Online Edge Caching in Fog-Aided Wireless Networks

Seyyed Mohammadreza Azimi, Osvaldo Simeone, Avik Sengupta and Ravi Tandon

Abstract

In a Fog Radio Access Network (F-RAN) architecture, edge nodes (ENs), such as base stations, are equipped with limited-capacity caches, as well as with fronthaul links that can support given transmission rates from a cloud processor. Existing information-theoretic analyses of content delivery in F-RANs have focused on offline caching with separate content placement and delivery phases. In contrast, this work considers an online caching set-up, in which the set of popular files is time-varying and both cache replenishment and content delivery can take place in each time slot. 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 caching and delivery schemes based on reactive and proactive caching are investigated, and their performance is compared to optimal offline caching schemes both analytically and via numerical results.

Index Terms

Edge caching, Online caching, C-RAN.

I. Introduction

To cope with the growing demand for content by mobile users, *edge caching* stores popular content at the edge nodes (ENs) of a wireless system, such as base stations, thereby reducing latency and backhaul usage [1]. In contrast, *Cloud Radio Access Network* (C-RAN) delivers

S. M. Azimi and O. Simeone are with the CWiP, Department of Electrical and Computer Engineering, New Jersey Institute of Technology, Newark, NJ, USA. E-mail: (sa677, osvaldo.simeone@njit.edu). Their work was partially funded by U.S. NSF through grant CCF-1525629. A. Sengupta is with Department of Electrical and Computer Engineering, Virginia Tech, Blacksburg, VA 24060 USA. Email: (aviksg@vt.edu). R. Tandon is with University of Arizona, Tucson, AZ, USA. E-mail: (tandonr@email.arizona.edu).

content by leveraging processing at a central “cloud” processing unit, which has access to the content library and can communicate to the ENs via dedicated finite-capacity fronthaul links [2].

Thanks to Network Function Virtualization (NFV) [3], 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 reviewed above. To study the optimal operation of networks that allow for both edge and cloud processing, references [4]–[6] investigate a *Fog Radio Access Network (F-RAN)* architecture, in which ENs are equipped with limited-capacity caches and with fronthaul links that can support given rates (see Fig. 1). The key design question for F-RANs is: What is the optimal way to use the available physical resources for communication, on the wireless channel and fronthaul, and for storage, at the ENs, so as to maximize the performance of content delivery?

Related Works: Partial answers to this key question were provided from an information-theoretic viewpoint in [4]–[11]. In particular, references [6]–[9] considered a scenario with edge caching only, i.e., with zero-capacity fronthaul links. They developed upper and lower bounds on the achievable number of degrees of freedom (DoF), or more precisely on its inverse, which can be thought of as a measure of coding latency (i.e., transmission time) in the high signal-to-noise ratio (SNR) regime. In contrast, references [4, 5, 11] investigated the full F-RAN scenario with both edge caching and cloud processing, and derived 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 [6]–[9].

As summarized in Fig. 1-(a), a key assumption made in all prior works reviewed above is that caching takes place *offline*. More specifically, 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.

Main Contributions: In this work, we consider an alternative *online* caching set-up, typically considered in the networking literature [12], in which the set of popular files is time-varying and both cache replenishment and content delivery take place at each time slot. The main contributions of this article are as follows:

• The performance metric of the long-term NDT, which captures the temporal dependence of the coding latencies accrued in different slots, is introduced (Sec. II);

• Online caching and delivery schemes based on both reactive and proactive caching principles (see, e.g., [13]) are proposed, and bounds on the corresponding achievable long-term NDTs are derived (Sec. IV);

• A lower bound on the achievable long-term NDT is obtained. Using this bound, the performance loss caused by the variations in the set of popular files in terms of delivery latency is quantified by comparing the NDTs achievable under offline and online caching (Sec. V);

• Numerical results are provided in which the performance of reactive and proactive online caching schemes are compared with offline caching. Also, the performance of different eviction mechanisms such as random, Least Recently Used (LRU) and First In First Out (FIFO) (see, e.g., [12]) are evaluated (Sec. VI).

II. Problem Definition

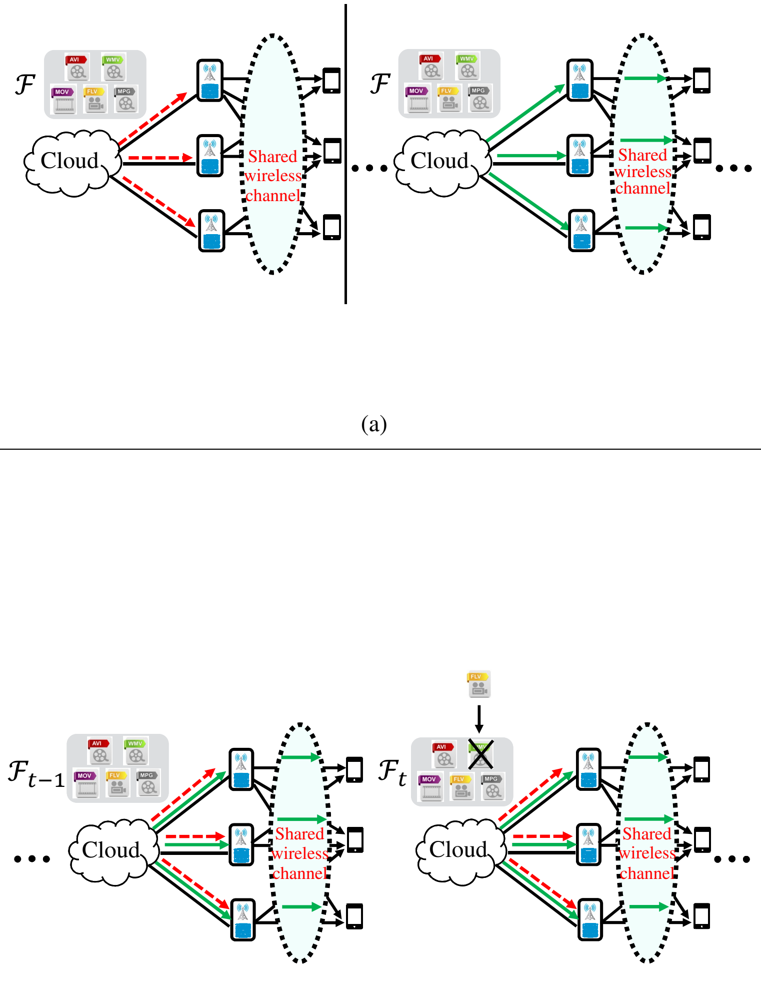
In this section, we first present the system model adopted for the analysis of F-RAN systems with online caching. Then, we introduce the long-term NDT as the performance measure of interest.

A. System Model

We consider the $M \times K$ F-RAN with online caching shown in Fig. 2, in which M ENs serve a total of $K \leq M$ users through a shared downlink wireless channel[1]. Each EN is connected to the cloud via a dedicated fronthaul link with capacity C_F bits per symbol, where a *symbol* henceforth refers to the duration of a channel use of the wireless channel. The ENs can cache content from a time-varying set of N popular files denoted as $\mathcal{F}_t$, with $t = 1, 2, ..$ indexing the time slots. All files are assumed to have the same size of L bits. The cache capacity of each EN is μNL bits, where μ, with $0 \leq \mu \leq 1$, is the *fractional cache capacity*. At each time slot t, each user $k \in [1 : K]$ requests a file $F_{d_{k,t}} \in \mathcal{F}_t$. As in [14], the requested file indices $d_t = (d_{1,t}, ..., d_{K,t})$ are chosen randomly without replacement from $[1 : N]$ following an arbitrary order.

The set of popular files $\mathcal{F}_t$ evolves according to the Markov model considered in [14]. Accordingly, given the popular set $\mathcal{F}_{t-1}$ at time $t-1$, with probability $1-p$ no new popular content is generated and we have $\mathcal{F}_t = \mathcal{F}_{t-1}$; while, with probability p, a new popular file is added to the set $\mathcal{F}_t$ by replacing a file selected uniformly at random from $\mathcal{F}_{t-1}$. We consider two cases, namely: (i) *known popular set*: the cloud is informed about the set $\mathcal{F}_t$ at time t, e.g.,

[1] The case $K > M$ will be addressed in future work.



(a)

(b)

Figure 1: (a) Offline edge caching comprises separate content placement (dashed arrows) and delivery phase (solid arrows), where the latter includes multiple transmission slots; (b) With online edge caching, online cache refreshment at the ENs and content delivery to the users can take place at each slot.

by leveraging data analytics tools; (ii) *unknown popular set*: the set $\mathcal{F}_t$ may only be inferred via the observation of the users' requests. We note that the latter assumption is typically made in the networking literature [12].

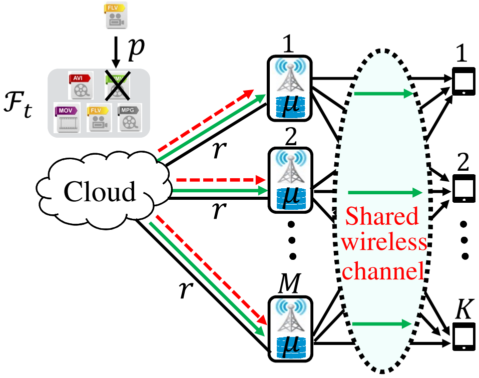


Figure 2: F-RAN with online edge caching and time-varying content library $\mathcal{F}_t$: cache refreshment (dashed arrow) at the ENs and content delivery (solid arrow) to the users can take place at each slot.

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

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

where $H_{k,m,t}$ is the channel gain between mth EN and kth user at time slot t; $X_{m,t}$ is the transmitted signal 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 $H_t = \{\{H_{k,m,t}\}_{k=1}^{K}\}_{m=1}^{M}$.

The system operates according to a fronthaul, caching, edge transmission and decoding policy $\Pi = (\Pi_C, \Pi_F, \Pi_E, \Pi_D)$, which is characterized by the following functions.

- *Fronthaul policy* Π_F: Two different cases are distinguished. In both cases, the fronthaul capacity limitations impose the condition $H(U_{m,t}) \leq T_{F,t} C_F$, where $T_{F,t}$ is the duration (in symbols) of the fronthaul transmission $U_{m,t}$ in time slot t for all ENs $m = 1, ..., M$.
  - Known popular set: The fronthaul policy with known popular set is defined by a function $\Pi_F : \{\mathcal{F}_t, d_t, H_t\} \rightarrow \{U_{1,t}, ..., U_{M,t}\}$, which maps the set $\mathcal{F}_t$, the instantaneous demand vector d_t and channel state information H_t to the fronthaul transmissions $(U_{1,t}, ..., U_{M,t})$ at time slot t.

  - Unknown popular set: The fronthaul policy with unknown popular set is defined by a function $\Pi_F : \{\{d_{t'}\}_{t' \leq t}, H_t\} \rightarrow \{U_{1,t}, ..., U_{M,t}\}$, which maps the set of demand vectors $d_{t'}$ up to time t and channel state information H_t to the fronthaul messages.

- *Caching policy* Π_C: The caching policy is defined by functions $\Pi_C = (\Pi_{C_1}, ..., \Pi_{C_M})$, where the function $\Pi_{C_m} : \{S_{m,t-1}, d_{t-1}, U_{m,t-1}\} \rightarrow S_t$ maps the cached content $S_{m,t-1}$ of the mth EN at the beginning of time slot $t-1$, demand vector d_{t-1} at time slot $t-1$ and the fronthaul message received in time slot $t-1$, to the cache content $S_{m,t}$ at the beginning of time slot t. Due to cache capacity constraints, we have the inequality $H(S_{m,t}) \leq \mu NL$, for all slots t and ENs m. More specifically, as in [5], we allow only for intra-file coding, which implies that the cache content $S_{m,t}$ can be partitioned into independent subcontents $S^f_{m,t}$, each obtained as a function of a single file $f \in \mathcal{F}_t$, with the condition $H(S^f_{m,t}) \leq \mu L$. We also assume that, at time $t = 1$, all the caches are empty.
- *Edge transmission policy* Π_E: The edge transmission policy is defined by the set of functions $\Pi_E = (\Pi_{E_1}, ..., \Pi_{E_M})$, where function $\Pi_{E_m} : \{d_t, H_t, U_{m,t}, S_{m,t}\} \rightarrow X_{m,t}$ defines the codeword $X_{m,t}$, of duration $T_{E,t}$ symbols, that is sent on the wireless channel by the mth EN 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* Π_D: Each user k maps its received signal $Y_{k,t}$ in (1) over a number $T_{E,t}$ of channel uses to an estimate $\hat{F}_{d_{t,k}}$ of the demanded file $F_{d_{t,k}}$.

The probability of error of a policy $\Pi = (\Pi_C, \Pi_F, \Pi_E, \Pi_D)$ at slot t is defined as

$$\mathrm{P}_{e,t} = \max_{k \in \{1,...,K\}} \Pr(\hat{F}_{d_{t,k}} \neq F_{d_{t,k}}), \tag{2}$$

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

B. Long-term Normalized Delivery Time (NDT)

For given parameters (M, K, N, μ, C_F, P), the average achievable delivery time per bit in slot t for a given sequence of feasible policies is defined as

$$\Delta_t(\mu, C_F, P) = \lim_{L \rightarrow \infty} \frac{1}{L} \mathrm{E}(T_{F,t} + T_{E,t}), \tag{3}$$

where the average is taken with respect to the distributions of $\mathcal{F}_t$, d_t and H_t, and we have made explicit only the dependence on the system resource parameters (μ, C_F, P) in order to simplify the notation. As in [4, 5], 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. For any achievable sequence $\Delta_t(\mu, C_F, P)$ for $t = 1, 2, ...$, the Normalized Delivery Time (NDT) of time slot t is defined as [4, 5]

$$\delta_t(\mu, r) = \lim_{P\to\infty} \frac{\Delta_t(\mu, r\log P, P)}{1/\log(P)}. \tag{4}$$

In (4), the delivery time per bit in (3) is normalized by the term $1/\log(P)$, which measures the delivery time per bit, at high SNR, of an ideal baseline system with no interference and unlimited caching [4, 5]. The *long-term NDT* for online caching is introduced here and defined as

$$\bar{\delta}_{\text{on}}(\mu, r) = \limsup_{T\to\infty} \frac{1}{T}\sum_{t=1}^{T} \delta_t(\mu, r). \tag{5}$$

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 [4, 5]. By construction, we have the inequalities $\delta^*_{\text{off}}(\mu, r) \leq \bar{\delta}^*_{\text{on,k}}(\mu, r) \leq \bar{\delta}^*_{\text{on,u}}(\mu, r)$.

III. Preliminaries

In this section, we first summarize for reference some key results on offline caching from [5]. With offline caching, the set of popular files $\mathcal{F}_t = \mathcal{F}$ is time invariant and caching takes place in a separate placement phase as seen in Fig. 1-(a). The following caching and delivery policies were found to be approximately optimal in [5], as summarized in Lemma 1 below.

Offline caching policy: With the definition

$$r_0 = \frac{K(M-1)}{M(K-1)}, \tag{6}$$

the offline caching strategy operates in the placement phase as follows:

- *Low fronthaul and small cache regime*: If $r \leq r_0$ and $\mu \leq 1/M$, different non-overlapping μ-fractions of each file are placed at different ENs as illustrated in Fig. 3-(a);

- *Low fronthaul and high cache regime*: If $r \leq r_0$ and $\mu \geq 1/M$, a fraction $(\mu M - 1)/(M - 1)$ of each file is placed at all ENs (shared part), while different non-overlapping $(1-\mu)/(M-1)$-fractions of the file are placed at different ENs, as seen in Fig. 3-(b);
- *High fronthaul*: If $r \geq r_0$, a common μ-fraction of each file is placed at all ENs, as illustrated in Fig. 3-(c).

We note that, unlike the low fronthaul regime, for the high fronthaul regime, the strategy caches the same fractions at all ENs so as to enable cooperative EN transmission, given the reduced overhead of fronthaul transmission, as detailed next.

Offline delivery policy: In the delivery phase, the policy operates as follows.

- *Low fronthaul and small cache regime*: If $r \leq r_0$ and $\mu \leq 1/M$, the μM-fractions of the requested files that are cached at ENs (see Fig. 3-(a)) are delivered using interference alignment for the resulting X-channel [15], since each EN has a sub-fraction $1/M$ of each such μM-fraction of each file. Instead, the remaining $(1 - \mu M)$-fractions of the requested files are delivered using cloud-aided zero-forcing (ZF) precoding. Specifically, precoding of these subfiles is performed at the cloud, which quantizes the precoded signals for transmission to the ENs. This approach is referred to as *soft-transfer* fronthaul transmission in [16].
- *Low fronthaul and high cache regime*: If $r \leq r_0$ and $\mu \geq 1/M$, the $M(1-\mu)/(M-1)$-fractions of the requested files that are cached at different ENs are delivered using interference alignment on the resulting X-channel as discussed above. Instead, the remaining $(\mu M - 1)/(M - 1)$-fractions of the requested files that are cached at all the ENs are delivered by employing cooperative ZF beamforming at the ENs.
- *High fronthaul*: If $r \geq r_0$, the μ-fractions of the requested files that are cached at all the ENs, are delivered using cooperative zero-forcing beamforming at the ENs, while the remaining $(1 - \mu)$-fractions of the requested files are delivered using cloud-aided soft-transfer fronthaul transmission.

The outlined caching and delivery policies achieves the following upper bound on the minimum offline NDT for offline caching.

Lemma 1. *(Achievable Offline NDT [5, Propositions 4 and 7]). For an $M \times K$ F-RAN with $N \geq M \geq K \geq 2$, the offline caching and delivery policy described above is order-optimal with* *respect to the minimum offline NDT* $\delta^*_{\text{off}}(\mu, r)$ *in the sense that the inequality*

$$\frac{\delta_{\text{off,ach}}(\mu, r)}{\delta^*_{\text{off}}(\mu, r)} \leq 2 \tag{7}$$

holds, where the achievable offline NDT $\delta_{\text{off,ach}}(\mu, r)$ *is given as*

$$\delta_{\text{off,ach}}(\mu, r) \triangleq \min\left\{(M+K-1)\mu + \left[1 + \frac{K}{Mr}\right](1-\mu M), 1 + \frac{K(1-\mu)}{Mr}\right\} \tag{8}$$

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

$$\delta_{\text{off,ach}}(\mu, r) \triangleq \min\left\{\frac{\mu M - 1}{M-1} + \left[\frac{M+K-1}{M-1}\right](1-\mu), 1 + \frac{K(1-\mu)}{Mr}\right\} \tag{9}$$

for $\mu \in [1/M, 1]$.

We finally recall, and slightly improve, the lower bound on the NDT of offline caching obtained in [5].

Lemma 2. *(Lower Bound on Minimum offline NDT). For an F-RAN with* M *ENs, each with a fractional cache size* $\mu \in [0, 1]$*,* K *users, a library of* $N \geq K$ *files and a fronthaul capacity of* $C_F = r\log(P)$ *bits per symbol, the minimum NDT is lower bounded as*

$$\delta^*_{\text{off}}(\mu, r) \geq \delta_{\text{off,lb}}(\mu, r) \tag{10}$$

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

$$\text{minimize} \quad \delta_E + \delta_F \tag{11}$$

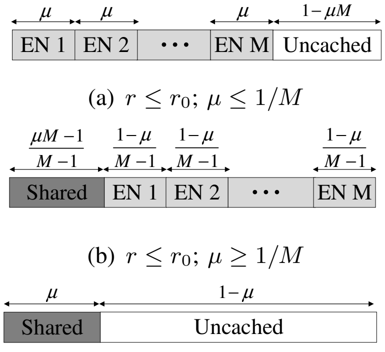
$$\text{subject to : } l\delta_E + (M-l)r\delta_F \geq K - \min\big((K-l), (M-l)(K-l)\mu\big) \tag{12}$$

$$\delta_F \geq 0, \delta_E \geq 1, \tag{13}$$

where (12) *is a family of constraints with* $0 \leq l \leq K$.

Proof. See Appendix A. □

The lower bound (10) is generally larger than that in [5], since in [5, Proposition 1] the right-hand side of (12) is given as $K - (M-l)(K-l)\mu$. This tightening of (12) will be instrumental in proving Proposition 3 below.



(a) $r \leq r_0$; $\mu \leq 1/M$

(b) $r \leq r_0$; $\mu \geq 1/M$

(c) $r > r_0$

Figure 3: Illustration of the offline caching policy proposed in [5] for each cached file. Note that each EN caches a fraction μ of each file.

IV. Achievable Long-Term NDT

In this section, we propose three online caching-fronthaul-edge transmission policies and evaluate their performance. Lower bounds on the minimum long-term NDT will be considered in the next section.

A. Greedy Delivery

For reference, we first consider a greedy strategy that, at each time slot t, aims at minimizing the current NDT $\delta_t(\mu, r)$ in the given slot t without accounting for the NDTs that will be accrued in future slots. The strategy is applicable for the case of unknown popular set and hence also for the known popular set case. As shown in [5, Sec. IV-B] (see also Sec. III), in order to minimize the NDT $\delta_t(\mu, r)$, it is preferable to transmit the non-cached fraction of the requested files after precoding and quantization by using the fronthaul links in the soft-transfer mode (see Sec. III), while it is strictly suboptimal to transmit hard copies of the non-cached contents to the ENs. Under the soft-transfer policy, however, the cache contents cannot be updated, resulting, in the long run, in a zero intersection between the set of cached files and the set $\mathcal{F}_t$ of popular files. Therefore, for sufficiently large t, the system becomes equivalent to the one with no caching, and the corresponding long-term NDT coincides with the offline NDT (8) when $\mu = 0$, namely

$$\bar{\delta}_{\text{greedy}}(\mu, r) = 1 + \frac{K}{Mr}. \tag{14}$$

We recall that the first term represents the normalized (in the sense of (4)) duration $T_{E,t}$ of edge transmission, while the second term accounts for the time $T_{F,t}$ needed for the fronthaul transfer of the quantized precoded signals.

B. Proactive Online Caching

Under the assumption of a known popular set $\mathcal{F}_t$, the cloud is able to observe the changes in the popular set and to proactively cache the new content at ENs as soon as it is generated. In caching the new content, the file that has become outdated, i.e., that is no longer in the set $\mathcal{F}_t$, is evicted from the caches. Specifically, we propose to perform caching of the μ-fraction of the new file at each EN by following the offline caching policy described in Sec. III and summarized in Fig. 3. Delivery can then be performed by following the offline delivery policy recalled in Sec. III. The following proposition presents the achievable long-term NDT of proactive online caching.

Proposition 1. *For an $M \times K$ F-RAN with $N \geq M \geq K \geq 2$, the online proactive caching scheme achieves the long-term NDT*

$$\bar{\delta}_{\text{proact}}(\mu, r) = \delta_{\text{off,ach}}(\mu, r) + \frac{p\mu}{r}, \tag{15}$$

where $\delta_{\text{off,ach}}(\mu, r)$ is given in (8)-(9). *We hence have the upper bound $\bar{\delta}^*_{\text{on,k}}(\mu, r) \leq \bar{\delta}_{\text{proact}}(\mu, r)$.*

Proof sketch: With probability of p there is a new file in the set $\mathcal{F}_t$ and a μ-fraction of the new file is proactively cached at all ENs. The NDT required for fronthaul transmission of the new file is by the definition (4) given by $(\mu L/(r \log P)) \times \log P = \mu/r$, and hence the NDT at a given time slot t is $\delta_t(\mu, r) = p(\delta_{\text{off,ach}}(\mu, r) + \mu/r) + (1-p)\delta_{\text{off,ach}}(\mu, r)$, since delivery requires the NDT $\delta_{\text{off,ach}}(\mu, r)$ achieved by the offline scheme described in Sec. III. Therefore, the long-term NDT of proactive caching is obtained as (15).

C. Reactive Online Caching

In contrast to the discussed greedy and proactive solutions, we now propose a novel scheme that reactively updates the ENs' caches every time a new file is requested by any user. This scheme does not require knowledge of the popular set $\mathcal{F}_t$ and hence operates also for the case of unknown popular set. The proposed reactive strategy delivers a portion of the requested and uncached files to all ENs, which then cache these fractions by evicting from the caches a randomly selected file.

To elaborate, in a manner similar to [14], each EN stores a μ/α-fraction of the same $N^{'} = \alpha N$ files for some $\alpha > 1$. Note that 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 $\mathcal{F}_t$ of N popular files. Caching $N^{'} > N$ files is instrumental in keeping the intersection between the set of cached files and $\mathcal{F}_t$ from vanishing [14].

If Y_t requested files, with $0 \leq Y_t \leq K$, are not cached at the ENs, a μ/α-fraction of each requested and uncached file is sent on the fronthaul link to each EN following the offline caching policy in Fig. 3 with μ/α in lieu of μ. Delivery then takes place by following the achievable offline delivery strategy reviewed in Sec. III, 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. The latter equals $(\mu/\alpha)/(r \log P) \times \log P = \mu/(\alpha r)$, and hence the overall achievable NDT at each time slot t is

$$\delta_t(\mu, r) = \delta_{\text{off,ach}}\Big(\frac{\mu}{\alpha}, r\Big) + \frac{\mu}{\alpha}\Big(\frac{\mathrm{E}[Y_t]}{r}\Big). \tag{16}$$

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

Proposition 2. *For an $M \times K$ F-RAN with $N \geq M \geq K \geq 2$, the online reactive caching scheme achieves a long-term NDT that is upper bounded as*

$$\bar{\delta}_{\text{react}}(\mu, r) \leq \delta_{\text{off,ach}}\Big(\frac{\mu}{\alpha}, r\Big) + \frac{p\mu}{r(1-p/N)(\alpha-1)}, \tag{17}$$

where $\delta_{\text{off,ach}}(\mu, r)$ is given in (8)-(9) *and $\alpha > 1$ is an arbitrary parameter. It follows that we have the inequalities $\bar{\delta}^*_{\text{on,k}}(\mu, r) \leq \bar{\delta}^*_{\text{on,u}}(\mu, r) \leq \bar{\delta}_{\text{react}}(\mu, r)$.*

Proof. Plugging the achievable NDT (16) into the definition of long-term NDT in (5), we have

$$\bar{\delta}_{\text{react}}(\mu, r) = \delta^*_{\text{off,ach}}\Big(\frac{\mu}{\alpha}, r\Big) + \Big(\frac{\mu}{\alpha r}\Big)\limsup_{T\to\infty}\frac{1}{T}\sum_{t=1}^{T}\mathrm{E}[Y_t]. \tag{18}$$

Furthermore, since the users' demand distribution, caching and random eviction policies are the same as in [14], we can leverage [14, Lemma 3] to obtain the following upper bound on the long-term average number of requested but not cached files as

$$\limsup_{T\to\infty}\frac{1}{T}\sum_{t=1}^{T}\mathrm{E}[Y_t]\leq\frac{p}{(1-p/N)(1-1/\alpha)}. \tag{19}$$

Plugging (19) into (18) completes the proof. □

V. Comparison between Online and Offline Caching

In this section, we compare the performance of online and offline caching for F-RAN. For reference, in [14], it was proved that the minimum transmission rate $\bar{R}^*_{\text{on}}$ for online caching in a multicast scenario with caching at the receivers satisfies the conditions $R^*_{\text{off}}\leq\bar{R}^*_{\text{on}}\leq 2R^*_{\text{off}}+6$, where R^*_{off} is the minimum rate required for offline caching. This shows that online caching has the same rate performance order-wise of offline caching. In the next proposition, we evaluate related bounds for the F-RAN under study, revealing the significantly different scaling of the NDT for online and offline caching in an F-RAN.

Proposition 3. *For an $M\times K$ F-RAN with $N\geq M\geq K\geq 2$, the long-term NDT satisfies the inequalities*

$$\frac{1-\frac{Kp}{N}}{2}\delta^*_{\text{off}}(\mu,r)+\frac{Kp}{N}\Big(1+\frac{\mu}{r}\Big)\leq\bar{\delta}^*_{\text{on,k}}(\mu,r)\leq\bar{\delta}^*_{\text{on,u}}(\mu,r)\leq 2\delta^*_{\text{off}}(\mu,r)+\frac{4}{r}. \tag{20}$$

Proof. The upper bound is obtained by comparing the performance (17) of the proposed reactive scheme with the lower bound in Lemma 2 on the minimum offline NDT $\delta^*_{\text{off}}(\mu,r)$. Instead, the lower bound is obtained by combining the arguments in the proof of Lemma 1 with the idea of enhancing the system performance by periodically replenishing the ENs' caches without adding any fronthaul latency. Details are provided in Appendix B. □

Propositions 3 shows that the long-term NDT with online caching is proportional to the minimum NDT for offline caching, with an additive gap that is inversely proportional to the fronthaul rate r. This contrasts with the scenario with caching at the receivers of a noiseless broadcast channel in [14], in which instead the additive gap between the performance of offline and online caching was found to be constant. To see intuitively why this is the case, 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 $\mathcal{F}_t$ 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, an additive gap as a function of r is inevitable as compared to the offline case.

Remark 1. *Using a similar proof technique, it can also be shown that* $\bar{\delta}^*_{\text{on,k}}(\mu, r) \leq 2\delta^*_{\text{off}}(\mu, r) + 1/r$ *by comparing the performance of the proposed reactive scheme* (15) *with the lower bound in Lemma 2.*

VI. Numerical Results

In this section, we complement the analysis with numerical experiments. We specifically consider the long-term NDT achievable by the greedy scheme (eq. (14)), the proposed proactive scheme (eq. (15)) and the proposed reactive scheme (eq. (18)). For the latter, we evaluate the limit in (18) via Monte Carlo simulations by averaging over a large number of realizations (i.e., 10,000) of the random process Y_t, which is simulated starting from empty caches at time $t = 1$.

The impact of the fronthaul rate r is first considered in Fig. 4. Here, we also plot for reference the achievable NDT for offline caching in (8)-(9), and we assume random eviction for reactive caching. Parameters are set as $p = 0.5$, $\mu = 0.3$, $M = K = 5$ and $N = 10$. It is seen that proactive and reactive caching can significantly improve over greedy delivery by storing content for future slots. Furthermore, the results confirm the main conclusion of Proposition 3: As the fronthaul rate r decreases to zero, the additive gap between online and offline caching grows without bound due to the impossibility for online caching to deliver new popular files. Next, we further investigate the performance comparison of reactive and proactive online caching schemes as a function of the probability of new content p. As shown in Fig. 5 for $r = 0.5$, $K = 5$, $M = 10$, $\mu = 0.8$, $N = 10$, when p is large enough, the reactive approach yields a smaller latency than the proactive scheme. The reason is that, when p is large, proactive caching uses the fronthaul link to deliver a large number of new popular contents, only a small fraction of which will actually be requested by the users.

The figure also compares the performance of reactive online caching under different eviction strategies, namely random, which is used in the proof of Proposition 2; 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. It is seen that LRU and FIFO are both able to improve over the randomized eviction, with the former generally outperforming the latter, especially for large values of p.

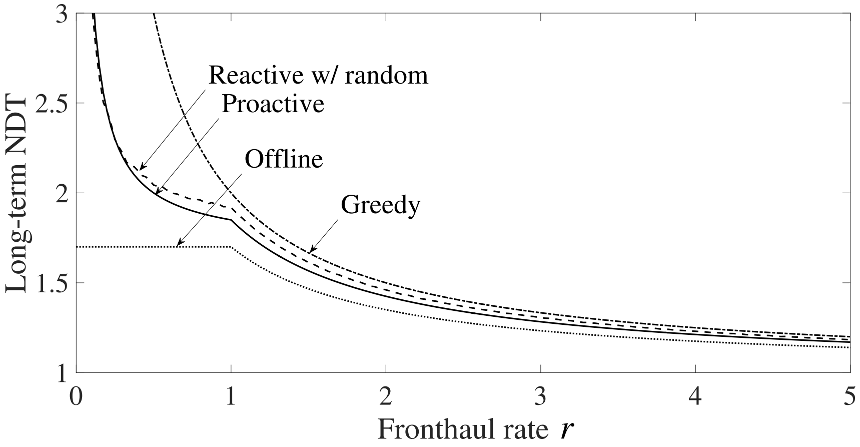


Figure 4: Achievable long-term NDT for greedy (14), proactive scheme (15) and reactive caching with random eviction (18) versus fronthaul rate r. For reference, the offline minimum NDT (8)-(9) is also shown. ($p = 0.5$, $\mu = 0.3$, $M = K = 5$, $N = 10$).

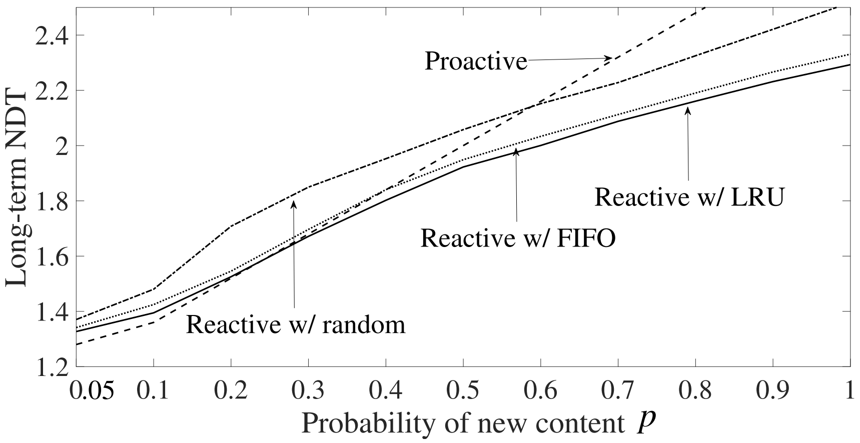


Figure 5: Achievable long-term NDT versus probability of new content p for proactive scheme (15) and reactive caching with random, LRU or FIFO eviction (18) ($r = 0.5$, $K = 5$, $M = 10$, $\mu = 0.8$, $N = 10$).

VII. Conclusions

In this work, we have analyzed the performance limits of content delivery in a fog architecture for an online caching set-up. Information-theoretic bounds have been derived on a high-SNR latency metric termed long-term Normalized Delivery Time (NDT) for both proactive and reactive caching. Analytical, as well as numerical, results have been presented to compare the performance of online and offline caching.

Appendix A
proof of Lemma 2

The proof follows that of [5, Proposition 1] and is based on the fact that the requested files should be decodable given the information subset $\{Y_{[1:l]}, S_{[1:(M-l)]}, U_{[1:(M-l)]}\}$ for any $l \in [0, K]$. Note that the quantities at hand are defined in a manner similar to Sec. II but the dependence on time is dropped in the offline scheme. The only caveat is that here instead of [5, Eq. (67c)] the following upper bound is used

$$H(S_{[1:(M-l)]}|F_{[1:l]}, F_{[K+1:N]}) \leq \min\Big((K-l), (M-l)(K-l)\mu\Big)L. \tag{21}$$

The first term in the right hand side of (21) is proved by using the fact that the $S_{[1:(M-l)]}$, when conditioned on the files $(F_{[1:l]}, F_{[K+1:N]})$ is only a function of the files $F_{[l+1:K]}$, whose entropy is $(K-l)L$ for $0 \leq l \leq K$. The second term as proved in [5] by using the fact that the joint entropy of $S_{[1:(M-l)]}$ cannot exceed the sum of the marginal entropies $H(S_i|F_{[1:l]}, F_{[K+1,N]})$ for $i \in [1:(M-l)]$. Plugging (21) into [5, Eq. (66)] and then taking the limit $L \to \infty$ and $P \to \infty$, results in (12), while (11) and (13) follow as in [5].

Appendix B
proof of Proposition 3

Lower bound: To prove the lower bound, we first introduce the following proposition.

Proposition 4. *(Lower bound on the Long-Term NDT of Online Caching). 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*

$$\text{minimize} \quad \delta_E + \delta_F \tag{22}$$

$$\text{subject to :}\ l\delta_E + (M-l)r\delta_F \geq K - \min\Big((K-l-1), (M-l)(K-l-1)\mu\Big) \tag{23}$$

$$\delta_F \geq 0, \delta_E \geq 1, \tag{24}$$

where (23) 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 2.

Proof. See Appendix C. □

Now, using Proposition 4, we have

$$\begin{aligned}\bar{\delta}^*_{\text{on,u}}(\mu,r) &\geq \bar{\delta}^*_{\text{on,k}}(\mu,r)\\ &\geq \Big(1-\frac{Kp}{N}\Big)\delta_{\text{off,lb}}(\mu,r)+\frac{Kp}{N}\delta_{\text{on,lb}}(\mu,r)\\ &\overset{(a)}{\geq} \frac{(1-\frac{Kp}{N})}{2}\delta^*_{\text{off}}(\mu,r)+\frac{Kp}{N}\delta_{\text{on,lb}}(\mu,r)\\ &\overset{(b)}{\geq} \frac{(1-\frac{Kp}{N})}{2}\delta^*_{\text{off}}(\mu,r)+\frac{Kp}{N}\Big(1+\frac{\mu}{r}\Big),\end{aligned} \tag{25}$$

where (a) is obtained using Lemma 1, namely $\delta^*_{\text{off}}(\mu,r)/\delta_{\text{off,lb}}(\mu,r) \leq 2$ and (b) follows by deriving a lower bound on the optimal solution of the LP (22) by setting $l=0$ in the constraint (23) and summing the result with constraint (24).

Upper bound: To prove the upper bound, we leverage the following lemma.

Lemma 3. *For any* $1<\alpha\leq 2$*, we have the following inequality*

$$\delta_{\text{off,ach}}\Big(\frac{\mu}{\alpha},r\Big) \leq 2\delta^*_{\text{off}}(\mu,r)+1+\frac{2}{r}(\alpha-1). \tag{26}$$

Proof. See Appendix D. □

Using Proposition 2 and Lemma 3, an upper bound on the long-term average NDT of the proposed reactive caching scheme is obtained as

$$\bar{\delta}_{\text{react}}(\mu,r) \leq 2\delta^*_{\text{off}}(\mu,r)+f(\alpha), \tag{27}$$

where

$$f(\alpha)=1+\frac{2}{r}(\alpha-1)+\frac{Np(\mu/r)}{(N-p)(\alpha-1)}. \tag{28}$$

Since the additive gap (28) 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 $\delta^*_{\text{on,u}}(\mu,r) \leq \bar{\delta}_{\text{react}}(\mu,r)$ the upper bound is proved.

Appendix C
Proof of Proposition 4

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 $\mathcal{F}_t$ 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 (4) in time slot t for the genie-aided system can be lower bounded as

$$\delta_t(\mu, r) \geq (1 - Kp/N)\delta_{\text{off,lb}}(\mu, r) + (Kp/N)\delta_{\text{on,lb}}(\mu, r), \tag{29}$$

where $\delta_{\text{off,lb}}(\mu, r)$ is the lower bound on the minimum NDT for offline caching in Lemma 2, 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 (29) follows since, in the genie-aided system, with probability $1-Kp/N$ the system is equivalent to the offline caching set-up studied in [5], 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 [5] 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 [5], we can write

$$H(S_{[1:(M-l)]}|F_{[1:l]}, F_{[K+1:N]}) \leq \min\Big(K - l - 1, (M - l)(K - l - 1)\mu\Big)L, \tag{30}$$

which is obtained by using the fact that the constrained entropy of the cached content cannot be larger than the overall size of files F_j with $j \in [l + 1, K - 1]$. Plugging (30) into [5, Eq. (66)] and then taking the limit $L \to \infty$ and $P \to \infty$, results in (23). The rest of proof is as in [5, Appendix I]. Using (29) in the definition of long-term average NDT (5) concludes the proof.

APPENDIX D
PROOF OF LEMMA 3

To prove Lemma 3, for any given $1 < \alpha \leq 2$, we consider separately small cache regime with $\mu/\alpha \in [0, 1/M]$ and the high cache regime with $\mu/\alpha \in [1/M, 1]$.

- **Small-cache Regime** ($\mu/\alpha \in [0, 1/M]$): Using Lemma 2 a lower bound on the minimum NDT can be obtained as

$$\delta^*_{\text{off}}(\mu, r) \geq 1 + \frac{K(1 - \mu M)}{Mr} \tag{31}$$

by considering the constraint the constraint (12) with $l = 0$ and constraint (13). Using (8) and (31), we have the following upper bound

$$\begin{aligned}\delta_{\text{off,ach}}\Big(\frac{\mu}{\alpha}, r\Big) &\leq \frac{(M+K-1)\mu}{\alpha} + \Big(1+\frac{K}{Mr}\Big) \times \Big(1-\frac{\mu M}{\alpha}\Big)\\ &= 1 + \frac{(K-1)\mu}{\alpha} + \frac{K}{Mr}\Big(1-\frac{\mu M}{\alpha}\Big).\end{aligned} \tag{32}$$

From (31) and (32), we have

$$\begin{aligned}&\delta_{\text{off,ach}}\Big(\frac{\mu}{\alpha}, r\Big) - 2\delta^*_{\text{off}}(\mu, r) \leq 1 + \frac{K}{Mr}\Big(1-\frac{\mu M}{\alpha}\Big)\\ &+ \frac{(K-1)\mu}{\alpha} - 2 - 2\frac{K(1-\mu M)}{Mr}\\ &= -1 + \frac{K}{Mr}\Big(-1-\frac{\mu M}{\alpha}+2\mu M\Big) + \frac{(K-1)\mu}{\alpha}\\ &\overset{(a)}{\leq} \frac{K\mu}{r}\Big(2-\frac{1}{\alpha}\Big) + \frac{(K-1)\mu}{\alpha} - \frac{K}{Mr}\\ &\overset{(b)}{\leq} \frac{2K}{Mr}(\alpha-1) + \frac{(K-1)}{M}\\ &\overset{(c)}{\leq} \frac{2}{r}(\alpha-1)+1,\end{aligned} \tag{33}$$

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

- **Large-cache regime** (**$\mu/\alpha \in [1/M, 1]$**): In this regime, upper bound on $\delta_{\text{off,ach}}(\mu/\alpha, r)$ in (17) is independent of μ/α and therefore in the same way as [5, Proposition 7], we have

$$\frac{\delta_{\text{off,ach}}\big(\mu/\alpha, r\big)}{\delta^*_{\text{off}}(\mu, r)} \leq 2. \tag{34}$$

Finally, using (33) and (34) concludes the proof.

References

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

[2] O. Simeone, A. Maeder, M. Peng, O. Sahin, and W. Yu, “Cloud radio access network: Virtualizing wireless access for dense heterogeneous systems,” *Journal Commun. and Netw.*, vol. 18, no. 2, pp. 135–149, Apr. 2016.

[3] J. G. Herrera and J. F. Botero, “Resource allocation in NFV: A comprehensive survey,” *IEEE Trans. Netw. Service Man.*, vol. 13, no. 3, pp. 518–532, Sep. 2016.

[4] R. Tandon and O. Simeone, “Cloud-aided wireless networks with edge caching: Fundamental latency trade offs in fog radio access networks,” in *Proc. IEEE Int. Symp. on Inform. Theory (ISIT)*, pp. 2029-2033, Barcelona, Spain, Jul. 2016.

[5] A. Sengupta, R. Tandon, and O. Simeone, “Cloud and cache-aided wireless networks: Fundamental latency trade-offs,” Mar. 2016. [Online]. Available: http://arxiv.org/abs/1605.01690

[6] M. A. Maddah-Ali and U. Niesen, “Cache-aided interference channels,” in *Proc. IEEE Int. Symp. on Inform. Theory (ISIT)*, pp. 809-813, Hong Kong, China, Jul. 2015.

[7] N. Naderializadeh, M. A. Maddah-Ali, and A. S. Avestimehr, “Fundamental limits of cache-aided interference management,” in *Proc. IEEE Int. Symp. on Inform. Theory (ISIT)*, pp. 2044-2048, Barcelona, Spain, Jul. 2016.

[8] F. Xu, K. Liu, and M. Tao, “Cooperative Tx/Rx caching in interference channels: A storage-latency tradeoff study,” in *Proc. IEEE Int. Symp. on Inform. Theory (ISIT)*, pp. 2034-2038, Barcelona, Spain, Jul. 2016.

[9] A. Sengupta, R. Tandon, and O. Simeone, “Cache-aided wireless networks: Tradeoffs between storage and latency,” in *Proc. Conf. Inform. Science and Systems (CISS)*, pp. 320-325, Princeton, NJ, Mar. 2016.

[10] S. M. Azimi, O. Simeone, and R. Tandon, “Fundamental limits on latency in small-cell caching systems: An information-theoretic analysis,” in *Proc. IEEE Global Commun. Conf. (GLOBECOM)*, Washington, DC, Dec. 2016.

[11] J. Koh, O. Simeone, R. Tandon, and J. Kang, “Cloud-aided edge caching with wireless multicast fronthauling in fog radio access networks,” in *Proc. IEEE Wireless Commun. and Netw. Conf. (WCNC)*, San Francisco, CA, Mar. 2017.

[12] V. Martina, M. Garetto, and E. Leonardi, “A unified approach to the performance analysis of caching systems,” Feb. 2016. [Online]. Available: http://arxiv.org/abs/1307.67026

[13] E. Bastug, M. Bennis, and M. Debbah, “Living on the edge: The role of proactive caching in 5G wireless networks,” *IEEE Commun. Mag.*, vol. 52, no. 8, pp. 82–89, Aug. 2014.

[14] R. Pedarsani, M. A. Maddah-Ali, and U. Niesen, “Online coded caching,” *IEEE Trans. Inf. Theory*, vol. 24, no. 2, pp. 836–845, 2016.

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

[16] S.-H. Park, O. Simeone, and 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.