following inequalities

$$\frac{1 - \frac{Kp}{N}}{2}\delta^*_{\text{off}}(\mu, r) + \frac{Kp}{N}\left(1 + \frac{\mu}{r}\right) \leq \bar{\delta}^*_{\text{on,k}}(\mu, r) \leq \bar{\delta}^*_{\text{on,u}}(\mu, r) \leq 2\delta^*_{\text{off}}(\mu, r) + \frac{4}{r}, \tag{C.24}$$

hold. Comparing the right-most and the left-most expressions will complete the proof. For the pipelined case, we have the following relationships with the minimum NDT for the serial case:

$$\bar{\delta}^{pl^*}_{\text{on,u}}(\mu, r) \leq \bar{\delta}^{*}_{\text{on,u}}(\mu, r) \overset{(a)}{\leq} 2\delta^{*}_{\text{off}}(\mu, r) + \frac{4}{r} \overset{(b)}{\leq} 4\delta^{pl^*}_{\text{off}}(\mu, r) + \frac{4}{r}, \tag{C.25}$$

where (a) is obtained by using the right-most inequality in (C.24), and (b) is obtained from [17, Lemma 4], namely from the inequality $\delta^{*}_{\text{off}}(\mu, r) \leq 2\delta^{pl^*}_{\text{off}}(\mu, r)$; and also

$$\bar{\delta}^{pl^*}_{\text{on,u}}(\mu, r) \geq \bar{\delta}^{pl^*}_{\text{on,k}}(\mu, r) \overset{(a)}{\geq} \frac{1}{2}\bar{\delta}^{*}_{\text{on,k}}(\mu, r) \overset{(b)}{\geq} \frac{1 - \frac{Kp}{N}}{4}\delta^{*}_{\text{off}}(\mu, r) + \frac{Kp}{2N}\Big(1 + \frac{\mu}{r}\Big), \tag{C.26}$$

where (a) is obtained using Lemma 4.2 and (b) is the first inequality in (C.24).

In what follows, we prove (C.24) for the serial transmission.

Lower bound: To prove the lower bound in (C.24), we first present the following lemma, which presents a slight improvement over the lower bound in [17, Proposition 1].

Lemma C.1. *(Lower Bound on Minimum offline NDT). For an $M \times K$ F-RAN with $N \geq K$ files, the minimum NDT is lower bounded as*

$$\delta^{*}_{\text{off}}(\mu, r) \geq \delta_{\text{off,lb}}(\mu, r) \tag{C.27}$$

where $\delta_{\text{off,lb}}(\mu, r)$ *is the minimum value of the following linear program (LP)*

$$\textit{minimize} \quad \delta_E + \delta_F \tag{C.28}$$

$$\textit{subject to :} \; l\delta_E + (M-l)r\delta_F \geq K - \min\big((K-l), (M-l)(K-l)\mu\big) \tag{C.29}$$

$$\delta_F \geq 0, \delta_E \geq 1, \tag{C.30}$$

where (C.29) *is a family of constraints with* $0 \leq l \leq \min\{M, K\}$.

Proof. It follows using the same steps as Proposition C.1 below. □

Next, we introduce the following lower bound on the minimum long-term NDT of online caching.

Proposition C.1. *(Lower bound on the Long-Term NDT of Online Caching for Serial Transmission). For an* $M \times K$ *F-RAN with a fronthaul rate of* $r \geq 0$*, the long-term NDT is lower bounded as* $\bar{\delta}^*_{\text{on,u}}(\mu, r) \geq \bar{\delta}^*_{\text{on,k}}(\mu, r) \geq (1 - Kp/N)\delta_{\text{off,lb}}(\mu, r) + (Kp/N)\delta_{\text{on,lb}}(\mu, r)$*, where* $\delta_{\text{on,lb}}(\mu, r)$ *is the solution of following LP*

$$\textit{minimize} \quad \delta_E + \delta_F \tag{C.31}$$

$$\textit{subject to :} \; l\delta_E + (M-l)r\delta_F \geq K - \min\Big((K-l-1), (M-l)(K-l-1)\mu\Big) \tag{C.32}$$

$$\delta_F \geq 0, \delta_E \geq 1, \tag{C.33}$$

where (C.32) *is a family of constraints with* $0 \leq l \leq K-1$ *and* $\delta_{\text{off,lb}}(\mu, r)$ *is the lower bound on the minimum NDT of offline caching defined in Lemma C.1.*

Proof. See Appendix C.5. □

Now, using Proposition C.1, we have

$$\bar{\delta}^*_{\text{on,u}}(\mu, r) \geq \bar{\delta}^*_{\text{on,k}}(\mu, r) \geq \Big(1 - \frac{Kp}{N}\Big)\delta_{\text{off,lb}}(\mu, r) + \frac{Kp}{N}\delta_{\text{on,lb}}(\mu, r) \tag{C.34}$$

$$\overset{(a)}{\geq} \frac{(1 - \frac{Kp}{N})}{2}\delta^*_{\text{off}}(\mu, r) + \frac{Kp}{N}\delta_{\text{on,lb}}(\mu, r)$$

$$\overset{(b)}{\geq} \frac{(1 - \frac{Kp}{N})}{2}\delta^*_{\text{off}}(\mu, r) + \frac{Kp}{N}\Big(1 + \frac{\min(\mu, 1/M)}{r}\Big), \tag{C.35}$$

where (a) is obtained using (4.13), namely $\delta^*_{\text{off}}(\mu, r)/\delta_{\text{off,lb}}(\mu, r) \leq 2$ and (b) follows by deriving the lower bound $\delta_F \geq \min(\mu, 1/M)/r$ on the optimal solution of the LP (C.31) by setting $l = 0$ in the constraint (C.32) and summing the result with constraint (C.33).

Upper bound: To prove the upper bound in (C.24), we leverage the following lemma.

Lemma C.2. *For any* $\alpha > 1$*, we have the following inequality*

$$\delta_{\text{off,ach}}\Big(\frac{\mu}{\alpha}, r\Big) \leq 2\delta^*_{\text{off}}(\mu, r) + \frac{1}{r} + \frac{1}{\alpha}\Big(1 - \frac{1}{r}\Big). \tag{C.36}$$

Proof. See Appendix C.6. □

Using Proposition 4.3 and Lemma C.2, an upper bound on the long-term average NDT of the proposed reactive caching scheme is obtained as

$$\bar{\delta}_{\text{react}}(\mu, r) \leq 2\delta^*_{\text{off}}(\mu, r) + f(\alpha), \tag{C.37}$$

where

$$f(\alpha) = \frac{1}{r} + \frac{1}{\alpha}\Big(1 - \frac{1}{r}\Big) + \frac{Np(\mu/r)}{(N-p)(\alpha-1)}. \tag{C.38}$$

Since the additive gap (C.38) is a decreasing function of N and an increasing function of p and μ, it can be further upper bounded by setting $N = 2$, $p = 1$ and $\mu = 1$. Finally, by plugging $\alpha = 2$, and using the inequality $\bar{\delta}^*_{\text{on,u}}(\mu, r) \leq \bar{\delta}_{\text{react}}(\mu, r)$ the upper bound is proved.

C.5 Proof of Proposition C.1

To obtain a lower bound on the long-term NDT, we consider a genie-aided system in which, at each time slot t, the ENs are provided with the optimal cache contents of an offline scheme tailored to the current popular set $\mathcal{L}_t$ at no cost in terms of fronthaul latency. In this system, as in the system under study, at each time slot t, with probability of p there is a new file in the set of popular files, and hence the probability that an uncached file is requested by one of the users is Kp/N. As a

result, the NDT in time slot t for the genie-aided system can be lower bounded as

$$\delta_t(\mu, r) \geq (1 - Kp/N)\delta_{\text{off,lb}}(\mu, r) + (Kp/N)\delta_{\text{on,lb}}(\mu, r), \tag{C.39}$$

where $\delta_{\text{off,lb}}(\mu, r)$ is the lower bound on the minimum NDT for offline caching in Lemma C.1, while $\delta_{\text{on,lb}}(\mu, r)$ is a lower bound on the minimum NDT for offline caching in which all files but one can be cached. The lower bound (C.39) follows since, in the genie-aided system, with probability $1 - Kp/N$ the system is equivalent to the offline caching set-up studied in [17], while, with probability of Kp/N, there is one file that cannot be present in the caches.

To obtain the lower bound $\delta_{\text{on,lb}}(\mu, r)$, we note that the set-up is equivalent to that in [17] with the only difference is that one of the requested files by users is no longer partially cached at ENs. Without loss of generality, we assume that file F_K is requested but it is not partially cached. Revising step (67c) in [17], we can write

$$\text{H}(S_{[1:(M-l)]}|W_{[1:l]}, W_{[K+1:N]}) \leq \min\Big((M-l)(K-l-1)\mu, K-l-1\Big)F, \tag{C.40}$$

which is obtained by using the fact that the constrained entropy of the cached content cannot be larger than the overall size of files W_j with $j \in [l+1, K-1]$. Plugging (C.40) into [17, Eq. (66)] and then taking the limit $F \to \infty$ and $P \to \infty$, results in

(C.32). The rest of proof is as in [17, Appendix I]. Using (C.39) in the definition of long-term average NDT (4.6) concludes the proof.

C.6 Proof of Lemma C.2

To prove Lemma C.2, for any given $\alpha > 1$ and $M \geq 2$, we consider separately small cache regime with $\mu \in [0, 1/M]$; intermediate cache regime with $\mu \in [1/M, \alpha/M]$ and the high cache regime with $\mu \in [\alpha/M, 1]$.

Small-cache Regime ($\mu \in [0, 1/M]$): Using Lemma C.1 a lower bound on the minimum NDT can be obtained as

$$\delta^*_{\text{off}}(\mu, r) \geq 1 + \frac{K(1-\mu M)}{Mr} \tag{C.41}$$

by considering the constraint (C.29) with $l = 0$ and constraint (C.30). Using the offline caching and delivery policy in Sec. 4.3 shown in the top left of Fig. 4.2, the NDT in the regime of interest is

$$\begin{aligned}
\delta_{\text{off,ach}}\Big(\frac{\mu}{\alpha}, r\Big) &= \Big(\frac{\mu M}{\alpha}\Big)\delta_{\text{E,Coor}} + \Big(1 - \frac{\mu M}{\alpha}\Big)\big[\delta_{\text{E,C-RAN}} + \delta_{\text{F,C-RAN}}\big] \\
&\overset{(a)}{\leq} \frac{(M+K-1)\mu}{\alpha} + \Big(1 + \frac{K}{Mr}\Big) \times \Big(1 - \frac{\mu M}{\alpha}\Big) \\
&= 1 + \frac{(K-1)\mu}{\alpha} + \frac{K}{Mr}\Big(1 - \frac{\mu M}{\alpha}\Big),
\end{aligned} \tag{C.42}$$

where (a) is obtained using (4.8) and (4.9). From (C.41) and (C.42), we have

$$\begin{aligned}\delta_{\text{off,ach}}\Big(\frac{\mu}{\alpha},r\Big)-2\delta^*_{\text{off}}(\mu,r) &\overset{(a)}{\leq} \frac{K\mu}{r}\Big(2-\frac{1}{\alpha}\Big)+\frac{(K-1)\mu}{\alpha}-\frac{K}{Mr}\\ &\overset{(b)}{\leq} \frac{K}{Mr}\Big(1-\frac{1}{\alpha}\Big)+\frac{(K-1)}{\alpha M}\\ &\overset{(c)}{\leq} \frac{1}{r}+\frac{1}{\alpha}\Big(1-\frac{1}{r}\Big),\end{aligned} \tag{C.43}$$

where (a) is obtained by omitting the first negative term; (b) is obtained by using the fact that $\mu \leq 1/M$ and (c) follows from $M \geq K$.

Intermediate cache Regime ($\mu \in [1/M, \alpha/M]$): Using Lemma C.1 a lower bound on the minimum NDT can be obtained as $\delta^*_{\text{off}}(\mu, r) \geq 1$ by considering the constraint (C.30). Using this lower bound and the offline caching and delivery policy in Sec. 4.3 shown in the bottom of Fig. 4.2, we have

$$\delta_{\text{off,ach}}\Big(\frac{\mu}{\alpha},r\Big)-2\delta^*_{\text{off}}(\mu,r) \leq \Big(\frac{\mu}{\alpha}\Big)\delta_{\text{E,Coop}}-2+\Big(1-\frac{\mu}{\alpha}\Big)[\delta_{\text{E,C-RAN}}+\delta_{\text{F,C-RAN}}] \overset{(a)}{\leq} \frac{1}{r} \tag{C.44}$$

where (a) is obtained using (4.7) and (4.9) and also the fact that $M \geq K$.

Large-cache regime ($\mu \in [\alpha/M, 1]$): In this regime, we have

$$\begin{aligned}\frac{\delta_{\text{off,ach}}\big(\mu/\alpha,r\big)}{\delta^*_{\text{off}}(\mu,r)} &\overset{(a)}{\leq} \delta_{\text{off,ach}}\big(\mu/\alpha,r\big) \overset{(b)}{=} \Big(\frac{\mu M/\alpha-1}{M-1}\Big)\delta_{\text{E,Coop}}+\Big(\frac{M(1-\mu/\alpha)}{M-1}\Big)\delta_{\text{E,Coor}}\\ &\overset{(c)}{\leq} \delta_{\text{E,Coor}} \overset{(d)}{\leq} \frac{M+K-1}{M} \overset{(e)}{\leq} 2\end{aligned} \tag{C.45}$$

where (a) is obtained using the fact that the lower bound $\delta^*_{\text{off}}(\mu, r) \geq 1$; (b) is obtained using the offline caching and delivery policy in Sec. 4.3 shown in the top right of Fig. 4.2; (c) is obtained using the fact that NDT is a decreasing function of μ and it is maximized by setting $\mu = \alpha/M$; (d) is obtained using (4.8) and (e) is obtained using $M \geq K$. Finally, using (C.43)-(C.45) concludes the proof.

BIBLIOGRAPHY

[1] https://www.cisco.com/c/en/us/solutions/collateral/service-provider/visual-networking-index-vni/complete-white-paper-c11-481360.pdf (accessed on February 7, 2017)

[2] S. M. Azimi, O. Simeone, R. Tandon, "Fundamental limits on latency in small-cell caching systems: An information-theoretic analysis," in *Proc. IEEE Global Communication Conference (GLOBECOM)*, pp. 1-6, Washington D.C., USA, Dec. 2016.

[3] R. Pedarsani, M. A. Maddah-Ali, U. Niesen, "Online coded caching," *IEEE/ACM Trans. Netw.*, vol. 24, no. 2, pp. 836-845, Feb. 2016.

[4] S. M. Azimi, O. Simeone, R. Tandon, "Content delivery in fog-Aided small-cell systems with offline and online caching: An information-theoretic analysis," *Entropy*, vol. 19, no. 7, pp. 1-23, Jul. 2017.

[5] S. M. Azimi, O. Simeone, A. Sengupta, R. Tandon, "Online edge caching in fog-aided wireless network," in *Proc. IEEE International Symposium on Information Theory* (ISIT), pp. 1217-1221, Aachen, Germany, Jun. 2017.

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

[7] E. Bastug, M. Bennis, M. Debbah, "Living on the edge: The role of proactive caching in 5G wireless networks," *IEEE Commun. Magazine,* vol. 52, no. 8, pp. 82-89, Aug. 2014.

[8] M. A. Maddah-Ali, U. Niesen, "Cache aided interference channels" in *Proc. IEEE International Symposium on Information Theory* (ISIT), pp. 7-12, Hong Kong, China, July. 2015.

[9] M. A. Maddah-Ali, U. Niesen, "Cache aided interference channels," http://arxiv.org/abs/1510.06121, 2015.

[10] A. Sengupta, R. Tandon, O. Simeone, "Cache-aided wireless networks: Tradeoffs between storage and latency," in *Proc. of the Annual Conference on Information Science and Systems* (CISS), pp. 320-325, Princeton, NJ, USA, Mar. 2016.

[11] N. Naderializadeh, M. A. Maddah-Ali, A. S. Avestimehr, "Fundamental limits of cache-aided interference management," *IEEE Trans. Info. Theory*, vol. 63, no. 5, 3092-3107, May 2017.

[12] F. Xu, K. Liu, M. Tao, "Cooperative Tx/Rx caching in interference channels: A storage-latency tradeoff study," in *Proc. IEEE International Symposium on Information Theory* (ISIT), pp. 2034-2038, Barcelona, Spain, Jul. 2016.

[13] J. S. P. Roig, D. Gunduz, F. Tosato, "Interference networks with caches at both ends," in *Proc. IEEE Int. Conference on Communications* (ICC), pp. 1-6, Paris, France, May 2017.

[14] N. Naderializadeh, M. A. Maddah-Ali, A. S. Avestimehr, "On the optimality of separation between caching and delivery in general cache networks," in *Proc. IEEE International Symposium on Information Theory* (ISIT), pp. 1232-1236, Aachen, Germany, June. 2017.

[15] Y. Cao, M. Tao, F. Xu, K. Liu, "Fundamental storage-latency tradeoff in cache-aided MIMO interference networks," *IEEE Trans Wireless Commun.*, vol. 16, no. 8, pp. 5061-5076, Aug. 2017.

[16] R. Tandon, O. Simeone,"Cloud aided wireless networks with edge caching: Fundamental latency trade offs in fog radio access networks," in *Proc. IEEE International Symposium on Information Theory* (ISIT), pp. 2029-2033, Barcelona, Spain, Jul. 2016.

[17] A. Sengupta, R. Tandon, O. Simeone,"Fog-aided wireless networks for content delivery: Fundamental latency tradeoffs," *IEEE Trans. Info. Theory*, vol. 63, no. 10, 6650-6678, Oct. 2017.

[18] J. Koh, O. Simeone, R. Tandon, J. Kang"Cloud-aided edge caching with wireless multicast fronthauling in fog radio access networks," in *Proc. IEEE Wireless Communications and Networking Conference* (WCNC), pp. 1-6, San Francisco, CA, USA, Mar. 2017.

[19] A. M. Girgis, O. Ercetin, M. Nafie, T. ElBatt, "Decentralized coded caching in wireless networks: Trade-off between storage and latency," in *Proc. IEEE International Symposium on Information Theory* (ISIT), pp. 1-5, Aachen, Germany, Jun. 2017.

[20] J. Goseling, O. Simeone, P. Popovski, "Delivery latency trade-offs of heterogeneous contents in fog radio access networks," in *Proc. IEEE Global Communication Conference (GLOBECOM)*, pp. 1-6, Singapore, Dec. 2017.

[21] J. Kakar, S. Gherekhloo, Z. H. Awan, A. Sezgin, "Fundamental limits on latency in cloud- and cache-aided HetNets," in *Proc. IEEE Int. Conference on Communications* (ICC), pp. 1-6, Paris, France, May 2017.

[22] X. Peng, J.C. Shen, J. Zhang, J. Kang, K.B. Letaief, "Joint data assignment and beamforming for backhaul limited caching networks," in *Proc. IEEE International Symposium on Personal, Indoor and Mobile Radio Communications*(PIMRC), pp. 1370-1374, Washington, D.C., Sep. 2014.

[23] S. H. Park, O. Simeone, S. Shamai, "Joint optimization of cloud and edge processing for fog radio access networks," *IEEE Trans Wireless Commun.*, vol. 15, no. 11, pp. 7621-7632, Nov. 2016.

[24] M. Tao, E. Chen, H. Zhou, W. Yu, "Content-centric sparse multicast beamforming for cache-enabled cloud RAN," *IEEE Trans Wireless Commun.*, vol. 15, no. 9, pp. 6118-6131, Sep. 2017.

[25] B. Azari, O. Simeone, U. Spagnolini, A. M. Tulino, "Hypergraph-based analysis of clustered cooperative beamforming with application to edge caching," *IEEE Wireless Comm. Lett.*, vol. 5, no. 1, pp. 84-87, Feb. 2016.

[26] S. H. Park, O. Simeone, S. Shamai, "Joint cloud and edge processing for latency minimization in fog radio access networks," in *Proc. IEEE International Workshop on Signal Processing Advances in Wireless Communications*(SPAWC), pp. 1-5, Edinburgh, UK, Jul. 2016.

[27] Y. Zhu, D. Guo, O. Simeone,"Ergodic fading Z-interference channels without state information at transmitters.," *IEEE Trans. Info. Theory*, vol. 57, no. 5, 2627-2647, May 2011.

[28] A. Vahid, M. A. Maddah-Ali, A.S. Avestimehr,"Capacity results for binary fading interference channels with delayed CSIT," *IEEE Trans. Info. Theory*, vol. 60, no. 10, 6093-6130, May 2014.

[29] A.S. Avestimehr, S.N. Diggavi, D.N.C. Tse,"Wireless network information flow: A deterministic approach," *IEEE Trans. Info. Theory*, vol. 57, no. 4, 1872-1905, Apr. 2011.

[30] D.N.C. Tse, R.D. Yates, "Fading broadcast channels with state information at the receivers," *IEEE Trans. Info. Theory*, vol. 58, no. 6, 3453-3471, Jun. 2012.

[31] V. Martina, M. Garetto, E. Leonardi, "A unified approach to the performance analysis of caching systems," in *Proc. IEEE Conference on Computer Communications*(INFOCOM), pp. 2040-2048, Toronto, Canada, Apr. 2014.

[32] V. W. S. Wong, R. Schober, D. W. K. Ng, *Key Technologies for 5G Wireless Systems*, Cambridge, United Kingdom, Cambridge University Press; 1st edition, 2017.

[33] M. A. Maddah-Ali, U. Niesen, "Fundamental limits of caching" *IEEE Trans. Info. Theory*, vol. 60, no. 5, 2856-2867, May 2014.

[34] R. Tandon and O. Simeone, "Harnessing cloud and edge synergies: Toward an information theory of fog radio access networks," *IEEE Commun. Magazine,* vol. 54, no. 8, pp. 44-50, Aug. 2016.

[35] J. Llorca, A. M. Tulino, K. Guan, J. Esteban, M. Varvello, N. Choi, and D. Kilper, "Dynamic in-network caching for energy efficient content delivery," in *Proc. IEEE Conference on Computer Communications*(INFOCOM), pp. 245-249, Turin, Italy, Apr. 2013.

[36] O. Simeone, A. Maeder, M. Peng, O. Sahin, W. Yu, "Cloud radio access network: Virtualizing wireless access for dense heterogeneous systems," *Journal of Communications and Networks,* vol. 18, no. 2, pp. 135-149, Apr. 2016.

[37] T. Q. S. Quek, M. Peng, O. Simeone, W. Yu, *Cloud Radio Access Networks Principles, Technologies, and Applications*, Cambridge, United Kingdom, Cambridge University Press; 1st edition, 2017.

[38] https://5g-ppp.eu/wp-content/uploads/2014/02/5G-PPP-5G-Architecture-WP-July-2016.pdf (accessed on July 1, 2016)

[39] J. G. Herrera, J. F. Botero, "Resource allocation in NFV: A comprehensive survey" *IEEE Trans. Netw. and Serv. Management*, vol. 13, no. 3, 518-532, Sep. 2016.

[40] N. Naderializadeh, M. A. Maddah-Ali, A. S. Avestimehr, "Cache-aided interference management in wireless cellular networks," in *Proc. IEEE Int. Conference on Communications* (ICC), pp. 1-6, Paris, France, May 2017.

[41] J. Hachem, U. Niesen, S. Diggavi, "A layered caching architecture for the interference channel," in *Proc. IEEE International Symposium on Information Theory* (ISIT), pp. 415-419, Barcelona, Spain, Jul. 2016.

[42] A. Sengupta, R. Tandon, O. Simeone, "Pipelined fronthaul-edge content delivery in fog radio access networks," in *Proc. IEEE Global Communication Conference (GLOBECOM)*, pp. 1-6, Washington D.C., USA, Dec. 2016.

[43] M. A. Maddah-Ali, A. S. Motahari, A. K. Khandani,"Communication over MIMO X channels: Interference alignment, decomposition, and performance analysis" *IEEE Trans. Info. Theory*, vol. 54, no. 8, pp. 3457-3470, Aug. 2008.

[44] W. Chang, R. Tandon, O. Simeone, "Cache-aided content delivery in fog-RAN systems with topological information and no CSI," in *Proc. Asilomar Conference on Signals, Systems and Computers*, pp. 1-5, Monterey, CA, Nov. 2017.

[45] M. Chiang, S. Ha, I. Chih-Lin, F. Risso, and T. Zhang, "Clarifying fog computing and networking: 10 questions and answers," *IEEE Commun. Magazine,* vol. 55, no. 4, pp. 1820, Apr. 2017.