Contents

1 Cache-Enabled Cloud Radio Access Networks *page* 2

1.1 Introduction 3

1.2 Cache-Enabled Cloud RAN Model 4

1.2.1 Network Model 4

1.2.2 Content-Centric BS Clustering 5

1.2.3 Caching at BSs 6

1.2.4 Backhauling 7

1.3 Caching at BSs for Cooperation in Access Link 9

1.3.1 Joint BS Clustering and Beamforming Design 9

1.3.2 Performance Evaluation 11

1.4 Caching at BSs for Multicasting in Backhaul Link 13

1.4.1 Joint BS Cache Allocation and Beamforming Design 13

1.4.2 Performance Evaluation 15

1.5 Conclusions and Open Issues 19

References 20

1 Cache-Enabled Cloud Radio Access Networks

Meixia Tao[a], Erkai Chen[a], Wei Yu[b], and Ya-Feng Liu[c]

This chapter presents a content-centric framework for transmission optimization in cloud radio access networks (RANs) by leveraging wireless edge caching and physical-layer multicasting. We consider a cache-enabled cloud RAN, where each base station (BS) is connected to a central processor (CP) via a potentially capacity-limited backhaul link and equipped with a local cache to alleviate the backhaul load. We first study the caching effects on multicast-enabled access downlink, where users that request the same content form a multicast group and are served by the same BS or BS cluster using multicasting. We study the cache-aware joint design of the content-centric BS clustering and multicast beamforming to minimize the system total power cost and backhaul cost under individual minimum transmission rate constraints for each multicast group. Through simulation results, we show that the proposed cache-aware content-centric multicast transmission is much superior to the traditional user-centric unicast transmission in terms of system total transmit power reduction and backhaul saving. We then study the caching effects on backhaul downlink with wireless multicast backhaul, where the CP delivers the requested contents to a single cluster of BSs via multicasting. Given a total cache size constraint, we study the joint cache size allocation at the BSs and the optimal multicast beamforming transmission at the CP to minimize the expected downloading time of requested contents from the CP to the BSs. Numerical results provide some useful insights into the BS caching design and show that the optimized cache size allocation scheme outperforms the uniform allocation and other heuristic schemes.

[a] M. Tao and E. Chen are with the Department of Electronic Engineering, Shanghai Jiao Tong University, Shanghai 200240, China (e-mails: {mxtao, cek1006}@sjtu.edu.cn).

[b] W. Yu are with The Edward S. Rogers Sr. Department of Electrical and Computer Engineering, University of Toronto, Toronto, ON M5S 3G4, Canada (e-mail: weiyu@comm.utoronto.ca).

[c] Y.-F. Liu is with the State Key Laboratory of Scientific and Engineering Computing, Institute of Computational Mathematics and Scientific/Engineering Computing, Academy of Mathematics and Systems Science, Chinese Academy of Sciences, Beijing 100190, China (e-mail: yafliu@lsec.cc.ac.cn).

1.1 Introduction

Cloud radio access network (cloud RAN) is a promising network architecture for the next generation of wireless cellular networks [1]. It can boost network capacity and increase energy efficiency by centralized signal processing among all the BSs that are connected to a central processor (CP) via potentially capacity-limited backhaul links. However, performing full joint processing requires the users' payload data to be shared among all the BSs, which can place a significant burden on backhaul links. As such, there is a fundamental tradeoff between the access link efficiency and the backhaul link consumption in cloud RANs. This chapter presents how to exploit wireless edge caching, in conjunction with physical-layer multicasting, in cloud RAN architectures to alleviate the backhaul requirement and improve system energy efficiency.

Wireless edge caching has emerged as a promising approach that can reduce peak traffic and backhaul burden for wireless content delivery by caching some popular contents at the local BSs or pushing directly the contents at user devices during the off-peak time [2]. On the other hand, multicasting provides an efficient capacity-offloading approach to deliver a common message to multiple receivers concurrently [3, 4]. It has great potential in many applications, e.g., video streaming, mobile application updates, and public group communications. It can also be exploited in wireless backhaul to push common information from a macro BS to multiple small BSs. Caching and multicasting are thus two important enabling techniques to accelerate content delivery in wireless networks.

This chapter presents a content-centric framework for transmission optimization in cloud RANs by collectively leveraging caching and multicasting. We consider a cache-enabled cloud RAN, where each BS has a local cache with limited storage size and is connected to a CP via a dedicated or shared backhaul link. If the requested contents are not cached in the local cache of a BS, it will acquire the content from the core network via the backhaul links. Users requesting a same content form a group and are served by the same BS cluster via multicast transmission This chapter shows that caching can improve the system-level performance of cloud RAN in two different ways: for both the access link and the backhaul link. The first part of the chapter studies the design of caching and multicasting in the access link. We study the cache-aware joint content-centric BS clustering and multicast beamforming design to minimize the system total network cost subject to a minimum rate constraint for each individual multicast group. Simulation results show that the proposed cache-aware content-centric multicast transmission is superior to the traditional user-centric unicast transmission in terms of system transmit power reduction and backhaul saving.

The second part of the chapter studies the design of caching and multicasting in the backhaul link, where the BSs fetch the requested contents from the CP through a shared wireless backhaul using joint cache-channel coding. Given a total cache size constraint, we study a mixed time-scale optimization for cache size allocation among all the BSs and multicast beamforming at the CP to mini-

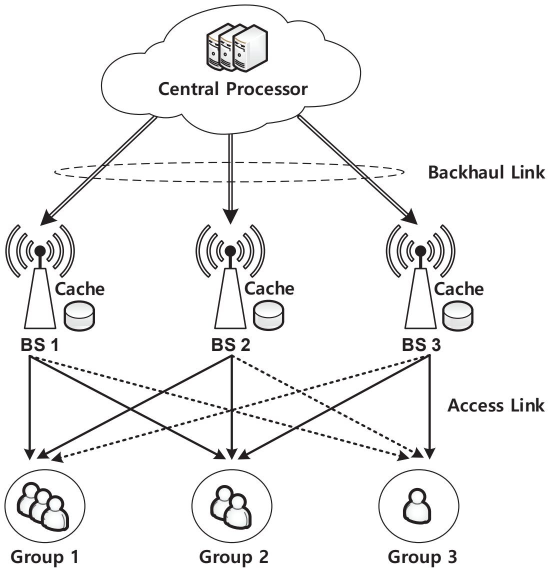

Figure 1.1 Downlink transmission of a cache-enabled cloud RAN.

mize the expected downloading time of requested contents in the backhaul phase. Numerical results provide some useful insights into the BS caching design and show that the optimized cache size allocation scheme outperforms the uniform allocation and other heuristic schemes.

The rest of this chapter is organized as follows. Section 1.2 introduces the model of the cache-enabled cloud RAN. Section 1.3 studies caching and multicasting in the access link. Section 1.4 studies the caching and multicasting in the backhaul link. Finally, we conclude this chapter in Section 1.5 and outline some possible directions for future research.

1.2 Cache-Enabled Cloud RAN Model

1.2.1 Network Model

As shown in Fig. 1.1, we consider the downlink transmission of a cloud RAN, where there are N BSs and K users. Each BS has a local cache and is connected to a cloud-based CP via a backhaul link. The CP has a database consisting of F files, where the size of each file is normalized as 1. Let p_f denote the request probability (i.e., popularity distribution) of the f-th file, which satisfies $0 \leq p_f \leq 1$ and $\sum_{f=1}^{F} p_f = 1$. Let C_n ($C_n \leq F$) denote the cache size of the n-th BS. Each BS can pre-store some file bits during off-peak time prior to user request. If the requested file of its serving user has been entirely stored in the local cache of this BS, the BS can access the file directly. Otherwise, it will download the requested file or the uncached part of this file from the CP via its backhaul link.

In this chapter, it is assumed that the channel state information (CSI) is perfectly known at the CP for joint signal processing and all BSs can precisely synchronize with each other for downlink cooperative transmission. Our focus is to illustrate a content-centric transmission framework in the cached-enabled cloud RAN and its baseband beamforming design.

1.2.2 Content-Centric BS Clustering

A prominent approach to mitigate the backhaul load in traditional cloud RANs is to serve each user using an individually selected subset of neighboring BSs, referred to as *user-centric BS clustering*, regardless of the contents each user requests. By adopting user-centric BS clustering, the CP only needs to deliver the user's payload data to its serving BSs rather than all the BSs, which can reduce the backhaul load significantly. In this case, different clusters for different users may overlap and there are no explicit cluster boundaries [5].

Generally, the users request contents according to some popularity distribution such as the Zipf distribution [6]. The more popular a content is, the more likely it will be requested and the more requests it will receive. By taking the content popularity into account, a *content-centric BS clustering* strategy is proposed in [7]. In the content-centric BS clustering, the users requesting a same content are grouped together and served by a cluster of BSs formed with respect to each content. Within each cluster, multicast transmission is then adopted to serve the users. The BS clusters for different contents can overlap with each other. Compared with user-centric BS clustering, content-centric BS clustering exploits the popularity of the request contents and benefits from multicast transmission, and thus can provide efficient content delivery in the considered networks.

In the following, we present the transmission model with content-centric BS clustering in detail. We assume that each user can request a content in each scheduling time slot. Denote $\mathcal{G}_m$ as the m-th multicast group formed by the users requesting file f_m, for all $m = 1, \ldots, M$, where M $(1 \leq M \leq \min\{K, F\})$ is the total number of the formed multicast groups. Denote the serving BS cluster of multicast group m as $\mathcal{Q}_m$, where $\mathcal{Q}_m \subseteq \mathcal{N}$. An example with three multicast groups is illustrated in Fig. 1.1, where the serving BSs of the three multicast groups are $\mathcal{Q}_1 = \{1, 2\}$, $\mathcal{Q}_2 = \{1, 2, 3\}$, and $\mathcal{Q}_3 = \{3\}$, respectively.

Define a binary matrix $\mathbf{S} \in \{0, 1\}^{M \times N}$ as the indicator of BS clustering, where $s_{m,n} = 1$ represents that BS n is within the BS cluster of multicast group m, otherwise $s_{m,n} = 0$. Denote $\mathbf{w}_m = [\mathbf{w}_{m,1}^H, \mathbf{w}_{m,2}^H, \ldots, \mathbf{w}_{m,N}^H]^H \in \mathbb{C}^{NL \times 1}$ as the network-wide beamformer for the m-th group, where $\mathbf{w}_{m,n} \in \mathbb{C}^{L \times 1}$ is the beamformer of group m at BS n. Note that $\mathbf{w}_{m,n} = \mathbf{0}$ if $s_{m,n} = 0$. Therefore, $\mathbf{w}_m$ is potentially (group) sparse. For each user $k \in \mathcal{G}_m$, the received signal can be written as

$$y_k = \mathbf{h}_k^H \mathbf{w}_m x_m + \sum_{j \neq m}^{M} \mathbf{h}_k^H \mathbf{w}_j x_j + n_k, \tag{1.1}$$

where $\mathbf{h}_k = [\mathbf{h}_{k,1}^H, \mathbf{h}_{k,2}^H, \ldots, \mathbf{h}_{k,N}^H]^H \in \mathbb{C}^{NL \times 1}$ is the composite channel vector between all BSs and the k-th user, $x_m \in \mathbb{C}$ is the message intended for group m, and $n_k \sim \mathcal{CN}(0, \sigma_k^2)$ is the additive white Gaussian noise. The corresponding SINR at user k can be expressed as

$$\text{SINR}_k = \frac{|\mathbf{h}_k^H \mathbf{w}_m|^2}{\sum_{j \neq m}^{M} |\mathbf{h}_k^H \mathbf{w}_j|^2 + \sigma_k^2}. \tag{1.2}$$

Accordingly, the total transmit power of the network can be expressed as:

$$C_P = \sum_{m=1}^{M} \sum_{n=1}^{N} \|\mathbf{w}_{m,n}\|_2^2. \tag{1.3}$$

Compared with the traditional user-centric BS clustering, where each user is served by its nearby BSs that have good channel conditions, the content-centric BS clustering is more complicated. In the content-centric BS clustering, since the users within the same multicast group may be dispersed geographically, it is no longer feasible to determine the BS clustering simply according to the received signal strength or the location closeness between each BS and each user. Moreover, by considering the local cache at each BS, the BS that has cached the requested file may have a higher chance to joint the cluster. As such, the content-centric BS clustering in the considered network must be aware of both channel states and cache states.

1.2.3 Caching at BSs

Caching at the BSs can enable more BSs to cooperatively transmit the same content to the users in the access link. What contents to cache at each BS is a crucial design factor in cache-enabled cloud RAN. Intuitively, in a sparse network where each user can access to only one single BS, it is optimal to cache contents with the largest popularities in each BS in terms of cache hit ratio maximization. While in a densely deployed network where each user can access to multiple BSs, finding the optimal cache placement is often intractable [2]. By allowing coded caching at each BS, one can find the optimal coded fraction of each file efficiently [8]. In this chapter, however, we restrict to uncoded caching for simplicity and consider three heuristic caching strategies as follows.

- *Popularity-aware Caching (PopC):* All the storage size of each BS are used to cache the contents with the largest popularities. This strategy can fully exploit the benefits of full cooperation. However, when the content popularity is uniformly distributed, it may cause high backhaul load due to the low cache hit ratio.

- *Random Caching (RanC):* All the contents are cached at the BSs randomly and equally without knowing their popularity. Due to the randomness in the cache placement, it is highly probable that each user can find its requested file from the caches of the BSs without resorting to CP via backhaul. However, since different BSs tend to cache distinct contents, there is little opportunity for cooperative transmission.
- *Probabilistic Caching (ProC):* Each BS randomly caches a content with a certain probability that is related to its popularity. With a higher popularity, the content is more likely to be cached at the BSs. In this caching strategy, a better tradeoff between the cooperation gain and the cache hit ratio can be made.

We shall evaluate the performance of the above three caching strategies via simulation in Section 1.3.2. Besides content placement, how much cache size to deploy at each BS is also an important design factor, which shall be discussed in detail in Section 1.4.

1.2.4 Backhauling

The backhaul with limited capacity has become a big concern for small-cell deployment. Although the traditional fiber-based backhaul solution can provide high data rates, the prohibitive cost is high and the geographical limitations also make it impossible to deploy in many practical scenarios. Instead, with low-cost and plug-and-play installation, wireless backhauling is a promising solution. It is worth to note that with wireless backhauling, the data-sharing strategy is preferred since it has the following two advantages. First, the CP can exploit the multicast transmission to deliver the user messages simultaneously to multiple BSs via the shared backhaul. Second the BSs can cache part of the user messages to further reduce the backhaul load. While for the compression strategy, sine the compressed signals generated for different BSs are different and they are also adaptive to the channel conditions, it cannot exploit the benefits of multicasting or caching. In this chapter, we assume that the backhaul connections can be dedicated fiber optic cables, or they can be a shared wireless link.

Dedicated Wired Backhaul

We model the cost of the dedicated backhaul link as the required transmission rate of this link. Define a binary matrix $\mathbf{C} \in \{0,1\}^{F \times N}$ to denote the cache status, where $c_{f,n} = 1$ represents that the f-th file is cached in the n-th BS, otherwise $c_{f,n} = 0$. For each BS, if the requested file is not cached in its local storage, it should fetch the file from the CP with the backhaul transmission rate as large as the content-delivery rate R_m. Therefore, we model the backhaul cost as the transmission rate of the corresponding file. Then the overall backhaul cost is

$$C_B = \sum_{m=1}^{M} \sum_{n=1}^{N} s_{m,n}(1 - c_{f_m,n})R_m, \tag{1.4}$$

where f_m denotes the file requested by multicast group m and R_m is the transmission rate for group m.

Shared Wireless Backhaul

Compared to the dedicated wired backhaul, shared wireless backhaul not only is much easier to deploy (when wireline infrastructure is not available) but also enjoys the crucial wireless multicast advantage which allows for efficient content delivery to multiple BSs using a same resource block. Wireless multicast is ideally suited for enabling the cooperative transmission benefit of C-RAN; but it also brings in the challenge of path-loss, fading, and shadowing effect of the wireless medium. In particular, because of the different locations of the BSs, there may be considerable disparity in the quality of their respective channels. Deploying caching at BSs [9] (i.e., BSs can pre-store contents of popular files) can handle the channel disparity issue in wireless multicast to aid the BSs with weak channels. For wireless backhauling, the backhaul efficiency is often modeled as the (expected) downloading time. In this chapter, we consider a cluster of N BSs cooperatively serving users. The CP delivers the user's message to all the BSs via multicasting. Suppose that each file has normalized size of 1 and each BS n has a local storage of size C_n that can cache some of the files. In other words, given cache size allocation C_n, each BS n can cache C_n fraction of the file. We assume that the channel coherent time is large enough such that the file delivery can be completed within one coherent time. According to [10, Lemma 1], by adopting the joint cache-channel coding strategy [11], the file delivery rate R with can be written as

$$R = \min_{n} \left\{ \frac{I\left(\mathbf{x}; y_n\right)}{1 - C_n} \right\}, \tag{1.5}$$

and the downloading time thus can be expressed as

$$T = \frac{1}{R} = \max_{n} \left\{ \frac{1 - C_n}{I\left(\mathbf{x}; y_n\right)} \right\}. \tag{1.6}$$

Here, $I\left(\mathbf{x}; y_n\right)$ denotes the mutual information between the transmit signal $\mathbf{x}$ and the received signal y_n. If the file size is S, then the real downloading time should be $S \times T$.

Notice that the above $\{I\left(\mathbf{x}; y_n\right)\}$ depend on the channel realizations and the beamforming vectors at the CP and hence change quickly in different fading blocks; while the cache size $\{C_n\}$ should be allocated based on the long-term statistics of the backhaul channel. Therefore, the BS cache size allocation and the beamforming design occur in different time scales. In the later part of this chapter, we shall focus on the downloading time T in (1.6).

1.3 Caching at BSs for Cooperation in Access Link

We now treat the optimization of caching and multicasting in the access link of cloud RAN. It is worthwhile to mention that the cache placement and content delivery occur in different timescales. Specifically, cache placement often happens in days or hours, while content delivery happens in a much shorter timescale [2, 12]. In the shorter timescale of each transmission slot, the cache placement is usually fixed according to some strategy. We can then optimize the content delivery scheme, which should be adaptive to the instantaneous channel realization and the cache placement. In the larger timescale, the cache placement can be optimized by taking into account the content popularity distribution as well as the long-term statistics of the wireless channel. In this section, we mainly focus on the short timescale problem in the access link, i.e., the joint optimization of content-centric BS clustering and multicast beamforming with given cache placement. The large timescale problem, i.e. the design of cache placement shall be briefly address via numerical results.

1.3.1 Joint BS Clustering and Beamforming Design

In this section, given the BS caching, we study the joint content-centric BS clustering and multicast beamforming design in access link to seek the minimum network cost. Specifically, in the considered network architecture, the network cost is modeled as the weighted sum of the backhaul cost and the transmission power:

$$C_N = C_B + \eta C_P, \tag{1.7}$$

where $\eta > 0$ is a weighting parameter.

The total network cost minimization problem with given cache placement can be formulated as:

$$\begin{aligned}
\mathcal{P}_0 : \quad \min_{\{\mathbf{w}_{m,n}\},\{s_{m,n}\}} \quad & \sum_{m=1}^{M}\sum_{n=1}^{N} s_{m,n}(1-c_{f_m,n})R_m + \eta\sum_{m=1}^{M}\sum_{n=1}^{N}\|\mathbf{w}_{m,n}\|_2^2 && \text{(1.8a)}\\
\text{s.t.} \quad & \mathrm{SINR}_k \geq \gamma_m, \ \forall\, k \in \mathcal{G}_m, \ \forall\, m && \text{(1.8b)}\\
& s_{m,n} \in \{0,1\}, \ \forall\, m,n && \text{(1.8c)}\\
& (1-s_{m,n})\mathbf{w}_{m,n} = \mathbf{0}, \ \forall\, m,n && \text{(1.8d)}
\end{aligned}$$

where $R_m = B\log(1+\gamma_m)$ is the transmission rate for group m, B is the channel bandwidth, and γ_m is the target SINR for group m.

Note that constraint (1.8d) indicates that if BS n is not in the BS clustering of group m, i.e., $s_{m,n} = 0$, then the beamformer $\mathbf{w}_{m,n}$ should be zero. We also note that problem $\mathcal{P}_0$ can be infeasible due to the QoS constraint (1.8b). In general, determining the feasibility of this problem is very difficult. Therefore, in this section, we only discuss $\mathcal{P}_0$ when it is feasible.

Problem $\mathcal{P}_0$ is a non-convex mixed-integer non-linear programming (MINLP) problem and is combinatorial in nature; it is in general challenging to find its global optimum solution. However, an exhaustive search can be adopted to find the global optimum BS clusters. Specifically, there are total 2^{MN} candidate BS clustering matrices $\{\mathbf{S}\}$. For each given $\mathbf{S}$, we can solve the following power minimization problem to obtain the power cost:

$$\begin{aligned} \mathcal{P}(\mathcal{Z}_{\mathbf{S}}): \quad \min_{\{\mathbf{w}_{m,n}\}} \quad & \sum_{m=1}^{M}\sum_{n=1}^{N}\|\mathbf{w}_{m,n}\|_2^2 && (1.9a) \\ \text{s.t.} \quad & (1.8b), \\ & \mathbf{w}_{m,n} = \mathbf{0},\ \forall (m,n) \in \mathcal{Z}_{\mathbf{S}}. && (1.9b) \end{aligned}$$

where $\mathcal{Z}_{\mathbf{S}} = \{(m,n) \mid s_{m,n} = 0\}$ denotes the set of inactive BS-content pairs. While the backhaul cost C_B reduces to a constant.

Similar to the traditional multicast beamforming problems [13, 14], $\mathcal{P}(\mathcal{Z}_{\mathbf{S}})$ is a non-convex quadratically constrained quadratic programming (QCQP) problem. Different from unicast beamforming problem which can be equivalently transformed into a second-order cone programming (SOCP) problem and hence solved efficiently. Multicast beamforming problem is generally NP-hard. A semi-definite relaxation (SDR) method is developed in [14] to obtain a near-optimal solution. After solving $\mathcal{P}(\mathcal{Z}_{\mathbf{S}})$ with all possible matrices $\mathbf{S}$'s, we can find the one with the minimum objective.

Another approach to deal with problem $\mathcal{P}_0$ is to reformulate it as a more tractable sparse multicast beamforming (SBF) problem. Specifically, when $\mathbf{w}_{m,n} = \mathbf{0}$, we have:

$$s_{m,n} = \begin{cases} 0, & \text{if } c_{f_m,n} = 0, \\ 0 \text{ or } 1, & \text{if } c_{f_m,n} = 1. \end{cases} \tag{1.10}$$

Otherwise, according to constraint (1.8d), there holds $s_{m,n} = 1$. Therefore, we have the following relationship between the BS cluster and the beamformer:

$$s_{m,n} = \left\| \|\mathbf{w}_{m,n}\|_2^2 \right\|_0. \tag{1.11}$$

Note that the ℓ_0-norm is defined as the number of non-zero elements of a vector. It reduces to the indicator function in the scalar case. By substituting (1.11) into the objective function (1.8a), $\mathcal{P}_0$ can be equivalently transformed into the following problem:

$$\begin{aligned} \mathcal{P}_{\text{SBF}}: \quad \min_{\{\mathbf{w}_{m,n}\}} \quad & \sum_{m=1}^{M}\sum_{n=1}^{N} \left\| \|\mathbf{w}_{m,n}\|_2^2 \right\|_0 (1 - c_{f_m,n}) R_m + \eta \sum_{m=1}^{M}\sum_{n=1}^{N}\|\mathbf{w}_{m,n}\|_2^2 && (1.12) \\ \text{s.t.} \quad & (1.8b). \end{aligned}$$

With ℓ_0-norm in the objective function, problem $\mathcal{P}_{\text{SBF}}$ is a sparse multicast beamforming problem. It considers the adaptive content-centric BS clustering inexplicitly, since by solving this problem, a sparse beamformer for each multicast group may be obtained, whose non-zero entries correspond to its serving BSs. The equivalent problem $\mathcal{P}_{\text{SBF}}$ is still difficult due to the non-convex discontinuous ℓ_0-norm in the objective and the non-convex QoS constraint (1.8b).

One way to tackle this issue is to first adopt smoothed ℓ_0-norm approximation to replace the discontinuous ℓ_0-norm with a concave smooth function. The problem after approximation then can be represented as a general form of difference of convex (DC) programming problem, for which the convex-concave procedure (CCP) [15] based algorithm can be adopted to find a stationary solution with convergence guarantee. The main idea behind CCP is to convexify the DC problem by approximating its concave parts with their first order Taylor expansions and then solve the approximated convex subproblems successively until convergence. The details of such an approach can be found in [7].

1.3.2 Performance Evaluation

This section provides numerical results to demonstrate the superiority of the proposed content-centric transmission framework. A hexagonal multi-cell cloud RAN consisting of $N = 7$ BSs is considered, where each BS has $L = 4$ antennas. The distance between BSs is 500 m. There are $K = 30$ users uniformly distributed within the network. The total number of contents is $F = 100$. The cache size of BS n is set to $C_n = C$ for all n. The channel bandwidth is 10 MHz. The BS antenna gain is 10 dBi. The noise power σ_k^2 is set to be -102 dBm for all users. The path-loss is modeled as $PL(\text{dB}) = 148.1 + 37.6\log_{10}(d)$, where d is the distance in km. The shadowing follows the log-normal distribution with parameter being 8 dB. The small-scale fading is modeled as the Rayleigh fading. The SINR target is $\gamma_m = 10$ dB for all multicast groups. All the results are averaged over 100 independent simulation trials.

In this section, we assume the following unequal content popularity distribution: there is one popular content accounting for 0.5 of the request probability, while the rest $F - 1$ contents follow a Zipf distribution with skewness parameter α and the sum probability being 0.5. In the following simulation, the skewness parameter is set to $\alpha = 1$. Each BS can caches up to $C = 10$ contents. More results with different setups can be found in [7].

Effects of Caching

We first evaluate the caching effects and compare the performance of different caching strategies in Fig. 1.2. We consider two scenarios with the number of users being $K = 30$ and $K = 7$, respectively. The skewness parameter α is set to $\alpha = 1$. It can be seen that by carefully designing the caching strategy, the proposed heuristic caching strategy can significantly reduce the backhaul cost, and hence improve the tradeoff performance between backhaul and power. In addition, it is observed that PopC is superior to ProC for most of the tradeoff parameter η, except the extreme case when $\eta \to 0$. Intuitively, in PopC, the most popular contents are cached in all BSs, the cooperative transmission gain

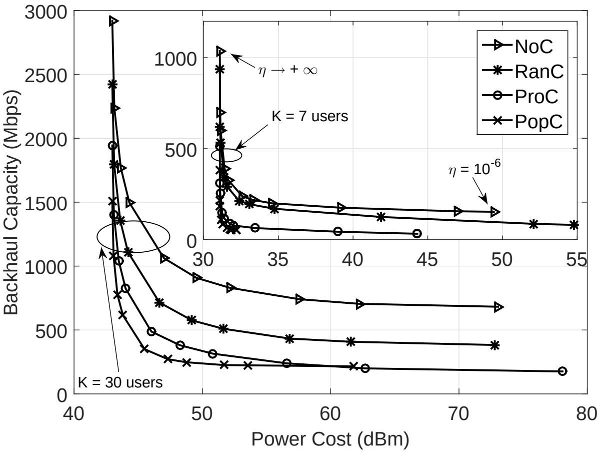

Figure 1.2 Performance comparison of different caching strategies.

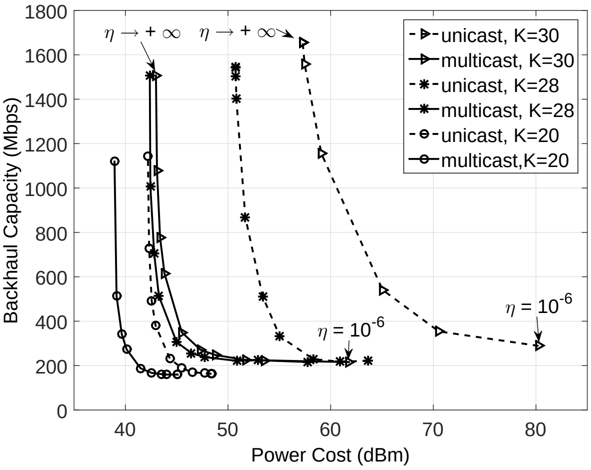

Figure 1.3 Performance comparison between multicast transmission and unicast transmission.

can then be fully exploited. This is very helpful when the network does not care about the backhaul overhead. However, when backhaul is the main concern of the network cost (i.e., $\eta \rightarrow 0$), ProC can outperform PopC. We can also see that all the caching strategies has the minimum transmit power. This is because the minimum transmit power only depends on the target SINRs of the multicast groups.

Effects of Multicasting

We also illustrate the performance comparison of multicast transmission and unicast transmission with different number of active users in Fig. 1.3. For unicast transmission, we design an individual beamformer for each of the users regardless of their requested contents. In order to ensure fairness of the backhaul link overhead, if multiple users that request a same content are served by a same BS, the BS only needs to fetch a copy of the content from the CP with the maximum requested rate if it does not cache the content. We adopt the iterative reweighted ℓ_1-norm based algorithm proposed in [16] to solve the sparse unicast beamforming problem.

From Fig. 1.3, it is seen that when $K = 30$, the unicast transmission performs very poorly. This is mainly due to that the number of transmit antennas is less than the number of users, and hence there is no enough design dimensions for the unicast beamforming. On the other hand, the performance of multicast transmission is much better since it can exploit the content reuse feature among different users and hence fewer beamformers are required. With the number of users decreasing, the performance of unicast transmission becomes better, but still far inferior to multicast transmission. Specifically, in the extreme case when $\eta \to +\infty$, which means only power cost is concerned, we can see that multicast transmission can save 3 dB power comparing with unicast transmission when $K = 20$.

1.4 Caching at BSs for Multicasting in Backhaul Link

1.4.1 Joint BS Cache Allocation and Beamforming Design

Next, we study the effect of caching to improve the wireless backhauling of cloud RAN. We consider the downlink transmission with wireless multicast backhaul, where each user is cooperatively served by a single cluster of BSs. The CP deliver the user's message to these BSs via multicasting. The BSs can also pre-store some fractions of the popular contents during the off-peak hours. The rest of the contents will be fetched from the CP using coded delivery via the wireless multicast backhaul. Assuming that the CP is equipped with multiple antennas and given a total cache size constraint, we study the joint design of cache size allocation at the BSs and the multicast beamforming transmission at the CP so that the expected downloading time of requested files in (1.6) from the CP to the BSs is minimized. It is worthwhile emphasizing that the designs of cache size allocation and the beamforming strategy occur in two different timescales. The cache size allocation is optimized in a much large timescale, which is adaptive to the long-term statistics of the wireless backhaul channel, while the beamforming design is performed under a given cache size allocation and adapts to the instantaneous channel conditions.

Single-File Case: We consider the single file case of normalized size and formulate a mixed-timescale problem for joint design of cache size allocation and multicast beamforming. We first focus on the beamforming design in the shorter timescale with fixed cache size allocation and given content placement. Suppose that $\mathbf{w}$ is the beamforming vector used by the CP and $\mathbf{h}_n$ is the channel between the n-th BS and the CP. The mutual information can be expressed as

$$I(\mathbf{x}; y_n) = \log\left(1 + \frac{\mathrm{Tr}\left(\mathbf{H}_n \mathbf{W}\right)}{\sigma^2}\right),$$

where σ^2 is the variance of the complex Gaussian noise, $\mathbf{H}_n = \mathbf{h}_n \mathbf{h}_n^H$ is the channel covariance matrix, $\mathbf{W} = \mathbf{w}\mathbf{w}^H$ is the covariance matrix for the transmit signal $\mathbf{x}$, where $\{\mathbf{W} \succeq \mathbf{0} \mid \mathrm{Tr}\left(\mathbf{W}\right) \leq P,\ \mathrm{rank}(\mathbf{W}) = 1\}$, and P is the peak power of the CP. We shall drop the rank-one constraint in the above set and define

$$\mathbb{W} = \left\{\mathbf{W} \succeq \mathbf{0} \mid \mathrm{Tr}\left(\mathbf{W}\right) \leq P\right\}.$$

With the given cache allocation $\{C_n\}$, the file downloading time (1.6) can be expressed as

$$T^* = \min_{\mathbf{W} \in \mathbb{W}} \max_n \left\{ \frac{1 - C_n}{\log\left(1 + \frac{\mathrm{Tr}(\mathbf{H}_n \mathbf{W})}{\sigma^2}\right)} \right\}. \tag{1.13}$$

Suppose that all $\mathbf{H}_n$ remain constant within a coherent block but change according to certain channel distribution in different coherent blocks, then T^* in (1.13) is a random variable. In this chapter, our aim is to find the optimal cache size allocation such that the long-term expected file downloading time is minimized. The problem can be mathematically formulated as [10]:

$$\min_{\{C_n\}} \quad \mathbb{E}_{\{\mathbf{H}_n\}}\left[T^*\right] \tag{1.14a}$$

$$\text{s.t.} \quad \sum_{n \in \mathcal{N}} C_n \leq C,\ 0 \leq C_n \leq 1,\ n \in \mathcal{N}. \tag{1.14b}$$

where $C(\leq N)$ is the total cache size across all the BSs.

This problem is difficult mainly due to expectation in the objective function (1.14a), which has no closed-form expression. A popular approach to handling this difficulty is to approximate the expectation in (1.14a) with its sample average [17]. By adopting the sample average approximation, the above problem can be approximated as:

$$\min_{\{C_n, \mathbf{W}^m\}} \quad \frac{1}{M_s} \sum_{m=1}^{M_s} \max_n \left\{ \frac{1 - C_n}{\log\left(1 + \frac{\mathrm{Tr}(\mathbf{H}_n^m \mathbf{W}^m)}{\sigma^2}\right)} \right\} \tag{1.15a}$$

$$\text{s.t.} \quad \sum_n C_n \leq C,\ 0 \leq C_n \leq 1,\ n \in \mathcal{N}, \tag{1.15b}$$

$$\mathrm{Tr}\left(\mathbf{W}^m\right) \leq P,\ \mathbf{W}^m \succeq \mathbf{0},\ m \in \mathcal{M}_s, \tag{1.15c}$$

where M_s is the sample size, $\mathcal{M}_s := \{1, 2, \ldots, M_s\}$, $\{\mathbf{H}_n^m\}_{m \in \mathcal{M}_s}$ are the samples of $\mathbf{H}_n$, and $\mathbf{W}^m$ is the covariance matrix corresponding to the samples

$\{\mathbf{H}_n^m\}_{n\in\mathcal{N}}$. Furthermore, dropping the constant $1/M_s$ in (1.15a) and introducing the auxiliary variable $\{\xi^m\}$, problem (1.15) can be reformulated as

$$\min_{\{C_n,\mathbf{W}^m,\xi^m\}} \quad \sum_{m=1}^{M_s} \frac{1}{\xi^m} \tag{1.16a}$$

$$\text{s.t.} \quad \log\left(1+\frac{\mathrm{Tr}\left(\mathbf{H}_n^m\mathbf{W}^m\right)}{\sigma^2}\right) \geq \xi^m(1-C_n),\ n\in\mathcal{N},\ m\in\mathcal{M}_s, \tag{1.16b}$$

$$\text{(1.15b) and (1.15c).}$$

The above problem (1.16) can be efficiently solved by the trust region method [18], where the nonconvex term $\xi^m(1-C_n)$ in (1.16b) is iteratively approximated by its first-order Taylor expansion and the approximation subproblem at each iteration is convex and can be solved by the ADMM approach [19]. For more details of solving problem (1.14), please refer to [10].

Multi-File Case: We now study the cache size allocation problem in the general case with multiple files and different popularities. We assume that the user requests file f with probability $p_f,\ f\in\mathcal{F}:=\{1,2,\ldots,F\}$, where $\sum_f p_f=1$. The fraction of file f cached in BS n is C_{nf}. Therefore, we have the total cache size constraint $\sum_n\sum_f C_{nf}\leq C$, where $C\leq NF$. If file f is requested, the downloading time, denoted as T_f^*, can be expressed as

$$T_f^* = \min_{\mathbf{W}_f\in\mathbb{W}} \max_n \left\{ \frac{1-C_{nf}}{\log\left(1+\frac{\mathrm{Tr}(\mathbf{H}_n\mathbf{W}_f)}{\sigma^2}\right)} \right\}. \tag{1.17}$$

Different from the downloading time (1.13) in the single file case, the above downloading time T_f^* depends on both the channel conditions and the requested file. We then formulate the cache size allocation problem with multiple files as [10]

$$\min_{\{C_{nf}\}} \quad \sum_f p_f\mathbb{E}_{\{\mathbf{H}_n\}}\left[T_f^*\right] \tag{1.18a}$$

$$\text{s.t.} \quad \sum_n\sum_f C_{nf}\leq C,\ 0\leq C_{nf}\leq 1,\ n\in\mathcal{N},\ f\in\mathcal{F}. \tag{1.18b}$$

This problem can be solved using the same sample approximation approach as in the single file case. Please see [10] for more details.

1.4.2 Performance Evaluation

In this section, we demonstrate the performance of the proposed cache size allocation scheme via simulations. As shown in Fig. 1.4, we consider a C-RAN with $N=5$ BSs, where the BSs are randomly distributed on one side of the CP. The distances between the CP and the BSs are $(398,278,473,286,267)$ meters,

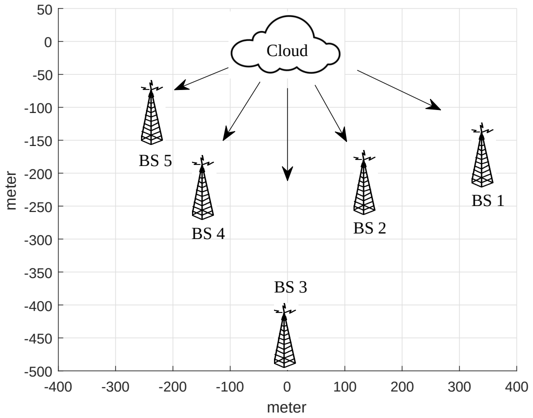

Figure 1.4 An example of C-RAN with 5 BSs.

respectively. We generate 1000 channel realizations of $\mathbf{h}_n$ according to the distribution $\mathbf{h}_n = \mathbf{K}_n^{1/2}\mathbf{v}_n$, where $\mathbf{K}_n$ denotes the large-scale path-loss component and $\mathbf{v}_n$ is the small-scale fading. The path-loss is modeled as $128.1+37.6\log_{10}(d)$ dB, where d is the distance in kilometers.The small-scale fading is modeled as a random vector following the independently and identically Gaussian distribution, i.e., $\mathbf{v}_n \sim \mathcal{CN}(0,1)$. We use the first 100 samples for the sample average approximation method to optimize the cache allocation while the rest 900 samples to evaluate the performance with the obtained cache allocation. More parameters settings can be found in Table I of [10].

Cache Size Allocation with Varying Channel Strengths

In this part, the superiority of the proposed scheme is demonstrate when caching a single file across multiple BSs with different channel strengths. The following schemes are considered as benchmark:

- *Uniform Cache Allocation*: Each BS has the same cache size of $C_n = C/N$;
- *Proportional Cache Allocation*: The allocated cache sizes among the BSs satisfy that $(F - C_n)/\log\left(1 + \frac{P\mathrm{Tr}(\mathbf{K}_n)}{N\sigma^2}\right)$ are equalized for all n;
- *Lower Bound*: We solve problem (1.13) to obtain the cache sizes by treating $\{C_n\}$ as the optimization variables for each channel realization. This is not practical, but can serve as a lower bound for the minimum expected file downloading time;
- *Rank-One Multicast Beamformer*: The cache sizes are the same as the optimized scheme, but with the multicast beamformer being rank-one obtained using eigenvector decomposition.

In Table 1.1, we show the cache size allocation obtained by different schemes under normalized total cache size constraint $C = 1$. It can be seen that the

Table 1.1 Cache allocation for different schemes under normalized total cache size $C = 1$.

Schemes BSs	Uniform	Proportional	Optimized
BS1	0.2	0.232	0.222
BS2	0.2	0.170	0.071
BS3	0.2	0.261	0.588
BS4	0.2	0.175	0.101
BS5	0.2	0.163	0.019

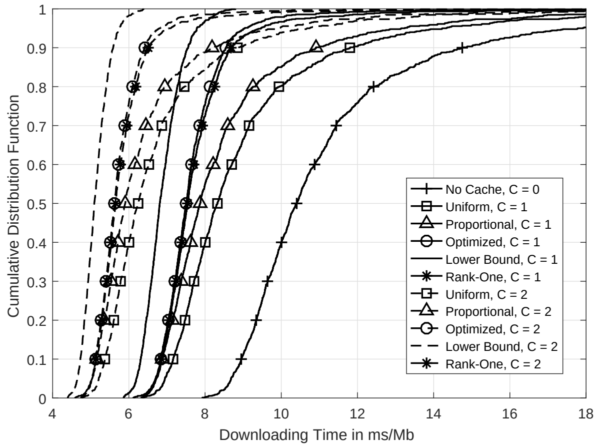

Figure 1.5 CDF of downloading time under different caching schemes.

proposed caching scheme and the proportional caching scheme allocate more cache size to the weaker BS 3 comparing with the uniform caching scheme, but our scheme is more aggressive. In Fig. 1.5, we compare the cumulative distribution function (CDF) of the downloading time between different caching schemes. From Fig. 1.5, we first see that the proposed caching scheme is superior to all the benchmark schemes in the high downloading time regime. It is also seen that the performance loss of the rank-one multicast beamformer is negligible compared to the solution obtained by solving (1.13).

Cache Allocation for Files of Varying Popularity

In this part, we show simulation results for the cache size allocation schemes with multiple files and different popularities. We first consider only two files with the request probabilities being (p_1, p_2) shown in the first row of Table 1.2. Each column denotes the cache size allocation of the BSs under different file popularities given in the first row. From Table 1.2 we first see that for different file popularities, the cache size of the weakest BS 3 is always the largest, as in the

Table 1.2 Optimized cache allocation for a 2-file case with different file popularities under $C = 1$.

BSs	File Popularity (p_1, p_2)				
	$(0.5, 0.5)$	$(0.6, 0.4)$	$(0.7, 0.3)$	$(0.8, 0.2)$	$(0.9, 0.1)$
BS1	$(0.082, 0.082)$	$(0.132, 0.027)$	$(0.168, 0)$	$(0.202, 0)$	$(0.222, 0)$
BS2	$(0, 0)$	$(0, 0)$	$(0, 0)$	$(0.046, 0)$	$(0.071, 0)$
BS3	$(0.418, 0.418)$	$(0.482, 0.359)$	$(0.536, 0.27)$	$(0.568, 0.109)$	$(0.588, 0)$
BS4	$(0, 0)$	$(0, 0)$	$(0.026, 0)$	$(0.075, 0)$	$(0.101, 0)$
BS5	$(0, 0)$	$(0, 0)$	$(0, 0)$	$(0, 0)$	$(0.018, 0)$
Total	$(0.5, 0.5)$	$(0.614, 0.386)$	$(0.73, 0.27)$	$(0.891, 0.109)$	$(1, 0)$

single file case shown in Table 1.1. We also seen that the more popular a file is, the more cache size it will be allocated. For example, when $(p_1, p_2) = (0.9, 0.1)$, file 1 occupies all the cache space without caching any fraction of file 2.

In Fig. 1.6, we compare the file downloading time of the optimized cache scheme with the following benchmarks:

- *Uniform Cache Allocation*: All the files has the same cache size of $C_{nf} = C/NF$ at all the BSs;
- *Proportional Cache Allocation*: The total allocated cache size of file f is first set as $p_f C$. The cache size of this file is then obtained according to the *Proportional Cache Allocation* scheme in the single file case;
- *Caching the Most Popular File*: We cache the most popular file in its entirety first, followed by caching of the second most popular file, and so on. When the remaining cache space is not enough for caching a whole file, we allocate the remaining cache space according to the *Proportional Cache Allocation* scheme.

In Fig. 1.6, we consider $F = 4$ files and assume the file popularity follows the the Zipf distribution [20], i.e., $p_f = \frac{f^{-\alpha}}{\sum_{i=1}^{F} i^{-\alpha}}, \forall\, f$. We compare the average downloading time of all the schemes with different α. Note that when α increases, the differences among the file popularities also increase. From Fig. 1.6, it can be seen that for all schemes, except the uniform scheme, the average downloading time decreases when α increases. This is expected, since in uniform cache allocation scheme, the cache sizes of all files are the same, the downloading time is the same for all files. While in other three schemes, more cache sizes are allocated to the files with larger popularities. We can also see that the proposed caching scheme outperforms other three schemes for different α, and it converges to the scheme of caching the most popular file when $\alpha = 1.5$.

To sum up, from the above simulation results and discussions, it is benefit to allocate more cache sizes to the files with larger popularities and our proposed

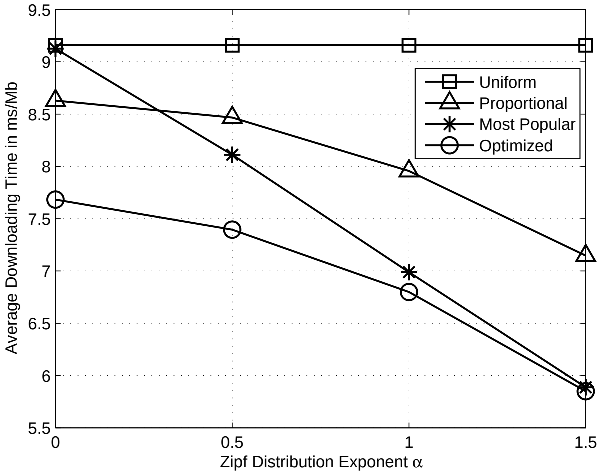

Figure 1.6 Average downloading time for different Zipf file distributions under the same number of files $F = 4$ and the total normalized cache size $C = 4$.

cache allocation scheme can provide a better cache allocation solution compared to the heuristic schemes.

1.5 Conclusions and Open Issues

This chapter presents a content-centric framework for transmission optimization in cloud RANs by leveraging caching and multicasting. We first study the effects of caching and multicasting on the access link in a cloud RAN with dedicated backhaul through the joint design of the content-centric BS clustering and multicast beamforming under different but given BS caching strategies. Simulation results show that our proposed content-centric multicast transmission is much superior to the traditional user-centric unicast transmission in terms of system total transmit power reduction and backhaul saving. We then study the effects of caching and multicasting on backhaul link in a cloud RAN with wireless backhaul through the joint design of cache size allocation at the BSs and the multicast beamforming at the CP. Numerical results show the optimized cache size allocation scheme can greatly improve the network performance comparing with other heuristic schemes.

To exploit the full potential of cache-enabled cloud RAN, it is worthwhile to investigate the joint design of access and backhaul links in the future. It is also of practical importance to seek a scalable solution for caching and multicasting in a large scale of cache-enabled cloud RAN.

References

[1] P. Rost, C. Bernardos, A. Domenico, M. Girolamo, M. Lalam, A. Maeder, D. Sabella, and D. Wübben, “Cloud technologies for flexible 5G radio access networks,” *IEEE Commun. Mag.*, vol. 52, no. 5, pp. 68–76, May 2014.

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

[3] D. Lecompte and F. Gabin, “Evolved multimedia broadcast/multicast service (eMBMS) in LTE-advanced: overview and rel-11 enhancements,” *IEEE Commun. Mag.*, vol. 50, no. 11, pp. 68–74, Nov. 2012.

[4] N. D. Sidiropoulos, T. N. Davidson, and Z.-Q. Luo, “Transmit beamforming for physical-layer multicastings,” *IEEE Trans. Signal Process.*, vol. 54, no. 6, pp. 2239–2251, Jun. 2006.

[5] B. Dai and W. Yu, “Sparse beamforming and user-centric clustering for downlink cloud radio access network,” *IEEE Access*, vol. 2, pp. 1326–1339, 2014.

[6] L. Breslau, P. Cao, L. Fan, G. Phillips, and S. Shenker, “Web caching and Zipf-like distributions: evidence and implications,” in *Proc. IEEE INFOCOM*, 1999, pp. 126–134.

[7] M. Tao, E. Chen, H. Zhou, and W. Yu, “Content-centric sparse multicast beamforming for cache-enabled cloud RAN,” *IEEE Trans. Wireless Commun.*, vol. 15, no. 9, pp. 6118–6131, Sep. 2016.

[8] X. Xu and M. Tao, “Modeling, analysis, and optimization of coded caching in small-cell networks,” *IEEE Trans. Communications*, vol. 65, no. 8, pp. 3415–3428, Aug. 2017.

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

[10] B. Dai, Y.-F. Liu, and W. Yu, “Optimized base-station cache allocation for cloud radio access network with multicast backhaul,” *IEEE J. Sel. Areas Commun.*, vol. 36, no. 8, 2018.

[11] S. S. Bidokhti, M. A. Wigger, and R. Timo, “Noisy broadcast networks with receiver caching,” *IEEE Trans. Inf. Theory*, 2018. [Online]. Available: http://arxiv.org/abs/1605.02317

[12] A. F. Molisch, G. Caire, D. Ott, J. R. Foerster, D. Bethanabhotla, and M. Ji, “Caching eliminates the wireless bottleneck in video aware wireless networks,” *Adv. Elect. Eng.*, vol. 2014, 2014.

[13] Z. Xiang, M. Tao, and X. Wang, “Coordinated multicast beamforming in multicell networks,” *IEEE Trans. Wireless Commun.*, vol. 12, no. 1, pp. 12–21, Jan. 2013.

[14] E. Karipidis, N. Sidiropoulos, and Z.-Q. Luo, “Quality of service and max-min fair transmit beamforming to multiple cochannel multicast groups,” *IEEE Trans. Signal Process.*, vol. 56, no. 3, pp. 1268–1279, Mar. 2008.

[15] A. L. Yuille and A. Rangarajan, “The concave-convex procedure,” *Neural Comput.*, vol. 15, no. 4, pp. 915–936, 2003.

[16] E. Chen and M. Tao, “User-centric base station clustering and sparse beamforming for cache-enabled cloud RAN,” in *Proc. IEEE/CIC ICCC*, Nov. 2015, pp. 1–6.

[17] J. R. Birge and F. Louveaux, *Introduction to Stochastic Programming*, 2nd ed. Springer, 2011.

[18] A. R. Conn, N. I. Gould, and P. L. Toint, *Trust Region Methods.* Society for Industrial and Applied Mathematics (SIAM), 2000.

[19] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, “Distributed optimization and statistical learning via the alternating direction method of multipliers,” *Found. Trends Mach. Learn.*, vol. 3, no. 1, pp. 1–122, 2011.

[20] M. Zink, K. Suh, Y. Gu, and J. Kurose, “Characteristics of YouTube network traffic at a campus network–Measurements, models, and implications,” *Comput. Netw.*, vol. 53, no. 4, pp. 501–514, 2009.