Adaptive Handover Scheme for Mulit-mode Device in Power Wireless Heterogeneous Networks

Weiping Shao[1], Donglei Zhang[2,3], Yang Lu[2,3], Yazhou Wang[5], Wei Bai[2,3](✉), Ping Ma[4], and Shuiyao Chen[1]

[1] State Grid Zhejiang Electric Power Co., Ltd., Hangzhou, China
[2] Electric Power Intelligent Sensing Technology and Application State Grid Corporation Joint Laboratory, Beijing, China
[3] Global Energy Interconnection Research Institute Co., Ltd., Beijing, China
baiwei@geiri.sgcc.com.cn
[4] State Grid Shaoxing Electric Power Supply Company Shaoxing, Shaoxing, China
[5] Beijing University of Posts and Telecommunications, Beijing, China

Abstract. In smart grid systems, heterogeneous networks are considered as a promising solution to address the expeditious growth of mobile traffic. Considering the different user preferences, how to efficiently utilize the limited resource of small base stations (SBSs) becomes a challenge. In this paper, we investigate a joint handover and transmission strategy for users. We formulate the handover and transmission problem to minimize the service delay, which considers the overlapped coverage of SBSs and the limited capacity of backhaul link. To solve this NP-hard problem, we design a heuristic algorithm with two phases. In content caching phase, the contents are cached at SBSs according to the greedy algorithm. In content delivery phase, a transmission strategy is designed to meet the user demands for videos with different quality level. Simulation results show that our proposed algorithm has the advantage of reducing video delivery delay and saving the backhaul traffic compared with other algorithms.

Keywords: Smart grid networks · Heterogeneous small cell networks · Cooperative caching and transmission · Layered-video caching

1 Introduction

At present, the fifth generation network can be used as a new framework to enhance public utilities and smart grids [1]. However, the increasing traffic has put great pressure on the capacity of smart grid [2]. Small cell networks (SCNs) are one of the promising solutions to meet this increasing demands, where heterogeneous small base stations (SBSs) are densely deployed near users to improve area spectral efficiency and network capacity. However, the cost of deploying high-speed backhaul may hinder the densification of network [3].

S. W. Loke et al. (Eds.): MONAMI 2020, LNICST 338, pp. 270–283, 2020.
https://doi.org/10.1007/978-3-030-64002-6_18

Recently, various research has shown that the demands from users are usually focused on a few popular contents. Therefore, these popular contents can be proactively cached at the SBSs during off-peak hours, reducing the backhaul traffic caused by duplicate transmission [4]. In cache-enabled small cell networks, a key challenge is to determine which contents should be cached at each SBSs to improve the user experience, considering the content popularity and network topology. The works [5,6] analyze the optimal content caching strategy to minimize the download time of files. In work [7], the expected backhaul rate and the energy consumption is minimized via optimizing cache placement.

Furthermore, as the dense deployment of SBSs, cooperative cache has been proposed to efficiently utilize the limited cache capacity of SBSs [8–10]. In cooperative cache, users can be served by multiple nearby SBSs, and these SBSs can cache different contents to improve cache hit rate. In [8,9], the authors design a cooperative content caching and delivery policy in small cell network. In content caching phase, the network hasn't been congested, and each SBSs proactively stores contents according to the caching policy. In content delivery phase, the demands of users have been revealed, and the problem is to decide which place can serve the user requests. The work [10] explores cooperative cache in user-centric mobile networks, where the cache-enabled SBSs in user-centric cluster can cooperatively serve the user requests, and the cluster size shows a trade-off between caching diversity and spectrum efficiency.

Due to the heterogeneity of user devices and the variation of the wireless channel, users may have different preferences for video quality [11]. Scalable video coding (SVC) has been proposed to provide adaptive video streaming service, which allows temporal, spatial, and quality scalability [12]. In SVC, each video can be encoded into multiple layers, and the combination of these layers can achieve different video quality. The layer 1 is also called base layer that provides the lowest video quality. The high video quality can be achieved by adding the layer 2 up to the highest necessary layer. In work [13], the videos encoded by SVC are cached at base station to serve the user with different demands for video quality, and the author also proposes the content caching and transmission schemes for this scalable videos. The work [14] designs a cooperative caching algorithm for video encoding layers, where the multiple network operators can share their cached video layers to reduce the video delivery delay. The work [15] investigates a cooperative caching problem for video encoding layers in heterogeneous network. The author formulates the delay minimization problem and solves it by greedy algorithm.

Although much works have been done on layered video caching, they don't consider the properties of small cell networks, i.e., the overlapped coverage of SBSs and the limited capacity of backhaul link. In this paper, we design a cooperative caching and transmission strategy for video encoding layers, where the user can be served by multiple nearby SBSs and SBSs can cooperate with each other to minimize the video delivery delay. The main contributions of this paper are summarized as follows:

– We investigate the cooperative caching and transmission problem for SVC-encoded videos in small cell networks, and formulate the video delivery delay minimization problem considering the overlapped coverage of SBSs and the limited backhaul capacity.
– The delay minimization problem is NP-hard, we design a greedy caching algorithm to cache the video encoding layers at each SBS. After the content caching phase, we also design a content delivery algorithm to decide which place should serve the demands of users.
– The simulation results shows that our algorithm achieves lower delivery delay and saves the backhaul traffic compared with other algorithm.

The rest of this paper is organized as follows: Section 2 describes the caching and delivery model. Section 3 formulates the delay minimization problem and propose a heuristic algorithm to solve it. Section 4 presents simulation results. Finally, Sect. 5 concludes this paper.

2 System Model

2.1 Network Model

In the heterogeneous smart grid network, a macro base station (MBS) is connected with a remote video server, and M SBSs are deployed densely to serve N wireless users, as shown in Fig. 1. We denote the SBSs set as: $\mathcal{M} = \{1, 2, \ldots, M\}$, and the users set as: $\mathcal{N} = \{1, 2, \ldots, N\}$. The SBSs are connected with MBS via backhaul link, through which the cached contents can be shared to avoid requesting these contents from remote servers. The specific cooperation strategy will explain in the latter.

2.2 Scalable Video Coding and Caching

The SBSs are equipped with a cache of size C_m bits, which is filled in advance during off-peak hours, according to the popularity of the videos. Each user independently requests videos of interest from a library $\mathcal{V} = \{1, 2, \ldots, V\}$, where each video has the same size. We utilize SVC to encode each video content into multiple layers, $\mathcal{L} = \{1, 2, \ldots, L\}$, and the combination of these layers can achieve a certain quality level of corresponding video. For example, layer 1 provider the basic quality level 1, and the quality level 2 can be achieved by adding layer 2, and so on. Typically, the quality level q requires all the lower layer $1, 2, \cdots, q$ successfully decoded. We use L_{vl} to represent the layer l of video v, and V_{vq} to represent the qth quality level of video v. The size of layer L_{vl} is denoted as s_{vl}, which decreases with the increasing of the layer, i.e., $s_{v1} \geq s_{v2} \geq \cdots \geq s_{vL}$.

2.3 Requests and Cooperative Transmission

We assume the probability that users request the qth quality level of video v is known in advance, denoted by p_{vq}, which can be learned by analyzing the historical request patterns of users.

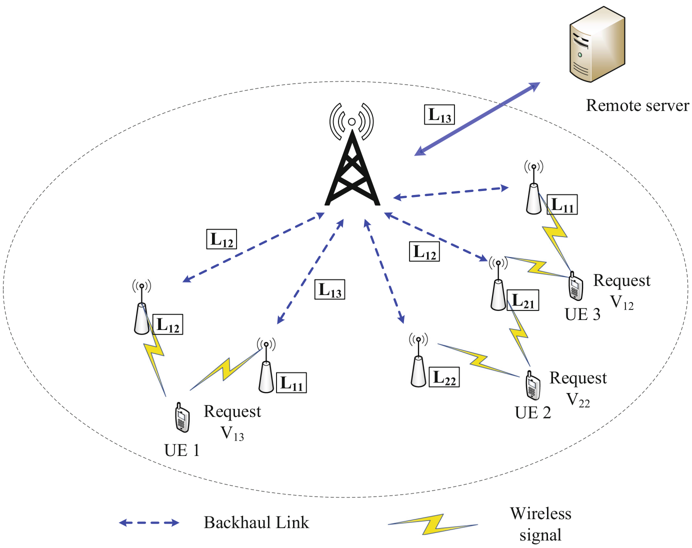


Fig. 1. A Smart Grid enabled heterogeneous network

As mentioned before, when a user requests video with quality level q, all the layers below q must be successfully received. To reduce the download delay, those layers are sent to the user at the same time. Thus the delivery delay of video is bounded by the maximum delivery delay among all the layers.

$$D_n^{vq} = \max_{1 \leq l \leq q} \left\{ d_n^{vl} \right\} \tag{1}$$

Where d_n^{vl} is the delivery delay of layer L_{vl}, and D_n^{vq} is the delivery delay of video V_{vq}.

In order to efficiently utilize the limited cache size of SBSs, the cached contents in SBS i can be shared with SBS j through the backhaul link. As illustrated in Fig. 1, the user 3 requests for video V_{12}, the layer L_{12} hasn't been cached in its nearby SBSs, then the SBS with layer L_{12} will send it to the nearest SBS of user 3, which will reduce the download delay compared with fetching L_{12} from the remote server. According to the different place where the user gets the requested content, we define the following three delivery modes.

- **Local delivery mode:** We denote the nearby SBSs set of user n as: $\mathcal{M}_n$. If the requested content of user n has been cached in the SBS m belonging to $\mathcal{M}_n$, this content is directly sent by SBS m. The download rate of this mode is given by

$$r_{mn}^{w} = W log_2 \left(1 + \frac{p_m g_{mn}}{I + \sigma^2} \right), \tag{2}$$

where g_{mn} is the channel coefficient from SBS m to user n, p_m is the transmission power of SBS m. In this paper, we assume that the simultaneous

transmission at the same SBS is handled by orthogonal spectrum of the same bandwidth W, and the advanced inter-cell interference coordination (ICIC) technology has been used so that the interference I is a constant.
- **Backhaul delivery mode:** If the requested content of user n hasn't been cached in SBSs set $\mathcal{M}_n$, then the nearest SBS of user n will attempt to fetch this content through backhaul link and send it to user. This process will cause a larger delay compared with the local delivery. We use r_k^b to denote the download rate, where k represents the SBS that sends the requested content.
- **Remote delivery mode:** If there is no copy of the requested content in any SBS, the request file is fetched from the remote servers. This fetching will cause the slowest download rate r_0.

3 Problem Formulation and Solution

In this paper, our goal is to minimize the download delay of video with different quality level. To describe the video layers placement, we define the binary variables $x_m^{vl} \in \{0, 1\}$, where $x_m^{vl} = 1$ indicates that the layer L_{vl} is cached at the SBS m, and $x_m^{vl} = 0$ otherwise. The cache size constraint of SBS can be expressed as,

$$\sum_{v \in \mathcal{V}} \sum_{l \in \mathcal{L}} x_m^{vl} s_{vl} \leq C_m, \quad \forall m \in \mathcal{M}. \tag{3}$$

To describe the three mentioned delivery modes, we introduce the binary variables $\{y_n^{vl}, z_{kn}^{vl}\} \in \{0, 1\}$, where the binary variable $y_n^{vl} = 1$ is used to indicate that the layer L_{vl} is fetched from SBS in set $\mathcal{M}_n$, and the binary variable $z_{kn}^{vl} = 1$ is used to denote the layer L_{vl} is fetched from SBS k out of the set $\mathcal{M}_n$. The download rate in local delivery mode should be larger than backhaul delivery mode, then we have following constraint,

$$r_{mn}^w > \max(r_k^b), \quad \forall m \in \mathcal{M}_n. \tag{4}$$

The delay for user n receiving the layer L_{vl} can be expressed as:

$$\begin{aligned} d_n^{vl} = y_n^{vl} & \min_{m \in \mathcal{M}_n} (x_m^{vl} \frac{s_{vl}}{r_{mn}^w}) + \\ & (1 - y_n^{vl}) \sum_{k \in \mathcal{M} \setminus \mathcal{M}_n} z_{kn}^{vl} \frac{s_{vl}}{r_k^b} + \\ & (1 - y_n^{vl}) \prod_{k \in \mathcal{M} \setminus \mathcal{M}_n} (1 - z_{kn}^{vl}) \frac{s_{vl}}{r_0}. \end{aligned} \tag{5}$$

The first term is the delay of local delivery mode. If the nearby SBSs of user have cached the requested content, the closest SBS will send this content to user. The second term is the delay of backhaul delivery mode. If requested content has been cached in nearby SBSs, user will try to fetch it from other SBSs through backhaul link. The last term is the delay that the requested content is delivered from the remote server.

Generally, the capacity of the backhaul link of each SBS is limited. Therefore the upstream traffic of SBSs k can not greater than its backhaul capacity B_k.

$$\sum_{n\in\mathcal{N}}\sum_{v\in\mathcal{V}}\sum_{1\le l\le q} z_{kn}^{vl} r_k^b \le B_k, \quad \forall k\in\mathcal{M}. \tag{6}$$

The overall problem is minimal the delay when users request for video with different quality level. Given the probability p_{vq} that users request for video v with quality level q, the objective is to determine the cache placement of different layers x_m^{vl}, and the delivery modes of requested content $\{y_n^{vl}, z_{kn}^{vl}\}$.

$$\min_{\boldsymbol{x},\boldsymbol{y},\boldsymbol{z}} \sum_{n\in\mathcal{N}}\sum_{v\in\mathcal{V}}\sum_{q\in\mathcal{Q}} p_{vq} \max_{1\le l\le q}\left\{d_n^{vl}\right\} \tag{7a}$$

$$\text{s.t.} \quad \sum_{v\in\mathcal{V}}\sum_{l\in\mathcal{L}} x_m^{vl} s_{vl} \le C_m, \quad \forall m\in\mathcal{M}, \tag{7b}$$

$$0\le y_n^{vl} \le \min\Big(\sum_{m\in\mathcal{M}_n} x_m^{vl}, 1\Big), \quad \forall n,v,l, \tag{7c}$$

$$0\le z_{kn}^{vl} \le x_k^{vl}, \quad \forall n,k,v,l, \tag{7d}$$

$$\sum_{n\in\mathcal{N}}\sum_{v\in\mathcal{V}}\sum_{1\le l\le q} z_{kn}^{vl} r_k^b \le B_k, \quad \forall k\in\mathcal{M}, \tag{7e}$$

$$r_{mn}^w > \max(r_k^b), \quad \forall m\in\mathcal{M}_n. \tag{7f}$$

$$\boldsymbol{x},\boldsymbol{y},\boldsymbol{z}\in\{0,1\}. \tag{7g}$$

Constraint (7b) means that the cached contents of each SBS can't exceed its capacity C_m. Constraint (7c) and (7d) ensure the availability of the layers, i.e., the requested layers should be sent by the SBS that caches them. Constraint (7e) ensures that the upstream traffic of each SBS can't exceed its backhaul capacity B_k. Constraint (7f) limits the size of nearby SBS set $\mathcal{M}_n$.

This problem is NP-hard, which can be proved by reduction from the knapsack problem, and the latter can be expressed as follows. Given a knapsack with capacity W, and N items with non-negative weights $w_1,\cdots,w_N$ and values $v_1,\cdots,v_N$. The problem is to find a subset with largest value from items N, when the total weight of this subset does not exceed the capacity W. We consider a special case of our problem where each video encodes to one base layer and there are only one SBS in the network. Then this problem is to cache a subset of videos from $\mathcal{V}$ to achieve minimal delay under the constraint of caching size of SBS, which is the standard knapsack problem.

3.1 Proposed Greedy Layer Caching Algorithm

In this subsection, we will present the proposed greedy caching algorithm for the video layers caching problem, which can find a sub-optimal solution within polynomial time. We first give the expression of expected delay savings when adding a new content to the SBS cache. Then we design a greedy caching algorithm which caches the content with maximal delay savings in each iteration.

We define the set $\mathcal{X} = \{x_1^{11}, \ldots, x_m^{vl}, \ldots, x_M^{VL}\}$ to describe the cache placement, and $G(\mathcal{X}^*, x_m^{vl})$ represents the expected delay savings by adding new content x_m^{vl} to the last caching placement $\mathcal{X}^*$. To describe the delay savings, we introduce the variables $\{D_{nvl}^*, r_{vl}^*\}$ to record the state of the last caching placement, which are explained as follows.

- D_{nvl}^* is the last download delay for user n requesting the video v with quality level l.
- r_{vl}^* records the remaining number of users, and the layer l they requested is sent by the remote server.

When a new content x_m^{vl} is added into the caching placement $\mathcal{X}^*$, the SBS m can send this content to its nearby users $\mathcal{N}_m$, which may save the download delay when these users request for video V_{vl}, and this part of delay savings can be expressed as:

$$G_1 = \sum_{n \in \mathcal{N}_m} p_{vl}[D_{nvl}^* - D_{nvl}]^+, \tag{8}$$

where operation $[x]^+ \triangleq \max(x, 0)$, and D_{nvl} is the current download delay, which is bounded by the last download delay of video $V_{v(l-1)}$.

$$D_{nvl} = \max(\frac{s_{vl}}{r_{mn}^w}, D_{nv(l-1)}^*). \tag{9}$$

Note the SBS m can also send content x_m^{vl} to the remote users via backhaul link, which can save the delay compared with the remote delivery mode.

$$G_2 = p_{vl} \min(\frac{B_m}{r_m^b}, r_{vl}^*)(\frac{s_{vl}}{r_m^b} - \frac{s_{vl}}{r_0}). \tag{10}$$

Here, $\min(\frac{B_m}{r_m^b}, r_{vl}^*)$ is the number of users that they requested content is fetched from the SBS m via backhaul link.

By adding this two part of delay savings, $G(\mathcal{X}^*, x_m^{vl})$ can be expressed as follows.

$$G(\mathcal{X}^*, x_m^{vl}) = G_1 + G_2. \tag{11}$$

Next, we illustrate the greedy caching algorithm for the layer caching problem (7a) in Algorithm 1. This algorithm starts with the empty cache at each SBS. In each iteration, it adds the content $x_{m^*}^{v^*l^*}$ with maximum utility $U(x_m^{vl})$ to the SBS m until the caches of all SBSs are filled, and the utility is defined as the delay savings per bit.

$$U(x_m^{vl}) = \frac{G(\mathcal{X}^*, x_m^{vl})}{s_{vl}}. \tag{12}$$

After adding the content $x_{m^*}^{v^*l^*}$ to the cache of SBS m^*, the delay record $D_{nv^*l^*}^*$ and the remaining number of users $r_{v^*l^*}^*$ will be updated to record the state of current iteration.

Algorithm 1. Proposed Greedy Layer Caching Algorithm

Require: $\mathcal{N}_m$, p_{vl}, B_m, r^w_{mn},r^b_m,r_0,s_{vl}.
Ensure: x^{vl}_m.
1: Initialization:
$x^{vl}_m = 0$;
$D^*_{nvl} = \frac{s_{vl}}{r_0}$, $D^*_{nv0} = 0$, $r^*_{vl} = N$;
2: **while** the caching space is not full **do**
3: $(m^*, v^*, l^*) \leftarrow_{m,v,l} U(x^{vl}_m)$, $x^{v^*l^*}_{m^*} = 1$;
4: **for** each $n \in \mathcal{N}_{m^*}$ **do**
5: **if** $\frac{s_{v^*l^*}}{r^w_{m^*n}} < D^*_{nv^*l^*}$ **then**
6: $D^*_{nv^*l^*} = \max(\frac{s_{v^*l^*}}{r^w_{m^*n}}, D^*_{nv^*(l^*-1)})$
7: **end if**
8: **end for**
9: $r^*_{v^*l^*} = \max(r^*_{v^*l^*} - \frac{B_{m^*}}{r^b_{m^*}}, 0)$;
10: **end while**

3.2 Proposed Caching and Cooperative Delivery Algorithm

In the following, we will introduce the overall caching and cooperative delivery algorithm which gives the caching placement of video layers and the places that serve the requests of users.

In this algorithm, we first initialize the neighbor set M_n (N_m) of each user (SBS), which satisfies $r^w_{mn} > \max(r^b_m)$. Then, each SBS caches video layers x^{vl}_m according to the result of Algorithm 1. Finally, this algorithm will decide where to serve the requests of users. If a user n requests video V_{vq}, all the layers $l \leq q$ should be received to recover the original video. To get the layer L_{vl}, the user will first examine whether the nearby SBSs $m \in M_n$ have cached this layer. If not, the nearest SBS of this user will try to get this layer from remote SBSs through backhaul link. If the nearest SBS still can't get the layer L_{vl} from other SBSs, it will fetch this layer from the remote server. The delay for request r_{nvq} is the maximum download delay among all the layers $l \leq q$. The specific algorithm is shown in Algorithm 2.

4 Simulation Results

In this section, we evaluate the performance of the proposed layer caching and cooperative delivery algorithm. We consider a small cell network consisting of 13 uniformly distributed SBSs and 60 users. The video library V consists of 200 unique videos, and each video has $Q = 5$ quality levels by using SVC. For simplicity, we consider that each video has already been divided to equal size 40 MB, and the size of layers 1 to 5 is 20,5,5,5,5 MB. The popularity of the videos follows the Zipf distribution, and we assume the quality level of video requested by users follows a uniform distribution. Then the probability that users request

Algorithm 2. Proposed Layer Caching and Cooperative Delivery Algorithm

Require: $\mathcal{N}$, $\mathcal{M}$, g_{mn}, r_{nvq}, r_m^b,r_0,s_{vl}.
Ensure: $x_m^{vl}, y_n^{vl}, z_{kn}^{vl}$.
1: Initialization:
Calculate $r_{mn}^w = Wlog_2\left(1 + \frac{p_m g_{mn}}{I+\sigma^2}\right)$;
Set $M_n = \emptyset$, $N_m = \emptyset$;
2: **for** each r_{mn}^w **do**
3: **if** $r_{mn}^w > \max(r_m^b)$ **then**
4: $\mathcal{M}_n \leftarrow M_n \cup m$, $\mathcal{N}_m \leftarrow N_m \cup n$;
5: **end if**
6: **end for**
7: Obtain the caching placement x_m^{vl} by algorithm 1;
8: **for** each user request r_{nvq} **do**
9: **for** $l = 1, \ldots, q$ **do**
10: $r_n^{vl} = r_0$;
11: **for** each SBS $m \in M_n$ **do**
12: **if** $x_m^{vl} = 1$ **then**
13: $r_n^{vl} = \max(r_{mn}^w)$, $y_n^{vl} = 1$;
14: Go to step 22;
15: **end if**
16: **end for**
17: **for** each SBS $k \notin M_n$ **do**
18: **if** $x_k^{vl} = 1$ and $\sum_{n\in\mathcal{N}}\sum_{v\in\mathcal{V}}\sum_{1\le l\le q} z_{kn}^{vl} r_k^b \le B_k$ **then**
19: $r_n^{vl} = \max(r_k^b)$
20: $k^* \leftarrow_k (r_k^b)$, $z_{k^*n}^{vl} = 1$;
21: **end if**
22: **end for**
23: **end for**
24: $D_{nvq} = \max\limits_{1\le l\le q}(\frac{s_{vl}}{r_n^{vl}})$;
25: **end for**

video v with quality level q can be expressed as:

$$p_{vq} = \frac{v^{-\delta}}{Q\sum_{i=1}^{V} i^{-\delta}}, \tag{13}$$

Where δ is the skew parameter. In this simulation, we generated 200 user requests r_{nvq} based on the popularity p_{vq} and calculated the corresponding performance indicators. The other settings of the simulation are summarized in Table 1.

We evaluate the performance in terms of average delivery delay and backhaul traffic load. The former is the average latency of delivering all the layers to decode the original video requested by the user, and the latter is defined as follows.

$$T_{bh} = s_{vl}\left(2\sum_n\sum_k z_{kn}^{vl} + (1 - y_n^{vl})\prod_k(1 - z_{kn}^{vl})\right). \tag{14}$$

The first term is the traffic caused by sharing layers over the backhaul link, and the last is caused by delivering layers from the remote server.

Table 1. Simulation parameters

Parameter	Value
Transmit power of SBSs p_m	200 mW
Path loss model	$140.7 + 36.7 \log_{10} d$
Bandwidth W	900 kHz
Noise power density σ^2	−174 dBm/Hz
Interference I	−100 dBm/Hz
Delivery rate over backhaul link r_m^b	5 Mbps
Delivery rate from remote server r_0	1 Mbps
Cache capacity of SBSs C_m	5%–35% (% of library size)
Backhaul capacity of SBSs B_m	10–80 MB
Zipf skewing parameter δ	0.3–1.0

For comparison, we implement the following three caching algorithm.

- *Most Popular Caching (MPC):* Each SBS independently caches layers of the videos with the highest popularity until the cache capacity is full.
- *Largest Content Diversity (LCD):* This algorithm try to cache more different layers in the network to achieve largest diversity. It traverses the video set V and caches the content in the SBS m with the maximum utility $U(x_m^{vl})$ each time.
- *Octopus:* We take the greedy idea of the proactive cache distribution algorithm in paper. Each user is associated with the closest SBS. Iteratively, it adds the layer with the maximum delay savings to the cache of SBSs, but does not consider the limited backhaul capacity.

We first compare the impact of different caching schemes on the average delivery delay in Fig. 2, where the default value is $C_m = 20\%$, $\delta = 0.6$, and $B_m = 20$ MB. It can be seen that our proposed algorithm always achieves the minimum delivery delay. This is because MPC redundantly caches the most popular content in each SBS, ignoring the cooperation opportunity between SBSs, and this can be seen from Fig. 2(c), the increasing of backhaul capacity has no impact on the delivery delay in MPC. Although LCD caches different contents as much as possible, it does not consider the popularity of the content. Thus from Fig. 2(b), the increasing of the Zipf parameter δ has no impact on the delivery delay in LCD. In Octopus, each user is only associated with the closest SBS, ignoring the overlapped coverage of SBSs, which leads to a larger delivery delay compared with our algorithm.

In addition, we can see that the marginal value of caching a new content becomes smaller as the cache size increases from Fig. 2(a). Therefore we should cache the content with more delay savings when the cache set is small. From Fig. 2(b), the average delivery delay decreases with the increasing of the Zipf skewing parameter δ, because the popularity is more concentrated in a few videos

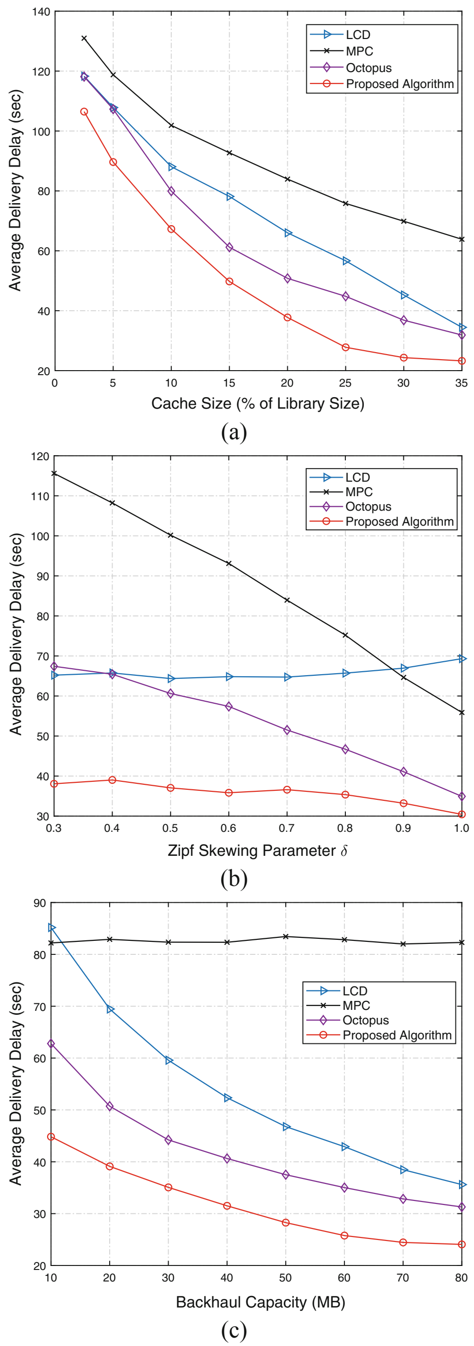
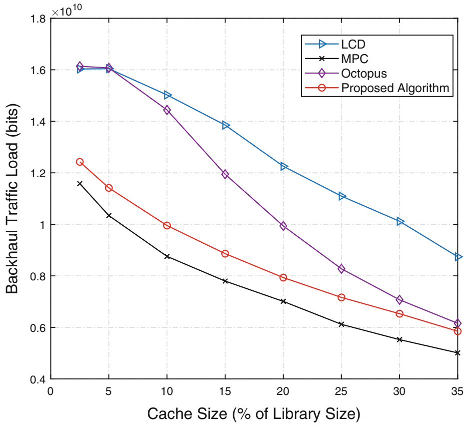
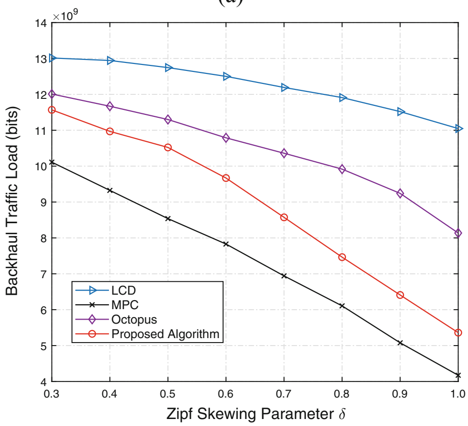


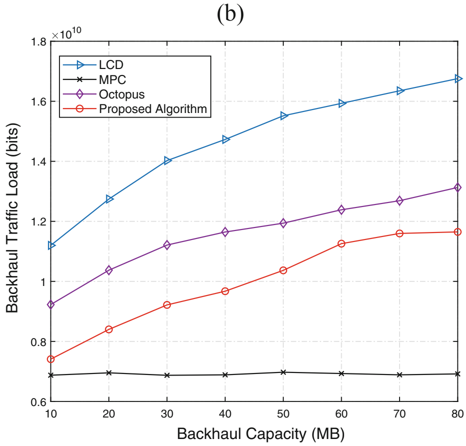
Fig. 2. The average delivery delay achieved by the different caching schemes as a function of (a) the cache sizes, (b) the skewing parameter δ of the Zipf distribution, and (c) the capacity of backhaul link.



(a)



(b)



(c)

Fig. 3. The backhaul traffic load achieved by the different caching schemes as a function of (a) the cache sizes, (b) the skewing parameter δ of the Zipf distribution, and (c) the capacity of backhaul link.

with a larger δ. From Fig. 2(c), as the backhaul capacity increases, the delivery delay decreases, because the SBSs with larger backhaul capacity can share more cached contents, thus avoiding fetching them from the remote server.

Figure 3 compares the impact of different caching schemes on the backhaul traffic load. It can be seen that MPC scheme always achieves the minimum backhaul traffic, because there is no cooperation between SBSs in MPC, thus reducing the traffic caused by sharing contents over the backhaul link. However, as can be seen from Fig. 2, MPC will result in the largest delivery delay. LCD try to cache more different contents, which results in more frequent sharing of cached contents, and it achieves the maximum backhaul traffic in Fig. 3. Our proposed algorithm results smaller backhaul traffic compared with Octopus. Because users only associate with the closest SBS in Octopus, the cached contents in the nearby SBS should be delivered through the backhaul link, which results a larger backhaul traffic. In particular, we can see that our proposed algorithm have a larger gap between Octopus when cache size is small from Fig. 3(a).

5 Conclusion

In this paper, we design a cooperative caching and transmission strategy for video layers encoded by SVC. We formulate the problem to minimize the video delivery delay, where SBSs can share the cached content with each other and users can be served by multiple nearby SBSs. To solve this NP-hard problem, we design a heuristic algorithm with two phase. In content caching phase, a greedy algorithm is designed to cache the video encoding layers at each SBSs. After this phase, we also design a transmission strategy to deliver the cached video layers, which considers the limited capacity of backhaul link. Simulation results show that our proposed algorithm not only reduce the video delivery delay, but also save the backhaul traffic.

Acknowledgment. This work has been supported by STATE GRID Corporation of China science and technology project "Research and application on key technologies of power wireless heterogeneous network convergence" (5700-201919236A-0-0-00).

References

1. Anandakumar, H., Arulmurugan, R.: Next generation wireless communication networks for smart grid. In: 2019 Third International conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud) (I-SMAC), pp. 260–264 (2019)
2. Pei, Y., Zhang, H., Gu, X., Wang, H.: Research on power grid information model based on artificial intelligence. In: 2019 International Conference on Computer Network, Electronic and Automation (ICCNEA), pp. 321–328 (2019)
3. Asuhaimi, F.A., Bu, S., Klaine, P.V., Imran, M.A.: Channel access and power control for energy-efficient delay-aware heterogeneous cellular networks for smart grid communications using deep reinforcement learning. IEEE Access **7**, 133474–133484 (2019)

4. Baştuğ, E., Bennis, M., Debbah, M.: Cache-enabled small cell networks: modeling and tradeoffs. In: 2014 11th International Symposium on Wireless Communications Systems (ISWCS), pp. 649–653, August 2014
5. Shanmugam, K., Golrezaei, N., Dimakis, A.G., Molisch, A.F., Caire, G.: Femtocaching: wireless content delivery through distributed caching helpers. IEEE Trans. Inf. Theory **59**(12), 8402–8413 (2013)
6. Poularakis, K., Iosifidis, G., Argyriou, A., Tassiulas, L.: Video delivery over heterogeneous cellular networks: optimizing cost and performance. In: IEEE INFOCOM 2014 - IEEE Conference on Computer Communications, pp. 1078–1086, April 2014
7. Gabry, F., Bioglio, V., Land, I.: On energy-efficient edge caching in heterogeneous networks. IEEE J. Sel. Areas Commun. **34**(12), 3288–3298 (2016)
8. Chen, Z., Lee, J., Quek, T.Q.S., Kountouris, M.: Cooperative caching and transmission design in cluster-centric small cell networks. IEEE Trans. Wireless Commun. **16**(5), 3401–3415 (2017)
9. Jiang, W., Feng, G., Qin, S.: Optimal cooperative content caching and delivery policy for heterogeneous cellular networks. IEEE Trans. Mob. Comput. **16**(5), 1382–1393 (2017)
10. Zhang, S., He, P., Suto, K., Yang, P., Zhao, L., Shen, X.: Cooperative edge caching in user-centric clustered mobile networks. IEEE Trans. Mob. Comput. **17**(8), 1791–1805 (2018)
11. Tran, T.X., Pandey, P., Hajisami, A., Pompili, D.: Collaborative multi-bitrate video caching and processing in mobile-edge computing networks. In: 2017 13th Annual Conference on Wireless On-demand Network Systems and Services (WONS), pp. 165–172, February 2017
12. Schwarz, H., Marpe, D., Wiegand, T.: Overview of the scalable video coding extension of the H.264/AVC standard. IEEE Trans. Circuits Syst. Video Technol. **17**(9), 1103–1120 (2007)
13. Wu, L., Zhang, W.: Caching-based scalable video transmission over cellular networks. IEEE Commun. Lett. **20**(6), 1156–1159 (2016)
14. Poularakis, K., Iosifidis, G., Argyriou, A., Koutsopoulos, I., Tassiulas, L.: Distributed caching algorithms in the realm of layered video streaming. IEEE Trans. Mob. Comput. **18**(4), 757–770 (2019)
15. Zhang, T., Mao, S.: Cooperative caching for scalable video transmissions over heterogeneous networks. IEEE Netw. Lett. **1**(2), 63–67 (2019)