Cloud Radio Access Networks With Coded Caching

Yiğit Uğur, Zohaib Hassan Awan and Aydin Sezgin
Institute of Digital Communication Systems
Ruhr-Universität Bochum, 44780, Germany
Email: {yigit.ugur, zohaib.awan, aydin.sezgin}@rub.de

Abstract—A cloud radio access network (C-RAN) is considered as a candidate to meet the expectations of higher data rate demands in wireless networks. In C-RAN, low energy base stations (BSs) are deployed over a small geography and are allowed to connect to the cloud via finite capacity backhaul links where the information is processed. A conventional C-RAN, however, requires high capacity backhaul links, since the requested files need to be transferred first from the cloud to the BS before conveying them to the users. One approach to overcome the limitations of the backhaul links is to introduce local storage caches at the BSs, in which the popular files are stored locally in order to reduce the load of the backhaul links. Furthermore, we utilize coded caching with the goal to minimize the total network cost, i.e., the transmit power and the cost associated with the backhaul links. The initial formulation of the optimization problem for this model is non-convex. We first reformulate and then convexify the problem through some relaxation techniques. In comparison to the uncoded caching at the BSs, our results highlight the benefits associated with coded caching and show that it decreases the backhaul cost.

I. INTRODUCTION

Mobile data traffic has experienced explosive growth in the last decade and is expected to increase further. Capabilities for supporting high data traffic and extremely low latency are some fundamental requirements identified for the future 5G mobile networks. A C-RAN model can be considered as a potential candidate to meet these requirements [1]. In C-RAN, due to economical reasons instead of a high power BS, several low power BSs are spread over a small area, which are allowed to connect to the cloud via backhaul links. High speed fiber cables are generally used to establish backhaul links for low latency. One drawback is that establishing all the links via fiber optical cables from the cloud to the BSs may lead to high infrastructure cost. A possible alternative to save on cost is by utilizing finite capacity wireless backhaul links. However, this may lead to higher latency. A potential solution to solve this issue is by introducing local storage or caches at the BSs [2], where the most popular contents, for example, files, movies, or multimedia, are stored, locally. When a user requests a file which is available at the cache of the BS, it can be directly downloaded from the BS. In other words, this approach reduces the load of backhaul links with relatively inexpensive local storage.

In general, the cache placement schemes can be roughly partitioned into two types, namely, uncoded and coded caching [3], [4]. In uncoded caching, BSs store the complete file; while, in coded caching, instead of complete file, small fractions of the files are stored at the caches using fountain or maximum distance separable (MDS) codes [3], [5], [6]. The main advantage of the coded caching compared to the uncoded caching is that the probability of re-constructing the file at the desired receiver without using the backhaul links is higher. Subsequently, it reduces the backhaul cost. In addition to this, if the users still need to use backhaul links to recover the files — for example, when the portion of the file required by the users are not available in the cache — only remaining portion of the files will be transferred from the cloud [4]; and, under secrecy constraints in [7]. In [8], minimizing the backhaul and transmission power cost is considered where uncoded caching is used at the BSs. This optimization problem is formalized with the help of semidefinite programming (SDP) relaxation and l_0-norm approximation. In [9], [10], the authors studied a C-RAN in the absence of caching and study the problem of resource allocation.

In this work, we consider a C-RAN model as shown in Fig. 1, where the BSs are equipped with multiple antennas. The users can download the files either locally from the BSs or from the central storage at the cloud via backhaul links. The choice of beamforming vectors and how to appropriately fill the caches are decided in the central processor at the cloud. The main goal of this work is to minimize the downlink network cost of the aforementioned C-RAN system. We refer to the downlink network cost as the sum of the total transmission power and the backhaul cost. We optimize the downlink cost over beamforming vectors such that each user has a minimum quality of service (QoS). In this work, we consider the case where coded caching at the BSs is used. Users that request the file, receive multiple fractions of the file from different caches of the BSs and try to reconstruct the desired file. The initial optimization problem consists of a non-convex indicator function and a parameter that models the missing fractions of the files at the serving BSs. We first reformulate the original problem by using a maximum function, then solve it with the help of SDP relaxation. The simulation results show that the coded caching outperforms the uncoded caching strategies in a variety of situations.

We structure this paper as follows. The system model is described in Section II. The optimization problem is formally presented in Section III. Simulation results are presented in Section IV. Finally, in Section V, we conclude the paper by summarizing its contributions.

Notation: Bold small and capital letters denote vector and matrices, respectively. The superscripts $(\cdot)^{\mathrm{H}}$ and $(\cdot)^{-1}$ repre-

sent the Hermitian transpose and the matrix inverse, respectively. The function $\{x\}^+$ denotes the maximum value between 0 and x. The l_p-norm is denoted by $\|\cdot\|_p$. The expectation is denoted by $E[\cdot]$, the trace operation is represented by $\mathrm{tr}(\cdot)$ and $\mathbb{C}^{x \times y}$ denotes the space of $x \times y$ complex matrices. For a square matrix $\mathbf{A}$, we use $\mathbf{A} \succeq \mathbf{0}$ to indicate that $\mathbf{A}$ is positive semidefinite. The zero vector of size $N \times 1$ is represented by $\mathbf{0}_N$.

II. SYSTEM MODEL AND PROBLEM FORMULATION

We consider a C-RAN composed of L BSs and M users. Each BS is equipped with N_t transmit antennas and each user has a single antenna as shown in Fig. 1. All the BSs are connected to the cloud via finite capacity backhaul links; and, there are no direct links between the users and the central cloud storage. In addition to the central file storage, local storage called cache are available at each BS. The caches have a smaller storage capacity compared to the cloud and are thus able to store only a limited portion of files. We focus on the downlink phase of C-RAN in which the users request files. The files can be delivered in two ways, 1) direct transmission from the BSs — if they are available locally in their cache, 2) by prefetching the files from the cloud to the BSs and then transmitting it to the users.

The channel vector from the l-th BS to the m-th user is denoted by $\mathbf{h}_{l,m} \in \mathbb{C}^{N_t \times 1}$. The overall channel vector from all the BSs to the m-th user can be concisely given by $\mathbf{h}_m = [\mathbf{h}_{1,m}^H, \mathbf{h}_{2,m}^H, \ldots, \mathbf{h}_{L,m}^H]^H \in \mathbb{C}^{LN_t \times 1}$. We assume that the channel is constant and do not change in a time slot. The channel state information is known to all nodes, globally. The received signal at the m-th user can be expressed as

$$y_m(t) = \mathbf{h}_m^H \mathbf{w}_m(t) s_m(t) + \sum_{j \neq m}^{M} \mathbf{h}_m^H \mathbf{w}_j(t) s_j(t) + z_m(t),$$
$$m = 1, \ldots, M, \quad 1 \leq t \leq T \quad (1)$$

where $\mathbf{w}_m(t) = [\mathbf{w}_{1,m}^H(t), \mathbf{w}_{2,m}^H(t), \ldots, \mathbf{w}_{L,m}^H(t)]^H \in \mathbb{C}^{LN_t \times 1}$ is the beamforming vector from all the BSs to the m-th user, $\mathbf{w}_{l,m}(t)$ is the beamforming vector from the l-th BS to the m-th user, $s_m(t)$ is the desired data symbol for the m-th user and $z_m(t)$ denotes the independent identically distributed additive complex Gaussian noise, with zero mean and variance σ^2 at users, $\forall$ m. Note that T is the number of channel uses needed to fulfill the demands of the users. For convenience, we skip the time index t in the rest of the paper. Without loss of generality, we assume that $E[|s_m|^2] = 1$ and s_m's are chosen independently from each other. By treating the contribution of undesired signals at the m-th user as noise, the corresponding signal-to-interference-plus-noise ratio (SINR) can be written as

$$\mathrm{SINR}_m = \frac{|\mathbf{h}_m^H \mathbf{w}_m|^2}{\sum_{j \neq m}^{M} |\mathbf{h}_m^H \mathbf{w}_j|^2 + \sigma^2}, \quad m = 1, \ldots, M. \quad (2)$$

We assume that each user has a minimum SINR requirement that needs to be fulfilled to satisfy a particular quality of
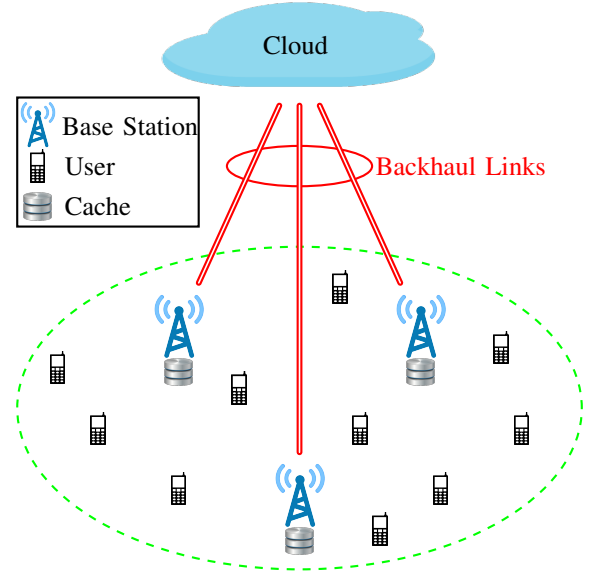

Fig. 1. System architecture of a C-RAN with local cache storage.

service constraint. We refer to the target SINR for the m-th user as γ_m.

A. Files Placement in the Caches

Since the storage capacity of caches is lower than that of the central cloud, how to fill them is of paramount importance. The main idea is to only store the popular contents in the caches, so as to avoid the data transfer cost that arises from the backhaul link. The popularity of the files can be measured based on the number and behavior of the requests; and, is modeled here as Zipf distribution [2]. We assume that the files are sorted from the most to the least popular ones, such that the most popular file has index 1 ($f = 1$) and the least popular file has index F ($f = F$). According to Zipf distribution, the probability of f-th file requested by the user is given as

$$Z_f = \frac{f^{-\alpha}}{\sum_{k=1}^{F} k^{-\alpha}}, \quad f = 1, \ldots, F \quad (3)$$

where α is the shaping parameter and is related to the skewness of the distribution [11]. A large value of α means that the distribution is more skewed and therefore the probability of requesting a small group of files is large. We assume that the popularity distribution is known at the cloud.

We define the cache placement matrix $\mathbf{R} \in [0,1]^{F \times L}$, where the f-th row and the l-th column of $\mathbf{R}$ is $\delta_{f,l}$, which corresponds to the fraction of the parity bits of the f-th file that is stored in the cache of the l-th BS. We denote the storage capacity of each cache as BS bits, where B is the size of each file. Due to the cache size constraint,

$$\sum_{f=1}^{F} \delta_{f,l} = BS \quad (4)$$

needs to be satisfied at the l-th BS, $\forall$ l [1]. The cache placement

---

[1] For convenience, in the rest of the paper, we assume that all files have normalized size ($B = 1$).

matrix $\mathbf{R}$ is fixed. This is due to the fact that, the popularity of files changes slowly in comparison to the channel variations. As mentioned before, MDS or fountain codes can be used to encode files which are then stored in the caches of the BSs. By using an ideal MDS code, a file is encoded into an arbitrary long number of parity bits, such that after recovering any of B parity bits, the receiver is able to recover the desired message. We assume that the cache at each BS stores a unique subset of parity bits. Then, the m-th user can download the f-th file if the following condition is fulfilled

$$\sum_{l \in \mathcal{H}(m)} \delta_{f,l} \geq 1 \tag{5}$$

where $\mathcal{H}(m)$ is the set of the BSs that serve the m-th user. If all elements of the beamforming vector from the l-th BS to the m-th user equals zero ($\mathbf{w}_{l,m} = \mathbf{0}_{N_t}$), i.e., the input power of beamforming vector $\mathbf{w}_{l,m}$ is zero, the l-th BS does not serve the m-th user. Subsequently, (5) can be rewritten as

$$\sum_{l=1}^{L} \delta_{f,l} \|\|\mathbf{w}_{l,m}\|_2^2\|_0 \geq 1 \tag{6}$$

where l_0-norm gives the number of nonzero entries in a vector. For notational simplicity, in the rest of the paper, we will use $\|\cdot\|_{2,0}$ to replace $\|\|\cdot\|_2^2\|_0$.

B. Network Cost Model

We define the total network cost of the C-RAN as the sum of backhaul and power consumption cost. The overall network cost C_N can be written as

$$C_N = \lambda C_{BH} + (1 - \lambda) C_P \tag{7}$$

where C_{BH} and C_P denote the backhaul cost and the power consumption cost, respectively. The parameter λ indicates the relative emphasis on the backhaul cost and the power consumption cost, where $\lambda \in [0, 1]$. Starting from (7), the overall network cost can be written as

$$C_N = \lambda \underbrace{\sum_{m=1}^{M} \sum_{f=1}^{F} Z_f \mathbf{1}_{m,f} C_{BH}^{(m,f)}}_{:=C_{BH}} + (1 - \lambda) \underbrace{\sum_{m=1}^{M} \|\mathbf{w}_m\|_2^2}_{:=C_P} \tag{8}$$

where $\mathbf{1}_{m,f}$ is the indicator function and is given by

$$\mathbf{1}_{m,f} = \begin{cases} 0, & \sum_{l=1}^{L} \delta_{f,l} \|\mathbf{w}_{l,m}\|_{2,0} \geq 1 \\ 1, & \text{otherwise} \end{cases} \tag{9}$$

and $C_{BH}^{(m,f)}$ is the associated backhaul cost for the case in which the f-th file required by the m-th user can not be provided by the BSs. We define $C_{BH}^{(m,f)}$ as

$$C_{BH}^{(m,f)} = X_{m,f} R_m \tag{10}$$

where $X_{m,f}$ denotes the missing portion of the f-th file not present in the serving caches of the m-th user. This fraction of the f-th file needs to be transfered via backhaul. The transmission rate R_m for the m-th user is given by $R_m = \log_2(1 + \gamma_m)$.

III. Optimization Problem

In this section, we study the network cost problem stated in (8). The goal is to minimize the total network cost and in doing so, we need to optimize the beamforming vectors such that 1) the quality of service constraints (in this case minimum SINR targets) of each user is fulfilled, and 2) the maximum transmit power constraints at each BS is met. Thus, we can formulate our optimization problem as follows

$$\min_{\mathbf{w}_m} \quad \lambda \sum_{m=1}^{M} \sum_{f=1}^{F} Z_f \mathbf{1}_{m,f} X_{m,f} R_m + (1 - \lambda) \sum_{m=1}^{M} \|\mathbf{w}_m\|_2^2 \tag{11a}$$

$$\text{s. t.} \quad \text{SINR}_m \geq \gamma_m, \quad \forall m \tag{11b}$$

$$\sum_{m=1}^{M} \|\mathbf{w}_{l,m}\|_2^2 \leq P_{\max}, \quad \forall l \tag{11c}$$

where $P_{\max}$ is the maximum transmission power of each BS. Note that the indicator function makes the objective non-convex. Starting from (11), the optimization problem can be identically expressed as

$$\min_{\mathbf{w}_m} \quad \lambda \sum_{m=1}^{M} \sum_{f=1}^{F} Z_f \left\{ 1 - \sum_{l=1}^{L} \delta_{f,l} \|\mathbf{w}_{l,m}\|_{2,0} \right\}^+ R_m$$
$$+ (1 - \lambda) \sum_{m=1}^{M} \|\mathbf{w}_m\|_2^2 \tag{12a}$$

$$\text{s. t.} \quad (11b), (11c). \tag{12b}$$

In (12), the parameter $X_{m,f}$ and the indicator function $\mathbf{1}_{m,f}$ from (11a) are condensed into the function $\{\cdot\}^+$. Next, we reformulate the objective function (12a) by introducing an auxiliary parameter $\beta_{f,m}$, as

$$\min_{\mathbf{w}_m, \beta_{f,m}} \quad \lambda \sum_{m=1}^{M} \sum_{f=1}^{F} Z_f \beta_{f,m} R_m + (1 - \lambda) \sum_{m=1}^{M} \|\mathbf{w}_m\|_2^2 \tag{13a}$$

$$\text{s. t.} \quad \beta_{f,m} \geq 0, \quad \forall f, \forall m \tag{13b}$$

$$1 - \sum_{l=1}^{L} \delta_{f,l} \|\mathbf{w}_{l,m}\|_{2,0} \leq \beta_{f,m}, \quad \forall f, \forall m \tag{13c}$$

$$(11b), (11c). \tag{13d}$$

Due to the presence of quadratic terms in (2), the SINR constraints in (11b) render the problem non-convex. We solve this issue by semidefinite programming relaxation to convexify the SINR constraints. Let $\mathbf{W}_m$ and $\mathbf{H}_m$ be given by

$$\mathbf{W}_m = \mathbf{w}_m \mathbf{w}_m^H \tag{14}$$

$$\mathbf{H}_m = \mathbf{h}_m \mathbf{h}_m^H \tag{15}$$

where $\mathbf{W}_m$ and $\mathbf{H}_m$ are positive semidefinite matrices. Then, $\|\mathbf{w}_{l,m}\|_2^2$ in (6) and (13c) can be concisely written as

$$\|\mathbf{w}_{l,m}\|_2^2 = \text{tr}(\mathbf{W}_m \mathbf{J}_l) \tag{16}$$

where $\mathbf{J}_l$ is selection matrix [8] such that

$$\mathbf{J}_l = \text{diag}\left(\left[\mathbf{0}_{(l-1)N_t}^H, \mathbf{1}_{N_t}^H, \mathbf{0}_{(L-l)N_t}^H \right] \right). \tag{17}$$

Algorithm 1 Randomization for (19)

1: Initialize the minimum objective value
2: **repeat**
3:      Generate random vectors $\mathbf{w}_m$ according to complex Gaussian distributions $\mathbf{w}_m \sim \mathcal{N}(\mathbf{0}, \mathbf{W}_m^*)$
4:      Check whether constraints (11b) and (11c) satisfies
5:      **if** all constraints satisfy **then**
6:          Plug in the $\mathbf{w}_m$ into the objective function (11a)
7:          Compare the current minimum objective value to the previous one and retain the smaller one
8:      **end if**
9: **until** number of iterations are reached

Fig. 2. Network geometry for the numerical example.

After the aforementioned SDP relaxation, the problem is still non-convex because of the presence of the l_0-norm in the constraints (13c). The l_0-norm can be approximated by the logarithmic function [12] as follows

$$\|\mathbf{X}\|_0 = \log(\mathrm{tr}(\mathbf{X}) + \theta) \tag{18}$$

where θ is a parameter to adjust the smoothness, $0 < \theta \leq 1$. Finally, after the SDP relaxation the optimization problem in (13) can be written as

$$\min_{\mathbf{W}_m, \beta_{f,m}} \lambda \sum_{m=1}^{M} \sum_{f=1}^{F} Z_f \beta_{f,m} R_m + (1-\lambda) \sum_{m=1}^{M} \mathrm{tr}(\mathbf{W}_m) \tag{19a}$$

s. t.

$$\beta_{f,m} \geq 0, \quad \forall f, \forall m \tag{19b}$$

$$1 - \sum_{l=1}^{L} \delta_{f,l} \log\left(\mathrm{tr}(\mathbf{W}_m \mathbf{J}_l) + \theta\right) \leq \beta_{f,m}, \quad \forall f, \forall m \tag{19c}$$

$$\frac{\mathrm{tr}(\mathbf{W}_m \mathbf{H}_k)}{\sum_{n \neq m}^{M} \mathrm{tr}(\mathbf{W}_n \mathbf{H}_k) + \sigma^2} \geq \gamma_m \quad \forall m, \forall k \tag{19d}$$

$$\sum_{m=1}^{M} \mathrm{tr}(\mathbf{W}_m \mathbf{J}_l) \leq P_{\max}, \quad \forall l \tag{19e}$$

$$\mathbf{W}_m \succeq \mathbf{0}, \quad \forall m. \tag{19f}$$

Let $\mathbf{W}_m^*$ denote the optimum solution of (19). If $\mathbf{W}_m^*$ is rank-one, we can readily apply eigenvalue decomposition to get the optimum beamforming vectors. Otherwise, randomization methods [13], [14] can be used to obtain the beamforming vectors from the optimum solution $\mathbf{W}_m^*$. Algorithm 1 concisely captures the steps for the randomization methods.

IV. Simulation Results

In this section, we illustrate the usefulness of the proposed optimization method with the help of following example. We considered a network geometry as shown in Fig. 2. The distance between adjacent BSs is fixed to 0.8 km such that 7 BSs form an equilateral triangular lattice ($L = 7$). We assume that 200 users are uniformly distributed within a circular network of radius 0.8 km and in each time slot 12 of them are served ($M = 12$). The ta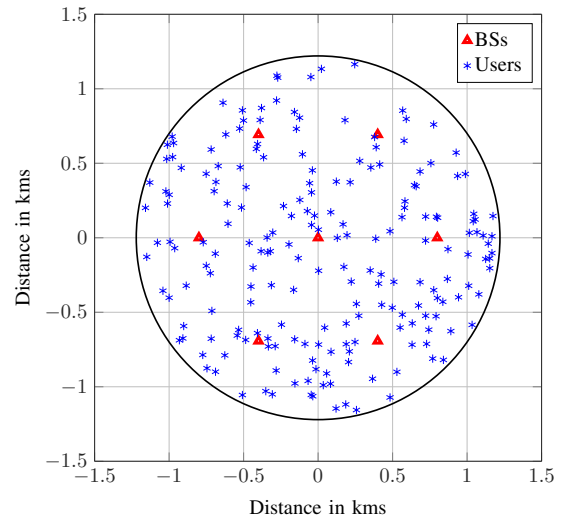rget SINR of each user is $\gamma_m = 10$ dB, $\forall m$. Each BS has two antennas ($N_t = 2$) and each user has a single antenna. Each BS has the same cache size S. The channel vector from the l-th BS to the m-th user is given by [15],

$$\mathbf{h}_{l,m} = \mathbf{g}_{l,m} \sqrt{10^{-\frac{\rho_{l,m}}{10}} \varphi \zeta_{l,m}} \tag{20}$$

where $\varphi = 10$ dBi is the transmit antenna power gain of each BS, $\rho_{l,m}$ is the path loss at the distance $d_{l,m}$, $\zeta_{l,m}$ is the log-normal coefficient — with a 0 dB mean and standard deviation of 8 dB — that models the large-scale fading (shadowing), and $\mathbf{g}_{l,m}$ is the complex Gaussian coefficient with variance 1 that models the small-scale fading. The channel vector changes independently at each time slot. For the path loss, the 3GPP Long Term Evolution (LTE) path loss model [16] with a path-loss exponent of 3.76 is used

$$\rho_{l,m} = 148.1 + 37.6 \log_{10}(d_{l,m}) \tag{21}$$

where $d_{l,m}$ is the distance between the l-th BS and m-th user in kms. All users are subject to white Gaussian noise with power spectral density -172 dBm/Hz over 10 MHz bandwidth. The total number of files available in the cloud is 20 ($F = 20$). We investigate the costs for both coded and uncoded caching for different cache sizes. As mentioned in the introduction, in the case of uncoded caching only complete files are stored at the BSs; and, thus each BS is able to store S ($BS/B = S$) files. As opposed to the uncoded caching case, in the coded caching setting, we assume that $0.5B$ parity bits of a particular file is stored in the caches of the BSs. This implies that, each BS stores the parity bits of $2S$ ($BS/0.5B = 2S$) files. For the simulation, the caches of the BSs are filled with the S most popular files for the uncoded caching and parity bits of the $2S$ most popular files for the coded caching, respectively. Note that each BS stores distinct parity bits of a particular file. The Zipf distribution parameter α is chosen as 1.2 in the simulations. The optimization software tool CVX [17] is used
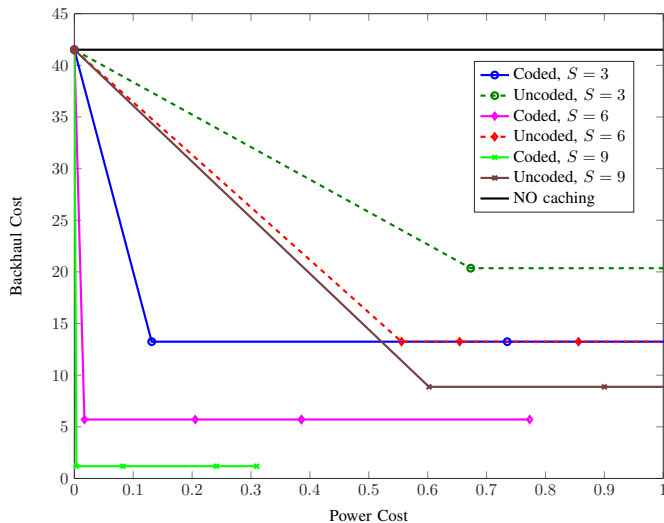
Fig. 3. Network cost trade-off.

	Coded vs. No caching	Uncoded vs. No caching	Coded vs. Uncoded
$S = 3$	68.1%	50.9%	34.9%
$S = 6$	86.2%	68.1%	56.9%
$S = 9$	97.1%	78.6%	86.6%

sized caches at BSs decreases the backhaul cost significantly. In comparison to the uncoded caching strategies studied in the existing literature, we show that coded caching schemes reduce the overall network backhaul.

to obtain the optimum beamforming vectors and the results are obtained by averaging over 100 time slots.

Fig. 3 shows the power consumption and backhaul costs trade-off for uncoded and coded caching with different cache sizes ($S = 3, 6, 9$). Data points of each line are obtained by changing the weighting parameter λ. As it can be seen from Fig. 3, the backhaul cost saturates if BSs transmit with a required power. For different cache sizes, Table I summarizes the decrease in the backhaul costs with the coded caching in comparison to uncoded and no caching. For sake of completeness, we also compare the uncoded caching with no caching. We observe that coded caching always outperforms the uncoded caching cases in terms of low backhaul cost for the same cache size. This is due to the fact that since more portions of files can be stored at the BSs, files can be downloaded directly from the BSs. It is observed that the coded caching decreases the backhaul cost approximately 86% compared to the uncoded caching for the cache size 9 ($S = 9$). We also compare the benefits of the coded caching over no caching case in Table I. By introducing a cache size of 3 ($S = 3$), the backhaul cost decreases by 68.1%. An additional decrease of 18.1% in backhaul cost is observed by doubling cache size from 3 to 6 ($S = 6$). Furthermore, only 10.9% additional decrease is obtained when the cache size is increased from 6 to 9 ($S = 9$). A similar trend is also observed for the uncoded case. Introducing even a small size of storage at the BSs cause a substantial decrease in the backhaul cost.

V. CONCLUSION

In this work, we consider a C-RAN model in which the BSs are connected to the cloud via finite capacity backhaul links and where local storage is also available at the BSs. We investigate this problem by minimizing the total network cost under quality of service constraints, where coded caching is allowed at the BSs. We show that even introducing small

REFERENCES

[1] China Mobile, "C-RAN: the road towards green RAN," White Paper, ver. 2.5, China Mobile Research Institute, Oct. 2011.
[2] N. Golrezaei, A. F. Molisch, A. G. Dimakis, and G. Caire, "Femtocaching and device-to-device collaboration: a new architecture for wireless video distribution," *IEEE Commun. Mag.*, vol. 51, no. 4, pp. 142–149, Apr. 2013.
[3] 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.
[4] M. A. Maddah-Ali and U. Niesen, "Fundamental limits of caching," *IEEE Trans. Inf. Theory*, vol. 60, no. 5, pp. 2856–2867, May 2014.
[5] A. Liu and V. K. Lau, "Mixed-timescale precoding and cache control in cached MIMO interference network," *IEEE Trans. Signal Process.*, vol. 61, no. 24, pp. 6320–6332, Dec. 2013.
[6] A. Liu and V. K. N. Lau, "Cache-enabled opportunistic cooperative MIMO for video streaming in wireless systems," *IEEE Trans. Signal Process.*, vol. 62, no. 2, pp. 390–402, Jan. 2014.
[7] Z. H. Awan and A. Sezgin, "Fundamental limits of caching in D2D networks with secure delivery," in *Proc. IEEE International Conference on Communication Workshop*, Jun. 2015, pp. 464–469.
[8] H. Zhou, M. Tao, E. Chen, and W. Yu, "Content-centric multicast beamforming in cache-enabled cloud radio access networks," Aug. 2015. [Online]. Available: http://arxiv.org/abs/1504.05663
[9] Y. Shi, J. Zhang, and K. Letaief, "Group sparse beamforming for green cloud-RAN," *IEEE Trans. Wireless Commun.*, vol. 13, no. 5, pp. 2809–2823, May 2014.
[10] S. Luo, R. Zhang, and T. J. Lim, "Downlink and uplink energy minimization through user association and beamforming in C-RAN," *IEEE Trans. Wireless Commun.*, vol. 14, no. 1, pp. 494–508, Jan. 2015.
[11] N. Golrezaei, P. Mansourifard, A. Molisch, and A. Dimakis, "Basestation assisted device-to-device communications for high-throughput wireless video networks," *IEEE Trans. Wireless Commun.*, vol. 13, no. 7, pp. 3665–3676, Jul. 2014.
[12] F. Rinaldi, F. Schoen, and M. Sciandrone, "Concave programming for minimizing the zero-norm over polyhedral sets," *Comput. Optim. Appl.*, vol. 46, no. 3, pp. 467–486, Jul. 2010.
[13] H. Dahrouj and W. Yu, "Multicell interference mitigation with joint beamforming and common message decoding," *IEEE Trans. Commun.*, vol. 59, no. 8, pp. 2264–2273, Aug. 2011.
[14] X. Fan, J. Song, D. P. Palomar, and O. C. Au, "Universal binary semidefinite relaxation for ML signal detection," *IEEE Trans. Commun.*, vol. 61, no. 11, pp. 4565–4576, Nov. 2013.
[15] X. Peng, J.-C. Shen, J. Zhang, and K. Letaief, "Joint data assignment and beamforming for backhaul limited caching networks," in *Proc. IEEE International Symposium on Personal, Indoor, and Mobile Radio Communication*, Sep. 2014, pp. 1370–1374.
[16] C. Botella, T. Svensson, X. Xu, and H. Zhang, "On the performance of joint processing schemes over the cluster area," in *Proc. IEEE Vehicular Technology Conference*, May 2010, pp. 1–5.
[17] M. Grant and S. Boyd, "CVX: Matlab software for disciplined convex programming, version 2.1," Mar. 2014. [Online]. Available: http://cvxr.com/cvx