


Research Article

Joint Optimization of Content Placement and User Association in Cache-Enabled Heterogeneous Cellular Networks Based on Flow-Level Models

Hua Qu,^{1,2} Gongye Ren ,¹ Jihong Zhao,^{1,2,3} Zhenjie Tan,¹ and Shuyuan Zhao ¹

¹School of Electronic and Information Engineering, Xi'an Jiaotong University, Xi'an 710054, China

²Suzhou Caiyun Network Technologies Co., Ltd, Suzhou 215000, China

³School of Telecommunication and Information Engineering, Xi'an University of Posts and Telecommunications, Xi'an 710061, China

Correspondence should be addressed to Gongye Ren; rengongye@stu.xjtu.edu.cn

Received 4 September 2018; Accepted 31 October 2018; Published 19 November 2018

Academic Editor: Patrick Seeling

Copyright © 2018 Hua Qu et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Cache-enabled heterogeneous cellular networks (HCNs) have been investigated extensively to alleviate backhaul congestion and reduce content delivery delay. In this paper, we jointly optimize content placement and user association to minimize the average content delivery delay in cache-enabled HCNs based on flow-level models. This formulation considers (1) different timescales of content placement and content delivery, (2) locality of content popularity, and (3) the heterogeneity of spatial traffic distribution, which are often neglected in existing researches. The joint optimization problem is formulated as a mixed integer nonlinear programming problem in load-non-coupled and load-coupled models, respectively. We decouple this problem into two interrelated subproblems and resolve them individually. For the user association problem under a given content placement situation, we propose a content-level selective association algorithm, which allows the requests for different contents at the same location to connect to different base stations (BSs). In addition, we propose a greedy content caching algorithm to add contents to the caches of BSs in an iterative manner. These two algorithms are alternately executed until the caches of all the BSs are filled to capacity. Simulation results show that the proposed algorithm achieves better performance in terms of average delay and backhaul usage compared with traditional content placement and user association approaches.

1. Introduction

Driven by the proliferation of smart devices and abundant applications, the past decade witnesses a sharp rise in mobile data traffic. It is predicted that the global mobile data traffic will reach 49 exabytes per month by 2021 [1]. An effective approach to address the explosively growing data volume is to deploy plenty of low-power small base stations (SBSs) together with traditional macrobase stations (MBSs) to form a heterogeneous cellular network (HCN) [2, 3]. The densely deployed SBSs can meet the huge demand for high-speed data traffic in hot-spots and fill coverage holes of macrocells. However, deploying high-speed backhaul links to connect massive SBSs to core networks brings about huge capital and operational expenditures, which are unaffordable for network operators.

Among the huge data traffic, mobile video streaming is expected to account for 78% of the total data traffic in 2021. Many studies have shown that video streaming in wireless networks exhibits significant regularity [4–6]. Particularly, a few popular contents are requested frequently by different users at different times, which is referred to as asynchronous content reuse [7]. Repetitive content transmission raises congestion in backhaul links and core networks, especially at peak hours, which increases the content retrieval delay and decreases the efficiency of content delivery. This issue is further aggravated by the limited-capacity backhaul links of SBSs. Borrowing the concept of information-centric networking in wired networks [8], caching at the wireless edge [9–12] has been proposed to reduce the backhaul usage via equipping BSs and mobile devices with low-cost cache units. In the cache-enabled cellular networks, majority

of requested contents can be directly obtained from local storage, nearby devices or BSs, which significantly alleviates the backhaul congestion and reduces the download delay. The implementation of content-centric networking paradigm in HCNs can unleash the potential of HCNs and is an important candidate technology for the fifth generation communication systems [10, 11].

1.1. Related Work. Many studies focus on the performance analysis of cache-enabled communication networks. In [13], the performance of a coded caching scheme is analyzed from the perspective of information theory. The work of [14] contrasts the cache-enabled device-to-device (D2D) content delivery with other alternative approaches and demonstrates its superiority. The authors of [15] investigate the scaling law of link rates with respect to some network size parameters in a cache-enabled wireless network and analyze the sustainability of this network. Using tools from stochastic geometry, [16] derives the expressions of outage probability and average delivery rate in a cache-enabled small cell network and analyzes the impacts of some network parameters on the system performance. Likewise, Yang *et al.* [17] deduce the outage probability and average ergodic rate in cache-enabled HCNs. Reference [18] investigates the energy efficiency of the cache-enabled wireless access networks and analyzes the effects of some factors on it.

Generally, the content delivery in cache-enabled networks consists of two phases [13]: content placement phase and content delivery phase. In the content placement phase, some strategic contents are prefetched via backhaul links and cached at BSs during off-peak hours. In the context of D2D networks, contents can also be predownloaded from BSs to devices and cached at devices. The content placement schemes are crucial to the performance of cache-enabled networks and are studied extensively. In [19], the content placement problem is formulated as maximizing a monotone submodular function over matroid constraints and a greedy algorithm is proposed to solve it. Reference [20] proposes a distributed belief propagation algorithm to solve the content placement problem, which is aimed at minimizing the download latency. Taking into account BS cooperation and the propagation delay in backhaul links, Peng *et al.* [21] propose a low-complexity algorithm to optimize the caching placement strategy. The above-mentioned content placement schemes are all implemented in concrete network scenarios, and the outputs of these schemes are caching states of given contents at specific BSs or devices. These schemes are termed deterministic caching policies. Another line of work focuses on the probabilistic caching policy, which optimizes a caching distribution for a group of cache-enabled nodes. These studies often model the networks based on stochastic geometry and find the caching policies that optimize the derived performance metrics. In [22], an optimal probabilistic caching policy is proposed to maximize the content hit probability, which can be defined based on either signal-to-interference-plus-noise ratio (SINR) model or Boolean model. Reference [23] proposes the tier-level content placement policies in HCNs. Taking into consideration millimeter-wave and full-duplex communications,

[24] proposes a content dissemination mechanism based on proactive content fetching in cache-enabled full-duplex D2D networks and analyzes its performance from an evolutionary perspective.

After contents have been cached at BSs and/or devices, content delivery schemes direct how to deliver the requested contents to users, e.g., routing, resource allocation, transmission schemes, and so on. In [25], the cache-aware user association problem is formulated as a one-to-many game and the objective is to minimize the backhaul usage at each SBS. By leveraging both physical and social characteristics, [26] jointly optimizes user pairing, channel allocation and power control in cache-enabled D2D networks. Cheng *et al.* [27] propose three power allocation algorithms with different objectives in cache-aided small cell networks with limited backhaul. In a given caching situation, [28] proposes a distributed relaxing-rounding algorithm to jointly optimize user association and resource allocation in small cell networks. Reference [29] studies the multicast scheduling in cache-enabled wireless networks. By formulating the optimization problem as a Markov decision process, the authors analyze the structure of the optimal policy and propose a low-complexity suboptimal policy.

The performance of content-centric networking paradigm highly depends on the cooperation between content placement and content delivery. Content placement determines the upper bound of the performance of content delivery, and content delivery is implemented in a given content placement situation. Joint optimization of content placement and content delivery takes into account the interaction between these two aspects and can improve the performance dramatically compared with separate approaches. Reference [30] formulates the joint routing and caching problem as a variant of the facility location problem, and proposes an approximation algorithm to solve this NP-hard problem. In [31], the joint optimization of request routing and content caching in a network with given topology is investigated. The authors propose approximate solutions for this problem in congestion-insensitive and congestion-sensitive models, respectively. In [32], the authors develop the optimal cooperative content caching and delivery policy in a network where both BSs and devices have cache capacity. The work in [33] jointly optimizes caching, routing, and channel assignment in a collaborative small cell networks, in which network coding is applied to enable multiple BSs to cooperatively transmit contents to a user. Reference [34] considers multicasting in cache-enabled HCNs and optimizes the content caching at different tiers to maximize the successful transmission probability. The authors of [35] design a probabilistic caching policy and a random scheduling policy to maximize the successful offloading probability in cache-enabled D2D networks.

1.2. Motivation, Contributions, and Organization. Although the joint optimization of content placement and content delivery has been investigated extensively in the literature, there are three important issues that are not considered in the state-of-the-art researches, as specified in the following.

First, the timescale of content placement is much larger than that of content delivery. The rate of content placement should accord with the variation of content library and the content popularity distribution over it, which can be deemed constant during a few days [7]. The timescale of content delivery, which is affected by user mobility and user activity, ranges from seconds to minutes. Existing works, such as [32, 33, 36, 37], often optimize content placement at the timescale of content delivery; i.e., the content placement is designed based on a snapshot of a network with given user distribution. When user distribution changes, the previously derived content placement scheme is no longer optimal and it should be updated based on the new user distribution. Frequent content replacement due to user mobility and user activity makes the backhaul links congested, and the advantages of caching diminish or even become negative. Moreover, because of the combinatorial characteristic of content placement problems, their solutions often have relatively high complexity, and they are not practicable in a highly dynamic scenario.

Second, the local content popularity often differs from the general content popularity. Content popularity distribution is a holistic statistical measure over a large area, and it may obscure meaningful difference in content popularity among small regions. References [38, 39] have verified this point based on analysis of the real YouTube datasets. Most current researches simply assume identical content request probabilities among all the users, and neglect the geographic locality of user interests.

Third, the spatial traffic distribution is uneven over a large area. Due to the unequal population sizes in different regions and the difference in user behaviors, the spatial traffic distribution exhibits evident heterogeneity [40, 41]. The traffic demand in hot-spots is much larger than that in suburbs. In existing works, especially the ones in which models are constructed based on stochastic geometry, such as [22–24, 34, 35], the traffic demand is uniformly distributed. This setting differs from real situation and the performance of these approaches will be degraded in practical application.

For addressing the above challenges, in this paper, we formulate a framework for joint optimization of content placement and user association in cache-enabled HCNs based on flow-level models [42, 43]. In the flow-level models, networks are modeled as queuing systems, in which BSs correspond to servers and user requests correspond to flows to be served by these servers. Different from traditional snapshot models, flow-level models focus on the spatial traffic demand distribution during a time period instead of the locations and demands of individual users at a certain moment. With the help of flow-level models, we jointly optimize content placement and user association based on the aggregated traffic demand during a long time period rather than instantaneous traffic demand. There are two kinds of flow-level models proposed in the literature: load-non-coupled (LNC) model [42] and load-coupled (LC) model [44]. In the LNC model, the intercell interference is assumed to be static, while, in the LC model, the intercell interference interacts with the loads of other BSs. In this paper, we propose a greedy content caching and content-level selective association (GCC-CSA) algorithm in LNC and LC model,

respectively, for joint optimization of content placement and user association in cache-enabled HCNs. The content-level spatial traffic distribution is modeled to simulate the difference in content popularity among different regions. The superposition of all the content-level spatial traffic distributions forms the overall spatial traffic distribution. This joint optimization problem is formulated as a mixed integer nonlinear programming (MINLP) problem and its objective is to minimize the average delay of a typical flow. This formulation takes into account the limited backhaul capacity of SBSs and its effect on the achievable data rate of each content. For tackling this problem, we decouple it into two interrelated subproblems. First, we propose a CSA algorithm to optimize user association in a given content placement situation. The requests for different contents at the same location are allowed to be served by different BSs due to different caching states of these contents at nearby BSs. Second, we propose a GCC algorithm to add the content that yields the maximum reduction in the value of cost function to each BS in a given user association situation. These two algorithms are alternately executed until the caches of all the BSs are filled to capacity. The derived content-level selective user association takes effect unless the content-level spatial traffic distribution changes, which means that user association is optimized at the timescale of content placement.

Our main contributions are summarized as follows.

(1) We formulate the joint optimization of content placement and user association based on flow-level models. In this formulation, content-level spatial traffic distribution is modeled to simulate the locality of content popularity and the heterogeneity of spatial traffic distribution, and user association is optimized at the timescale of content placement. To the best of our knowledge, this is the first work that addresses all the issues summarized above.

(2) We jointly optimize content placement and user association in LNC and LC model, respectively. LNC model and LC model are two typical network models. Most of existing works based on flow-level models focus on the LNC model due to its favorable properties and elegant solution. In addition to the formulation and solution in LNC model, in this paper, we also propose corresponding formulation and solution in LC model.

(3) We propose a GCC-CSA algorithm to tackle the joint optimization problem. We decouple the complex MINLP problem into two interrelated subproblems. The CSA algorithm is proposed to find the optimal user association in a given content placement situation and the GCC algorithm is proposed to update the content placement. Some properties of these algorithms are also proved.

The rest of this paper is organized as follows. Section 2 constructs the LNC and LC models for the cache-enabled HCNs. Section 3 formulates the joint optimization of cache placement and user association as an MINLP problem. In Section 4, we present the GCC-CSA algorithm in LNC and LC models, respectively. Section 5 defines two performance metrics, average delay and occupied backhaul data rates, to evaluate the performance of the GCC-CSA algorithm. In Section 6, we give the implementation details and complexity

analysis. Section 7 compares the performance of the proposed algorithm with that of other schemes through simulations. Finally, Section 8 concludes this paper.

2. System Model

We consider the downlink transmission in a cache-enabled HCN. A geographic area $\mathcal{L} \subset \mathbb{R}^2$ is covered by a set of BSs $\mathcal{M} = \{1, 2, \dots, M\}$, which includes MBSs and SBSs. \mathcal{M}_M and \mathcal{M}_S denote the set of MBSs and SBSs, respectively. For alleviating congestion in limited backhaul links, each SBS $j \in \mathcal{M}_S$ is equipped with a storage unit with capacity s_j to cache strategic contents. Since MBSs are often equipped with high-capacity backhaul links, we do not consider caching at MBSs in this paper. The total transmission bandwidth is W . The transmit power and backhaul capacity of BS $i \in \mathcal{M}$ are denoted by P_i and B_i , respectively.

In a certain time period, all the contents possibly requested by users in \mathcal{L} constitute a file library $\mathcal{F} = \{1, 2, \dots, F\}$. The size of content $f \in \mathcal{F}$ is denoted by v_f . According to statistical analysis [5, 6], the probability that a certain content is requested in \mathcal{L} can be modeled by Zipf distribution. If these contents are sorted according to their popularity in descending order, the probability that the f -th content is requested is given by

$$q_f = \frac{f^{-z}}{\sum_{k=1}^F k^{-z}}, \quad f = 1, 2, \dots, F, \quad (1)$$

where $z \geq 0$ characterizes the skewness of Zipf distribution. A large z means that a small number of popular contents account for most content requests. In the sequel, contents and files are used interchangeably.

In the flow-level model, requests for content f are assumed to follow an inhomogeneous Poisson point process with arrival rate per unit area $\lambda_f(x)$ at location $x \in \mathcal{L}$. We assume that the Poisson point processes with respect to all the contents are independent from each other. According to superposition theorem, the process characterizing requests generated at x is also a Poisson process, and its intensity is $\lambda(x) = \sum_{f \in \mathcal{F}} \lambda_f(x)$. The probability that f is requested at x is obtained as $p_f(x) = \lambda_f(x)/\lambda(x)$, and the probability that f is requested in \mathcal{L} , $p_{f,\mathcal{L}}$, is calculated by

$$p_{f,\mathcal{L}} = \frac{\int_{\mathcal{L}} \lambda_f(x) dx}{\int_{\mathcal{L}} \lambda(x) dx}. \quad (2)$$

Obviously, $\{p_{f,\mathcal{L}}\}$ follow Zipf distribution. The content request probabilities in a subset $\mathcal{L}' \subset \mathcal{L}$ also follow Zipf distribution, but the order of contents and (or) the skewness parameter may change because of locality of content popularity. The traffic density of f at x , $\gamma_f(x)$, is defined as the average required data rate of f at x per unit area, and it is obtained as $\gamma_f(x) = v_f \lambda_f(x)$. In other words, $\gamma_f(x)$ is the average required amount of data with respect to f at x per unit area per unit time, and it characterizes the content-level spatial traffic distribution. The traffic density at x is simply given by $\gamma(x) = \sum_{f \in \mathcal{F}} \gamma_f(x)$ and it captures the overall spatial traffic variability.

According to Shannon's formula, the radio link data rate from BS i to a device located at x is $c_i(x) = W \log_2(1 + \eta_i(x))$. $\eta_i(x)$ denotes the SINR experienced by a device at x with respect to BS i , and it is given by

$$\eta_i(x) = \frac{P_i g_i(x)}{I_i(x) + \sigma^2}, \quad (3)$$

where $g_i(x)$ is the channel gain from BS i to location x , $I_i(x)$ is the interference received from other BSs except BS i , σ^2 is the power of background noise. Since the data rate is evaluated at the timescale of content placement, which is much larger than the coherence time of wireless channels, fast fading is not contained in $g_i(x)$.

When a data flow requesting content f at location x is served by BS i , the load density of BS i at x with respect to f is defined as

$$\varphi_{i,f}(x) = \frac{\gamma_f(x)}{c_{i,f}(x)}, \quad (4)$$

where

$$c_{i,f}(x) \triangleq d_{i,f} c_i(x) + (1 - d_{i,f}) \min \{c_i(x), B_i\} \quad (5)$$

denotes the achievable data rate of f from BS i to x . $d_{i,f}$ is a binary variable indicating whether BS i stores f . When $d_{i,f} = 1$, content f is cached at BS i and it can be transmitted to the receiver without using the backhaul link. In this case, the achievable data rate of f from BS i to x is the data rate in the radio link. If $d_{i,f} = 0$, BS i does not store f and content f must be retrieved via the backhaul link. Accordingly, the achievable data rate of f from BS i is limited by backhaul capacity B_i . The physical meaning of $\varphi_{i,f}(x)$ is the fraction of time required to deliver traffic density $\gamma_f(x)$ from BS i to x in unit time.

Let $\delta_{i,f}(x)$ denote the probability that a data flow requesting content f at location x is routed to BS i . Of course we have $\delta_{i,f}(x) \in [0, 1]$ and $\sum_{i \in \mathcal{M}} \delta_{i,f}(x) = 1$ for any f and x . This definition allows content-level selective association and it contains traditional user association policies that are insensitive to requested contents. Base on $\{\varphi_{i,f}(x)\}$ and $\{\delta_{i,f}(x)\}$, the load of BS i can be expressed as

$$\begin{aligned} \rho_i &= \min \left\{ \int_{\mathcal{L}} \left(\sum_{f \in \mathcal{F}} \varphi_{i,f}(x) \delta_{i,f}(x) \right) dx, 1 - \varepsilon \right\} \\ &= \min \left\{ \sum_{f \in \mathcal{F}} \left(\int_{\mathcal{L}} \varphi_{i,f}(x) \delta_{i,f}(x) dx \right), 1 - \varepsilon \right\} \\ &= \min \left\{ \sum_{f \in \mathcal{F}} \rho_{i,f}, 1 - \varepsilon \right\}, \end{aligned} \quad (6)$$

where ε is an arbitrarily small positive constant and it is introduced to avoid some intractable and trivial situations in the following formulation and solution. According to the above definition, ρ_i can be interpreted as the total fraction of time needed for BS i to serve all the associated flows in unit time, which cannot be larger than 1.

Given $\{\delta_{i,f}(x)\}$, the arrival process of data flows to BS i follows Poisson process with arrival rate $\lambda_i = \int_{\mathcal{L}} \sum_{f \in \mathcal{F}} (\lambda_f(x) \delta_{i,f}(x)) dx$. If multiple flows associated with a BS are scheduled in a round robin manner, the BS can be modeled as an M/G/1 multiclass processor sharing (MCPS) queue [45]. Multiclass means that users at different locations receive different data rates depending on channel conditions and caching states, and processor sharing means that the associated flows are scheduled in a round robin manner.

2.1. Load-Non-Coupled Model. In the LNC model, the interference received by a device is assumed to be static and it is independent of the activity of other BSs. This assumption is reasonable in a system with fractional frequency reuse or enhanced intercell interference cancellation. When these techniques are applied, the intercell interference itself and its variation is reasonably negligible [42]. In this model, $I_i(x)$ can be calculated as $I_i(x) = \eta \sum_{j \neq i} P_j g_j(x)$, where $\eta \in (0, 1)$ characterizes the average received interference.

According to definition in (6), the feasible set of $\boldsymbol{\rho} = [\rho_1, \rho_2, \dots, \rho_M]^T$ in the LNC model is obtained as

$$\mathcal{P}_{\text{NC}} = \left\{ \boldsymbol{\rho} \left| \begin{array}{l} \rho_i = \int_{\mathcal{L}} \left(\sum_{f \in \mathcal{F}} \varphi_{i,f}(x) \delta_{i,f}(x) \right) dx, \\ 0 \leq \rho_i \leq 1 - \varepsilon, \\ \sum_{i \in \mathcal{M}} \delta_{i,f}(x) = 1, \\ 0 \leq \delta_{i,f}(x) \leq 1, \forall i \in \mathcal{M}, \forall x \in \mathcal{L}, \forall f \in \mathcal{F} \end{array} \right. \right\}. \quad (7)$$

With given routing probabilities $\{\delta_{i,f}(x)\}$, the load of BS i is independent of the loads of other BSs, thus this model is termed load-non-coupled model. \mathcal{P}_{NC} has the following property.

Lemma 1. \mathcal{P}_{NC} is a convex set.

Proof. Assume $\boldsymbol{\rho}^1, \boldsymbol{\rho}^2 \in \mathcal{P}_{\text{NC}}$ and $\boldsymbol{\rho}^1 \neq \boldsymbol{\rho}^2$. For all $i \in \mathcal{M}$, we have $\rho_i^1 = \int_{\mathcal{L}} (\sum_{f \in \mathcal{F}} \varphi_{i,f}(x) \delta_{i,f}^1(x)) dx$ and $\rho_i^2 = \int_{\mathcal{L}} (\sum_{f \in \mathcal{F}} \varphi_{i,f}(x) \delta_{i,f}^2(x)) dx$. $\{\delta_{i,f}^1(x)\}$ and $\{\delta_{i,f}^2(x)\}$ are corresponding routing probabilities of $\boldsymbol{\rho}^1$ and $\boldsymbol{\rho}^2$, respectively. Let $\boldsymbol{\rho}$ be a convex combination of $\boldsymbol{\rho}^1$ and $\boldsymbol{\rho}^2$. Given $\theta \in [0, 1]$, for all $i \in \mathcal{M}$, we have

$$\begin{aligned} \rho_i &= \theta \rho_i^1 + (1 - \theta) \rho_i^2 \\ &= \theta \int_{\mathcal{L}} \left(\sum_{f \in \mathcal{F}} \varphi_{i,f}(x) \delta_{i,f}^1(x) \right) dx \\ &\quad + (1 - \theta) \int_{\mathcal{L}} \left(\sum_{f \in \mathcal{F}} \varphi_{i,f}(x) \delta_{i,f}^2(x) \right) dx \end{aligned}$$

$$\begin{aligned} &= \int_{\mathcal{L}} \left(\sum_{f \in \mathcal{F}} \varphi_{i,f}(x) (\theta \delta_{i,f}^1(x) + (1 - \theta) \delta_{i,f}^2(x)) \right) dx \\ &= \int_{\mathcal{L}} \left(\sum_{f \in \mathcal{F}} \varphi_{i,f}(x) \delta_{i,f}(x) \right) dx. \end{aligned} \quad (8)$$

$\boldsymbol{\rho}$ and its routing probabilities $\{\delta_{i,f}(x)\}$ satisfy all conditions in (7). Thus $\boldsymbol{\rho} \in \mathcal{P}_{\text{NC}}$ and \mathcal{P}_{NC} is a convex set. \square

2.2. Load-Coupled Model. In a cellular network with frequency reuse factor of 1, all the BSs work on the same frequency band. In this scenario, the interference received by a user varies considerably depending on the activity of other BSs. It varies at the timescale of flow dynamics and accurately modeling these correlations is intractable [44]. For tackling this issue, Fehske *et al.* [46] propose to model the dynamic interference by the time-averaged interference. If the load of BS i is treated as the probability that BS i is transmitting, the SINR can be expressed as

$$\eta_i(x, \boldsymbol{\rho}) = \frac{P_i g_i(x)}{\sum_{j \in \mathcal{M}, j \neq i} \rho_j P_j g_j(x) + \sigma^2}, \quad (9)$$

where $\sum_{j \in \mathcal{M}, j \neq i} \rho_j P_j g_j(x)$ is the average interference in a long time period and it replaces the instantaneous interference at any moment in this period. Note that $\eta_i(x, \boldsymbol{\rho})$ relates to the loads of all BSs except BS i . The radio link data rate in the LC model is $c_i(x, \boldsymbol{\rho}) = W \log_2(1 + \eta_i(x, \boldsymbol{\rho}))$ and the load density of BS i at x with respect to f is written as

$$\varphi_{i,f}(x, \boldsymbol{\rho}) = \frac{\gamma_f(x)}{c_{i,f}(x, \boldsymbol{\rho})}, \quad (10)$$

where

$$\begin{aligned} c_{i,f}(x, \boldsymbol{\rho}) &\triangleq d_{i,f} c_i(x, \boldsymbol{\rho}) \\ &\quad + (1 - d_{i,f}) \min \{c_i(x, \boldsymbol{\rho}), B_i\}. \end{aligned} \quad (11)$$

Similarly, the feasible set of $\boldsymbol{\rho}$ in the LC model is given by

$$\mathcal{P}_{\text{C}} = \left\{ \boldsymbol{\rho} \left| \begin{array}{l} \rho_i = \int_{\mathcal{L}} \left(\sum_{f \in \mathcal{F}} \varphi_{i,f}(x, \boldsymbol{\rho}) \delta_{i,f}(x) \right) dx, \\ 0 \leq \rho_i \leq 1 - \varepsilon, \\ \sum_{i \in \mathcal{M}} \delta_{i,f}(x) = 1, \\ 0 \leq \delta_{i,f}(x) \leq 1, \forall i \in \mathcal{M}, \forall x \in \mathcal{L}, \forall f \in \mathcal{F} \end{array} \right. \right\}. \quad (12)$$

Since ρ_i is derived based on $\eta_i(x, \boldsymbol{\rho})$, it is coupled with the loads of all the other BSs. Let $S_i(\boldsymbol{\rho}) = \min \{ \int_{\mathcal{L}} (\sum_{f \in \mathcal{F}} \varphi_{i,f}(x, \boldsymbol{\rho}) \delta_{i,f}(x)) dx, 1 - \varepsilon \}$ and $S(\boldsymbol{\rho}) = [S_1(\boldsymbol{\rho}), S_2(\boldsymbol{\rho}), \dots, S_M(\boldsymbol{\rho})]^T$; then a load vector $\boldsymbol{\rho} \in \mathcal{P}_{\text{C}}$ must be a fixed point of $S(\cdot)$; i.e., $\boldsymbol{\rho} = S(\boldsymbol{\rho})$. For given routing probabilities $\{\delta_{i,f}(x)\}$ and caching states $\{d_{i,f}\}$, we can find a unique fixed point in $[0, 1]^M$ according to Theorem 1 in [46].

3. Problem Formulation

We aim to find the optimal content placement scheme $\{d_{i,f}\}$ and user association policy $\{\delta_{i,f}(x)\}$ that minimize the average content delivery delay in a cache-enabled HCN. Although routing probabilities $\{\delta_{i,f}(x)\}$ describe user association in a probabilistic manner, the values of optimal routing probabilities are binary under given caching states, as shown in Section 4. In the M/G/1 MCPS queue, the average number of flows at BS i is given by $\mathbb{E}[N_i] = \rho_i/(1 - \rho_i)$ [47], and the total number of flows at all BSs is $\sum_{i \in \mathcal{M}} \rho_i/(1 - \rho_i)$. According to Little's formula, minimizing the average number of flows in a BS is equivalent to minimizing the average delay of a typical flow in this BS. For minimizing the average delay of all flows, the joint optimization of content placement and user association can be formulated as

$$\begin{aligned} \min_{\boldsymbol{\rho}, \mathbf{D}} \quad & f(\boldsymbol{\rho}) = \sum_{i \in \mathcal{M}} \frac{1}{1 - \rho_i} \\ \text{s.t.} \quad & \boldsymbol{\rho} \in \mathcal{P}, \\ & \mathbf{D}\mathbf{v} \leq \mathbf{s}. \end{aligned} \quad (13)$$

Because $1/(1 - \rho_i) = \rho_i/(1 - \rho_i) + 1$, minimizing $\sum_{i \in \mathcal{M}} \rho_i/(1 - \rho_i)$ is equivalent to minimizing the cost function $f(\boldsymbol{\rho})$. \mathcal{P} denotes \mathcal{P}_{NC} or \mathcal{P}_{C} depending on whether loads are coupled. $\mathbf{D} = [d_{i,f}]_{M \times F}$ denotes the caching state of each file at each BS, $\mathbf{v} = [v_1, v_2, \dots, v_F]^T$ and $\mathbf{s} = [s_1, s_2, \dots, s_M]^T$. The second constraint ensures that the total size of cached files at each BS cannot exceed corresponding cache capacity. Due to the correlation between $\boldsymbol{\rho}$ and $\{\delta_{i,f}(x)\}$, we can obtain the optimal content-level selective user association during the course of finding the optimal $\boldsymbol{\rho}$. The details are presented in Section 4

Problem (13) belongs to MINLP problems and it is NP-hard. If \mathbf{D} is given, however, this problem degenerates into a user association problem, which is similar to the ones studied in [42, 44]. In the following section, we present the proposed GCC-CSA algorithm to solve problem (13) in LNC model and LC model, respectively.

4. GCC-CSA Algorithm

In this section, we present the GCC-CSA algorithm in LNC model and LC model, respectively. The GCC-CSA algorithm is an iterative algorithm, and each iteration is composed of two steps: CSA and GCC. In the CSA step, the optimal user association is derived under given caching states. Based on the derived user association results, the GCC step adds the file that yields the maximum cost reduction to each BS. Beginning with empty caches, these two steps are alternately executed until all the caches cannot accommodate any more contents.

4.1. Load-Non-Coupled Model. At first, we find the optimal user association under given caching states. In the LNC model, given \mathbf{D} , problem (13) is simplified as the following form:

$$\begin{aligned} \min_{\boldsymbol{\rho}} \quad & f(\boldsymbol{\rho}) = \sum_{i \in \mathcal{M}} \frac{1}{1 - \rho_i} \\ \text{s.t.} \quad & \boldsymbol{\rho} \in \mathcal{P}_{\text{NC}}. \end{aligned} \quad (14)$$

Inspired by the work in [42], we first propose a CSA algorithm to solve problem (14), as shown in Algorithm 1, and then prove its optimality.

The following theorem states the convergence of Algorithm 1 and the optimality of the output.

Theorem 2. *In the LNC model, the sequence $\{\boldsymbol{\rho}^{(k)}\}$ derived from Algorithm 1 converges to the fixed point of $\boldsymbol{\rho} = T(\boldsymbol{\rho})$, and it is the unique optimal solution of problem (14).*

Proof. We first prove that the fixed point of $\boldsymbol{\rho} = T(\boldsymbol{\rho})$, denoted by $\boldsymbol{\rho}^*$, is the unique optimal solution of problem (14). Since \mathcal{P}_{NC} is a convex set and $f(\boldsymbol{\rho})$ is a convex function, problem (14) is a convex optimization problem. Let $\{\delta_{i,f}^*(x)\}$ and $\{\delta_{i,f}(x)\}$ be the routing probabilities associated with $\boldsymbol{\rho}^*$ and $\forall \boldsymbol{\rho} \in \mathcal{P}_{\text{NC}}$, respectively; then we have

$$\begin{aligned} \langle \nabla f(\boldsymbol{\rho}^*), (\boldsymbol{\rho} - \boldsymbol{\rho}^*) \rangle &= \sum_{i \in \mathcal{M}} \frac{1}{(1 - \rho_i^*)^2} (\rho_i - \rho_i^*) = \sum_{i \in \mathcal{M}} \frac{\int_{\mathcal{X}} (\sum_{f \in \mathcal{F}} \varphi_{i,f}(x) \delta_{i,f}(x)) dx - \int_{\mathcal{X}} (\sum_{f \in \mathcal{F}} \varphi_{i,f}(x) \delta_{i,f}^*(x)) dx}{(1 - \rho_i^*)^2} \\ &= \sum_{i \in \mathcal{M}} \frac{\int_{\mathcal{X}} (\sum_{f \in \mathcal{F}} \varphi_{i,f}(x) (\delta_{i,f}(x) - \delta_{i,f}^*(x))) dx}{(1 - \rho_i^*)^2} = \int_{\mathcal{X}} \left(\sum_{f \in \mathcal{F}} \gamma_f(x) \sum_{i \in \mathcal{M}} \frac{\delta_{i,f}(x) - \delta_{i,f}^*(x)}{c_{i,f}(x) (1 - \rho_i^*)^2} \right) dx. \end{aligned} \quad (15)$$

From Algorithm 1, $\delta_{i,f}^*(x) = \mathbf{1}\{i = \arg \max_{j \in \mathcal{M}} c_{j,f}(x) (1 - \rho_j^*)^2\}$; thus $\langle \nabla f(\boldsymbol{\rho}^*), (\boldsymbol{\rho} - \boldsymbol{\rho}^*) \rangle \geq 0$. According to the optimality condition of convex optimization problem [48], $\boldsymbol{\rho}^*$ is the optimal solution of problem (14). Since $f(\boldsymbol{\rho})$ is a strictly convex function, the optimal solution of (14) is unique, and so is the fixed point of $\boldsymbol{\rho} = T(\boldsymbol{\rho})$.

Following the similar steps in (15), we can conclude $\langle \nabla f(\boldsymbol{\rho}), (T(\boldsymbol{\rho}) - \boldsymbol{\rho}) \rangle < 0$ for $\forall \boldsymbol{\rho} \in \mathcal{P}_{\text{NC}}$ and $\boldsymbol{\rho} \neq \boldsymbol{\rho}^*$. In other words, $T(\boldsymbol{\rho})$ gives a descent direction and $f(\boldsymbol{\rho}^{(k+1)}) <$

$f(\boldsymbol{\rho}^{(k)})$ with proper β . Since $f(\boldsymbol{\rho})$ is a continuous function on a compact set, $\{f(\boldsymbol{\rho}^{(k)})\}$ must converge to its minimum value $f(\boldsymbol{\rho}^*)$. If $\{f(\boldsymbol{\rho}^{(k)})\}$ converges to some value that is larger than $f(\boldsymbol{\rho}^*)$, $T(\boldsymbol{\rho})$ will generate a descent direction, and the function value will decrease. Thus $\{f(\boldsymbol{\rho}^{(k)})\}$ must converge to $f(\boldsymbol{\rho}^*)$, and $\{\boldsymbol{\rho}^{(k)}\}$ must converge to $\boldsymbol{\rho}^*$. \square

Although user association is defined as probabilities in (7), Algorithm 1 shows that the optimal user association

Initialization: D , small positive constant ε and ξ , $\mu > \xi$, stepsize $\beta \in [0, 1)$, $\rho^{(0)} \in (0, 1 - \varepsilon)^M$, $k = 0$
while $\mu > \xi$ **do**
 for all location $x \in \mathcal{L}$ and content $f \in \mathcal{F}$, the flow requesting f at x connects to BS
 $i_f^{(k)}(x) = \arg \max_{j \in \mathcal{M}} c_{j,f}(x)(1 - \rho_j^{(k)})^2$;
 for all BS $i \in \mathcal{M}$ and content $f \in \mathcal{F}$, calculate the coverage area of BS i with respect to f
 $\mathcal{L}_{i,f}^{(k)} = \{x \in \mathcal{L} \mid i = \arg \max_{j \in \mathcal{M}} c_{j,f}(x)(1 - \rho_j^{(k)})^2\}$;
 for all BS $i \in \mathcal{M}$, calculate its new load $T_i(\rho^{(k)}) = \min\{\sum_{f \in \mathcal{F}} \int_{\mathcal{L}_{i,f}^{(k)}} \varphi_{i,f}(x) dx, 1 - \varepsilon\}$;
 $\rho^{(k+1)} = \beta \rho^{(k)} + (1 - \beta)T(\rho^{(k)})$;
 $\mu = \|\rho^{(k+1)} - \rho^{(k)}\|_2$, $k := k + 1$;
end while
Outputs: the optimal load $\rho^* = \rho^{(k)}$ and the optimal coverage area $\mathcal{L}_{i,f}^* = \mathcal{L}_{i,f}^{(k)}$ for all $i \in \mathcal{M}$ and $f \in \mathcal{F}$

ALGORITHM 1: The CSA algorithm for solving problem (14).

Initialization: $D_0 = [0]_{M \times F}$, $\bar{s}_0 = [0]_{M \times 1}$, $\bar{\mathcal{F}}_{i,0} = \mathcal{F}$ for all $i \in \mathcal{M}$, $\mathcal{G}_{i,0} = \mathcal{F}$ for all $i \in \mathcal{M}$, $t = 0$
while $\{i \in \mathcal{M} \mid \mathcal{G}_{i,t} \neq \emptyset\} \neq \emptyset$ **do**
 get the optimal ρ_i^* and corresponding $\{\mathcal{L}_{i,f,t}^*\}$ under D_t according to Algorithm 1;
 for all BS $i \in \{j \in \mathcal{M} \mid \mathcal{G}_{j,t} \neq \emptyset\}$ **do**
 for all $f \in \mathcal{G}_{i,t}$, calculate $\rho_{i,f,t}^* = \int_{\mathcal{L}_{i,f,t}^*} (\gamma_f(x) / \min\{c_i(x), B_i\}) dx$ and $\bar{\rho}_{i,f,t} = \int_{\mathcal{L}_{i,f,t}^*} (\gamma_f(x) / c_i(x)) dx$;
 find $f_{i,t} = \arg \max_{f \in \mathcal{G}_{i,t}} (\rho_{i,f,t}^* - \bar{\rho}_{i,f,t})$;
 $\bar{s}_{i,t+1} = \bar{s}_{i,t} + v_{f_{i,t}}$; % update $\bar{s}_{i,t}$
 $\bar{\mathcal{F}}_{i,t+1} = \bar{\mathcal{F}}_{i,t} \setminus \{f_{i,t}\}$, $\mathbf{d}_{i,t+1} = \mathbf{d}_{i,t}$, $d_{i,f_{i,t},t+1} = 1$; % update the cached contents of BS i
 $\mathcal{G}_{i,t+1} = \{f \in \bar{\mathcal{F}}_{i,t+1} \mid \bar{s}_{i,t+1} + v_f \leq s_i\}$; % update $\mathcal{G}_{i,t}$
 end for
 $t := t + 1$;
end while
 get the optimal ρ_i^* and corresponding $\{\mathcal{L}_{i,f,t}^*\}$ under D_t according to Algorithm 1;
Outputs: the content placement scheme $D^* = D_t$ and corresponding optimal load $\rho^* = \rho_i^*$

ALGORITHM 2: GCC-CSA algorithm for solving problem (13) in LNC model.

is deterministic. $\mathcal{L}_{i,f}^*$ indicates the coverage area of BS i with respect to content f , and the coverage areas associated with different contents may be different from each other depending on the caching states and backhaul capacity.

Given certain D , we can obtain the optimal load ρ^* and the optimal coverage area $\{\mathcal{L}_{i,f}^*\}$ according to Algorithm 1. Thus, for any BS $i \in \mathcal{M}$, we have

$$\begin{aligned} \rho_i^* &= \sum_{f \in \mathcal{F}} \left(\int_{\mathcal{L}_{i,f}^*} \frac{\gamma_f(x)}{d_{i,f} c_i(x) + (1 - d_{i,f}) \min\{c_i(x), B_i\}} dx \right) \quad (16) \\ &= \sum_{f \in \mathcal{F}} \rho_{i,f}^*. \end{aligned}$$

For a content f with $d_{i,f} = 0$, $\rho_{i,f}^* = \int_{\mathcal{L}_{i,f}^*} (\gamma_f(x) / \min\{c_i(x), B_i\}) dx$. If $d_{i,f}$ is changed from 0 to 1 and other elements in D remain unchanged, with the same coverage $\mathcal{L}_{i,f}^*$, the new load of BS i with respect to f is $\bar{\rho}_{i,f} = \int_{\mathcal{L}_{i,f}^*} (\gamma_f(x) / c_i(x)) dx$. Since $\min\{c_i(x), B_i\} \leq c_i(x)$, we have $\bar{\rho}_{i,f} \leq \rho_{i,f}^*$ and the cost function also decreases. Base on this fact, we propose a GCC algorithm for content placement in an iterative

manner, as shown in Algorithm 2. In the GCC algorithm, the content that achieves the maximum cost reduction is added to each BS at each iteration. After all the BSs cache new contents, the content placement scheme D is updated and Algorithm 1 is executed again to obtain the new cell coverage areas associated with the updated D . This process continues until no more contents can be cached in any BSs.

In Algorithm 2, $\bar{\mathbf{s}}_t = [\bar{s}_{1,t}, \bar{s}_{2,t}, \dots, \bar{s}_{M,t}]^T$ denotes the occupied cache capacity of these BSs at the beginning of the t -th iteration. $\bar{\mathcal{F}}_{i,t}$ denotes the set of noncached contents of BS i at the beginning of the t -th iteration. $\mathcal{G}_{i,t}$ denotes the set of contents that can be cached at BS i at the t -th iteration. $\mathbf{d}_{i,t}$ denotes the i -th row vector in D_t . With given $\{\mathcal{L}_{i,f,t}^*\}$, the loads of a BS with respect to different contents are independent of each other, thus caching $f_{i,t}$ at BS i produces the maximum reduction in its load. Each BS caches the content that produces the maximum reduction in its load, and the cost function $f(\rho)$ also achieves the maximum reduction at a single iteration. The following theorem gives the convergence property of Algorithm 2.

Theorem 3. *The sequence of cost function values $\{f(\rho_t^*)\}$ derived from Algorithm 2 decreases monotonically.*

Proof. At the t -th iteration, the optimal load ρ_t^* and coverage $\{\mathcal{L}_{i,f,t}^*\}$ are derived under \mathbf{D}_t . With given $\{\mathcal{L}_{i,f,t}^*\}$, for all BS $i \in \{j \in \mathcal{M} \mid \mathcal{G}_{j,t} \neq \emptyset\}$, we have $\tilde{\rho}_{i,f_i,t} \leq \rho_{i,f_i,t}^*$. Let $\tilde{\rho}_{i,t} \triangleq \tilde{\rho}_{i,f_i,t} + \sum_{f \neq f_i,t} \rho_{i,f,t}^*$ and $\tilde{\rho}_t = [\tilde{\rho}_{1,t}, \tilde{\rho}_{2,t}, \dots, \tilde{\rho}_{M,t}]^T$, then $\tilde{\rho}_{i,t} \leq \rho_{i,t}^*$ and $f(\tilde{\rho}_t) \leq f(\rho_t^*)$. Note that $\tilde{\rho}_t$ is derived based on \mathbf{D}_{t+1} and $\{\mathcal{L}_{i,f,t}^*\}$. The optimal load ρ_{t+1}^* under \mathbf{D}_{t+1} must obtain the minimum value of $f(\rho)$ and $f(\rho_{t+1}^*) \leq f(\tilde{\rho}_t)$. Thus $f(\rho_{t+1}^*) \leq f(\rho_t^*)$ and $\{f(\rho_t^*)\}$ is a monotonically decreasing sequence. \square

4.2. Load-Coupled Model. In the LC model with given \mathbf{D} , problem (13) is simplified as follows:

$$\begin{aligned} \min_{\rho} \quad & f(\rho) = \sum_{i \in \mathcal{M}} \frac{1}{1 - \rho_i} \\ \text{s.t.} \quad & \rho \in \mathcal{P}_C. \end{aligned} \quad (17)$$

Problem (17) shares the same form with problem (14), and its solution can also be derived from Algorithm 1 with slight modifications that $c_{j,f}(x)$ and $\varphi_{i,f}(x)$ are replaced by $c_{j,f}(x, \rho^{(k)})$ and $\varphi_{i,f}(x, \rho^{(k)})$, respectively. However, for proving the optimality of the solution derived from Algorithm 1 in LC model, \mathcal{P}_C must have the following property [44].

Property 4 (full convertibility). \mathcal{P}_C is said to have the property of full convertibility if for $\forall \rho, \rho' \in \mathcal{P}_C$, there exist valid $\{\tilde{\delta}_{i,f}(x)\}$ that make the following equation hold for all $i \in \mathcal{M}$:

$$\begin{aligned} \rho_i &= \int_{\mathcal{L}} \left(\sum_{f \in \mathcal{F}} \varphi_{i,f}(x, \rho) \delta_{i,f}(x) \right) dx \\ &= \int_{\mathcal{L}} \left(\sum_{f \in \mathcal{F}} \varphi_{i,f}(x, \rho') \tilde{\delta}_{i,f}(x) \right) dx. \end{aligned} \quad (18)$$

With the property of full convertibility, the following theorem guarantees that the solution derived from Algorithm 1 is also optimal in LC model.

Theorem 5. *In the LC model, if \mathcal{P}_C has the property of full convertibility, then the sequence $\{\rho^{(k)}\}$ derived from Algorithm 1 converges to the fixed point of $\rho = T(\rho)$, and it is the unique optimal solution of problem (17).*

Proof. This proof is similar to the proof of Theorem 2. Denote by ρ^* the fixed point of $\rho = T(\rho)$. Let $\{\delta_{i,f}^*(x)\}$ and $\{\tilde{\delta}_{i,f}(x)\}$ be the associated routing probabilities of ρ^* and $\forall \rho \in \mathcal{P}_C$; then we have

$$\begin{aligned} \langle \nabla f(\rho^*), (\rho - \rho^*) \rangle &= \sum_{i \in \mathcal{M}} \frac{1}{(1 - \rho_i^*)^2} (\rho_i - \rho_i^*) \\ &= \sum_{i \in \mathcal{M}} \frac{\int_{\mathcal{L}} (\sum_{f \in \mathcal{F}} \varphi_{i,f}(x, \rho) \delta_{i,f}(x)) dx - \int_{\mathcal{L}} (\sum_{f \in \mathcal{F}} \varphi_{i,f}(x, \rho^*) \delta_{i,f}^*(x)) dx}{(1 - \rho_i^*)^2} \\ &= \sum_{i \in \mathcal{M}} \frac{\int_{\mathcal{L}} (\sum_{f \in \mathcal{F}} \varphi_{i,f}(x, \rho^*) \tilde{\delta}_{i,f}(x)) dx - \int_{\mathcal{L}} (\sum_{f \in \mathcal{F}} \varphi_{i,f}(x, \rho^*) \delta_{i,f}^*(x)) dx}{(1 - \rho_i^*)^2} \\ &= \sum_{i \in \mathcal{M}} \frac{\int_{\mathcal{L}} (\sum_{f \in \mathcal{F}} \varphi_{i,f}(x, \rho^*) (\tilde{\delta}_{i,f}(x) - \delta_{i,f}^*(x))) dx}{(1 - \rho_i^*)^2} \\ &= \int_{\mathcal{L}} \left(\sum_{f \in \mathcal{F}} \gamma_f(x) \sum_{i \in \mathcal{M}} \frac{\tilde{\delta}_{i,f}(x) - \delta_{i,f}^*(x)}{c_{i,f}(x, \rho^*) (1 - \rho_i^*)^2} \right) dx. \end{aligned} \quad (19)$$

The fourth line applies full convertibility property. Since $\delta_{i,f}^*(x) = \mathbf{1}\{i = \arg \max_{j \in \mathcal{M}} c_{j,f}(x, \rho^*) (1 - \rho_j^*)^2\}$, so $\langle \nabla f(\rho^*), (\rho - \rho^*) \rangle \geq 0$. Unlike \mathcal{P}_{NC} , \mathcal{P}_C is not necessarily a convex set. According to the approach proposed in [44], for $\forall \rho', \rho'' \in \mathcal{P}_C$ and $\forall \theta \in [0, 1]$, $\rho = \theta \rho' + (1 - \theta) \rho''$ satisfies

$$\begin{aligned} \langle \nabla f(\rho^*), (\rho - \rho^*) \rangle &= \langle \nabla f(\rho^*), ((\theta \rho' + (1 - \theta) \rho'') - \rho^*) \rangle \end{aligned}$$

$$\begin{aligned} &= \theta \langle \nabla f(\rho^*), (\rho' - \rho^*) \rangle \\ &\quad + (1 - \theta) \langle \nabla f(\rho^*), (\rho'' - \rho^*) \rangle \geq 0. \end{aligned} \quad (20)$$

Hence any $\rho \in \text{Conv}(\mathcal{P}_C)$ satisfies $\langle \nabla f(\rho^*), (\rho - \rho^*) \rangle \geq 0$, and the minimum value of $f(\rho)$ is achieved at $\rho^* \in \text{Conv}(\mathcal{P}_C)$. Since $f(\rho)$ is a convex function and $\mathcal{P}_C \subset \text{Conv}(\mathcal{P}_C)$, the minimum value of $f(\rho)$ in \mathcal{P}_C is also achieved at ρ^* . Since $f(\rho)$ is a strictly convex function, the optimal solution of (17)


```

Initialization:  $\mathbf{D}_0 = [0]_{M \times F}$ ,  $\tilde{\mathbf{s}}_0 = [0]_{M \times 1}$ ,  $\overline{\mathcal{F}}_{i,0} = \mathcal{F}$  for all  $i \in \mathcal{M}$ ,  $\mathcal{G}_{i,0} = \mathcal{F}$  for all  $i \in \mathcal{M}$ ,  $t = 0$ 
while  $\{i \in \mathcal{M} \mid \mathcal{G}_{i,t} \neq \emptyset\} \neq \emptyset$  do
  get the optimal  $\boldsymbol{\rho}_t^*$  and corresponding  $\{\mathcal{L}_{i,f,t}^*\}$  under  $\mathbf{D}_t$  according to Algorithm 1;
  for all BS  $i \in \{j \in \mathcal{M} \mid \mathcal{G}_{j,t} \neq \emptyset\}$  do
    for all  $f \in \mathcal{G}_{i,t}$ , calculate  $\boldsymbol{\rho}_{d_{i,f,t}}$  according to (22);
    find  $f_{i,t} = \arg \min_{f \in \mathcal{G}_{i,t}} f(\boldsymbol{\rho}_{d_{i,f,t}})$ ;
     $\tilde{\mathbf{s}}_{i,t+1} = \tilde{\mathbf{s}}_{i,t} + \mathbf{v}_{f_{i,t}}$ ; % update  $\tilde{\mathbf{s}}_{i,t}$ 
     $\overline{\mathcal{F}}_{i,t+1} = \overline{\mathcal{F}}_{i,t} \setminus \{f_{i,t}\}$ ,  $\mathbf{d}_{i,t+1} = \mathbf{d}_{i,t}$ ,  $d_{i,f_{i,t},t+1} = 1$ ; % update the cached contents of BS  $i$ 
     $\mathcal{G}_{i,t+1} = \{f \in \overline{\mathcal{F}}_{i,t+1} \mid \tilde{\mathbf{s}}_{i,t+1} + \mathbf{v}_f \leq \mathbf{s}_i\}$ ; % update  $\mathcal{G}_{i,t}$ 
  end for
   $t := t + 1$ ;
end while
get the optimal  $\boldsymbol{\rho}_t^*$  and corresponding  $\{\mathcal{L}_{i,f,t}^*\}$  under  $\mathbf{D}_t$  according to Algorithm 1;
Outputs: the content placement scheme  $\mathbf{D}^* = \mathbf{D}_t$  and corresponding optimal load  $\boldsymbol{\rho}^* = \boldsymbol{\rho}_t^*$ 

```

ALGORITHM 3: GCC-CSA algorithm for solving problem (13) in LC model.

is unique, and so is the fixed point of $\boldsymbol{\rho} = T(\boldsymbol{\rho})$. Proving that $\{\boldsymbol{\rho}^{(k)}\}$ converges to $\boldsymbol{\rho}^*$ follows the same steps in the proof of Theorem 2, and they are omitted here. \square

Given \mathbf{D} , the optimal load $\boldsymbol{\rho}^*$ and the optimal cell coverage $\{\mathcal{L}_{j,f}^*\}$ satisfies

$$\begin{aligned} \rho_j^* &= \sum_{f \in \mathcal{F}} \left(\int_{\mathcal{L}_{j,f}^*} \frac{\gamma_f(x)}{d_{j,f} c_j(x, \boldsymbol{\rho}^*) + (1 - d_{j,f}) \min\{c_j(x, \boldsymbol{\rho}^*), B_j\}} dx \right) \\ &= \sum_{f \in \mathcal{F}} \rho_{j,f}^*, \quad j = 1, 2, \dots, M. \end{aligned} \quad (21)$$

If a certain $d_{i,f}$ is changed from 0 to 1, with the same coverage $\{\mathcal{L}_{j,f}^*\}$, ρ_i^* will probably change and it further influences the loads of other BSs in the LC model. The new load vector that makes the system stable, $\boldsymbol{\rho}_{d_{i,f}} = [\rho_{1,d_{i,f}}, \rho_{2,d_{i,f}}, \dots, \rho_{M,d_{i,f}}]^T$, is obtained from the following iterative formula:

$$\begin{aligned} \rho_{j,d_{i,f}}^{(k+1)} &= T_j(\boldsymbol{\rho}_{d_{i,f}}^{(k)}) \\ &= \min \left\{ \sum_{f \in \mathcal{F}} \int_{\mathcal{L}_{j,f}^*} \varphi_{j,f}(x, \boldsymbol{\rho}_{d_{i,f}}^{(k)}) dx, 1 - \varepsilon \right\}, \quad (22) \\ & \quad j = 1, 2, \dots, M, \end{aligned}$$

in which $\boldsymbol{\rho}_{d_{i,f}}^{(0)} = \boldsymbol{\rho}^*$. In fact, the loads of all the BSs will decrease or remain unchanged if a new content is added to a BS, as showed later in Proposition 6. This motivates us to design a similar GCC algorithm as in LNC model. At each iteration, each SBS chooses to cache the content that produces the maximum cost reduction and the content placement is updated. The GCC-CSA algorithm in LC model is given in Algorithm 3.

Before proving the convergence property of Algorithm 3, we present the following two propositions at first.

Proposition 6. *At the t -th iteration, for any BS $i \in \{j \in \mathcal{M} \mid \mathcal{G}_{j,t} \neq \emptyset\}$ and any $f \in \mathcal{G}_{i,t}$, the sequence $\{\boldsymbol{\rho}_{d_{i,f,t}}^{(k)}\}$ derived from (22) converges to, say $\boldsymbol{\rho}_{d_{i,f,t}}$, and $f(\boldsymbol{\rho}_{d_{i,f,t}}) \leq f(\boldsymbol{\rho}_t^*)$.*

Proof. For brevity, we omit the subscript “ t ” that indicates the number of iterations in this proof. At a certain iteration, for any BS $i \in \{j \in \mathcal{M} \mid \mathcal{G}_j \neq \emptyset\}$ and any $f \in \mathcal{G}_i$, we have $d_{i,f} = 0$ and $\rho_{i,f}^* = \int_{\mathcal{L}_{i,f}^*} (\gamma_f(x) / \min\{c_i(x, \boldsymbol{\rho}^*), B_i\}) dx$. If $d_{i,f}$ is changed to 1, according to (22), we have

$$\begin{aligned} \rho_{i,d_{i,f}}^{(1)} &= T_i(\boldsymbol{\rho}^*) = \min \left\{ \sum_{f' \in \mathcal{F} \setminus \{f\}} \int_{\mathcal{L}_{i,f'}^*} \varphi_{i,f'}(x, \boldsymbol{\rho}^*) dx \right. \\ & \quad \left. + \int_{\mathcal{L}_{i,f}^*} \frac{\gamma_f(x)}{c_i(x, \boldsymbol{\rho}^*)} dx, 1 - \varepsilon \right\} \leq \rho_i^*. \end{aligned} \quad (23)$$

For other BS $j \neq i$,

$$\begin{aligned} \rho_{j,d_{i,f}}^{(1)} &= T_j(\boldsymbol{\rho}^*) \\ &= \min \left\{ \sum_{f' \in \mathcal{F}} \int_{\mathcal{L}_{j,f'}^*} \varphi_{j,f'}(x, \boldsymbol{\rho}^*) dx, 1 - \varepsilon \right\} \\ &= \rho_j^*. \end{aligned} \quad (24)$$

Since the data rates of flows associated with BS i relate to the loads of other BSs rather than the load of BS i , we have

$$\begin{aligned} \rho_{i,d_{i,f}}^{(2)} &= T_i(\boldsymbol{\rho}_{d_{i,f}}^{(1)}) \\ &= \min \left\{ \sum_{f' \in \mathcal{F} \setminus \{f\}} \int_{\mathcal{L}_{i,f'}^*} \varphi_{i,f'}(x, \boldsymbol{\rho}_{d_{i,f}}^{(1)}) dx \right. \\ & \quad \left. + \int_{\mathcal{L}_{i,f}^*} \frac{\gamma_f(x)}{c_i(x, \boldsymbol{\rho}_{d_{i,f}}^{(1)})} dx, 1 - \varepsilon \right\} = \rho_{i,d_{i,f}}^{(1)} \leq \rho_i^*. \end{aligned} \quad (25)$$

Since $\rho_{i,d_{i,f}}^{(1)} \leq \rho_i^*$, for other BS $j \neq i$, we have

$$\begin{aligned} \rho_{j,d_{i,f}}^{(2)} &= T_j \left(\rho_{d_{i,f}}^{(1)} \right) \\ &= \min \left\{ \sum_{f' \in \mathcal{F}} \int_{\mathcal{L}_{j,f'}}^* \varphi_{j,f'} \left(x, \rho_{d_{i,f}}^{(1)} \right) dx, 1 - \varepsilon \right\} \\ &\leq \min \left\{ \sum_{f' \in \mathcal{F}} \int_{\mathcal{L}_{j,f'}}^* \varphi_{j,f'} \left(x, \rho^* \right) dx, 1 - \varepsilon \right\} \\ &= \rho_{j,d_{i,f}}^{(1)}. \end{aligned} \quad (26)$$

Thus we conclude $\rho_{d_{i,f}}^{(2)} \leq \rho_{d_{i,f}}^{(1)}$. Assuming $\rho_{d_{i,f}}^{(k)} \leq \rho_{d_{i,f}}^{(k-1)}$ for $k = 2, 3, \dots$, for BS i , we have

$$\begin{aligned} \rho_{i,d_{i,f}}^{(k+1)} &= T_i \left(\rho_{d_{i,f}}^{(k)} \right) \\ &= \min \left\{ \sum_{f' \in \mathcal{F} \setminus \{f\}} \int_{\mathcal{L}_{i,f'}}^* \varphi_{i,f'} \left(x, \rho_{d_{i,f}}^{(k)} \right) dx \right. \\ &\quad \left. + \int_{\mathcal{L}_{i,f}}^* \frac{\gamma_f(x)}{c_i \left(x, \rho_{d_{i,f}}^{(k)} \right)} dx, 1 - \varepsilon \right\} \\ &\leq \min \left\{ \sum_{f' \in \mathcal{F} \setminus \{f\}} \int_{\mathcal{L}_{i,f'}}^* \varphi_{i,f'} \left(x, \rho_{d_{i,f}}^{(k-1)} \right) dx \right. \\ &\quad \left. + \int_{\mathcal{L}_{i,f}}^* \frac{\gamma_f(x)}{c_i \left(x, \rho_{d_{i,f}}^{(k-1)} \right)} dx, 1 - \varepsilon \right\} = \rho_{i,d_{i,f}}^{(k)}, \end{aligned} \quad (27)$$

and for BS $j \neq i$, we have

$$\begin{aligned} \rho_{j,d_{i,f}}^{(k+1)} &= T_j \left(\rho_{d_{i,f}}^{(k)} \right) \\ &= \min \left\{ \sum_{f' \in \mathcal{F}} \int_{\mathcal{L}_{j,f'}}^* \varphi_{j,f'} \left(x, \rho_{d_{i,f}}^{(k)} \right) dx, 1 - \varepsilon \right\} \\ &\leq \min \left\{ \sum_{f' \in \mathcal{F}} \int_{\mathcal{L}_{j,f'}}^* \varphi_{j,f'} \left(x, \rho_{d_{i,f}}^{(k-1)} \right) dx, 1 - \varepsilon \right\} \\ &= \rho_{j,d_{i,f}}^{(k)}. \end{aligned} \quad (28)$$

So we get $\rho_{d_{i,f}}^{(k+1)} \leq \rho_{d_{i,f}}^{(k)}$. Since $\rho_{d_{i,f,t}}^{(k)}$ is lower-bounded by $\mathbf{0}$, $\{\rho_{d_{i,f,t}}^{(k)}\}$ must converge to, say, $\rho_{d_{i,f,t}^*}$. Since $\rho_{d_{i,f,t}} \leq \rho_t^*$, $f(\rho_{d_{i,f,t}^*}) \leq f(\rho_t^*)$ is obtained. \square

Proposition 7. *At the t -th iteration, after all the BS $i \in \{j \in \mathcal{M} \mid \mathcal{G}_{j,t} \neq \emptyset\}$ add a new content to their caches, the load vector ρ_t obtained from (22) under the updated \mathbf{D}_{t+1} and $\{\mathcal{L}_{i,f,t}^*\}$ satisfies $f(\rho_t) \leq f(\rho_t^*)$.*

Proof. Assume that there are M_t BSs in the set $\{j \in \mathcal{M} \mid \mathcal{G}_{j,t} \neq \emptyset\}$. These M_t BSs update their cached contents in a certain order. We get $\rho_{t,1}$ according to (22) after the first BS caches a new content. According to Proposition 6, we have $f(\rho_{t,1}) \leq f(\rho_t^*)$. Then the second BS caches a new content (a new content has already cached at the first BS), and $\rho_{t,2}$ is got according to (22) with initial load vector $\rho_{t,1}$. Similarly, we have $f(\rho_{t,2}) \leq f(\rho_{t,1})$. Each BS caches a new content after the former BS, and we finally get $f(\rho_{t,M_t}) \leq f(\rho_{t,M_t-1})$. Thus we have $f(\rho_{t,M_t}) \leq f(\rho_t^*)$. Note that ρ_{t,M_t} is the stable load vector under the updated caching states \mathbf{D}_{t+1} and $\{\mathcal{L}_{i,f,t}^*\}$, i.e., $\rho_{t,M_t} = \rho_t$. So we have $f(\rho_t) \leq f(\rho_t^*)$. \square

Based on Propositions 6 and 7, the following theorem gives the convergence property of Algorithm 3.

Theorem 8. *The sequence $\{f(\rho_t^*)\}$ derived from Algorithm 3 decreases monotonically.*

Proof. At the t -th iteration, the optimal load ρ_t^* and coverage $\{\mathcal{L}_{i,f,t}^*\}$ are derived under \mathbf{D}_t . After all the BS $i \in \{j \in \mathcal{M} \mid \mathcal{G}_{j,t} \neq \emptyset\}$ update their cached contents, the stable load vector ρ_t under \mathbf{D}_{t+1} and $\{\mathcal{L}_{i,f,t}^*\}$ satisfies $f(\rho_t) \leq f(\rho_t^*)$ according to Proposition 7. Note that $\{\mathcal{L}_{i,f,t}^*\}$ is not the optimal cell coverage associated with \mathbf{D}_{t+1} . The optimal load vector ρ_{t+1}^* under \mathbf{D}_{t+1} satisfies $f(\rho_{t+1}^*) \leq f(\rho_t)$. Thus we have $f(\rho_{t+1}^*) \leq f(\rho_t^*)$, and $\{f(\rho_t^*)\}$ is a monotonically decreasing sequence. \square

5. Performance Metrics

As shown in (13), the objective of GCC-CSA algorithm is to minimize the average content delivery delay in a cache-enabled HCN. However, the objective function $f(\rho)$ in (13) corresponds to the total number of flows at all BSs, and it cannot be used to characterize delay in the flow-level models. For evaluating the performance of GCC-CSA algorithm in terms of delay, in this section, we define the average delay at a given location and the average delay in the whole area. In addition, we also define the occupied backhaul data rates of BSs to demonstrate the advantage of GCC-CSA algorithm in terms of decreasing backhaul load.

5.1. Average Delay. For ease of definition and computation, we partition the continuous area \mathcal{L} into massive pixels and user association policy is derived for each pixel. The traffic densities and the radio link parameters at the center of a pixel are viewed as the average traffic densities and average radio link parameters in this pixel. Let pixels be indexed by y , and let \mathcal{Y} denote the set of all pixels. In the M/G/1 MCPS queues, the time-averaged throughput of a flow associated with BS $i_f(y)$ for content f at location y is given by $c_{i_f(y),f}(y)(1 - \rho_{i_f(y)})$ [49] (In the LC model, this expression is given by $c_{i_f(y),f}(y, \rho)(1 - \rho_{i_f(y)})$). Thus, the average delay for requesting any content at location y can be obtained as

$$\mathbb{E}[D \mid y] = \sum_{f \in \mathcal{F}} p_f(y) \frac{v_f}{c_{i_f(y),f}(y) (1 - \rho_{i_f(y)})}. \quad (29)$$

Define a set $\mathcal{Y}_M \triangleq \{y \in \mathcal{Y} \mid i_f(y) \in \mathcal{M}_M, \forall f \in \mathcal{F}\}$, and it denotes the set of locations that only connect to MBSs. The complement of \mathcal{Y}_M is denoted by \mathcal{Y}_S , and it denotes the set of locations that are probably associated with SBSs for requesting some contents. Apparently, $\mathcal{Y}_M \cup \mathcal{Y}_S = \mathcal{Y}$ and $\mathcal{Y}_M \cap \mathcal{Y}_S = \emptyset$. The average delay of flows at locations in \mathcal{Y}_M and \mathcal{Y}_S are calculated by $\bar{D}_M = \sum_{y \in \mathcal{Y}_M} \mathbb{E}[D \mid y] / |\mathcal{Y}_M|$ and $\bar{D}_S = \sum_{y \in \mathcal{Y}_S} \mathbb{E}[D \mid y] / |\mathcal{Y}_S|$, respectively. The average delay of flows at all locations is calculated by $\bar{D} = \sum_{y \in \mathcal{Y}} \mathbb{E}[D \mid y] / |\mathcal{Y}|$ correspondingly.

5.2. Occupied Backhaul Data Rates. Backhaul usage can be significantly reduced by proper content placement and user association strategies. In the flow-level models constructed in this paper, the occupied backhaul data rate of BS i is obtained as

$$B_{u,i} \triangleq \sum_{f=1}^F (1 - d_{i,f}) \int_{\mathcal{L}_{i,f}} \gamma_f(x) dx. \quad (30)$$

The average occupied backhaul data rates of MBSs and SBSs are calculated by $\bar{B}_{u,M} = \sum_{i \in \mathcal{M}_M} B_{u,i} / |\mathcal{M}_M|$ and $\bar{B}_{u,S} = \sum_{i \in \mathcal{M}_S} B_{u,i} / |\mathcal{M}_S|$, respectively.

6. Implementation and Complexity

In this paper, the formulation of joint optimization of content placement and user association is based on the content-level spatial traffic distribution $\gamma_f(x)$ of each content f in a given area \mathcal{L} . Time is divided into multiple time periods, which range from several hours to several days. The content-level spatial traffic distribution is assumed to be static during each time period, and it changes when the next time period starts. Before a time period starts, $\{\gamma_f(x)\}$ in the forthcoming time period should be estimated in advance. Generally, $\{\gamma_f(x)\}$ correlates with the content popularity distribution and spatial traffic distribution. There have been many studies on the prediction of content popularity distribution [50–52] and the analysis of spatial traffic distribution [40, 53]. With the aid of these methods, we can precisely estimate $\{\gamma_f(x)\}$.

Once $\{\gamma_f(x)\}$ is obtained, it is imported into the GCC-CSA algorithm to derive the content placement and user association policy that take effect during the next time period. Thus, the GCC-CSA algorithm is an offline algorithm. Although the CSA algorithm is essentially a distributed online algorithm [42, 44], it must be executed at the centralized controller in our scheme. The user association policy $i_f(y)$ of each content f at each pixel y can be calculated by parallel computing, e.g., NVIDIA CUDA toolkit, to reduce the running time. The simulations in this paper are conducted by this approach.

The associated BS $i_f(y)$ of a data flow depends on the requested file f and the location y . Contents can be identified by naming contents at the network layer [8]. With positioning techniques in cellular networks [54], the locations of users can also be easily obtained. Based on these techniques, when a content request arrives, it will be routed to corresponding BS with limited signaling overhead.

In the following, we analyze the complexity of the GCC-CSA algorithm in LC model, and the complexity of it in LNC model can be derived similarly. At each iteration in Algorithm 1, the number of operations for the first three steps is $3|\mathcal{Y}|MF$, and the number of operations for the last two steps is $2M$. Denote by I_{\max} the number of iterations that makes Algorithm 1 converge, then the total number of operations of Algorithm 1 is $C_1 \triangleq I_{\max}(3|\mathcal{Y}|MF + 2M)$. Denote by J_{\max} the number of iterations that makes equation (22) converge; then the total number of operations of (22) is $C_2 \triangleq J_{\max}|\mathcal{Y}|MF$. To capture the essence of the complexity of Algorithm 3, we make two assumptions: (1) all the files have the same size; (2) all the M BSs can cache S files at most. Considering the upper bound of complexity of Algorithm 3, the total number of operations of Algorithm 3 is given by $S(C_1 + M(FC_2 + FM + 1)) + C_1$. After some mathematical manipulations, the complexity of the GCC-CSA algorithm in LC model is obtained as $O(\max\{I_{\max}, J_{\max}\}|\mathcal{Y}|SM^2F^2)$.

7. Simulation Results

In this section, we validate the performance of the proposed GCC-CSA algorithm based on the spatial traffic distribution derived from a real network. For comparison, we also evaluate the performance of another two schemes: (1) most popular caching and max-SINR association scheme, denoted by MPC-MSA, and (2) most popular caching and content-level selective association scheme, denoted by MPC-CSA. MPC means all the BSs cache the most popular contents in a given area, and it has been considered in [16, 17]. The MPC policy only considers the overall content popularity distribution, and it ignores the heterogeneity of user preference over a large area. MSA is a user association method generally implemented in the current networks, and it serves as a baseline user association scheme.

7.1. Simulation Setup. We consider a square area with side length 2km, as shown in Figure 1. 7 MBSs and 10 SBSs are located in this area. MBSs are deployed based on hexagonal grid model with an intersite distance of 800 m. SBSs are randomly deployed in this area. The minimum intersite distance between SBS and SBS (MBS) is set to 400 m. The bandwidth is $W = 10$ MHz, and the noise power spectral density is -174 dBm/Hz. The spatial traffic distribution during a certain time period derived from a network operator [53] is also shown in Figure 1. The content library contains $F = 50$ files. For simulating the locality of content popularity distribution, this area is partitioned into 9 regions, and the content popularity distributions in these regions are different. All these regions share the same skewness parameter but differ in the order of content popularity. For example, Content 1 is the most popular content in Region 1, but it is the fifth most popular content in Region 2. Two values of skewness parameter $z \in \{0.8, 1.2\}$ are considered. All the files are assumed to have the same size, i.e., $v_f = 10$ MB for all $f \in \mathcal{F}$. For conducting the experiments on computers, the whole area is divided into 200×200 pixels. Other simulation parameters are given in Table 1.

TABLE 1: Simulation parameters.

Parameters	MBS	SBS
Transmit power	$P_M = 43$ dBm	$P_S = 33$ dBm
Path loss model	$128.1 + 37.6 \log_{10}(R [\text{km}])$	$140.7 + 36.7 \log_{10}(R [\text{km}])$
Backhaul capacity	$B_M = 1$ Gbps	$B_S \in \{1, 10, 100\}$ Mbps
Cache capacity (the number of contents)	$s_M = 0$	$s_S \in \{5, 10\}$

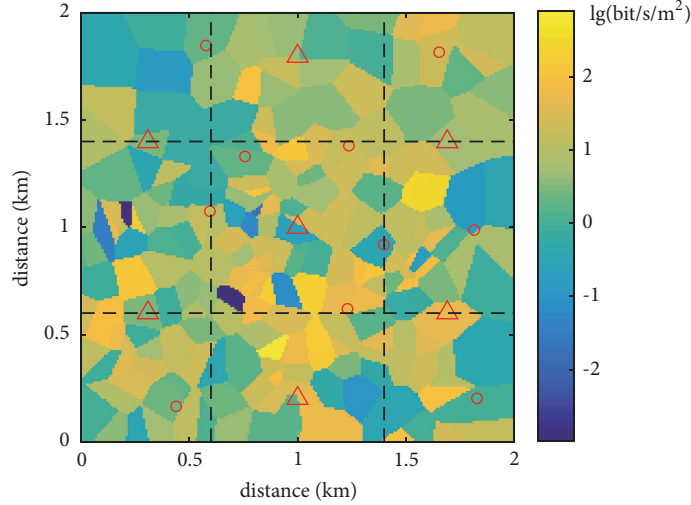


FIGURE 1: The simulation scenario. Red triangles denote MBSs and red circles denote SBSs.

7.2. Illustration of Content-Level Selective Association. Figures 2 and 3 illustrate the coverage areas associated with Content 1 and Content 2 derived from GCC-CSA algorithm in LNC model and LC model, respectively. From these figures, we can observe that the coverage areas of SBS1, SBS4 and SBS8 associated with Content 1 and Content 2 are quite different, especially in LC model. The difference in caching state of Content 1 and Content 2 results in the different coverage areas. In these two models, SBS1 and SBS8 cache Content 1 but do not cache Content 2, and SBS4 caches Content 2 but does not cache Content 1. The caching states of these two contents are identical in other SBSs. Taking SBS8 as an example, since it does not cache Content 2, the coverage area with respect to Content 2 must shrink to prevent heavy load.

7.3. Statistical Properties of Delay. Tables 2 and 3 give the average delay in the three schemes under various network configurations in LNC model and LC model, respectively. \bar{D}_S , \bar{D}_M , and \bar{D} are defined in Section 5.1. We can draw the following conclusions from these tables.

(1) Under the same configuration, \bar{D} in GCC-CSA is smaller than that in MPC-MSA and MPC-CSA, especially when B_S is small. The proposed GCC algorithm always makes BSs cache the contents that produce the maximum reduction in average delay, and the CSA algorithm enables flows requesting different contents to connect to different BSs, which reduces the average delay further. When B_S is small, CSA algorithm avoids many flows being associated with the SBSs that do not cache the requested files, and therefore the loads of SBSs decrease tremendously. When B_S is large,

however, the caching states of SBSs have minor influence on the loads of SBSs according to (5), and the advantage of GCC-CSA scheme over other schemes diminishes.

(2) In the same scenario, the three kinds of average delay in the three schemes decrease as B_S increases (except \bar{D}_M in MPC-MSA scheme in LNC model). When B_S increases, the achievable data rates from SBSs probably increase, and the loads of SBSs and \bar{D}_S decrease accordingly. In the GCC-CSA and MPC-CSA scheme, increased B_S leads to expanded coverage areas of SBSs. Thus, the coverage areas of MBSs shrink and \bar{D}_M decreases. In the LNC model, the radio link data rates provided by MBSs are independent of the loads of other BSs, and thus \bar{D}_M in MPC-MSA remains unchanged as B_S changes. In the LC model, the interference received by flows associated with MBSs is attenuated as the loads of SBSs decrease, and \bar{D}_M in MPC-MSA decreases accordingly.

(3) For given B_S , the three kinds of average delay in GCC-CSA scheme decrease (or remain unchanged) as s_S or z increases. The increase of s_S means that more contents can be cached at SBSs, and the increase of z implies that more requests are aimed at a few popular contents. Both of them increase the probability of finding the requested files at SBSs and reduce the average delay in GCC-CSA scheme. Because of the locality of content popularity distribution, the cached contents in MPC-MSA and MPC-CSA scheme are not always popular in local areas, and the increase of s_S or z possibly enlarges the average delay.

(4) In all scenarios, \bar{D}_S in GCC-CSA is smaller than that in MPC-MSA when B_S is small. When B_S is large, however, \bar{D}_S in GCC-CSA is larger than that in MPC-MSA. CSA

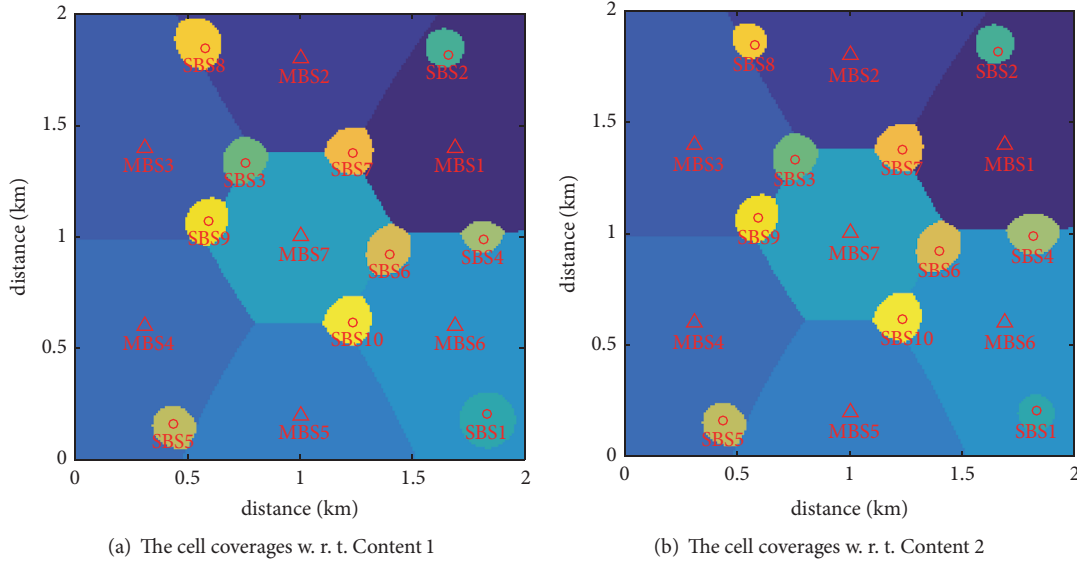


FIGURE 2: The coverage areas with respect to Content 1 and Content 2 in LNC model. $s_s = 10$, $z = 0.8$, and $B_S = 10$ Mbps.

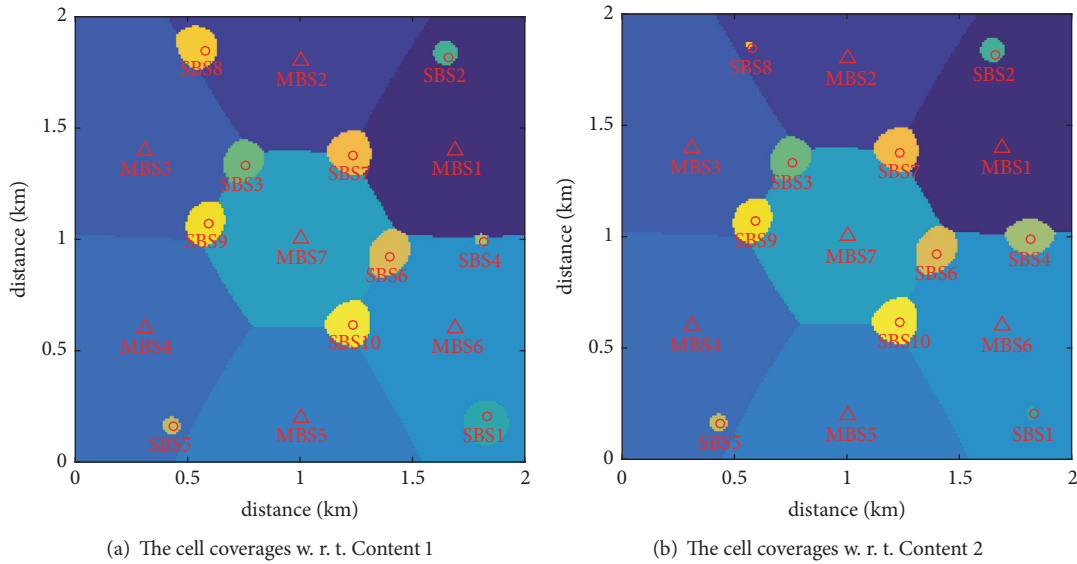


FIGURE 3: The coverage areas with respect to Content 1 and Content 2 in LC model. $s_s = 10$, $z = 0.8$, and $B_S = 10$ Mbps.

mechanism restricts the coverage areas of SBSs when B_S is small, which brings about smaller \overline{D}_S compared with MSA mechanism. When B_S becomes large, the data flows that were originally associated with MBSs probably transfer to SBSs to reduce the overall average delay in the GCC-CSA scheme. In this case, \overline{D}_S in GCC-CSA is larger than that in MPC-MSA, but \overline{D} in GCC-CSA is always smaller than that in MPC-MSA.

For visually showing the advantages of GCC-CSA scheme in reducing average delay, Figures 4 and 5 illustrate the three kinds of average delay in LNC and LC models when $B_S = 10$ Mbps, respectively. We can observe that \overline{D}_S in GCC-CSA is smaller than that in MPC-MSA and MPC-CSA. The reduction in \overline{D}_S in GCC-CSA compared with the values in MPC-MSA becomes apparent when s_s is small. When $s_s = 5$

and $z = 0.8$, the proposed scheme achieves a reduction of 19.4% and 36.1% in \overline{D}_S compared with MPC-CSA scheme in LNC and LC models, respectively. When $s_s = 5$ and $z = 1.2$, the proposed scheme achieves a reduction of 24.3% and 37.6% in \overline{D}_S compared with MPC-CSA scheme in LNC and LC models, respectively. Moreover, \overline{D}_M and \overline{D} in these schemes are almost equal. This is because the coverage areas of MBSs are much larger than those of SBSs and \overline{D}_M in these schemes are close to each other.

Figures 6 and 7 show the cumulative distribution functions (CDFs) of $\mathbb{E}[D | y]$ under a given network configuration in LNC and LC models, respectively. The CDFs of $\mathbb{E}[D | y]$ in \mathcal{Y}_M in these three schemes almost overlap. However, the distributions of $\mathbb{E}[D | y]$ in \mathcal{Y}_S in GCC-CSA

TABLE 2: The average delay in the three schemes in LNC model.

Scenario	B_S (Mbps)	\overline{D}_S (s)			\overline{D}_M (s)			\overline{D} (s)		
		GCC-CSA	MPC-MSA	MPC-CSA	GCC-CSA	MPC-MSA	MPC-CSA	GCC-CSA	MPC-MSA	MPC-CSA
$s_S = 5$ $z = 0.8$	1	12.802	160.54	14.781	2.2614	2.2710	2.2871	3.3414	13.963	3.5848
	10	5.5067	6.8358	6.0531	2.1799	2.2710	2.1871	2.5060	2.6082	2.5668
	100	2.9788	2.3292	2.9792	2.1559	2.2710	2.1572	2.2358	2.2753	2.2369
$s_S = 5$ $z = 1.2$	1	9.5583	160.66	12.890	2.2247	2.2710	2.2675	2.9620	13.972	3.3597
	10	4.6681	6.1678	5.5866	2.1718	2.2710	2.1837	2.4158	2.5589	2.5176
	100	2.9778	2.3284	2.9799	2.1559	2.1710	2.1571	2.2357	2.2753	2.2368
$s_S = 10$ $z = 0.8$	1	10.340	68.724	11.984	2.2319	2.2710	2.2507	3.0528	7.1803	3.2444
	10	4.8670	5.7330	5.3061	2.1739	2.2710	2.1794	2.4373	2.5268	2.4857
	100	2.9778	2.3282	2.9797	2.1557	2.2710	2.1571	2.2355	2.2752	2.2368
$s_S = 10$ $z = 1.2$	1	7.2567	43.406	9.3310	2.2003	2.2710	2.2227	2.7011	5.3099	2.9365
	10	4.0710	4.7195	4.6161	2.1664	2.2710	2.1738	2.3521	2.4519	2.4121
	100	2.9769	2.3271	2.9788	2.1557	2.2710	2.1571	2.2354	2.2752	2.2367

TABLE 3: The average delay in the three schemes in LC model.

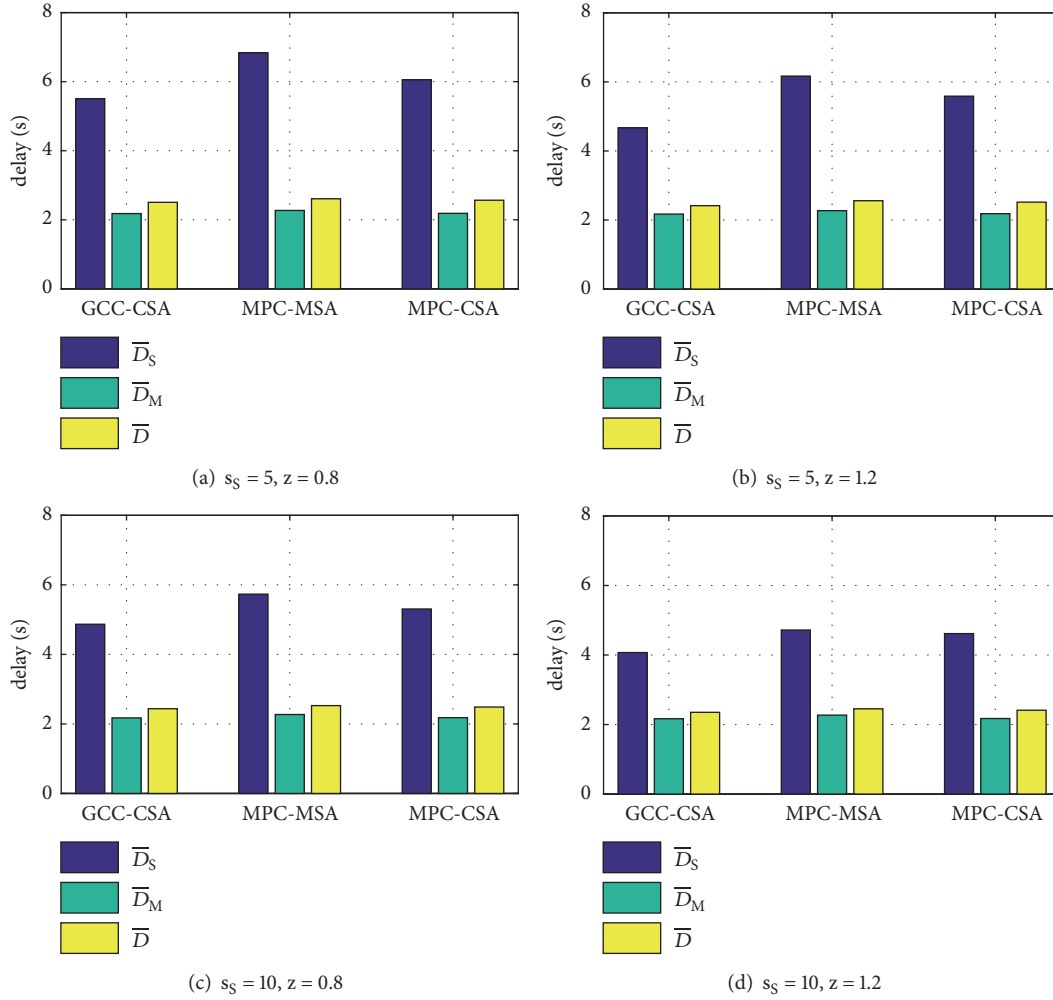
Scenario	B_S (Mbps)	\overline{D}_S (s)			\overline{D}_M (s)			\overline{D} (s)		
		GCC-CSA	MPC-MSA	MPC-CSA	GCC-CSA	MPC-MSA	MPC-CSA	GCC-CSA	MPC-MSA	MPC-CSA
$s_S = 5$ $z = 0.8$	1	7.3088	161.37	8.3352	2.5229	2.7115	2.5873	2.9197	14.433	3.0721
	10	4.3990	6.8890	4.7461	2.3257	2.3161	2.3457	2.4889	2.6539	2.5358
	100	2.7915	2.5017	2.8042	2.2574	2.2841	2.2643	2.2989	2.3001	2.3063
$s_S = 5$ $z = 1.2$	1	5.7128	161.55	7.3231	2.4270	2.6568	2.5213	2.6937	14.395	2.9204
	10	3.8943	6.2456	4.4253	2.3024	2.3123	2.3370	2.4272	2.6029	2.5022
	100	2.7903	2.5009	2.8035	2.2574	2.2841	2.2643	2.2988	2.3001	2.3063
$s_S = 10$ $z = 0.8$	1	6.0494	69.106	6.8370	2.4427	2.6058	2.4882	2.7361	7.5185	2.8459
	10	4.0119	5.8228	4.2879	2.3082	2.3081	2.3263	2.4419	2.5678	2.4811
	100	2.7891	2.5007	2.8033	2.2566	2.2841	2.2643	2.2980	2.3001	2.3062
$s_S = 10$ $z = 1.2$	1	4.6031	43.715	5.5650	2.3578	2.5175	2.4139	2.5367	5.5610	2.6692
	10	3.5125	4.8380	3.8622	2.2862	2.3015	2.3092	2.3821	2.4889	2.4312
	100	2.7880	2.4997	2.8023	2.2566	2.2840	2.2643	2.2979	2.3000	2.3062

scheme improve significantly compared with those in MPC-MSA and MPC-CSA scheme.

7.4. Backhaul Usage. Figures 8 and 9 compare $\overline{B}_{u,M}$ and $\overline{B}_{u,S}$ in these schemes in LNC and LC models, respectively. For given s_S and z , $\overline{B}_{u,M}$ and $\overline{B}_{u,S}$ in MPC-MSA do not change with the variation of B_S because MSA method is independent of backhaul capacity. $\overline{B}_{u,M}$ in GCC-CSA and MPC-CSA decrease with the increase of B_S , and $\overline{B}_{u,S}$ in GCC-CSA and MPC-CSA increase with the increase of B_S . As B_S increases, some data flows that originally connected to MBSs are associated with SBSs to lower the overall average delay, which leads to decreased $\overline{B}_{u,M}$ and increased $\overline{B}_{u,S}$. We specify the advantage of GCC-CSA scheme in backhaul usage based on Figure 8(c), and other subfigures show the same results. When $B_S = 1$ Mbps, $\overline{B}_{u,S}$ in GCC-CSA is 1722.5 bit/s less than that in MPC-MSA, but $\overline{B}_{u,M}$ in GCC-CSA is only 680.1 bit/s greater than that in MPC-MSA. When $B_S = 10$ Mbps, $\overline{B}_{u,S}$ and $\overline{B}_{u,M}$ in GCC-CSA are all smaller than those in MPC-MSA.

When $B_S = 100$ Mbps, $\overline{B}_{u,M}$ in GCC-CSA is 2078.5 bit/s less than that in MPC-MSA, but $\overline{B}_{u,S}$ in GCC-CSA is only 320.0 bit/s greater than that in MPC-MSA. On the whole, the proposed scheme occupies less backhaul capacity than MPC-MSA scheme. Furthermore, $\overline{B}_{u,S}$ and $\overline{B}_{u,M}$ in GCC-CSA are always smaller than those in MPC-CSA, which demonstrates the advantage of GCC algorithm. These conclusions can also be drawn in other scenarios in LNC and LC models. The saved backhaul capacity can be used to provide other services, such as live streaming, video calls and online games.

7.5. Optimality of GCC Algorithm. We demonstrate the optimality of GCC algorithm in a simple network as shown in Figure 10. The simplified content library contains $F = 10$ files and the cache capacity of SBS is $s_S = 2$. The four regions separated by dashed lines share the same skewness parameter but differ in the order of content popularity. We compare the GCC algorithm with exhaustive search and MPC policy, and the CSA algorithm is applied together with these content placement schemes. Table 4 lists the

FIGURE 4: Comparison of average delay in LNC model when $B_s = 10$ Mbps.

obtained cost function values of these schemes under various network configurations. In this simple network, GCC can always find the optimal content placement with much lower complexity than exhaustive search. All these schemes obtain the optimal solutions when $s_s = 100$ Mbps. This is because the implication of content placement at SBSs becomes weak when the backhaul capacity is large enough according to (5).

8. Conclusions

In this paper, we have proposed a GCC-CSA algorithm for joint optimization of content placement and user association in cache-enabled HCNs based on flow-level models. By modeling cellular networks as queuing systems, we have taken into consideration the discrepancy in the timescales of content placement and user association, the locality of content popularity and the heterogeneity of spatial traffic distribution, which are often neglected in the literature. The objective of joint optimization is to minimize the average delay of data flows, and this problem is formulated as an MINLP problem in LNC and LC models, respectively. Given the contents cached at BSs, we have proposed a CSA

algorithm that allows data flows requesting different contents to connect to different BSs. A heuristic GCC algorithm is also proposed to tackle the content placement problem, and its convergence property is proved. Simulation results show that the proposed GCC-CSA algorithm can reduce the average content delivery delay compared with traditional approaches. Especially, when the backhaul capacity of SBSs is stringent, the proposed algorithm can significantly decrease the average content delivery delay in coverage areas of SBSs compared with traditional MPC-MSA scheme. In addition, the proposed algorithm can reserve larger backhaul capacities for transmission of contents that are not reusable.

Data Availability

The data used to support the findings of this study are included within the article.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

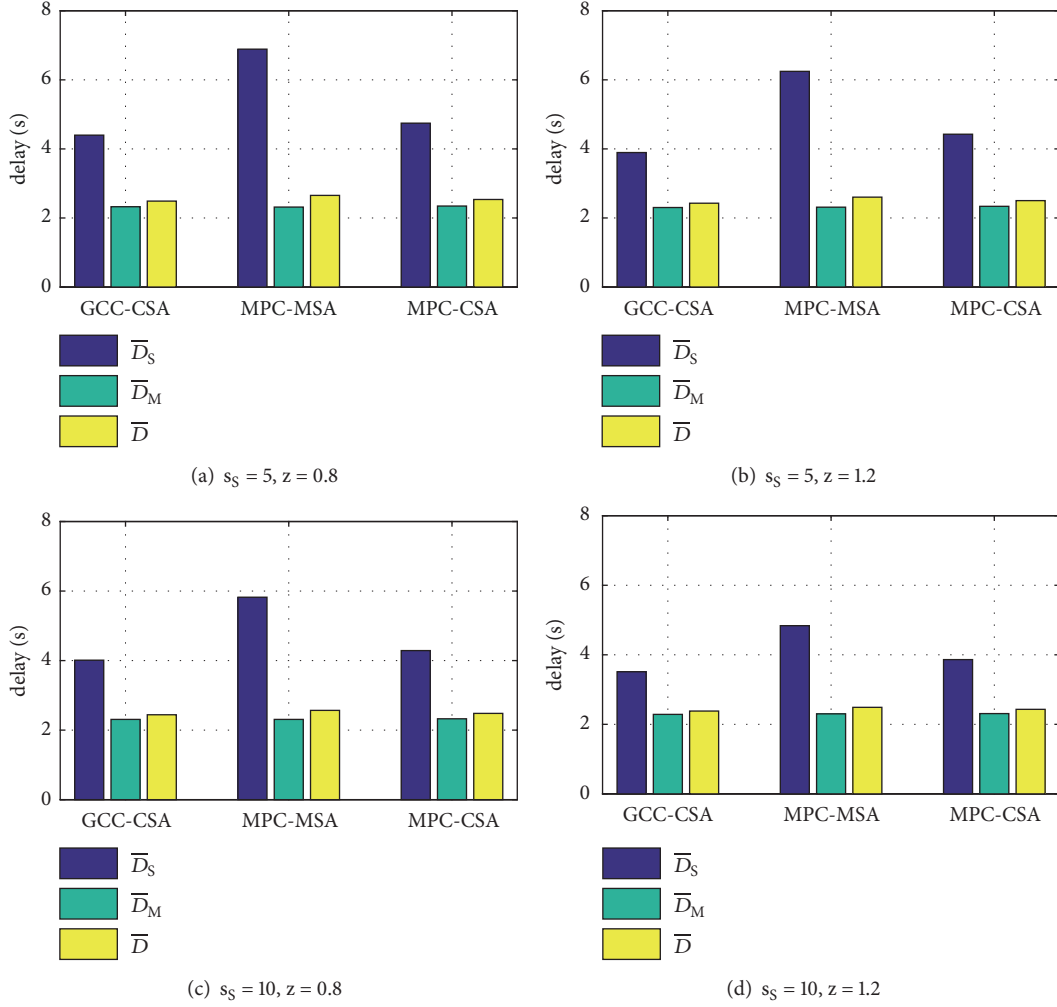


FIGURE 5: Comparison of average delay in LC model when $B_S = 10$ Mbps.

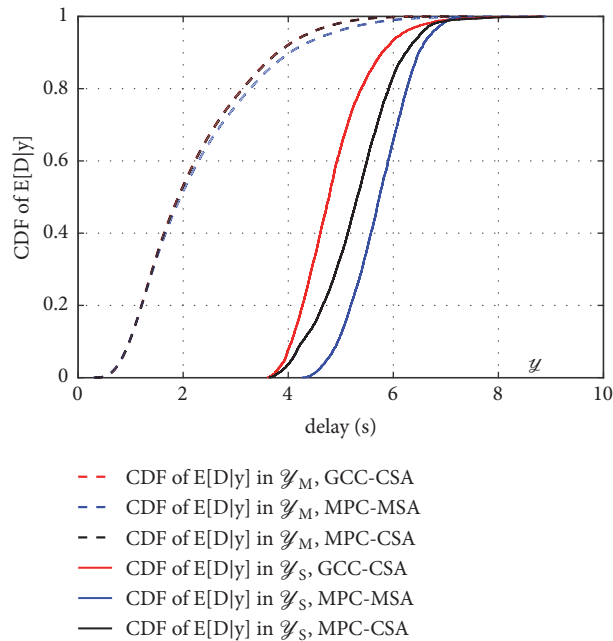


FIGURE 6: The CDFs of $E[D | y]$ in these three schemes in LNC model. $s_S = 10$, $z = 0.8$, and $B_S = 10$ Mbps.

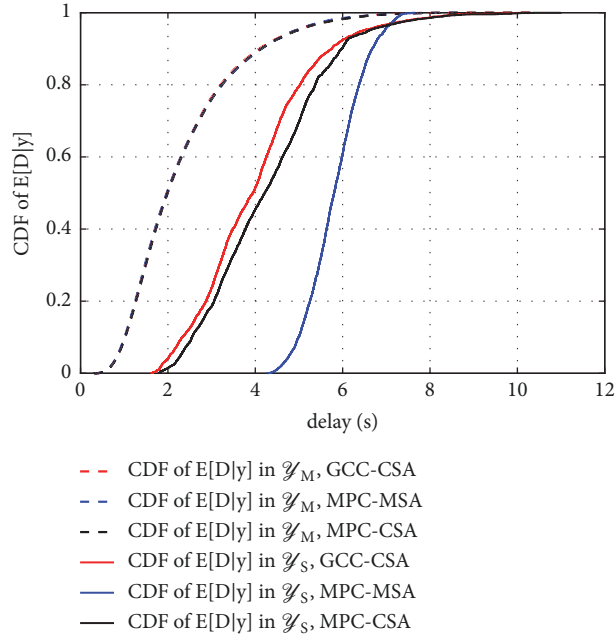


FIGURE 7: The CDFs of $E[D | y]$ in these three schemes in LC model. $s_s = 10$, $z = 0.8$, and $B_S = 10$ Mbps.

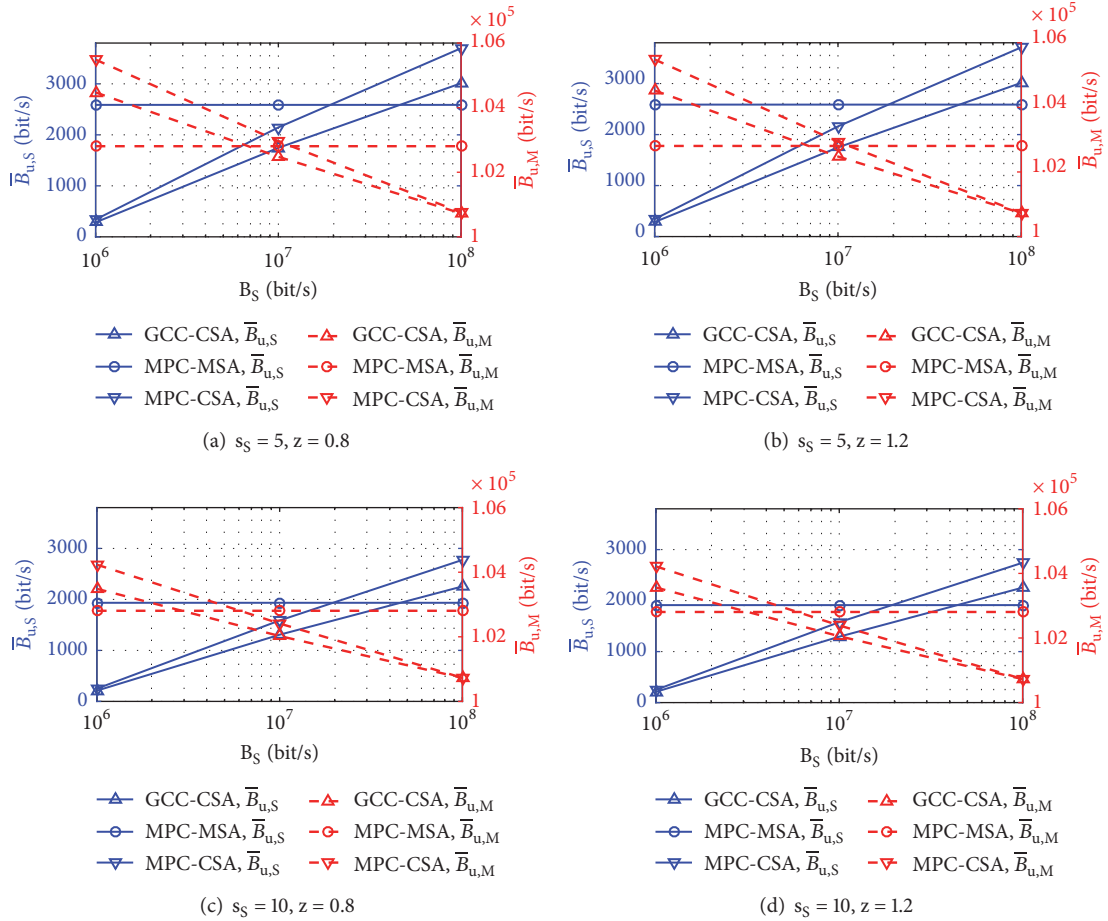


FIGURE 8: The backhaul usage of the three schemes in LNC model.

Acknowledgments

This work is supported by the National Natural Science Foundation of China under Grant No. 61531013 and National Science and Technology Major Project under Grant No. 2018ZX03001016.

References

- [1] "Cisco Visual Networking Index: Global Mobile Data Traffic Forecast Update, 2016–2021," San Jose, CA, USA, 2017.
- [2] A. Ghosh, N. Mangalvedhe, R. Ratasuk et al., "Heterogeneous cellular networks: from theory to practice," *IEEE Communications Magazine*, vol. 50, no. 6, pp. 54–64, 2012.
- [3] Z. Luan, H. Qu, J. Zhao, and B. Chen, "Low complexity distributed max-throughput algorithm for user association in heterogeneous network," *Wireless Personal Communications*, vol. 87, no. 4, pp. 1147–1156, 2016.
- [4] G. Ma, Z. Wang, M. Zhang, J. Ye, M. Chen, and W. Zhu, "Understanding performance of edge content caching for mobile video streaming," *IEEE Journal on Selected Areas in Communications*, vol. 35, no. 5, pp. 1076–1089, 2017.
- [5] H. Yu, D. Zheng, B. Y. Zhao, and W. Zheng, "Understanding user behavior in large-scale video-on-demand systems," in *Proceedings of the 1st ACM SIGOPS/EuroSys European Conference on Computer Systems*, pp. 333–344, 2006.
- [6] M. Cha, H. Kwak, P. Rodriguez, Y. Y. Ahnt, and S. Moon, "I tube, you tube, everybody tubes: analyzing the world's largest user generated content video system," in *Proceedings of the ACM SIGCOMM Internet Measurement Conference*, pp. 1–14, October 2007.
- [7] A. F. Molisch, G. Caire, D. Ott, J. R. Foerster, D. Bethanabhotla, and M. Ji, "Caching eliminates the wireless bottleneck in video aware wireless networks," *Advances in Electrical Engineering*, vol. 2014, Article ID 261390, 13 pages, 2014.
- [8] I. U. Din, S. Hassan, M. K. Khan, M. Guizani, O. Ghazali, and A. Habbal, "Caching in information-centric networking: strategies, challenges, and future research directions," *IEEE Communications Surveys and Tutorials*, vol. 20, no. 2, pp. 1443–1474, 2018.
- [9] N. Golrezaei, A. Molisch, A. G. Dimakis, and G. Caire, "Femto-caching and device-to-device collaboration: a new architecture for wireless video distribution," *IEEE Communications Magazine*, vol. 51, no. 4, pp. 142–149, 2013.
- [10] E. Bastug, M. Bennis, and M. Debbah, "Living on the edge: the role of proactive caching in 5G wireless networks," *IEEE Communications Magazine*, vol. 52, no. 8, pp. 82–89, 2014.
- [11] X. Wang, M. Chen, T. Taleb, A. Ksentini, and V. C. M. Leung, "Cache in the air: exploiting content caching and delivery techniques for 5G systems," *IEEE Communications Magazine*, vol. 52, no. 2, pp. 131–139, 2014.
- [12] D. Liu, B. Chen, C. Yang, and A. F. Molisch, "Caching at the wireless edge: design aspects, challenges, and future directions," *IEEE Communications Magazine*, vol. 54, no. 9, pp. 22–28, 2016.
- [13] M. A. Maddah-Ali and U. Niesen, "Fundamental limits of caching," *IEEE Transactions on Information Theory*, vol. 60, no. 5, pp. 2856–2867, 2014.
- [14] M. Ji, G. Caire, and A. F. Molisch, "Wireless device-to-device caching networks: basic principles and system performance," *IEEE Journal on Selected Areas in Communications*, vol. 34, no. 1, pp. 176–189, 2016.
- [15] S. Gitisenis, G. S. Paschos, and L. Tassioulas, "Enhancing wireless networks with caching: asymptotic laws, sustainability and trade-offs," *Computer Networks*, vol. 64, pp. 353–368, 2014.
- [16] E. Baştuğ, M. Bennis, M. Kountouris, and M. Debbah, "Cache-enabled small cell networks: modeling and tradeoffs," *EURASIP Journal on Wireless Communications and Networking*, vol. 2015, p. 41, 2015.
- [17] C. Yang, Y. Yao, Z. Chen, and B. Xia, "Analysis on cache-enabled wireless heterogeneous networks," *IEEE Transactions on Wireless Communications*, vol. 15, no. 1, pp. 131–145, 2016.
- [18] D. Liu and C. Yang, "Energy efficiency of downlink networks with caching at base stations," *IEEE Journal on Selected Areas in Communications*, vol. 34, no. 4, pp. 907–922, 2016.
- [19] N. Golrezaei, K. Shanmugam, A. G. Dimakis, A. F. Molisch, and G. Caire, "FemtoCaching: Wireless video content delivery through distributed caching helpers," in *Proceedings of the IEEE Conference on Computer Communications (INFOCOM)*, vol. 59, pp. 1107–1115, 2012.
- [20] J. Li, Y. Chen, Z. Lin, W. Chen, B. Vucetic, and L. Hanzo, "Distributed caching for data dissemination in the downlink of heterogeneous networks," *IEEE Transactions on Communications*, vol. 63, no. 10, pp. 3553–3568, 2015.
- [21] X. Peng, J.-C. Shen, J. Zhang, and K. B. Letaief, "Backhaul-aware caching placement for wireless networks," in *Proceedings of the 58th IEEE Global Communications Conference, GLOBECOM '15*, pp. 1–6, 2015.
- [22] B. Blaszczyszyn and A. Giovanidis, "Optimal geographic caching in cellular networks," in *Proceedings of the IEEE International Conference on Communications, ICC 2015*, pp. 3358–3363, UK, June 2015.
- [23] J. Wen, K. Huang, S. Yang, and V. O. Li, "Cache-enabled heterogeneous cellular networks: optimal tier-level content placement," *IEEE Transactions on Wireless Communications*, vol. 16, no. 9, pp. 5939–5952, 2017.
- [24] H. Qu, G. Ren, J. Zhao, Y. Shi, and Z. Tan, "Performance analysis of the content dissemination mechanism with proactive content fetching and full-duplex D2D communication: an evolutionary perspective," *Mobile Networks and Applications*, 2018.
- [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 *Proceedings of the IEEE International Conference on Communications, ICC 2015*, pp. 3082–3087, UK, June 2015.
- [26] H. Wu, L. Wang, T. Svensson, and Z. Han, "Resource allocation for wireless caching in socially-enabled D2D communications," in *Proceedings of the 2016 IEEE International Conference on Communications, ICC '16*, pp. 1–6, 2016.
- [27] F. Cheng, Y. Yu, Z. Zhao, N. Zhao, Y. Chen, and H. Lin, "Power allocation for cache-aided small-cell networks with limited backhaul," *IEEE Access*, vol. 5, pp. 1272–1283, 2017.
- [28] G. Ren, H. Qu, J. Zhao, S. Zhao, and Z. Luan, "A distributed user association and resource allocation method in cache-enabled small cell networks," *China Communications*, vol. 14, no. 10, pp. 95–107, 2017.
- [29] B. Zhou, Y. Cui, and M. Tao, "Optimal dynamic multicast scheduling for cache-enabled content-centric wireless networks," *IEEE Transactions on Communications*, vol. 65, no. 7, pp. 2956–2970, 2017.
- [30] K. Poularakis, G. Iosifidis, and L. Tassioulas, "Approximation algorithms for mobile data caching in small cell networks," *IEEE Transactions on Communications*, vol. 62, no. 10, pp. 3665–3677, 2014.

- [31] M. Dehghan, A. Seetharam, B. Jiang et al., "On the complexity of optimal routing and content caching in heterogeneous networks," in *Proceedings of the 34th IEEE Annual Conference on Computer Communications and Networks, IEEE INFOCOM 2015*, pp. 936–944, Hong Kong, May 2015.
- [32] W. Jiang, G. Feng, and S. Qin, "Optimal cooperative content caching and delivery policy for heterogeneous cellular networks," *IEEE Transactions on Mobile Computing*, vol. 16, no. 5, pp. 1382–1393, 2017.
- [33] A. Khreishah, J. Chakareski, and A. Gharaibeh, "Joint caching, routing, and channel assignment for collaborative small-cell cellular networks," *IEEE Journal on Selected Areas in Communications*, vol. 34, no. 8, pp. 2275–2284, 2016.
- [34] Y. Cui and D. Jiang, "Analysis and optimization of caching and multicasting in large-scale cache-enabled heterogeneous wireless networks," *IEEE Transactions on Wireless Communications*, vol. 16, no. 1, pp. 250–264, 2017.
- [35] B. Chen, C. Yang, and Z. Xiong, "Optimal caching and scheduling for cache-enabled D2D communications," *IEEE Communications Letters*, vol. 21, no. 5, pp. 1155–1158, 2017.
- [36] B. Dai and W. Yu, "Joint user association and content placement for Cache-enabled wireless access networks," in *Proceedings of the 41st IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP 2016*, pp. 3521–3525, China, March 2016.
- [37] Y. Wang, X. Tao, X. Zhang, and G. Mao, "Joint caching placement and user association for minimizing user download delay," *IEEE Access*, vol. 4, pp. 8625–8633, 2016.
- [38] M. Zink, K. Suh, Y. Gu, and J. Kurose, "Watch global, cache local: YouTube network traffic at a campus network - Measurements and implications," in *Proceedings of the Multimedia Computing and Networking Conference, MMCN '08*, pp. 1–13, 2008.
- [39] A. Brodersen, S. Scellato, and M. Wattenhofer, "YouTube around the world: Geographic popularity of videos," in *Proceedings of the 21st Annual Conference on World Wide Web, WWW'12*, pp. 241–250, France, April 2012.
- [40] H. Wang, J. Ding, Y. Li, P. Hui, J. Yuan, and D. Jin, "Characterizing the spatio-temporal inhomogeneity of mobile traffic in large-scale cellular data networks," in *Proceedings of the the 7th International Workshop*, pp. 19–24, June 2015.
- [41] D. Lee, S. Zhou, X. Zhong, Z. Niu, X. Zhou, and H. Zhang, "Spatial modeling of the traffic density in cellular networks," *IEEE Wireless Communications Magazine*, vol. 21, no. 1, pp. 80–88, 2014.
- [42] H. Kim, G. De Veciana, X. Yang, and M. Venkatachalam, "Distributed-optimal user association and cell load balancing in wireless networks," *IEEE/ACM Transactions on Networking*, vol. 20, no. 1, pp. 177–190, 2012.
- [43] A. Fehske, H. Klessig, J. Voigt, and G. Fettweis, "Flow-level models for capacity planning and management in interference-coupled wireless data networks," *IEEE Communications Magazine*, vol. 52, no. 2, pp. 164–171, 2014.
- [44] A. J. Fehske, H. Klessig, J. Voigt, and G. P. Fettweis, "Concurrent load-aware adjustment of user association and antenna tilts in self-organizing radio networks," *IEEE Transactions on Vehicular Technology*, vol. 62, no. 5, pp. 1974–1988, 2013.
- [45] L. Kleinrock, *Queueing Systems: Computer Applications*, John Wiley and Sons Inc, 1976.
- [46] A. J. Fehske and G. P. Fettweis, "Aggregation of variables in load models for interference-coupled cellular data networks," in *Proceedings of the 2012 IEEE International Conference on Communications, ICC 2012*, pp. 5102–5107, Canada, June 2012.
- [47] M. Harchol-Balter, *Performance Modeling and Design of Computer Systems*, Cambridge University Press, Cambridge, UK, 2013.
- [48] D. P. Bertsekas, *Nonlinear Programming*, Athena Scientific, Massachusetts, Mass, USA, 1999.
- [49] N. Sapountzis, T. Spyropoulos, N. Nikaein, and U. Salim, "Optimal downlink and uplink user association in backhaul-limited HetNets," in *Proceedings of the 35th Annual IEEE International Conference on Computer Communications, IEEE INFOCOM '16*, pp. 1–9, April 2016.
- [50] S. D. Roy, T. Mei, W. Zeng, and S. Li, "Towards cross-domain learning for social video popularity prediction," *IEEE Transactions on Multimedia*, vol. 15, no. 6, pp. 1255–1267, 2013.
- [51] H. Li, X. Ma, F. Wang, J. Liu, and K. Xu, "On popularity prediction of videos shared in online social networks," in *Proceedings of the 22nd ACM international conference on Conference on information knowledge management - CIKM*, vol. 13, pp. 169–178, 2013.
- [52] S. He, H. Tian, and X. Lyu, "Edge popularity prediction based on social-driven propagation dynamics," *IEEE Communications Letters*, vol. 21, no. 5, pp. 1027–1030, 2017.
- [53] X. Chen, Y. Jin, S. Qiang, W. Hu, and K. Jiang, "Analyzing and modeling spatio-temporal dependence of cellular traffic at city scale," in *Proceedings of the IEEE International Conference on Communications, ICC 2015*, pp. 3585–3591, UK, June 2015.
- [54] R. S. Campos, "Evolution of Positioning Techniques in Cellular Networks, from 2G to 4G," *Wireless Communications and Mobile Computing*, vol. 2017, Article ID 2315036, 17 pages, 2017.



Hindawi

Submit your manuscripts at
www.hindawi.com

