Optimized Content Caching and User Association for Edge Computing in Densely Deployed Heterogeneous Networks

Yun Li, *Member, IEEE*, Hui Ma, Lei Wang,
Shiwen Mao, *Fellow, IEEE*, and Guoyin Wang, *Senior Member, IEEE*

Abstract—Deploying small cell base stations (SBS) under the coverage area of a macro base station (MBS), and caching popular contents at the SBSs in advance, are effective means to provide high-speed and low-latency services in next generation mobile communication networks. In this paper, we investigate the problem of content caching (CC) and user association (UA) for edge computing. A joint CC and UA optimization problem is formulated to minimize the content download latency. We prove that the joint CC and UA optimization problem is NP-hard. Then, we propose a CC and UA algorithm (JCC-UA) to reduce the content download latency. JCC-UA includes a smart content caching policy (SCCP) and dynamic user association (DUA). SCCP utilizes the exponential smoothing method to predict content popularity and cache contents according to prediction results. DUA includes a rapid association (RA) method and a delayed association (DA) method. Simulation results demonstrate that the proposed JCC-UA algorithm can effectively reduce the latency of user content downloading and improve the hit rates of contents cached at the BSs as compared to several baseline schemes.

Index Terms—Content caching, content download latency, heterogeneous networks, user association

1 Introduction

The global mobile traffic has grown to more than three times of the wireline network traffic, along with the rapid increase of the number of mobile users and mobile business [1]. The rapid growth of mobile business brings about great challenges to the architecture of the existing and emerging mobile communication networks [2]. In a densely populated urban area, especially the residential areas with complex buildings structures, the indoor signal coverage is usually poor and the depth of signal coverage is also seriously insufficient due to the propagation loss due to the walls. The difficulty and high cost for building extension base stations can bring a great budget burden on the network operators [3].

With simple structure and flexible deployment features, small cell base stations (SBSs) can provide high data rate and high quality of service (QoS) to mobile services. Therefore, the "MBS + SBS" heterogeneous cellular network (HetNet) is one of the most important architectures in the next generation mobile communication systems. In this architecture, SBSs commonly connect to a core network through backhaul links with great transmission capacity [4], and the mobile users associated with the SBSs download contents from the content servers through the backhauls. When the contents are popular, the same popular content will be repeatedly transmitted through the backhaul links, which sharply increases the traffic load on the backhaul links and may lead to congestion in the backhaul links. Distributed content caching technique is an emerging and effective means to solve this problem [5], [6], [7]. According to a statistical study, distributed content caching can reduce 1/3 to 2/3 mobile data volume [8]. Furthermore, selectively caching contents at SBSs can significantly reduce traffic load on the backhaul links and decrease the content download latency [9].

In recent years, there have been some interesting works conducted on content caching in mobile cellular works [7], [10], [11], [12]. Different content caching methods were proposed for different purposes, such as reducing the content download latency [18], alleviating the traffic load on backhaul links [17], increasing the profits of service retailers or providers [19], making better tradeoff between different network performance [31], etc.

The prior works have demonstrated that content caching in heterogeneous mobile networks can effectively improve the QoS of mobile users. In HetNets, the content download

- *Yun Li is with the Chongqing Key Laboratory of Mobile Communications Technology, Chongqing University of Post and Telecommunications, Chongqing 400065, China, and also with the National Mobile Communications Research Laboratory, Southeast University, Nanjing 210018, China. E-mail: liyun@cqupt.edu.cn.*
- *Hui Ma and Lei Wang are with the Chongqing Key Laboratory of Mobile Communications Technology, Chongqing University of Post and Telecommunications, Chongqing 400065, China. E-mail: 15176367285@163.com, 18772001210@139.com.*
- *Shiwen Mao is with the Department of Electrical and Computer Engineering, Auburn University1383, Auburn, AL 36849-5201 USA. E-mail: smao@ieee.org.*
- *Guoyin Wang is with the Chongqing Key Laboratory of Computational Intelligence, Chongqing University of Posts and Telecommunications12419, Chongqing 400065, China. E-mail: wanggy@cqupt.edu.cn.*

Manuscript received 6 Feb. 2020; revised 18 Sept. 2020; accepted 21 Oct. 2020. Date of publication 26 Oct. 2020; date of current version 5 May 2022. (Corresponding author: Yun Li.) Digital Object Identifier no. 10.1109/TMC.2020.3033563

latency not only depends on the content caching strategy, but also hinges upon the user association strategy, especially in the scenario where the coverage areas of different SBSs overlap with each other. However, most of the content caching works cache contents according to the prediction results of the content requirement probability. However, the effect of mobile user association on the QoS of mobile users, especially on the content download latency, has been largely ignored. Although caching contents in SBSs can decrease the content download latency, an inappropriate user association will increase the content download latency. Therefore, we should cache contents in the SBSs with which more mobile user are associated with, to download these contents, and associate mobile users to the SBSs through which they can download contents with a smaller latency. This means that content caching and user association affect each other and jointly determine the content download latency.

In view of the above issue, in this paper, we investigate the optimization problem of content caching and user association. The main contributions of this paper are summarized as follows.

1) Considering of caching contents distributedly in cloud center and base stations (MBSs and SBSs), a joint content caching and user association (JCC-UA) optimization problem is formulated to minimize the average content download latency. We also prove that the JCC-UA problem is NP-hard.
2) In order to solve the JCC-UA problem, a smart content caching policy (SCCP) based on cubic exponential smoothing is proposed for content caching, while a rapid association (RA) algorithm and a delayed association (DA) algorithm are proposed for user association.
3) The performance of the proposed JCC-UA scheme, including the SCCP, RA, and DA algorithms, is comprehensively evaluated in our simulation study, in terms of cache hit rate and content download latency. The proposed scheme outperforms the several baseline schemes with considerable margins in our simulation study.

The rest of this paper are organized as follows. In Section 3, the system model is presented and the JCC-UA optimization problem is formulated. Section 4 proposes the SCCP and user association algorithms. The performances of the proposed algorithms are evaluated in Section 5. Section 6 concludes this paper.

2 Related Work

The authors in [13] showed that caching popular data at SBSs as far as possible can reduce the data transmission delay and offload the redundant data streams from an MBS. The authors in [14] developed a framework for jointly optimizing resource allocation and content caching for HetNets. In [15], a long short-term memory (LSTM) deep learning model was proposed to cache the data that was most likely requested by end users to reduce service latency. In [16] and [17], a pre-fetching strategy was proposed to cache contents on the edge of a mobile network to reduce the traffic load on the wireless link. The authors in [18] optimally assigned contents to SBSs to minimize the content download latency. In [19], a Stackelberg game was formulated to optimally cache content in SBSs to maximize the profit of video retailers and network service providers. Based on demand history, the works in [20] and [21] optimized content placement to make the best use of the cache capacity of SBSs. In [22], the authors improved the content caching strategy by considering user mobility and the randomness of contact duration.

Taking into account bandwidth limitations, the authors in [23] proposed a content caching strategy to maximize the number of content requests served by SBSs. Under the capacity constraints of the backhaul link, the authors in [24] exploited the caching capability of SBSs to improve the QoS of mobile users. In [25], a collaborative filtering scheme was proposed to improve the backhaul efficiency and increase the cache hit ratio. Considering the wired backhaul and wireless channel quality, the work in [26] studied the effect of backhaul delay on averaging content downloaded latency in HetNets. In [27], an optimal cooperative content caching and delivering strategy was proposed for the femtocell and device-to-device (D2D) communication architecture. The authors in [28] analyzed the probability that mobile users successfully downloaded contents from SBSs with distributed content caching. Considering that the paths to the back-end server were either congestion-sensitive or congestion-insensitive, the work in [29] investigated the joint content caching and routing problem to minimize the average content access delay. Based on the partial caching scheme, a joint subcarrier assignment and user association scheme was proposed in [30] to minimize the average content delivery time.

In addition, the tradeoff between network utility and backhaul saving was investigated in [31]. The tradeoff was measured by a utility function, and a jointly optimized cache placement and user-BS association algorithm was proposed to maximize the utility function. In [32], a Stackelberg game based framework was formulated to model the competition between video providers (VP) and mobile network operators (MNO). A joint video pricing and cache placement algorithm was developed to maximize the profits of the VP and the MNO. In [33], the authors investigated the problem of cache storage allocation among BSs, as well as multicast beamforming transmission in a wireless network with multicast and BS caching. The cache-channel coding scheme and cache size allocation algorithm were proposed to improve the message delivery efficiency. In [34], a framework for minimizing the system delivery time of full-duplex enabled MEC was built and two iterative optimization algorithms were proposed to solve the problem with sub-optimal solution. The authors in [35] also proposed a cache-channel coding scheme and a cache allocation algorithm to maximize the BS expected file downloading rate.

3 System Model and Problem Formulation

3.1 System Model

The two-tier content caching and service model of a heterogeneous network are shown in Fig. 1. The upper layer of the content cache is in the cloud center connected to the core

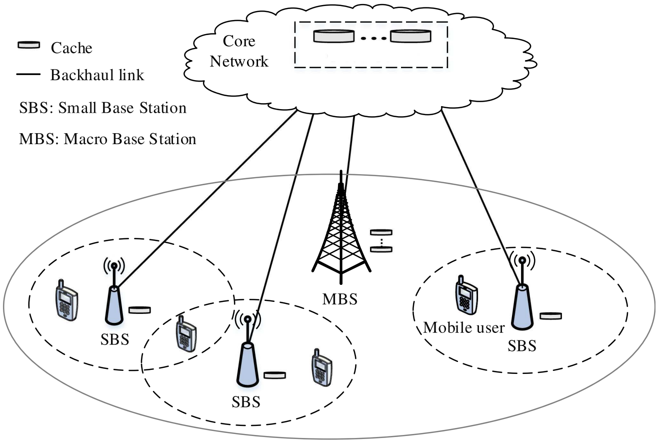


Fig. 1. Architecture of the two-tier heterogeneous network (HetNet).

network, while the lower layer content cache is located in the cellular network including some MBSs and SBSs. In this paper, we assume that all the SBSs are connected to the core network through backhaul links, and there is no direct link between SBSs, which is the common scenario in a real-world network. For simplification, we show only one MBS in Fig. 1. The set of the MBS and SBSs is denoted as $F = \{f_0, f_1, \ldots, f_N\}$, where f_0 represents the MBS and $f_1, f_2, \ldots, f_N$ represent the N SBSs. Let $U = \{u_1, u_2, \ldots, u_M\}$ denote the M mobile users in the cellular network.

We consider that the frequency sub-channels allocated to different BSs are orthogonal, so the inter-cell interference is not needed in this setting [27]. The available bandwidth at BS f_n is $B_n (n = 0, 1, \ldots, N)$, which is divided into I_n sub-channels. The bandwidth of each sub-channel of f_n is $b_n = B_n/I_n$. Each sub-channel serves one mobile user, which means that the maximum number of mobile users served by f_n is I_n. Due to overlapped coverage between SBSs and MBS, a user in an overlapped coverage area can access the cellular network by associating with the MBS or one of the SBSs. SNR_{mn_j} is the signal-to-noise ratio (SNR) from BS f_n to mobile user u_m on sub-channel j, $j \in \{1, 2, \ldots, I_n\}$, and $SNR_{mn_j} = P_{n_j} G_{mn_j} / \sigma_N^2$, where P_{n_j} is the transmission power of f_n in sub-channel j, G_{mn_j} is the channel gain of the wireless link from f_n to u_m on sub-channel j, and σ_N^2 is the Gaussian white noise power. The fading of the communication links is composed of Rayleigh fading and path loss [30], and thus we have $G_{mn_j} = \kappa \cdot \tau_{mn_j} \cdot d_{mn_j}^{-\varepsilon}$, where τ_{mn_j} and d_{mn_j} denote the fading factor on sub-channel j and the distance between u_m and f_n, respectively; and κ and ε denote the pathloss constant and pathloss exponent, respectively. Let r_{mn_j} denote the data rate of mobile user u_m for downloading content from f_n through sub-channel j, which can be calculated by the Shannon formula as

$$r_{mn_j} = b_n \cdot \log_2\Big(1 + SNR_{mn_j}\Big). \tag{1}$$

Let $C = \{c_1, c_2, \ldots, c_K\}$ denote the set of K contents in the system. The size of c_k is l_k bits, for $k = 1, 2, \ldots, K$. The content can be distributedly stored in the cloud center or in the BSs, including the MBS and SBSs. The caching capacity of f_n is denoted as $S_n (n = 0, 1, \ldots, N)$ bits. If the content c_k is cached at BS f_n and mobile user u_m is associated with f_n, then u_m can directly download the content from f_n. Otherwise, u_m will download the content from the cloud center f_n. As in [17], we assume the latency of the backhaul links from the BSs to the core network equals to T_C. Note that a content will be cached at a BS only when this content has been downloaded by mobile users through this BS. This type of content caching can be termed as "passive content caching." Therefore, we do not consider the additional cost for the BS to fetch contents.

In order to satisfy the QoS requirements of mobile users, we assume that the minimum guaranteed data rate for a mobile user u_m is $r_{m,min}$. A mobile user u_m may associate with BS f_n on sub-channel j, if and only if $r_{m,min} \leq r_{mn_j}$. If there are no BSs that can satisfy the minimum guaranteed date rate, the user call will be rejected. In this case, the user may initiate a new association request again at a later time.

3.2 Problem Formulation

In this section, we formulate the JCC-UA problem aiming to minimize the average content download latency for mobile users. The JCC-UA problem is defined as follows. There are K contents that will be distributedly cached in the cloud center and $N + 1$ BSs (including the MBS and N SBSs), and M mobile users will download these contents through the content center and the BSs. The JCC-UA problem is to make decision on which BS each content should be cached and on which BS each mobile user will be associated with.

According to the system model and symbol definitions, for a mobile user u_m requesting content $c_k \in C$, the content download latency is l_k / r_{mn_j}, if u_m is associated with an SBS f_n on sub-channel j and c_k is cached in f_n. Otherwise, the content download latency will become $l_k / r_{mn_j} + T_C$. We define $X = \{x_{nk} | f_n \in F, c_k \in C\}$ as the content caching decision matrix, where $x_{nk} \in \{0, 1\}$; $x_{nk} = 1$ means that content c_k is cached at BS f_n, and $x_{nk} = 0$ indicates that c_k is not cached at f_n. We also define $Y = \Big\{y_{mn_j} | u_m \in U, f_n \in F, j \in I_n\}$ as the user association decision matrix, where $y_{mn_j} \in \{0, 1\}$; $y_{mn_j} = 1$ means that user u_m is associated with f_n on sub-channel j, and $y_{mn_j} = 0$ indicates that u_m is not associated with f_n on sub-channel j.

Then, the content download latency for a mobile user requesting content c_k is

$$t_{mk} = \sum_{n=0}^{N} \sum_{j=1}^{I_n} y_{mn_j} \cdot \left(\frac{l_k}{r_{mn_j}} + (1 - x_{nk}) \cdot T_C \right). \tag{2}$$

Let $q_{mk} \in \{1, 0\}$ denote whether a mobile user u_m requests content c_k or not; $q_{mk} = 1$ means that u_m requests content c_k, and $q_{mk} = 0$ otherwise. The total content download latency for user u_m to download all the contents is

$$t_m = \sum_{k=1}^{K} q_{mk} \cdot t_{mk}. \tag{3}$$

In this paper, we aim to minimize the average content download latency, which is given by

$$\bar{t} = \frac{1}{|M|} \sum_{m=1}^{M} t_m. \tag{4}$$

Considering the system constraints, the JCC-UA optimization problem can be formulated as follows in (5), (6), (7), (8), (10), and (11). In the formulated problem, inequality (6) ensures that the sum of contents stored at a BS is not more than the storage capacity of the BS. Constraint (7) guarantees that the number of mobile users being served by a BS is not greater than the maximum number of mobile users that the BS can serve. Inequality (9) indicates that the data rate of a mobile user u_m will be no less than the required minimum data rate if user u_m is associated with a BS. Constraint (8) indicates that a mobile user can be associated with at most one BS.

$$\min_{\{x_{nk}, y_{mn_j}\}} \quad \bar{t} = \frac{1}{|M|}\sum_{m=1}^{M} t_m \tag{5}$$

$$\text{s.t.} \quad \sum_{k=1}^{K} x_{nk} \cdot l_k \leq S_n, \ \forall \ f_n \in F \tag{6}$$

$$\sum_{m=1}^{M}\sum_{j=1}^{I_n} y_{mn_j} \leq I_n, \ \forall \ f_n \in F \tag{7}$$

$$\sum_{n=0}^{N}\sum_{j=1}^{I_n} y_{mn_j} \leq 1, \ \forall \ u_m \in U \tag{8}$$

$$r_{mn_j} \geq r_{m,min}, \ \forall \ y_{mn_j} = 1 \tag{9}$$

$$x_{nk} \in \{0,1\}, \ \forall f_n \in F, \ \forall c_k \in C \tag{10}$$

$$y_{mn_j} \in \{0,1\}, \ \forall \ f_n \in F, \ \forall \ u_m \in U. \tag{11}$$

Lemma 1. *The JCC-UA problem formulated in (5), (6), (7), (8), (10), and (11) is NP-Hard.*

Proof. In order to prove Lemma 1, we consider a special case of the JCC-UA problem as follows. Each BS can cache all contents, that is $x_{nk} = 1$, for all $f_n \in F$, $c_k \in C$, and $\sum_{k=1}^{K} l_k \leq S_n$, for all $f_n \in F$. Meanwhile, the minimum guaranteed data rate for all mobile users is zero, that is $r_{m,min} = 0$, for all $u_m \in U$. For this special case, the optimization problem given in (5), (6), (7), (8), (10), and (11) can be simplified as follows.

$$\begin{aligned} &\min_{y_{mn_j}} \ \bar{t} \\ &\text{s.t.} \ (7), \ (8), \ (11). \end{aligned} \tag{12}$$

The optimization problem formulated in (12) is a classical assignment problem that has been proved to be NP-hard [36]. This means that a special case of the JCC-UA problem is NP-hard. Therefore, the JCC-UA problem formulated in (5), (6), (7), (8), (10), and (11) NP-Hard. □

4 Policy for Joint Content Caching and User Association

In Section 3, we have formulated the JCC-UA problem and proved that it is NP-Hard. In this section, we propose effective heuristic algorithms to solve the JCC-UA problem. The heuristic algorithms include (i) a smart content caching policy based on cubic exponential smoothing, and (ii) two dynamic user association algorithms. The smart content caching policy makes caching decision based on the history of content requests, which is related to user association; and dynamic user association algorithms associate users to SBSs according to cached content. Compared with user association, in most cases, content caching can be executed at a more coarser time granularity. This way, we decouple content caching and user association in the proposed JCC-UA algorithm.

4.1 Smart Content Caching Policy

The smart content caching policy (SCCP) is based on the prediction to the download counts of contents. We use the exponential smoothing method to predict the download count of a content (DCC) that is downloaded by mobile users through a BS. Due to the random behavior when a mobile user downloads contents, the data series of DCC is nonlinear. Therefore, the cubic exponential smoothing method is more suitable for predicting the value of DCC than the single exponential smoothing method [37].

In order to use exponential smoothing to predict the DCC value, we divide the time into discrete time slots. Let $z_{nk}(i)$ denote the download count of c_k that is download by mobile users through BS f_n in a time slot i, then

$$z_{nk}(i) = \sum_{m=1}^{M}\sum_{j=1}^{I_n} y_{mn_j}(i) \cdot q_{mk}(i), \tag{13}$$

where $y_{mn_j}(i) = 1$ (or 0) means that a mobile user u_m is associated with (or not associated with) BS f_n on the sub-channel j in time slot i, and $q_{mk}(i) = 1$ (or 0) means that a mobile user u_m requests (or does not request) a content c_k through BS f_n in time slot i.

The smoothed value of the download count of c_k that is downloaded by mobile users through BS f_n in time slot i is denoted by $F_{nk}(i)$, and the predicted value of $z_{nk}(i+1)$ is denoted by $\hat{z}_{nk}(i+1)$. $F_{nk}^{(\zeta)}(i)$ denotes the ζ-th value of $F_{nk}(i)$. We have

$$\begin{cases} F_{nk}^{(1)}(i) = \alpha \cdot z_{nk}(i) + (1-\alpha) \cdot F_{nk}^{(1)}(i-1) \\ F_{nk}^{(2)}(i) = \alpha \cdot F_{nk}^{(1)}(i) + (1-\alpha) \cdot F_{nk}^{(2)}(i-1) \\ F_{nk}^{(3)}(i) = \alpha \cdot F_{nk}^{(2)}(i) + (1-\alpha) \cdot F_{nk}^{(3)}(i-1), \end{cases} \tag{14}$$

where α is the smoothing parameter. The larger the α, the greater the weight of the new observed data. In cubic exponential smoothing, Formula (14) is further used for the calculation of the coefficients of prediction equations with nonlinear trends. The mathematical model of cubic exponential smoothing for predicting $\hat{z}_{nk}(i+1)$ is given in (15).

$$\begin{cases} \hat{z}_{nk}(i+1) = a_{nk}(i) + b_{nk}(i) + c_{nk}(i) \\ a_{nk}(i) = 3F_{nk}^{(1)}(i) - 3F_{nk}^{(2)}(i) + F_{nk}^{(3)}(i) \\ b_{nk}(i) = \frac{\alpha}{2(1-\alpha)^2}\Big[(6-5\alpha)F_{nk}^{(1)}(i) - \\ \qquad (10-8\alpha)F_{nk}^{(2)}(i) + (4-3\alpha)F_{nk}^{(3)}(i)\Big] \\ c_{nk}(i) = \frac{\alpha^2}{2(1-\alpha)^2}\Big[F_{nk}^{(1)}(i) - 2F_{nk}^{(2)}(i) + F_{nk}^{(3)}(i)\Big]. \end{cases} \tag{15}$$

The cubic exponential smoothing method is essentially an iterative process, and the reasonable value of the smoothing coefficient α has an important effect on the accuracy of the prediction of the DCC value. When the long-term trend of the observed data is relatively stable, the value of α should be small; Otherwise, a large value of α can work better to track the changes in DCC.

After the value of DCC is predicted by the cubic exponential smoothing method as mentioned above, we cache every content according to the following simple policy. *At time slot i when we want to cache contents, we cache the contents at each BS according to the descending order of the predicted DCC value (i.e., $\hat{z}_{nk}(i+1)$) until the caching capacity of the BS is fully occupied.*

4.2 Dynamic User Association Methods

In a heterogeneous cellular network with macrocells and smell cells, a mobile user in the overlaid coverage area of some BSs can select one BS to associate with. When a mobile user wants to download a content, it is important for the mobile user to select an appropriate BS to associate with to reduce the content download latency. This means that a mobile user associates with an appropriate sub-channel of an appropriate BS. In this sub-section, we propose the dynamic user association methods that include a rapid association method and a delayed association method.

In traditional user association strategies, mobile users can be associated according to signal strength, transmission rate, etc. In this paper, a mobile user is associated with a BS according to whether a content is download with the minimum latency. It is worth noting that the proposed scheme does not change the association signaling process. Therefore, our user association strategy can be smoothly integrated into the association process of a real-world mobile communications system, and it will not bring about an obvious additional cost.

4.2.1 The Rapid Association (RA) Method

Let u_m be a mobile user that requests content c_k. The set of sub-channels that can serve u_m is denoted as $CH = \{ch_1, ch_2, \ldots, ch_v\}$. From CH, the mobile user u_m first selects the channels that can satisfy its required minimum data rate ($r_{m,min}$), and denotes these channels as CH'. Then, a channel in CH' on which the mobile user u_m can download content c_k with the minimum content download latency will be associated with mobile user u_m. The detailed RA algorithm is presented in Algorithm 1.

For the RA algorithm, the computation complexity is determined by the two "for" loops, which are given in Lines 2 – 7 and Lines 10 – 21 of Algorithm 1. The complexities of both loops are O($|CH_t|$), so the complexity of the RA algorithm is $\mathcal{O}(|CH_t|)$.

4.2.2 Delayed Association (DA) Method

In the RA method, a mobile user with content request will be immediately associated with a BS if there is a BS that can satisfy the minimum requested data rate of the mobile user. The RA method can rapidly associate a mobile user with a BS, and is efficient from the perspective of mobile users. However, the RA method only associates one mobile user once making the association decision, which may cause local optimization in terms of the overall average content download latency of the entire network. In this sub-section, we propose the delayed association (DA) method for a more efficient user association.

Algorithm 1. The Rapid Association (RA) Algorithm

```
Input: $u_m$, a mobile user that requests content $c_k$; $CH = \{ch_1, ch_2, \ldots, ch_v\}$, the set of sub-channels that can serve $u_m$ ;
Output: $y_{mn_j}$, the sub-channel $j$ of BS $f_n$ with which $u_m$ is associated ;
1  $CH' = \emptyset$;
2  for $i = 1$ to $v$ do
3      Calculate the date rate on sub-channel $ch_i$, denoted as $r_{mi}$ ;
4      if $r_{mi} \geq r_{m\ min}$ then
5          $CH' = CH' \cup \{ch_i\}$ ;
6  $v' = |CH'|$ ;
7  $t_{mk} = \infty$ ;
8  for $i = 1$ to $v$ do
9      if $c_k$ is cached at the BSs to which sub-channel $ch_i$ belongs then
10         // $ch_i$ is the $i$th element of $CH'$
11         $t_{mk_i} = l_k / r_{mi}$
12     else
13         $t_{mk_i} = l_k / r_{mi} + T_C$ ;
14     if $t_{mk_i} < t_{mk}$ then
15         $t_{mk} := t_{mk_i}$ ;
16         $ch^* = ch_i$ ;
17 Let $ch^*$ be the sub-channel $j$ of BS $f_n$;
18 $y_{mn_j} = 1$;
```

In the DA method, the user association decision is made in every time slice, which is called the delayed time window (T_s). In every T_s, there may be more than one mobile users waiting for their association decisions. We aim to associate these mobile users to appropriate BSs to optimize the average content download latency of these mobile users. This optimization can be modeled as an optimal matching problem in graph theory.

Let U_t be the set of mobile users that requests contents in time slice T_s, and C_t be the set of contents requested by mobile users in U_t. For a mobile user $u_t \in U_t$, $R_t(u_t) = c_t, (c_t \in C_t)$, which means that the content c_t is requested by u_t. We define a weighted bipartite graph $G_t = (U_t, CH_t, E_t, W_t)$, where CH_t is the set of available channels of BSs and E_t is the set of edges. If a mobile user $u_t \in U_t$ can access a BS through a sub-channel $ch_t \in CH_t$ of the BS, and the data rate from the BS to u_t is larger than u_t's minimum required data rate, then there is an edge e_t between u_t and ch_t, i.e., $e_t \in E$. The weight of e_t, denoted as $w(e_t)$, is the content download latency taken for u_t to download content $R_t(u_t)$. W_t is the set of weights of the edges. According to the definition of G_t, the delayed user association problem for minimizing the average content download latency is transformed to finding out the perfect matching of G_t, which can be solved by the Kuhn-Munkers (KM) algorithm [38].

Before applying the KM algorithm to obtain the perfect matching of G_t, we should transform G_t to a regular bipartite graph, by adding virtual vertex and virtual edges to G_t. We denote the regular bipartite graph as $G_t^r = \left(U_t^r, CH_t^r, E_t^r, W_t^r\right)$.

First, virtual channels or virtual mobile users are added to make $\left|U_t^r\right| = \left|CH_t^r\right|$. If $\left|U_t\right| > \left|CH_t\right|$, then $|U_t| - |CH_t|$ virtual channels are added. Otherwise, $|CH_t| - |U_t|$ virtual users are added. Then a virtual edge with a weight of ∞ is added to connect the channel $ch_t^r \in CH_t^r$ and the mobile user $u_t^r \in U_t^r$, if there is no edge connecting ch_t^r and u_t^r in G_t^r.

Algorithm 2. The Delayed Association (DA) Algorithm: Part I

```
Input: $U_t = \{u_{t,1}, u_{t,2}, \ldots, u_{t,M}\}$, the set of mobile users that request contents in time slice $t$; $C_t = \{c_t, c_t = R_t(u_{t,m}), u_{t,m} \in U_t\}$, the set of contents that are requested by mobile users in time slice $t$; $CH_t = \{ch_{t,1}, ch_{t,2}, \ldots, ch_{t,V}\}$, the set of available sub-channels of the BSs ;
Output: $G_t^r = (U_t^r, CH_t^r, E_t^r, W_t^r)$ ;
  Step 1: Construct weighted bipartite graph $G_t = (U_t, CH_t, E_t, W_t)$, $W_t = \{w_{t,m,v}\}$ , $m \in \{1, 2, \ldots, M\}$, $v \in \{1, 2, \ldots, V\}$, which is the set of weights of edges of Graph $G_t$ ;
Calculate data rate $r_{m,ch_{t,v}}$ as in (1), assuming that $u_{t,M}$ is associated with sub-channel $ch_{t,v}$ ;
if $r_{m,ch_{t,v}} \geq r_{m\min}$ then
  if $R_t(u_{t_m})$ is cached in the BS that has available sub-channel $ch_{t,v}$ then
    $w_{t,m,v} = l(R_t(u_{t,m})) / r_{m,ch_{t,v}}$ ;
  else
    $w_{t,m,v} = l(R_t(u_{t,m})) / r_{m,ch_{t,v}} + T_C$ ;
else
  $w_{t,m,v} = \infty$
 Step 2: Transfer $G_t$ to a regular bipartite graph $G_t^r = (U_t^r, CH_t^r, E_t^r, W_t^r)$ by adding virtual nodes and virtual edges to $G_t$ ;
$M = |U_t|$; $V = |CH_t|$; $H = \max(M, V)$;
Let $W_t^r = \{w_{t,m,v}^r | m, v \in [1, H]\}$ be the weight matrix of $G_t^r$ ;
if $M > V$ then
  $U_t^r \leftarrow U_t$ ;
  $CH_t^r \leftarrow CH_t + \{ch_{t,V+1}, ch_{t,V+2}, \ldots, ch_{t,M}\}$ ;
  for $m = 1$ to $M$ do
    for $v = V + 1$ to $M$ do
      $w_{t,m,v}^r = \infty$ ;
if $M < V$ then
  $CH_t^r \leftarrow CH_t$ ;
  $U_t^r \leftarrow U_t + \{u_{t,M+1}, u_{t,M+2}, \ldots, u_{t,V}\}$ ;
  for $v = 1$ to $V$ do
    for $m = M + 1$ to $V$ do
      $w_{t,m,v}^r = \infty$ ;
$a = \max(w_{t,m,v}^r | m, v \in [1, H], w_{t,m,v}^r \neq \infty)$ ;
for $m = 1$ to $H$ do
  for $v = 1$ to $H$ do
    $w_{t,m,v}^r \leftarrow a - w_{t,m,v}^r$ ;
    if $w_{t,m,v}^r = -\infty$ then
      $w_{t,m,v}^r \leftarrow 0$
```

As the KM algorithm is used to find out the maximum matching of G_t^r, but our algorithm is to obtain the optimal user association to minimize the average content download latency, we revise the edge weights in G_t^r as follows. Define $W_{|U_t^r|\times|CH_t^r|}$ as the weight matrix. Let a be the element of $W_{|U_t^r|\times|CH_t^r|}$ with the largest value besides ∞, we compute

$$W_{|U_t^r|\times|CH_t^r|} = aJ - W_{|U_t^r|\times|CH_t^r|}, \tag{16}$$

where J is an identity matrix with order $\left|G_t^r\right|$. We also set the elements with a ∞ value in $W_{|U_t^r|\times|CH_t^r|}$ to zero. After revising the edge weights in G_t^r, we apply the KM algorithm to obtain the perfect matching of $W_{|U_t^r|\times|CH_t^r|}$, which is denote as $\Phi_{|U_t^r|\times|CH_t^r|}$. Finally, we delete the virtual nodes and virtual edges in $\Phi_{|U_t^r|\times|CH_t^r|}$ to obtain the optimal user association of G_t^r.

The detailed DA algorithm is presented in Algorithms 2 and 3, which includes the following four steps:

Algorithm 3. The Delayed Association (DA) Algorithm: Part II

```
Input: $G_t^r = (U_t^r, CH_t^r, E_t^r, W_t^r)$ ;
Output: $y_{mn_j}^t$ ;
Use the KM algorithm [38] to process $G_t^r$ to obtain the perfect matching of $G_t^r$, denoted by $\Phi_{|U_t^r|\times|CH_t^r|}$ ;
Deleting the virtual nodes and virtual edges in $\Phi_{|U_t^r|\times|CH_t^r|}$, and the remaining matching of $\Phi_{|U_t^r|\times|CH_t^r|}$ denotes the optimal user association ;
for $m = 1$ to $M$ do
  if $u_{t,m}$ is matched to $ch_{t,v}$ in the optimal user association and $ch_{t,v}$ is the sub-channel $j$ of BS $f_n$ then
    $y_{mn_j}^t = 1$ ;
```

Step 1: Construct the weighted bipartite graph $G_t = (U_t, CH_t, E_t, W_t)$.

Step 2: Transfer G_t to a regular bipartite graph $G_t^r = (U_t^r, CH_t^r, E_t^r, W_t^r)$.

Step 3: Apply the KM algorithm to G_t^r to obtain the perfect matching of G_t^r.

Step 4: Delete the virtual nodes and virtual edges in $\Phi_{|U_t^r|\times|CH_t^r|}$ to obtain the matching with the minimum total weights of G_t.

Denote the matching with the minimum total weights of G_t as $\Phi_{|U_t|\times|CH_t|}$, which is obtained in Step 4 above and provides the optimal user association.

The DA algorithm includes two parts, i.e., Part I and Part II. Furthermore, the Part I algorithm consists of Step 1 and Step 2, whose computation complexities are $\mathcal{O}(|U_t| \times |CH_t|)$ and $\mathcal{O}((\max(|U_t|, |CH_t|))^2)$, respectively. For Part II of the DA algorithm, the computation complexity is determined by the KM algorithm, whose complexity is $\mathcal{O}((\max(|U_t|, |CH_t|))^3)$. Therefore, the overall computation complexity of DA algorithm is $\mathcal{O}((\max(|U_t|, |CH_t|))^3)$.

5 Performance Evaluation

5.1 Simulation Configuration

In this section, we evaluate the performance of the proposed JCC-UA algorithm and compare it with several baseline schemes. In the simulation study, a HetNet with one MBS and ten randomly deployed SBSs is created. The coverage radiuses of an SBS and the MBS are 70m and 350m, respectively [39]. The MBS is located at the center of the HetNet. There are $M = 600$ mobile users, which are randomly distributed in the network. The total system bandwidth is 20MHz. The transmit powers of the MBS and SBSs are 43dBm and 23dBm, respectively [27]. The backhaul latency T_C is set to 1s [26]. There are $K = 100$ content items. The length of each content is 10Mbits [40]. The cache capacities

TABLE 1
System Parameters Used in the Simulation Study

Parameter	*Value*
Coverage radius of MBS (R_{MBS})	350 m
Coverage radius of SBS (R_{SBS})	70 m
Number of SBSs (N)	10
Number of mobile users (M)	600
Average arrive rate of requests (λ)	20 – 100
Number of content items (K)	100
Transmission power of MBS (P_{MBS})	43 dBm
Transmission power of SBS (P_{SBS})	23 dBm
Noise power (σ_N^2)	-174 dBm/Hz
Pathloss constant (κ)	10^{-2}
Pathloss exponent (ε)	4
System bandwith (B)	20 MHz
Backhaul delay (T_C)	1 s
Content size (l_k)	10 Mbits
Cache capacity of SBS ($S_{n_{SBS}}$)	0 – 500 Mbits
Cache capacity of MBS ($S_{n_{MBS}}$)	1000 Mbits
Zipf parameter (θ)	0.2 – 2
Smoothing parameter (α)	0.1 – 0.9
Delayed time window (T_s)	0 – 0.5 s
The length of a time slot	10 – 100
The residence time of a mobile user in a cell	1 – 10

of an SBS and the MBS are 0-500Mbits and 400Mbits, respectively. A user requests a content item one at a time. The probability that contents are requested by users is subject to the Zipf distribution, and the distribution probability is given by

$$p_k = \frac{1/k^{\theta}}{\sum_{j=1}^{K} 1/j^{\theta}}, \tag{17}$$

with the shape parameter θ is set to $\theta = 0.8$ [41]. The arrival of content requests of mobile users obeys the Poisson process model, and the average arrival rate is denoted by λ. The required minimum data rate of all mobile users is $r_{m,min} = 180$Kbit/s. The detailed simulation parameters are summarized in Table 1.

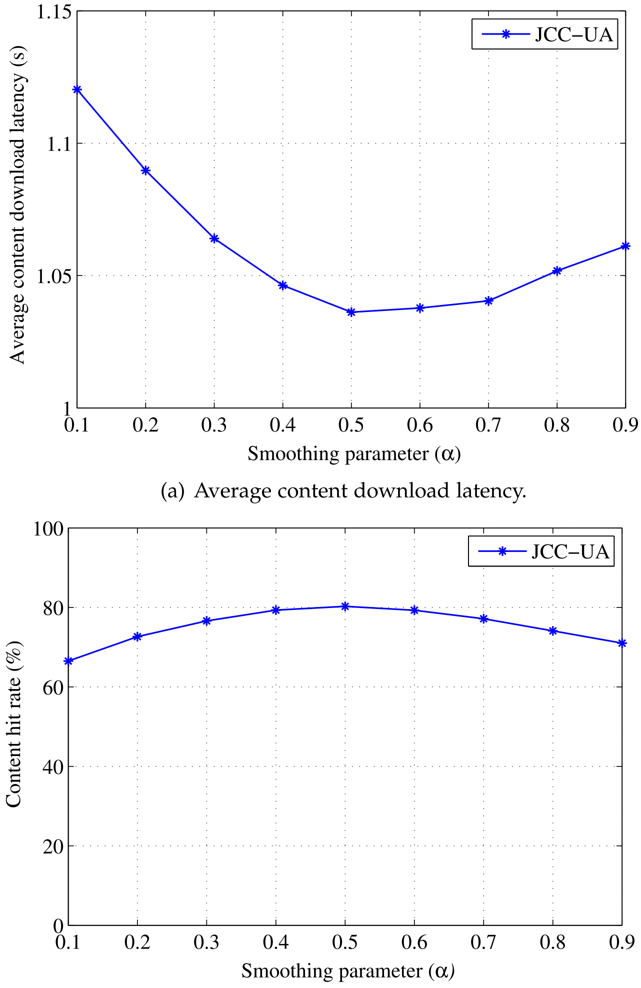
In the evaluation study, the performance metrics include average content download latency and content hit rate. The average content download latency is defined as the sum of the download latency of all the contents divided by the number of contents downloaded by the mobile users. The content hit rate is the ratio of contents directly being downloaded through the BSs to the total number of downloaded contents [27].

5.2 Impact of Design Parameters

The performance of the JCC-UA algorithm is mainly dependent on two important design parameters, i.e., the smoothing parameter (α) and the delayed time window (T_s). In this sub-section, we evaluate the impact of α and T_s on the performance of the JCC-UA algorithm.

5.2.1 Impact of Smoothing Parameters

The average content download latency and content hit rate for different values of the smoothing parameter, i.e., α, are shown in Figs. 2a and 2b, respectively. Note that the RA algorithm shown in Algorithm 1 is used for user association in this simulation. The number of mobile users is $M = 600$, and the average arrive rate of content requests is $\lambda = 20$. As shown in Fig. 2, the smoothing parameter affects in some degree the average content download latency and content hit rate. However, when the range of α is in [0.4, 0.7], the



(a) Average content download latency.

(b) Cache hit rate.

Fig. 2. Impact of the smoothing parameter α on the JCC-UA performance.

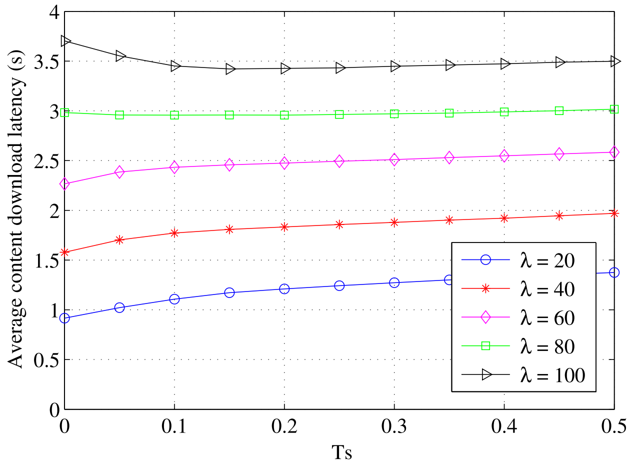


Fig. 3. Impact of the delayed time window T_s on average download latency of JCC-UA.

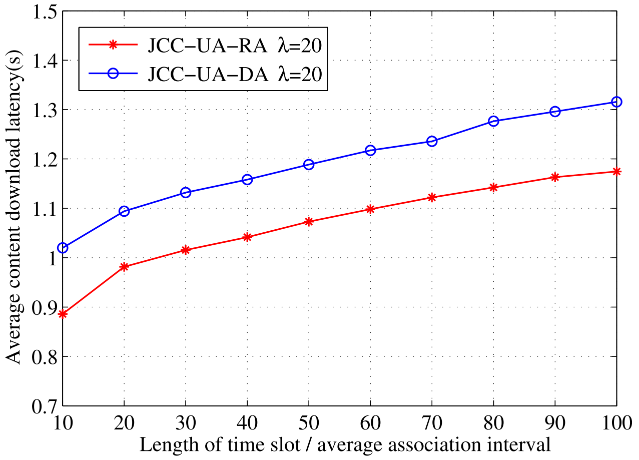


Fig. 4. Performance under different lengths of time slot i: $\theta = 0.8$, $S_{n_{SBS}} = 200$, and $\alpha = 0.5$.

average content download latency and content hit rate exhibit no obvious change, while the average content download latency achieves its mimimum values and the content hit rate achieves its maximum values for α values in this range. This means that we can accurately predict the content requests of mobile users and reasonably cache the contents in BSs with the proposed approach. Therefore, we set $\alpha = 0.5$ in the following simulations.

5.2.2 *Impact of the Delayed Time Window*

The influence of the delayed time window, i.e., T_s, on the average download latency of the JCC-UA algorithm is presented in Fig. 3, where the number of users is also set to 600. When $T_s = 0$, it means that the association method is RA shown in Algorithm 1; Otherwise, the association method is DA shown in Algorithms 2 and 3. As shown in Fig. 3, when λ is small, the average download latency increases with the grow of T_s. When λ is large, i.e., $\lambda = 80$ in Fig. 3, which means that the load of content requests is heavy, the average download latency first decreases, then increases with the grow of T_s. When λ is small, the content requests are sparse and the traffic load is light. It is unnecessary to let a mobile user wait for some time before being associated with a BS. Therefore, the RA association is a better choice than the DA association. When λ is large, there are a large number of mobile users with content requests in a T_s. We can optimally associate these mobile users to BSs in one short by the DA association method. Therefore, the DA

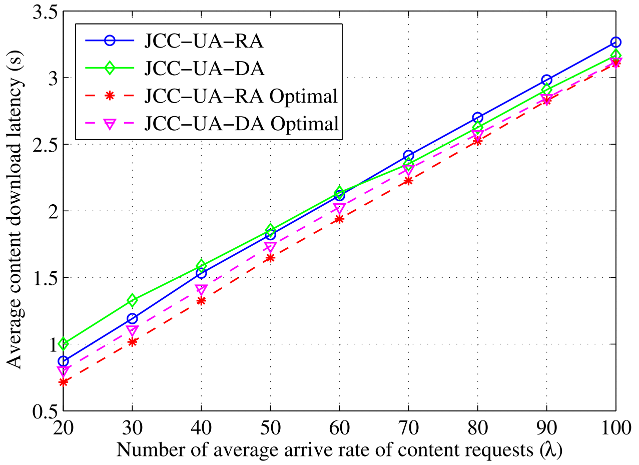


Fig. 5. Performance achieved by the optimal solution and JCC-UA: $\theta = 0.8$, $S_{n_{SBS}} = 200$, and $\alpha = 0.5$.

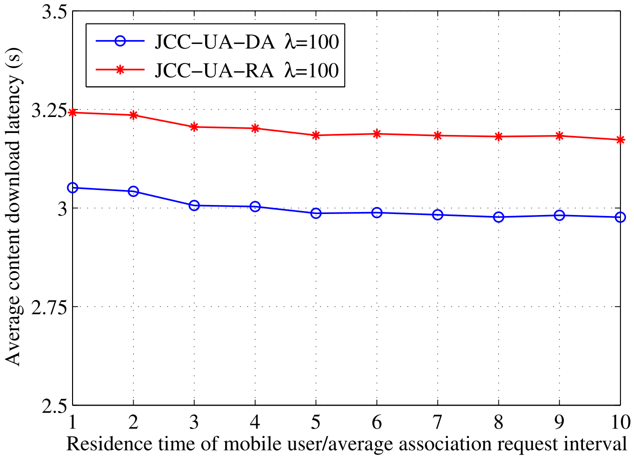
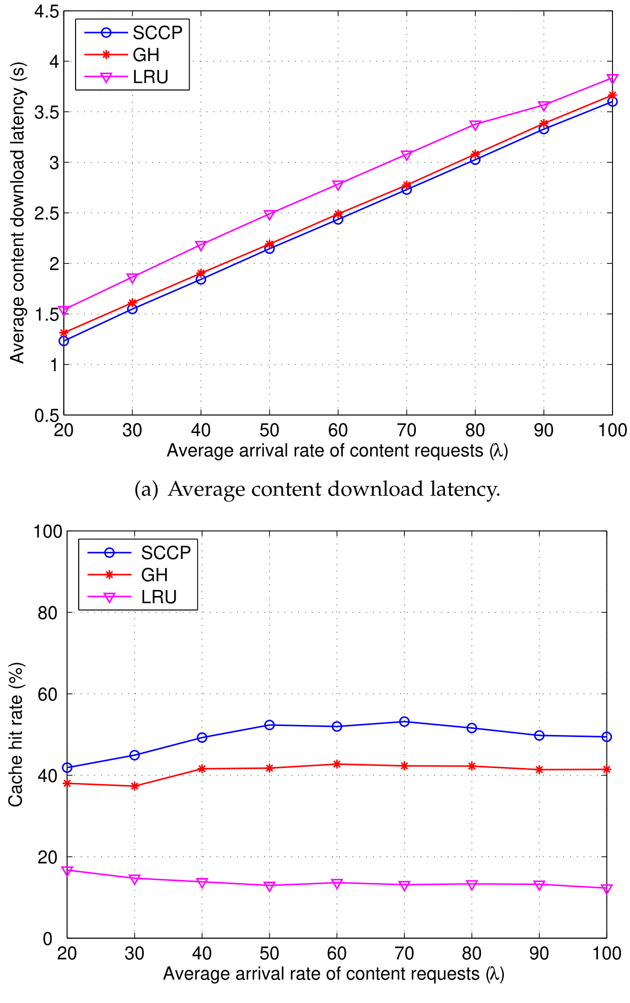


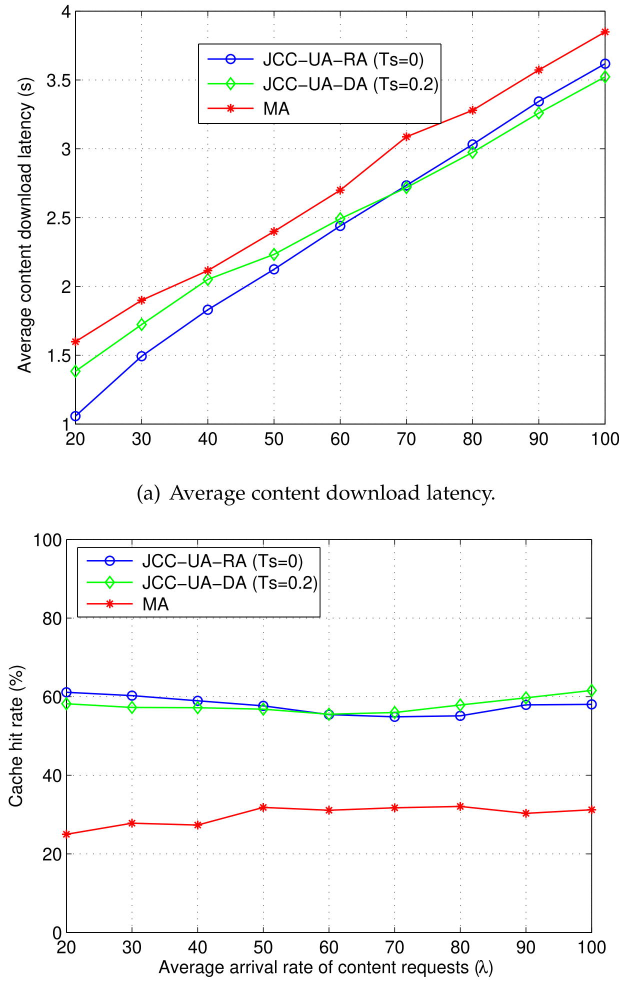
Fig. 6. Performance with the residence time of a mobile user in a cell: $\lambda = 100$, $T_s = 0.2$, $S_{n_{SBS}} = 200$, and $\alpha = 0.5$.



(a) Average content download latency.

(b) Cache hit rate.

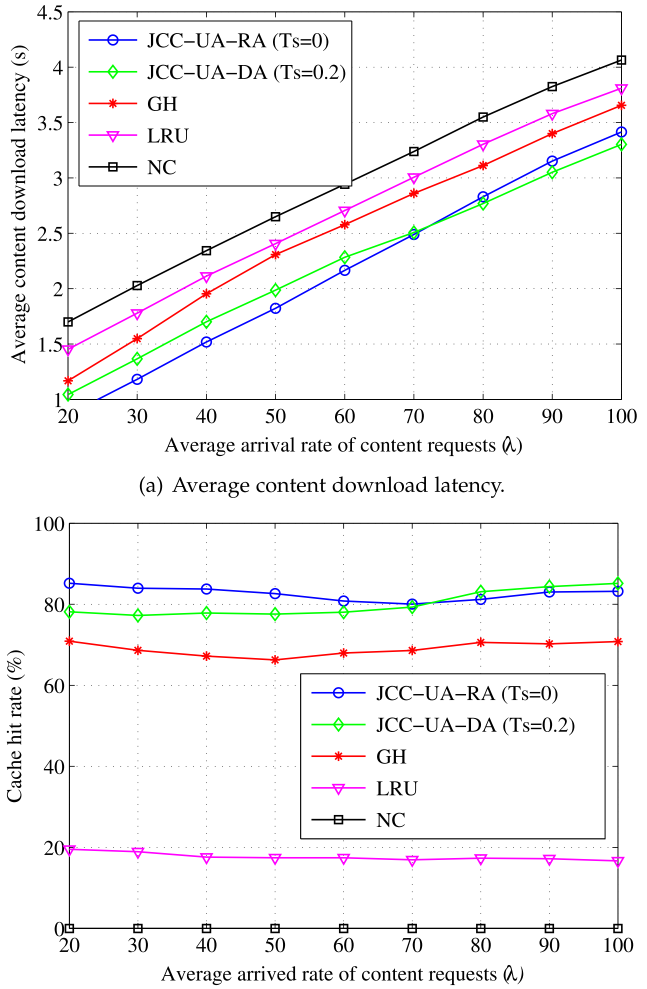
Fig. 7. Performance of the content caching policies with different arrival rates of content requests: $\theta = 0.8$, $S_{n_{SBS}} = 200$, and $\alpha = 0.5$.



(a) Average content download latency.

(b) Cache hit rate.

Fig. 8. Performance of the content caching algorithms with different arrival rates of content requests: $\theta = 0.8$, $S_{n_{SBS}} = 200$, and $\alpha = 0.5$.



(a) Average content download latency.

(b) Cache hit rate.

Fig. 9. The performance with different arrival rate of content requests: $\theta = 0.8$, $S_{n_{SBS}} = 200$, and $\alpha = 0.5$.

association is a better choice than the RA association when the λ value is large.

5.2.3 *Impact of Time Slot Length*

The length of a time slot does influence the performance of the proposed algorithms, as shown in Fig. 4. In the figure, the length of a time slot is related to the average association internal of mobile users. On one hand, while increasing the length of time slot, the content caching update may not be able to catch up with the changing speed of content popularity, which will increase the content download latency. On the other hand, a too short time slot may cause frequent content caching updates, which is also undesirable. Therefore, we should select a rational region for the time slot length. From this simulation, it is appropriate to set the length of a time slot to 20–50 times of the average user association interval.

5.2.4 *Comparison With the Optimal Solution*

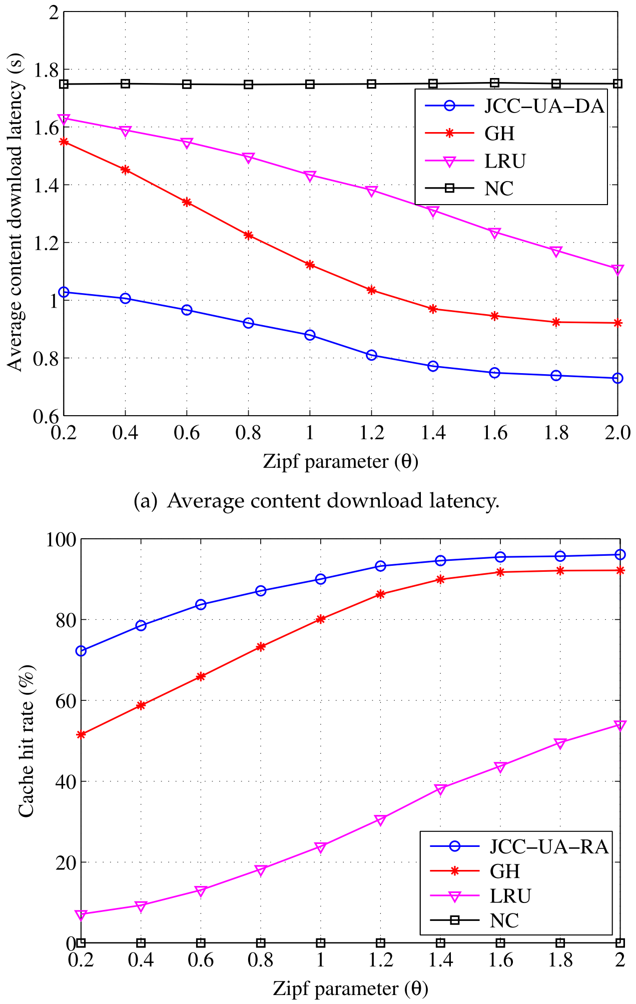
We compare the performance obtained by the proposed algorithms with the optimal solution derived for a small scenario, with $N = 10$, $M = 100$, $\alpha = 0.5$, $\theta = 0.8$, and $S_{n_{SBS}} = 200$. The comparison results are presented in Fig. 5. From the figure, it can be seen that the gap between the optimal solution curve and the proposed algorithm curve is generally small, and the gap decreases with the increased arrival rate of content requests (i.e., λ). For example, when λ is 70, the relative optimality gap is less than 6 percent.

5.2.5 *Impact of User Mobility*

In fact, user mobility will affect the residence time of a mobile user in a cell, which in turn affects the user association decision. To evaluate the impact of user mobility on the performance of the proposed algorithms, we simulate the content download latency for different residence times of mobile users. The simulation results are presented in Fig. 6, which shows that the content download latency increases when the residence time is decreased. This is because when the residence time is decreased (i.e., with higher user mobility), it is more difficult to catch up with the mobility of mobile users and make accurate content caching decisions. As a result, more mobile users download contents from the cloud center, which in turn increase the content download latency.

5.3 Comparison With Baseline Schemes

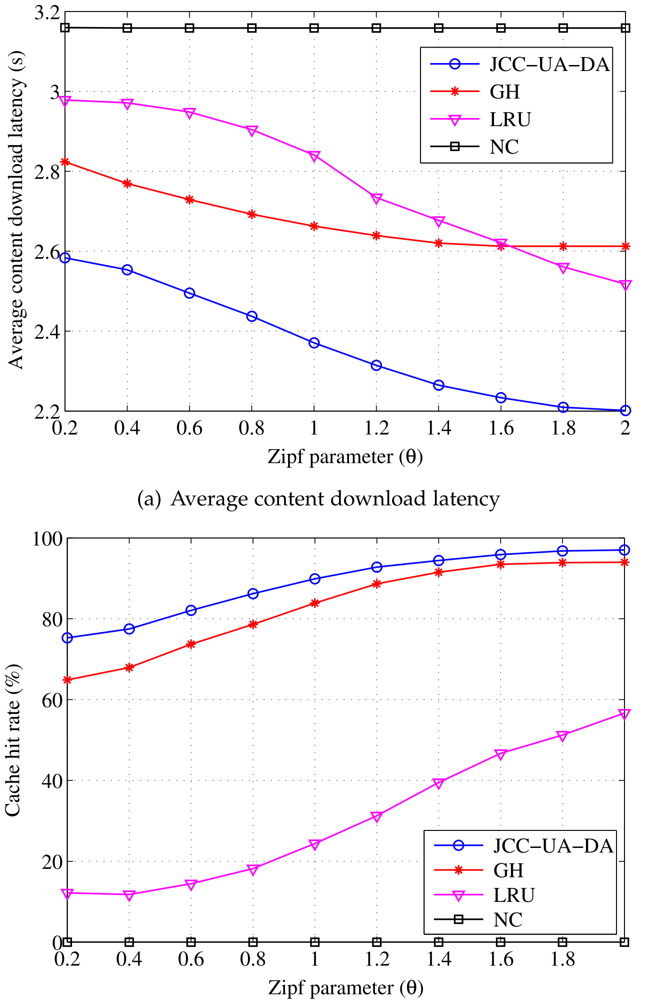
In this section, we compare the proposed algorithms with several baseline schemes, including (i) the *no caching* (NC)



(a) Average content download latency.

(b) Cache hit rate.

Fig. 10. The performance with various Zipf parameters for the light content request scenario: $\lambda = 20$, $T_s = 0$, $S_{n_{SBS}} = 200$, and $\alpha = 0.5$.



(a) Average content download latency

(b) Cache hit rate

Fig. 11. The performance with various Zipf parameters for the heavy content request scenario, $\lambda = 100$, $T_s = 0.2$, $S_{n_{SBS}} = 200$, and $\alpha = 0.5$.

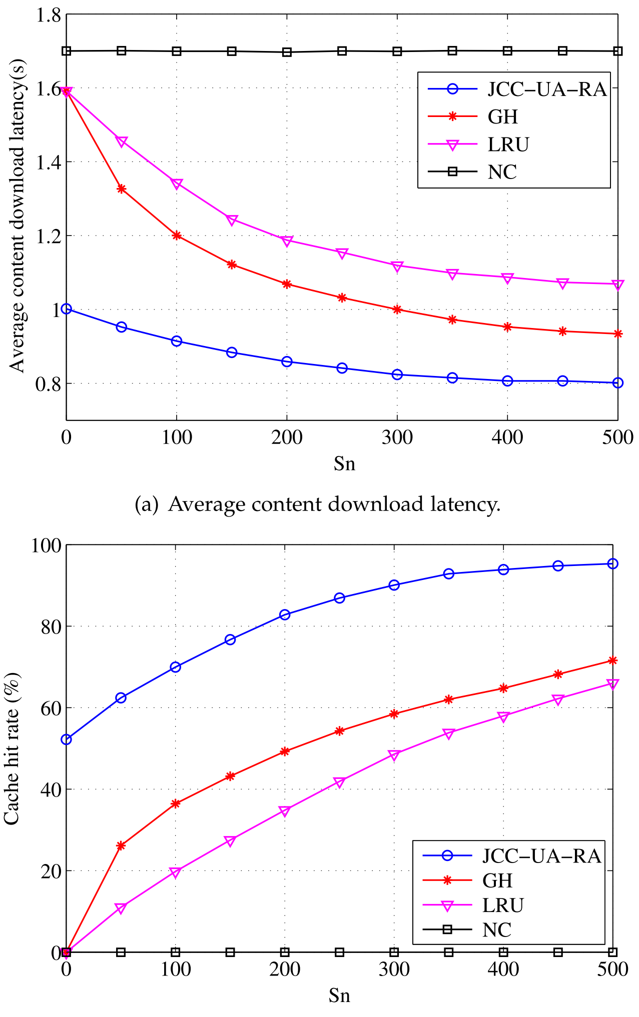
algorithm that does not cache contents in the SBSs and MBS, (ii) the *least recently used* (LRU) algorithm that discards the least used contents and caches the more recently requested contents [20], and (iii) the *greedy helper* (GH) algorithm that caches contents in the BSs via an iteration manner according to the popularity of contents [18]. We first evaluate the performance of the proposed content caching policy and the user association algorithms separately to demonstrate the benefit of jointly considering both content caching and user association in the proposed scheme. We then evaluate the performance of JCC-UA algorithm under various practical settings.

5.3.1 Evaluation of the Separate Content Caching and the User Association Algorithms

We first compare the proposed content caching policy and the user association algorithms separately with related works by simulations. The simulation results for different content caching policies are shown in Fig. 7, where the traditional maximum-rate association algorithm (MA) is used. In MA, a mobile user is associated to the BS with which the associated mobile user can obtain the maximum data rate. In Fig. 7, both the average content download latency and the cache hit rate of the proposed SCCP policy are better than that of the compared GH and LRU policies. Fig. 8 shows the performance of different user association algorithms, where the random content caching policy is used with them. As the proposed RA and DA algorithms associate mobile users based the the content location and the data rate, RA and DA outperform MA with considerable margins.

5.3.2 Performance Under Different Arrival Rates of Content Requests

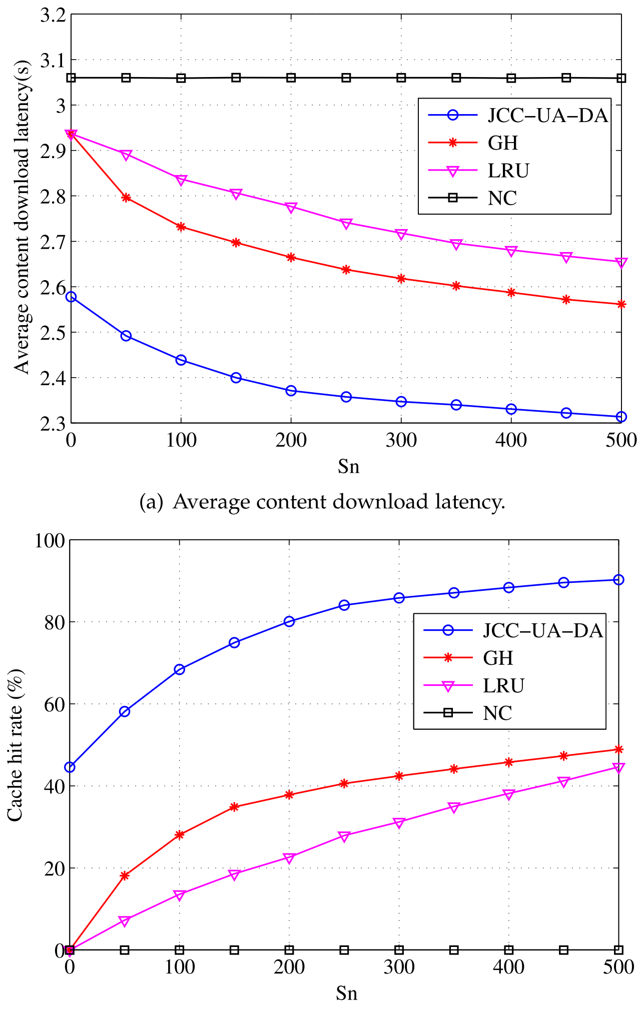
The average content download latency and the content hit rate for different arrival rates of content requests, i.e., λ, are presented in Figs. 9a and 9b, respectively. As the bandwidth of the BSs is limited, when the value of λ is increased, there are more mobile users that cannot be associated to the BSs that cache the request contents. Thus they have to download the requested contents from the cloud center, which increases the average content download latency. However, it is also interesting to see that the cache hit rate only changes lightly as the content request rate λ is increased.



(a) Average content download latency.

(b) Cache hit rate.

Fig. 12. The performance with various cache capacities at the BSs for the light content request scenario: $\theta = 0.8$ and $\alpha = 0.5$.



(a) Average content download latency.

(b) Cache hit rate.

Fig. 13. The performance with various cache capacities at the BSs for the heavy content request scenario: $\lambda = 100$, $T_s = 0.2$, and $\alpha = 0.5$.

Comparing to the NC, LRU, and GH algorithms, the JCC-UA algorithm obviously reduces the average content download latency and increases the content hit rate. This is because the JCC-UA algorithm can smartly cache the contents and associate mobile users to the BSs. In Fig. 9, it is worth noting that when λ is large, the DA algorithm (i.e., when $T_s = 0.2$) outperforms the RA algorithm (i.e., when $T_s = 0$) as DA makes more globally optimal decision by associating more mobile users in a time slice.

5.3.3 Performance With Different Zipf Parameters

In this simulation, we evaluate the performance of JCC-UA under two scenarios, i.e., (i) a light content request scenario ($\lambda = 20$) and (ii) a heavy content request scenario ($\lambda = 100$).

Figs. 10 and 11 present the average content download latency and content hit rate for different values of the Zipf parameter θ. Figs. 10a and 10b are for the light content request scenario ($\lambda = 20$), where the RA association method is applied. Figs. 11a and 11b are for the heavy content request scenario ($\lambda = 100$), where DA association method is used. When the value of θ is increased, the users' requests are more concentrated to some contents, which means that most mobile users request the small set of popular contents. Therefore, we can only cache these popular contents in the BSs to satisfy the requirements of many mobile users. This reduces the average content download latency and increase the content hit rate. Figs. 10 and 11 also demonstrate that the JCC-UA algorithm outperforms all the three baseline schemes with considerable margins in all the cases examined in this simulation.

5.3.4 Performance With Different Cache Capacities

Figs. 12 and 13 present the average content download latency and content hit rate for different values of the cache capacity S_n at the SBSs. Similar to the previous sub-section, we consider the light content request scenario ($\lambda = 20$) and the heavy content request scenario ($\lambda = 100$). The RA association and the DA association are respectively used for these two scenarios. Figs. 12 and 13 plot the simulation results. When S_n is increased, more contents can be stored at the BSs, and in turn more mobile users can directly download the requested contents from the SBSs. Thus the average content download latency decreases and the content hit rate increases. In Figs. 12 and 13, the performance of the NC algorithm is irrelevant to S_n as the NC algorithm does not cache contents at the SBSs. These two figures also demonstrate that the JCC-UA algorithm outperforms the all the three baseline schemes with considerable margins in all the cases examined in this simulation.

6 Conclusion

This paper investigated the content caching and user association problem in the context of edge computing in HetNets. The joint optimization problem was formulated and proved to be NP-hard. Then a content caching algorithm based on the cubic exponential smoothing was proposed to smartly cache contents in BSs, and two user association algorithms, i.e., the RA algorithm and the DA algorithm, were proposed to dynamically associate mobile users to different BSs to minimize the average content download latency. The comprehensive evaluation study showed our proposed algorithm could achieve the lowest average content download latency and the highest cache hit rate compared to the three baseline schemes. For future work, it would be interesting to jointly consider user mobility, content caching, and user association to reduce the content download latency. It would also be interesting to derive performance bounds for the proposed algorithms.

Acknowledgments

This work was supported in part by the Program for the National Science Foundation of China (61671096 and 62071077), the Chongqing Research Program of Basic Science and Frontier Technology (cstc2017jcyjBX0005), Chongqing Science and Technology Innovation Leading Talent Support Program (CSTCCXLJRC201710), and the NSF under Grant ECCS-1923717.

References

[1] Cisco VNI, "Cisco visual networking index: Forecast and methodology 2017–2022 white paper," *Cisco*, 2020. [Online]. Available: https://www.cisco.com/c/en/us/solutions/service-provider/visual-networking-index-vni/index.html

[2] X. Wang, A. V. Vasilakos, M. Chen, Y. Liu, and T. T. Kwon, "A survey of green mobile networks: Opportunities and challenges," *Springer Mobile Netw. Appl. J.*, vol. 17, no. 1, pp. 4–20, Feb. 2012.

[3] V. Chandrasekhar, J. Andrews, and A. Gatherer, "Femtocell networks: A survey," *IEEE Commun. Mag.*, vol. 46, no. 9, pp. 59–67, Sep. 2008.

[4] D. C. Chen, T. Q. S. Quek, and M. Kountouris, "Backhauling in heterogeneous cellular networks: Modeling and tradeoffs," *IEEE Trans. Wireless Commun.*, vol. 14, no. 6, pp. 3194–3206, Jun. 2015.

[5] 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. Magazine*, vol. 51, no. 4, pp. 142–149, Apr. 2013.

[6] X. Wang, M. Chen, T. Taleb, A. Ksentini, and V. C. Leung, "Cache in the air: Exploiting content caching and delivery techniques for 5G systems," *IEEE Commun. Mag.*, vol. 52, no. 2, pp. 131–139, Feb. 2014.

[7] Z. Chang, L. Lei, Z. Zhou, S. Mao, and T. Ristaniemi, "Learn to cache: Machine learning for network edge caching in the big data era," *IEEE Wireless Commun. Mag.*, vol. 25, no. 3, pp. 28–35, Jun. 2018.

[8] B. A. Ramanan, L. M. Drabeck, M. Haner, N. Nithi, T. E. Klein, and C. Sawkar, "Cacheability analysis of HTTP traffic in an operational LTE network," in *Proc. IEEE Wireless Telecommun. Symp.*, 2013, pp. 1–8.

[9] F. Boccardi, R. W. H. Jr, A. Lozano, T. L. Marzetta, and P. Popovski, "Five disruptive technology directions for 5G," *IEEE Commun. Mag.*, vol. 52, no. 2, pp. 74–80, Feb. 2013.

[10] Y. Mao, C. You, J. Zhang, K. Huang, and K. B. Letaief, "A survey on mobile edge computing: The communication perspective," *IEEE Commun. Surveys Tutl.*, vol. 19, no. 4, pp. 2322—2358, Fourth-Quarter 2017.

[11] Y. Sun, M. Peng, and S. Mao, "A game-theoretic approach to cache and radio resource management in fog radio access networks," *IEEE Trans. Veh. Technol.*, vol. 68, no. 10, pp. 10 145–10 159, Oct. 2019.

[12] T. Zhang and S. Mao, "Cooperative caching for scalable video transmissions over heterogeneous networks," *IEEE Netw. Lett.*, vol. 1, no. 2, pp. 63–67, Jun. 2019.

[13] L. T. Tan, R. Q. Hu, and L. Hanzo, "Heterogeneous networks relying on full-duplex relays and mobility-aware probabilistic caching," *IEEE Trans. Commun.*, vol. 67, no. 7, pp. 5037–5052, Jul. 2019.

[14] K. C. Tsai, L. L. Wang, and Z. Han, "Caching for mobile social networks with deep learning: Twitter analysis for 2016 US election," *IEEE Trans. Netw. Sci. Eng.*, vol. 7, no. 1, pp. 193–204, First Quarte 2020.

[15] J. Li, Y. Chen, Z. Lin, C. Wen, and L. Hanzo, "Distributed caching for data dissemination in the downlink of heterogeneous networks," *IEEE Trans. Commun.*, vol. 63, no. 10, pp. 3553–3568, Oct. 2015.

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

[17] R. Tandon and O. Simeone, "Cloud-aided wireless networks with edge caching: Fundamental latency trade-offs in fog radio access networks," in *Proc. IEEE Int. Symp. Inf. Theory*, 2016, pp. 2029–2033.

[18] 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.

[19] J. Li, W. Chen, M. Xiao, F. Shu, and X. Liu, "Efficient video pricing and caching in heterogeneous networks," *IEEE Trans. Veh. Technol.*, vol. 65, no. 10, pp. 8744–8751, Oct. 2016.

[20] P. Blasco and D. Gunduz, "Learning-based optimization of cache content in a small cell base station," in *Proc. IEEE Int. Conf. Commun.*, 2014, pp. 1897–1903.

[21] B. N. Bharath, K. G. Nagananda, and H. V. Poor, "A learning-based approach to caching in heterogenous small cell networks," *IEEE Trans. Commun.*, vol. 64, no. 4, pp. 1674–1686, Apr. 2016.

[22] C. Min, Y. Hao, H. Long, L. Zheng, and V. Lau, "Green and mobility-aware caching in 5G networks," *IEEE Trans. Wireless Commun.*, vol. 16, no. 12, pp. 8347–8361, Dec. 2017.

[23] K. Poularakis, "Approximation algorithms for mobile data caching in small cell networks," *IEEE Trans. Commun.*, vol. 62, no. 10, pp. 3665–3677, Oct. 2014.

[24] F. Pantisano, M. Bennis, W. Saad, and M. Debbah, "Cache-aware user association in backhaul-constrained small cell networks," in *Proc. 12th Int. Symp. Model. Optim. Mobile Ad Hoc Wireless Netw.*, 2014, pp. 37–42.

[25] F. Pantisano, M. Bennis, W. Saad, and M. Debbah, "Match to cache: Joint user association and backhaul allocation in cache-aware small cell networks," in *Proc. IEEE Int. Conf. Commun.*, 2015, pp. 3082–3087.

[26] Y. Wang, X. Tao, X. Zhang, and G. Mao, "Joint caching placement and user association for minimizing user download delay," *IEEE Access J.*, vol. 4, pp. 8625–8633, 2017.

[27] J. Wei, F. Gang, and Q. Shuang, "Optimal cooperative content caching and delivery policy for heterogeneous cellular networks," *IEEE Trans. Mobile Comput.*, vol. 16, no. 5, pp. 1382–1393, May 2017.

[28] Y. Chen, M. Ding, J. Li, Z. Lin, G. Mao, and L. Hanzo, "Probabilistic small-cell caching: Performance analysis and optimization," *IEEE Trans. Veh. Technol.*, vol. 66, no. 5, pp. 4341–4354, May 2017.

[29] M. Dehghan *et al.*, "On the complexity of optimal request routing and content caching in heterogeneous cache networks," *IEEE/ACM Trans. Netw.*, vol. 25, no. 3, pp. 1635–1648, Jun. 2017.

[30] S. Chen, L. Qiu, and X. Liang, "Joint subcarrier assignment and user association for partial caching-based small cell networks," in *Proc. 3rd IEEE Int. Conf. Comput. Commun.*, 2017, pp. 595–599.

[31] B. Dai and W. Yu, "Joint user association and content placement for cache-enabled wireless access networks," in *Proc. IEEE Int. Conf. Acoust. Speech Signal Process.*, 2016, pp. 3521–3525.

[32] J. Zou, C. Li, C. Zhai, H. Xiong, and E. Steinbach, "Joint pricing and cache placement for video caching: A game theoretic approach," *IEEE J. Sel. Areas Commun.*, vol. 37, no. 7, pp. 1566–1583, Jul. 2019.

[33] 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, pp. 1737–1750, Aug. 2018.

[34] T. X. Vu, S. Chatzinotas, B. Ottersten, and A. V. Trinh, "Full-duplex enabled mobile edge caching: From distributed to cooperative caching," *IEEE Trans. Wireless Commun.*, vol. 19, no. 2, pp. 1141–1153, Feb. 2020.

[35] B. Dai, W. Yu, and Y.-F. Liu, "Cloud radio access network with optimized base-station caching," in *Proc. IEEE Int. Conf. Acoust. Speech Signal Process.*, 2018, pp. 3764–3768.

[36] H. W. Kuhn, "The Hungarian method for the assignment problem," *Naval Res. Logistics*, vol. 52, no. 1, pp. 7–21, Jan. 2010.

[37] J. Mohammed, S. Bahadoorsingh, N. Ramsamooj, and C. Sharma, "Performance of exponential smoothing, a neural network and a hybrid algorithm to the short term load forecasting of batch and continuous loads," in *Proc. IEEE Manchester PowerTech*, 2017, pp. 1–6.

[38] H. Zhu, R. Alkins, and R. Alkins, "Group role assignment via a kuhn-munkres algorithm-based solution," *IEEE Trans. Syst. Man Cybern., A*, vol. 42, no. 3, pp. 739–750, May 2012.

[39] J. Gong, S. Zhou, Z. Zhou, and Z. Niu, "Joint optimization of content caching and push in renewable energy powered small cells," in *Proc. IEEE Int. Conf. Commun.*, 2016, pp. 1–6.

[40] J. Kwak, B. L. Long, and X. Wang, "Two time-scale content caching and user association in 5G heterogeneous networks," in *Proc. IEEE Global Commun. Conf.*, 2018, pp. 1–6.

[41] Y. Zhi, C. Shuang, O. Yang, and H. Liu, "Energy efficiency analysis of cache-enabled two-tier HetNets under different spectrum deployment strategies," *IEEE Access J.*, vol. 5, no. 1, pp. 6791–6800, Mar. 2017.

Yun Li (Member, IEEE) received the PhD degree in communication engineering from the University of Electronic Science and Technology of China. He is currently a professor of electrical engineering at the College of Communications, Chongqing University of Posts and Telecommunication, China. His research interests include mobile cloud computing, cooperative/relay communications, green wireless communications, wireless ad hoc networks, sensor networks, and virtual wireless networks. He (co-)authored more than 150 journal/conference articles. He is the executive associate editor of Elsevier/CQUPT Digital Communications and Networks (DCN). He is on the editorial board for IEEE Access and Wiley Security and Communication Networks. He served as a co-chair for ChinaCom 2010 WCN Symposium, IEEE RWS2011 DSPAW Symposium. He has also served as a TPC member for numerous conferences including IEEE GLOBECOM, IEEE WCNC, WiCON, CNC2012, WOCC, IWCMC, and WiCOM.

Hui Ma is currently working toward the master's degree in the School of Electronics and Communications Engineering, Chongqing University of Posts and Telecommunications, Chongqing, China. His research interests include edge cache and mobile cloud computing in heterogeneous networks.

Lei Wang received the MS degree in electronics and communications engineering from the Chongqing University of Posts and Telecommunications, Chongqing, China, in 2018. She is currently working as an engineer with China Mobile Group Hubei Co., Ltd. Her current research interest includes cloud and edge computing.

Shiwen Mao (Fellow, IEEE) received the PhD degree in electrical engineering from Polytechnic University, Brooklyn, NY, in 2004. Currently, he is a professor and Earle C. Williams eminent scholar, and director of the Wireless Engineering Research and Education Center with Auburn University, Auburn, AL. His research interests include wireless networks, multimedia communications, and smart grid. He is on the editorial board of the *IEEE Transactions on Wireless Communications*, the *IEEE Transactions on Network Science and Engineering*, the *IEEE Transactions on Mobile Computing*, the *IEEE Internet of Things Journal*, the *IEEE Open Journal of the Communications Society*, the *IEEE Multimedia*, the *IEEE Networking Letters*, ACM GetMobile, and KeAi Digital Communications and Networks Journal, etc. He is the TPC co-chair of IEEE INFOCOM 2018 and TPC vice chair of IEEE GLOBECOM 2022. He received the IEEE ComSoc TC-CSR Distinguished Technical Achievement Award in 2019 and NSF CAREER Award in 2010. He is a co-recipient of the IEEE Vehicular Technology Society 2020 Jack Neubauer Memorial Award, the IEEE ComSoc MMTC 2018 Best Journal Award and 2017 Best Conference Paper Award, the Best Demo Award from IEEE SECON 2017, the Best Paper Awards from IEEE GLOBECOM 2019, 2016 & 2015, IEEE WCNC 2015, and IEEE ICC 2013, and the 2004 IEEE Communications Society Leonard G. Abraham Prize in the Field of Communications Systems.

Guoyin Wang (Senior Member, IEEE) received the BS, MS, and PhD degrees in computer science and technology from Xi'an Jiaotong University, Xi'an, China, in 1992, 1994, and 1996, respectively. During 1998–1999, he was a visiting scholar with the University of North Texas, Denton, TX, and the University of Regina, Reginca, SK, Canada. Since 1996, he has been working at the Chongqing University of Posts and Telecommunications, where he is currently a professor and PhD supervisor, the director of the Chongqing Key Laboratory of Computational Intelligence, and a vice-president of the University. His research interests include data mining, machine learning, rough set, granular computing, cognitive computing, etc. He had been the president of International Rough Set Society (IRSS) from 2014 to 2017. He is currently a vice-president of the Chinese Association for Artificial Intelligence (CAAI), a fellow of IRSS/CAAI/CCF.