SCIENCE CHINA
Information Sciences

• RESEARCH PAPER •

June 2017, Vol. 60 062302:1–062302:15
doi: 10.1007/s11432-016-0391-2

Real-world traffic analysis and joint caching and scheduling for in-RAN caching networks

Zejue WANG, Hongjia LI* & Zhen XU

State Key Laboratory of Information Security, Institute of Information Engineering, Chinese Academy of Sciences, Beijing 100093, *China*

Received October 9, 2016; accepted December 8, 2016; published online February 7, 2017

Abstract This paper analyzes the traffic of a current LTE network in China and investigates the joint optimization of content object caching and scheduling for in-radio access network (RAN) caches. Cooperative caching has been well recognized as a way of unleashing the ultimate potential of in-RAN caches, yet its feasibility is still unexplored. Moreover, content object caching and scheduling are two key issues for cache deployment, which are usually jointly considered and resolved. However, they are triggered by different events with different time granularities. Therefore, on the basis of the real-world dataset, the feasibility of in-RAN cooperative caching is proved from aspects of network topology, traffic load difference among small base stations (SBSs) and correlation analysis of content objects requested at different SBSs. Then, it is verified that different time scales should be considered in making content object caching and scheduling decisions. To exploit in-RAN cooperative caching while meeting the time scale requirement in making caching and scheduling decisions, an optimization problem is constructed considering practical transmission constraints in wireless and backhaul. It is proved to be a quadratic assignment problem, and then, a joint caching, and wireless and backhaul scheduling algorithm is proposed based on Lagrangian relaxation and decomposition, and hastening branch and bound. The performance of the proposed algorithm is evaluated based on the real-world dataset. Results depict the relationship among the cache capacity, the number of SBSs, the connection probability of SBSs and the objective performance, and show that the proposed algorithm can achieve better performance, compared with the existing algorithms.

Keywords 5G, mobile edge caching, mobile traffic analysis, optimization, cooperative caching

Citation Wang Z J, Li H J, Xu Z. Real-world traffic analysis and joint caching and scheduling for in-RAN caching networks. Sci China Inf Sci, 2017, 60(6): 062302, doi: 10.1007/s11432-016-0391-2

1 Introduction

More than 50 billion wireless devices are expected to utilize the cellular network services by the end of the year 2020, resulting in an unprecedented growth of global mobile data traffic with 30.5 exabytes per month [1]. With the goal of sustaining the vast array of connected wireless devices and the surging traffic, as well as enriching quality of experience (QoE) for mobile users (MUs), the current cellular networks are evolving towards 5th generation (5G) networks. One of the emerging trends in the evolution of the

* Corresponding author (email: lihongjia@iie.ac.cn)

envisioned 5G networks is the deployment of small-cell networks (SCNs), where femto-, pico- and micro-cells are overlaid in traditional macro-cell aiming to increase network capacity by making cells smaller and thus reducing the distances between small base stations (SBSs) and MUs. It has been widely recognized that the SCNs can significantly improve network coverage and MUs' QoE, and boost system capacity [2].

Meanwhile, the importance of backhaul network is underscored with the unabated proliferation of terminal devices with huge rise of data-hungry services, e.g., multimedia streaming. Under this context, in-radio access network (RAN) caching, which integrates caches into the RAN to cache popular content objects, e.g., associates caches with SBSs, has recently attracted considerable attentions. As the Internet content objects, such as videos, audios, APPs, etc., are accessed by MUs through cellular networks, the trips of them from the content distribution network (CDN) or Internet data center (DC) to MUs may make MUs' content requests and transports experience backhaul link bottleneck, variable round trip time (RTT) and content server response latency. By distributing storage resources near MUs, in-RAN caching yields less access delays of the requested content objects and releases the backhaul usage. As a result, in-RAN caching is considered as one of the most disruptive paradigms in 5G networks, and has begun to be commercialized by wireless industries, e.g., Altobridge's "Data at the Edge" solution [3] and Saguna Networks' Open RAN platform [4].

Excellent works have been done to improve the performance of the caching system, yet very few works explore the following problems.

Problem 1: Is it feasible to improve the performance of in-RAN caching system by means of cooperative caching? Cooperative caching has been widely recognized as an efficient method for improving overall caching performance. However, the feasibility of cooperative caching in RAN has rarely been investigated.

Problem 2: Should content object caching and scheduling be jointly considered? Content object caching and scheduling are two key issues for deploying in-RAN caches, which often in the relevant literatures are either separately determined, or jointly considered and resolved within the same time scale. Which way is more reasonable considering the differences and correlations between them?

Problem 3: How to unleash the ultimate potential of in-RAN caches? The well tuned content object caching and scheduling strategies for in-RAN caches should consider the features of RAN, e.g., the traffic conditions of SBSs, the degrees of freedom in selecting wireless and backhaul links, the possibility of global optimization of caching resource, etc.

Keeping these motivation problems in mind, we analyze the measured dataset which is a six-day trace of all requests and corresponding replies of an LTE network in China, and then study the joint optimization of content object caching and scheduling for in-RAN caches based on the findings. Specifically, the main contributions of this paper are summarized as follows.

(1) Traffic analysis of a current LTE network: On the basis of the real-world dataset, we prove the feasibility of in-RAN cooperative caching, and find that the content caching and scheduling problems should be resolved at different time scales for being triggered by different events with different time scales.

(2) Formulation of transmission delay minimization problem: To minimize the content object transmission delay, the joint content object caching and scheduling problem is constructed, considering practical transmission constraints in wireless and backhaul, as well as the time granularity requirement in making caching and scheduling decisions.

(3) Joint caching, and wireless and backhaul scheduling (JCWBS) algorithm: The formulated problem is proved to be a quadratic assignment problem (QAP), and the binary solution is obtained based on Lagrangian relaxation and decomposition, and hastening branch and bound (B&B). By utilizing the real-world dataset, the effectiveness of the proposed algorithm is verified.

The rest of this paper is organized as follows. Section 2 overviews the related works. Section 3 depicts the analysis of the obtained real-world dataset. Section 4 presents system framework, models and constraints. Section 5 provides problem formulation, solution methodology and proposed algorithm. Section 6 evaluates the performance of the proposed algorithm. Finally, Section 7 concludes the paper and depicts the future work.

2 Related work

A significant body of works have tried to investigate the performance of caching system. In [5], the evidence of the potential gains associated with the deployment of micro CDN technologies in Internet Service Provider (ISP) networks is provided on the basis of traffic analysis of Orange network in France. In [6], the detailed characterization and comparison of content delivery systems are provided from the perspective of traffic flowing in and out of the University of Washington. However, very few works focus on the analysis of RAN traffic, and address the caching and scheduling problems based on their characteristics.

In parallel with the aforementioned works which mainly focus on the traffic characteristic analysis, the content object caching strategy, and its associated scheduling problem have also been extensively studied for web caching and distributed storage systems, e.g., CDNs [7] and Peer-to-Peer (P2P) networks [8]. The efficiency of the caching strategy is limited by the number of content objects that can be stored, but caches associated with SBSs are more tend to be designed much smaller in comparison with storage units in CDN or ISP DC, considering the fact that the overall cost of distributed caches can be high due to the huge number of deployed SBSs if caches with larger capacity are utilized.

In in-network caching architectures, operators can cache popular content objects in the Evolved Packet Core (EPC) or the RAN, e.g., in dedicated core network (CN) elements or at the cellular BSs. In [9], a cost model is proposed to estimate caching at different levels in the 3G CN hierarchy. In [10], the gains for caching different types of content objects at serving gateway (S-GW) in an LTE network are compared. Compared with EPC caching, one important feature of in-RAN caching is that the number of MUs served by individual BSs is usually small and they usually move frequently, resulting in low-to-moderate hit ratios. More intelligent caching and scheduling strategies are mandatory for in-RAN caches.

In context of in-RAN caches, prior arts have delved into the content object caching and scheduling problems in [11–15]. In [11], the content object caching problem is formulated to minimize the access delay. In [12], an user preference profile based caching strategy is proposed. However, the focus of above researches is mainly on the optimization of content object caching, while ignoring the content object scheduling decisions. The problem of how to deliver the cached content object to ultimate destinations is studied from perspectives of minimizing transmission delay in [13], and reducing energy consumption in [14,15]. However, where to cache each content object (the caching strategy) affects how to fetch it (the scheduling strategy), while how to schedule each content object also in turn affects the effectiveness of the caching strategy. Thus, the two problems are interrelated and interact with each other, and should be jointly considered.

The problem of joint content object caching and scheduling has been investigated in [16,17]. In [16], the problem is investigated with the aim of minimizing the average transmission delay. In [17], the caching domains which support joint caching and scheduling are formed to improve the cache hit ratio. Even the two problems interact with each other, content object caching and scheduling are usually triggered by different events, where the content object caching decision is based on the change of content popularity which might be on time scale of several or a dozen hours, while the content object scheduling decision should be made instantaneously at the time the request arrives.

Our work differs from above existing works with the following three aspects: (1) we extensively analyze the mobile traffic recorded at a current LTE network in China, and study the feasibility of in-RAN cooperative caching and the difference between "triggers" for content object caching and scheduling; (2) to minimize the content object transmission delay, we construct the joint content object caching and scheduling problem based on the findings; (3) to find the optimal binary solution, the joint caching, and wireless and backhaul scheduling (JCWBS) algorithm is proposed based on Lagrangian relaxation and decomposition, and hastening B&B, and the effectiveness of the proposed algorithm is verified by utilizing the real-world dataset.

Table 1 Explanations of fields

Field	Description
Time stamp	Time instant when a Http request is initiated
User ID	Anonymous user ID
Cell ID	Indicating the serving cell of each MU
Content ID	Requested uniform resource locator (URL)
Content type	Application, audio, image, video, text or other
Downstream traffic	Actual downstream traffic generated by the content request

3 Traffic analysis

In this section, an overview of the dataset traced at a current cellular network in China is provided. Then, a detailed analysis of the obtained dataset is presented, which is performed from perspectives of investigating the feasibility of in-RAN cooperative caching and comparing the difference between "triggers" for content object caching and scheduling. Besides, two findings are provided corresponding to Problems 1 and 2 in Section 1.

3.1 Measured dataset

We trace all hypertext transfer protocol (Http) requests and corresponding replies of a real-world LTE network in China, which covers a comprehensive university campus including areas of teaching, sporting, living, entertainment, etc, from 00:00 on April 10, 2015, to 23:00 on April 15, 2015. Each Http request/reply transaction has a record of several fields, e.g., time stamp, user identity (ID), cell ID, etc, the explanations of which are provided in Table 1. During the six-day trace, it is recorded about 62 million Http sessions issued from 3876 MUs which are distributed over 10 LTE SBSs. The total amount of observed downlink traffic is more than 1.7 TB.

3.2 Feasibility of in-RAN cooperative caching

In the following, we study the feasibility of in-RAN cooperative caching, starting by investigating possible physical links among SBSs. Then, on the basis of the real-world dataset, we present the traffic load difference among SBSs over time, and the correlation of content objects requested at different SBSs.

3.2.1 *Physical links among SBSs*

SBSs in the same network domain could connect indirectly via a "U-turn" through an aggregation gateway, which acts as concentrator for S1 interface, along the path to the CN. Access options, e.g., point-to-point fiber, gigabit-capable passive optical network (GPON) and bonded high-rate digital subscriber line (DSL), as well as line-of-sight (LOS) wireless and non-LOS are possible solutions for the above backhaul links. Other than the indirect connections, SBSs in LTE (-A) can directly communicate with their neighbor SBSs with X2 interface, using point-to-point microwave, millimeter wave or optical fiber as the physical X2 link solution [18].

The direct and indirect links among SBSs lay the foundation for their associated caches to cooperate with each other.

3.2.2 *Traffic load difference among SBSs*

The snapshots of the traffic loads at different SBSs are provided in Figure 1, which illustrates the traffic loads of 10 SBS at 9:00, 13:00, 17:00 and 21:00, on April 12 (weekend) and April 15 (workday). Each SBS is represented by a circle, the color of which is corresponding to the traffic load: the heavier the traffic load is, the lighter the color is. As we focus on the difference of the traffic loads among SBSs, the values of the traffic loads are normalized to $[0, 1]$. It can be observed that the traffic load of one SBS changes over time which may result from the change of MUs' locations or interests. In the space, the traffic load differs from SBS to SBS, which may caused by the difference of MUs' distribution on space.

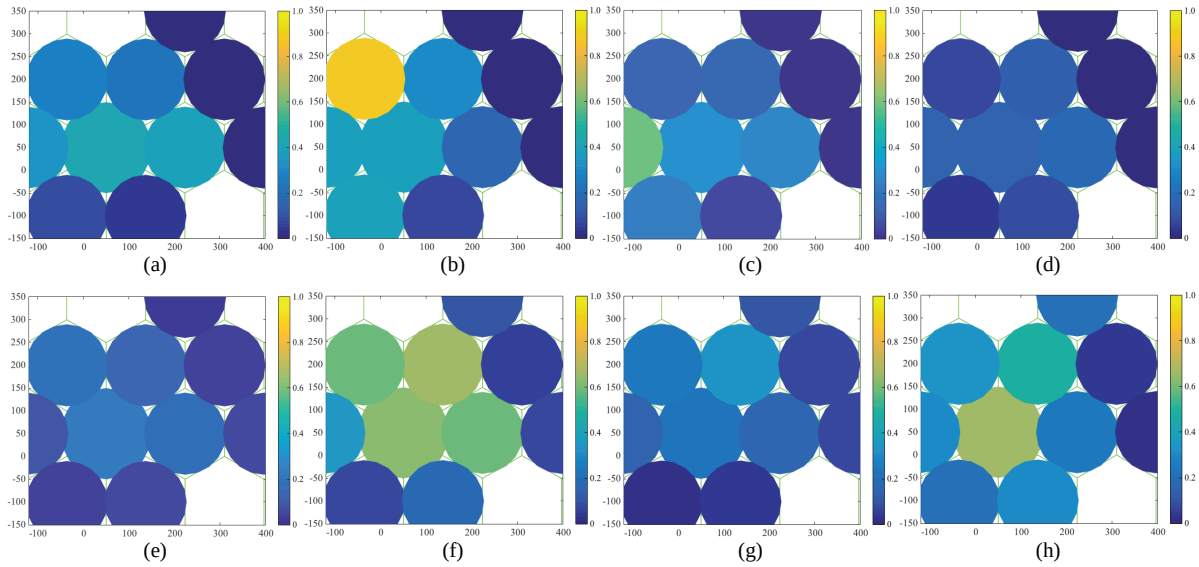

Figure 1 Snapshots of different SBSs' traffic loads on April 12 (weekend) and April 15 (workday). (a) At 9:00 on April 12; (b) at 13:00 on April 12; (c) at 17:00 on April 12; (d) at 21:00 on April 12; (e) at 9:00 on April 15; (f) at 13:00 on April 15; (g) at 17:00 on April 15; (h) at 21:00 on April 15.

To quantify the traffic load difference among SBSs, we then depict the maximum and minimum values of the traffic loads among SBSs in Figure 2. The data is plotted with a resolution of one hour. At the same time instant, the maximum and minimum values of the traffic loads among SBSs are connected by a vertical line, thus the longer the length of the vertical line, the bigger the difference of traffic loads among SBSs. It is shown that the traffic load is usually higher in the daytime (10 a.m.–9 p.m.) than at night (2 a.m.–6 a.m.). Besides, the traffic loads of different SBSs are quite different in the daytime, implying that the available resources, e.g., bandwidth and computing resources, are different for different SBSs.

Thus, except for serving its own MUs, SBSs with light traffic are capable of cooperating with others.

3.2.3 *Cross correlation of content objects requested at different SBSs*

We use cross correlation ratio (CCR) to estimate the degree of correlation of the content object sets requested at different SBSs. The CCR is a measure of cross similarity based on the Jaccard coefficient which indicates the proportion of objects in common between two given sets and is expressed as

$$\mathcal{J}\left(\psi_x, \psi_y\right) = \frac{|\psi_x \cap \psi_y|}{|\psi_x \cup \psi_y|}, \tag{1}$$

where ψ_x and ψ_y denote any two different sets, $|\cdot|$ is the cardinality of the set and $\mathcal{J}\left(\psi_x, \psi_y\right) = 0$ if $\psi_x = \psi_y = \emptyset$. Then the cross correlation function is defined as

$$\mathrm{CCR}\left(n\right) = \frac{1}{|\mathcal{I}|}\sum_{i=1}^{|\mathcal{I}|}\left[\frac{1}{J}\sum_{j=1}^{J}\mathcal{J}\left(\mathcal{C}_i, \Pi_{i,n}^{j}\right)\right]. \tag{2}$$

In (2), $\mathcal{C}_i$ is set of the unique content objects requested at SBS i during one day; $\Pi_{i,n}^{j}$ is the set of the unique content objects requested at other n SBSs at the same time period except for SBS i; j is the index of the set; J is the total number of the set $\Pi_{i,n}^{j}$; $\mathcal{I}$ is the set of all the SBS and $|\mathcal{I}| = 10$ in this paper.

Figure 3 shows CCR(n) for $n = \{1, 2, \ldots, 9\}$ averaged over six days. The results of CCR in terms of the number of content requests and the generated traffic are also provided. It can be observed that CCRs increase as the number of SBSs in the set $\Pi_{i,n}$ increases. For instance, about 14% of the content objects requested at one SBS in one day is the same with that requested at other $n = 1$ SBSs in the same time horizon, accounting for 22% and 11% of the total number of the content requests and the generated

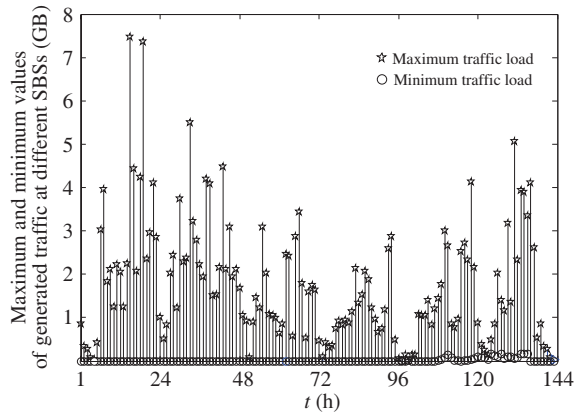

Figure 2 Maximum difference of traffic load among SBSs.

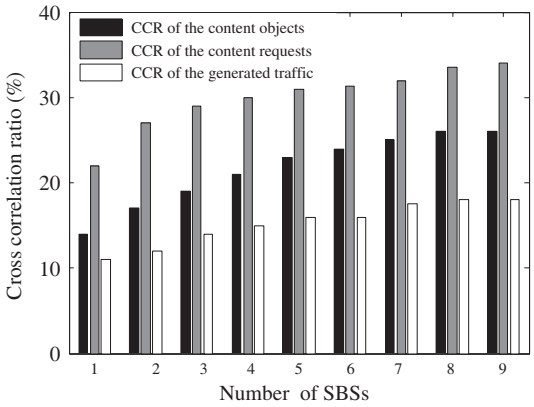

Figure 3 CCRs for different number of SBSs.

traffic, respectively. While for $n = 9$ SBSs, the CCRs in terms of the content objects, the number of content requests and the generated traffic are 24%, 32% and 18%, respectively. The higher the value of the CCR, the larger the probability that SBSs can cooperate with each other.

After investigating the possible physical links, the traffic load difference among SBSs and the cross correlation of content objects requested at different SBSs, here we can obtain the following finding.

Finding 1. It is feasible to leverage cooperation to unleash the ultimate potential of in-RAN caches.

3.3 Difference between "triggers" for content object caching and scheduling

The change of request arrival often acts as a "trigger" for the updating of the content object scheduling algorithm, while the change of content popularity is generally the determining factor that drives the updating of the content object caching algorithm. Figure 4 provides the number of content requests received at different SBSs, where the horizontal axis shows the particular point in time and each asterisk represents the number of content requests received at one SBS within one hour. It can be observed that other than the deep sleeping time (0 a.m.–5 a.m.), the average number of arrived content requests per hour is changed on the order of several hundred, implying a great fluctuation in available resources at SBS, and thus the content object scheduling algorithm should be adjusted accordingly to exploit the freedom of serve node selection.

Next, the quantization index of auto correlation ratio (ACR) is used to demonstrate the temporal evolution of the requested content objects. The ACR is a measure of auto similarity of one set and is also based on the Jaccard coefficient. The auto correlation function can be described as

$$\mathrm{ACR}\left(\alpha\right) = \frac{1}{|\mathcal{I}|}\sum_{i=1}^{|\mathcal{I}|}\left[\frac{1}{N_\alpha}\sum_{\alpha 1,\alpha 2:|\alpha 1-\alpha 2|=\alpha}\mathcal{J}\left(\mathcal{C}_{i,\alpha 1},\mathcal{C}_{i,\alpha 2}\right)\right], \tag{3}$$

where $\mathcal{C}_{i,\alpha 1}$ is the set of content objects requested at SBS i at time $t = \alpha 1$; $\mathcal{C}_{i,\alpha 2}$ is the set of content objects requested at SBS i after $t = \alpha$ hours later; $\alpha 1$ and $\alpha 2$ are integers; N_α is the number of pairs of $\alpha 1$ and $\alpha 2$ which satisfy $|\alpha 1 - \alpha 2| = \alpha$ and $\alpha 1, \alpha 2 < 24$. Figure 5 depicts the ACR(α) for $\alpha = \{1, 2, \ldots, 23\}$, where the gray area shows standard deviation as error bars and the horizontal axis expresses α. It can be observed that the ACR of the content objects changes slowly which at least over several hours, e.g., it changes about 0.3% within 12 h. Considering the results in Figure 4, Finding 2 can be obtained.

Finding 2. Content object caching and scheduling decisions should be made at different time scales, since they are triggered by different events with different time granularities.

In the following, we focus on the problem—how should in-RAN caches cooperate with each other based on Findings 1 and 2.

Notations: We use the bold curlicue characters to denote sets, e.g, $\mathcal{I}$ is the set of all SBSs. It is assumed that each SBS is associated with only one cache, the index of SBS, i.e., $i \in \mathcal{I}$, is reused for the cache associated with SBS i (short for cache i). Some other notations are summarized in Table 2.

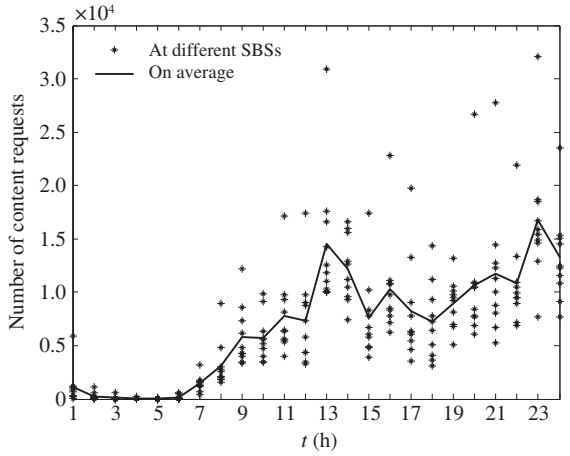

Figure 4 Hourly received content requests.

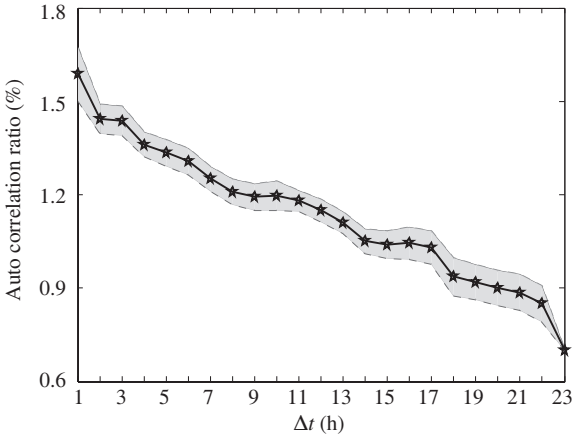

Figure 5 Auto correlation ratio of the content objects.

Table 2 A list of notations

$\mathcal{K}$	The set of MUs
$\mathcal{L}$	The set of backhaul links
M	Number of time slot (TS) per time interval (TI)
N_i	The maximal number of MUs served by SBS i
$\hat{N}_j$	The maximal number of users served by cache j
$N_{i,m}^{\text{rec}}$	Number of content-receiving MUs served by SBS i at the mth TS
$\hat{N}_{j,m}^{\text{rec}}$	Number of content-receiving users served by cache j at the mth TS
$\phi(k)$	The content object requested by MU k
$d_{i,k}^m$	The wireless transmission delay between SBS i and MU k at the mth TS
$\hat{d}_{j,i,k}^m$	The backhaul transmission delay between cache j and SBS i at the mth TS
Θ_i	The set of content objects stored in cache i
s_k	The size of content object $\phi(k)$
Φ_i	The capacity of cache i

4 System framework, models and constraints

4.1 System framework based on software defined features

We study the downlink operation of a cache enabled cellular network with software defined features [13], like the one presented in Figure 6. SBSs can be connected with each other in the same network domain with a probability, and then converged to the CN via backhaul links. The SBS is able to send periodical statistics of link states and load information to the central orchestrator. The orchestrator located behind Packet Data Network Gateway (P-GW) is an edge node in the CDN near SCN, and can communicate with caches via out-of-band communication links, e.g., in the optical-fibre based network, a single circuit can provide multiple interfaces sharing one physical link, and thus one or more of these interfaces can be used for out-of-band data transmission between the central orchestrator and the cache.

SBS associated with cache: Each SBS is equipped with the ability of filtering MU's packets, extracting the destination address and redirecting the content request. Caches are able to store popular content object to reduce content object transmission delay.

Orchestrator: The orchestrator is able to periodically make and give out caching decisions, and coordinate all in-RAN caches through wireless and in-RAN backhaul links to schedule the content objects to the content-requesting MUs.

The content requests issued by MUs are redirected to the orchestrator under a series of standard procedures within CDN. According to the type of the content server node, there are three cases when

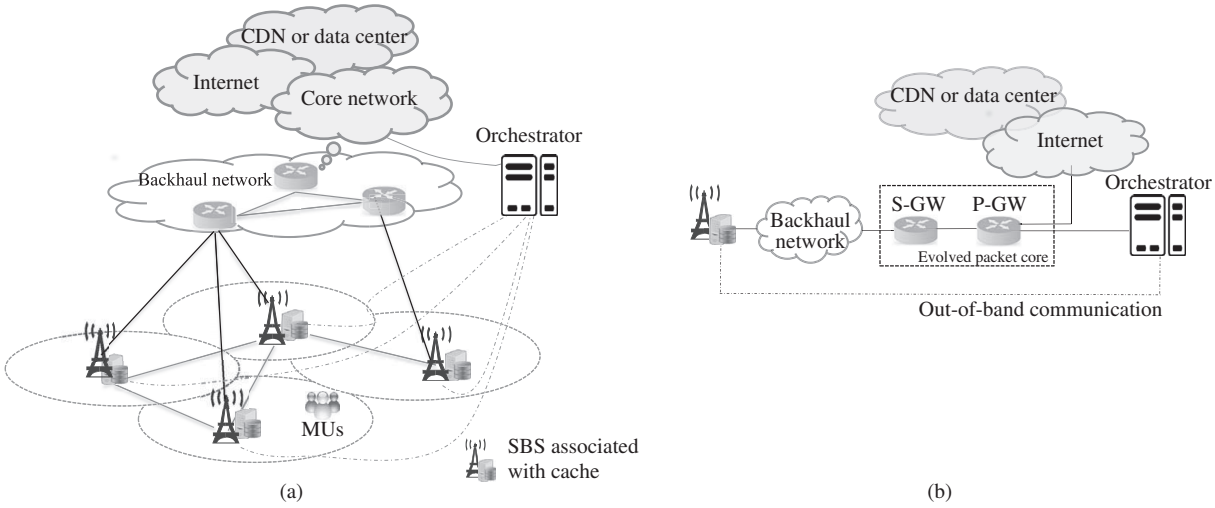

Figure 6 An illustration of the discussed framework. (a) An overview of the system framework; (b) an example of the caching system.

the MU gets service.

Case 1. If the requested content object is stored in the cache of the serving SBS, i.e., the local cache hit happens, the content request is locally served.

Case 2. If the orchestrator cannot retrieve the requested content object from the global content object log, the request is forwarded to the CDN or the ISP DC.

Case 3. If not, the requested content object will be extracted from other caches and sent back to the serving SBS under the schedule of the orchestrator utilizing the backhaul link log, the wireless link log and the global content object log which are defined in Subsection 4.2.

As Case 3 contains cooperation among caches and is the key and the most challenging case of in-RAN caching, it thus will be mainly discussed in the following.

4.2 System model

(1) The backhaul link log: The backhaul network can be modeled as a connected graph $\mathcal{B} = \{\mathcal{I}, \mathcal{L}\}$ named as the backhaul link log, where $\mathcal{I}$ is the set of all SBSs and $\mathcal{L}$ labels all links among SBSs.

(2) The wireless link log: The set of all MUs is $\mathcal{K}$ with integer index k. The wireless link log can be mathematically described as a matrix $\mathcal{W} = [e_{i,k}]_{|\mathcal{I}|\times|\mathcal{K}|}$,which expresses whether SBS $i \in \mathcal{I}$ is available for MU $k \in \mathcal{K}$. If the reference signal received power (RSRP) recieved by MU k from SBS i is greater than a certain threshold δ, $e_{i,k} = 1$, otherwise $e_{i,k} = 0$. The wireless link log can be maintained in the orchestrator by utilizing results from the MU assisted cell reselection measurement [19].

(3) The global content object log: The global content object log is defined as $\boldsymbol{\Theta} = \{\boldsymbol{\Theta}_1, \ldots, \boldsymbol{\Theta}_i, \ldots \boldsymbol{\Theta}_{|\mathcal{I}|}\}$, where $\boldsymbol{\Theta}_i$ is a list of content objects cached in SBS i consisting of the index of the content object and its corresponding request times at SBS i.

4.3 Caching and cooperation constraints

Without loss of generality, the time is slotted and divided into time intervals (TIs) with l_{TI} hours consisting of several time slots (TSs) with length l_{TS}. It is assumed that the length of TI is an integral multiple of TS, denoted as $l_{\mathrm{TI}} = M \cdot l_{\mathrm{TS}}$ where M is a positive integer. The caching decisions can be updated with period of TI, e.g., at predictably low traffic times, such as at midnight or early morning hour, while all in-RAN caches can be coordinated to perform scheduling decisions at the beginning of any TS. For clarity, the following paper is discussed within one reference TI, e.g., the sth TI.

Let $\phi(k)$ denote the content object requested by MU k and binary variable $y_{i,k}$ indicate whether the requested content object $\phi(k)$ is stored in cache $i \in \mathcal{I}$ ($y_{i,k} = 1$) or not ($y_{i,k} = 0$). Since the capacity of

each cache is limited, the total size of the content objects stored in cache i is constrained which can be expressed as

$$\sum_{k\in\mathcal{K}} y_{i,k}\cdot s_k \leqslant \Phi_i,\ \ \forall i, \tag{4}$$

where s_k denotes the size of the content object $\phi(k)$ and Φ_i is the capacity of cache i.

Further, denote binary scheduling indicator variable at the mth TS as $x^m_{j,i,k},\ \forall j,i\in\mathcal{I},\ k\in\mathcal{K},\ m\in\{1,2,\ldots,M\}$. $x^m_{j,i,k}=1$ denotes that the content-requesting MU k is associated with SBS i and cache j is selected as the content server, at the mth TS, otherwise, $x_{j,i,k,m}=0$. Note that $j=i$ denotes a local hit event, i.e., the content-requesting MU k is associated with and directly served by SBS i.

As each content request is satisfied with one cache, $x^m_{j,i,k}$ meets with

$$\sum_{j\in\mathcal{I}}\sum_{i\in\mathcal{I}} x^m_{j,i,k}=1,\ \ \forall k,m. \tag{5}$$

Besides, only the cache which stores the content object $\phi(k)$ can be selected, thus

$$x^m_{j,i,k}\leqslant y_{j,k},\ \ \forall j,i,k,m. \tag{6}$$

In addition, to guarantee the minimal backhaul and wireless transmission rates, the following constraints should be satisfied,

$$\sum_{i\in\mathcal{I}}\sum_{k\in\mathcal{K}} x^m_{j,i,k}\leqslant \hat{N}_j-\hat{N}^{\rm rec}_{j,m},\ \ \forall j, \tag{7}$$

$$\sum_{j\in\mathcal{I}}\sum_{k\in\mathcal{K}} x^m_{j,i,k}\leqslant N_i-N^{\rm rec}_{i,m},\ \ \forall i, \tag{8}$$

where $\hat{N}_j$ and N_i are the maximal numbers of users served by cache j and SBS i, respectively, and $\hat{N}^{\rm rec}_{j,m}$ and $N^{\rm rec}_{i,m}$ are the numbers of content-receiving users associated with cache j and SBS i at the mth TS.

4.4 Wireless and backhaul transmission models

The total transmission delay includes the wireless transmission delay of delivering the content object from the serving SBS to the content-requesting MU, and the backhaul transmission delay of delivering the requested content object from the content object server cache to the serving SBS.

Taking into consideration common content objects such as video chunks and APPs, both the time of the first chunk of the content object received by the MU (i.e., the startup time), and the time of the whole content object received by the MU, are important factors of MU's QoS. Since the period of content object transmission is much greater than the time scale of resource scheduling (e.g., that in LTE (-A) physical and multimedia access control layers), it is more rational that the orchestrator only considers the average wireless transmission rate, and each SBS implements the fine resource scheduling, coding, modulation and interference management [20]. Specifically, the average wireless transmission delay of delivering the content object from SBS i to MU k at the mth TS, $d^m_{i,k}$, can be calculated as

$$d^m_{i,k}=\frac{s_k}{\omega^m_{i,k}\cdot c_i}, \tag{9}$$

where $\omega^m_{i,k}$ denotes the fraction of bandwidth that SBS i uses to serve MU k at the mth TS, and satisfies $\sum_{k\in\mathcal{K}}\omega^m_{i,k}=1$; c_i is the total bandwidth of SBS i. Assuming the content object transmission meets with the fairness criteria , i.e., all content object transmissions equally share the total bandwidth, thus

$$\omega^m_{i,k}=\frac{1}{\sum_{j\in\mathcal{I}}\sum_{k\in\mathcal{K}} x^m_{j,i,k}+N^{\rm rec}_{i,m}}.$$

The backhaul transmission delay can be obtained with the method in [21]. Every SBS periodically probes the RTT from its reachable in-RAN caches and sends the results to the orchestrator. Then the backhaul transmission delay of delivering content object $\phi(k)$ from cache j to SBS i at the mth TS is

$$\hat{d}^m_{j,i,k}=s_k\cdot\min_{l\in\mathcal{L}^m_{j,i}}\left\{t^{\rm b}_h\right\}, \tag{10}$$

where t_h^{b} is the transmission delay per bit at link $l \in \mathcal{L}_{j,i}^m$ at the mth TS; $\mathcal{L}_{j,i}^m \subset \mathcal{L}$ is the set of all links from SBS i to SBS j without loops where the requested content object $\phi(k)$ can be found along the links.

5 Problem formulation, solution methodology and proposed algorithm

5.1 Optimization problem formulation

The transmission delay minimization problem is formulated as

$$\begin{aligned} \mathcal{P} \quad & \min_{\{x_{j,i,k}^m, y_{j,k}\}} \ \mathcal{O} = \sum_{j\in\mathcal{I}} \sum_{i\in\mathcal{I}} \sum_{k\in\mathcal{K}} \sum_{m=1}^{M} x_{j,i,k}^m \cdot \left(d_{i,k}^m + \hat{d}_{j,i,k}^m\right) \cdot p_{i,k}^m \\ & \text{s.t.} \qquad (4), (5), (6), (7), (8). \end{aligned} \tag{11}$$

In (11), $p_{i,k}^m$ is the requesting demand of MU k at SBS i at the mth TS, and can be predicted [22]. Since predictive analytics is not the topic considered in this paper, it is assumed that the value of $p_{i,k}^m$ has been known by using one of the effective prediction methods, such as linear regression and Bayesian forecasting, based on the obtained historical data.

The optimization problem (11) is a quadratic assignment problem (QAP) and NP-hard.

Proof. The objective function of (11) is a combination of a set of quadratic functions, each of which can be reformulated as

$$x_{j,i,k}^m \cdot \left(d_{i,k}^m + \hat{d}_{j,i,k}^m\right) \cdot p_{i,k}^m = \left((x_{j,i,k}^m)^2 \frac{s_k}{c_i} + x_{j,i,k}^m g_{j,i,k}^m + x_{j,i,k}^m \hat{d}_{j,i,k}^m\right) \cdot p_{i,k}^m, \tag{12}$$

where

$$g_{j,i,k}^m = \frac{s_k}{c_i} \left(x_{j,i,1}^m + \cdots + x_{j,i,k-1}^m + x_{j,i,k+1}^m + \cdots + x_{j,i,|\mathcal{K}|}^m\right). \tag{13}$$

Thus, combining (12) and the binary nature of the optimization variables $\{x_{j,i,k}^m, y_{j,k}\}$ yield the optimizaition problem corresponding to the format of QAP, which has been well proved NP-hard in [23].

Considering the complexity of problem (11), it is unpractical to solve it directly. In the following, the binary variables are first relaxed, and then the problem is decomposed and the solutions can be derived.

5.2 Relaxation and problem decomposition

Relaxing all binary variables $\{x_{j,i,k}^m, y_{j,k}\}$ to real continuous region $[0, 1]$. By associating with Lagrangian multipliers $\mu = \{\mu_{j,i,k}^m, \forall j, i \in \mathcal{I}, k \in \mathcal{K}, m \in \mathcal{M}\}$, constraint (6) which consists of two optimization variables, can be incorporated into the objective function of problem (11). Then, the optimization problem (11) can be divided in to a set of joint wireless and backhaul scheduling problems, $\mathcal{P}1^m$, $\forall m \in \{1, 2, \ldots, M\}$, and a content object caching problem $\mathcal{P}2$, which are described as

$$\begin{aligned} \mathcal{P}1^m \quad & \min_{\boldsymbol{x}^m} \ \mathcal{O}1^m = \sum_{j\in\mathcal{I}} \sum_{i\in\mathcal{I}} \sum_{k\in\mathcal{K}} x_{j,i,k}^m \cdot \left[\left(d_{i,k}^m + \hat{d}_{j,i,k}^m\right) p_{i,k}^m + \mu_{j,i,k}^m\right] \\ & \text{s.t.} \quad (5), (7), (8), \ 0 \leqslant x_{j,i,k}^m \leqslant 1 \ \ \forall j, i, k, \end{aligned} \tag{14}$$

$$\begin{aligned} \mathcal{P}2 \quad & \min_{\boldsymbol{y}} \ \mathcal{O}2 = -\sum_{j\in\mathcal{I}} \sum_{i\in\mathcal{I}} \sum_{k\in\mathcal{K}} \left(\sum_{m=1}^{M} \mu_{j,i,k}^m\right) \cdot y_{j,k} \\ & \text{s.t.} \quad (4), \ 0 \leqslant y_{j,k} \leqslant 1, \ \ \forall k, j, \end{aligned} \tag{15}$$

respectively, where $\boldsymbol{x}^m = \{x_{j,i,k}^m, \ \forall j, \ i \in \mathcal{I}, \ k \in \mathcal{K}\}$ is the content scheduling decision for the mth TS and $\boldsymbol{y} = \{y_{i,k}^m, \forall i \in \mathcal{I}, k \in \mathcal{K}\}$ is the content caching decision for the reference TI. As shown in Figure 7, take the sth TI as the reference TI, at the beginning of the sth TI, $\boldsymbol{y}$ is obtained by solving $\mathcal{P}2$ and used for content object caching at the sth TI, i.e., the following M TSs. Rewrite the caching decision with respective to each TS as $\{y_{j,k}\} = \{y_{j,k,m}, \forall j \in \mathcal{I}, k \in \mathcal{K}, m \in \{1, 2, \ldots, M\}\}$, then $\{y_{i,k,1}\} = \cdots \{y_{i,k,m}\} \cdots = \{y_{i,k,M}\}$. While the scheduling decision for the mth TS of the reference TI, $\boldsymbol{x}^m$, is obtained by solving $\mathcal{P}1^m$ at the beginning of the mth TS.

Next, we first explore the optimal solutions of $\mathcal{P}1^m$ and $\mathcal{P}2$ without binary variable regression which can be used as a benchmark, and then provide their binary solutions.

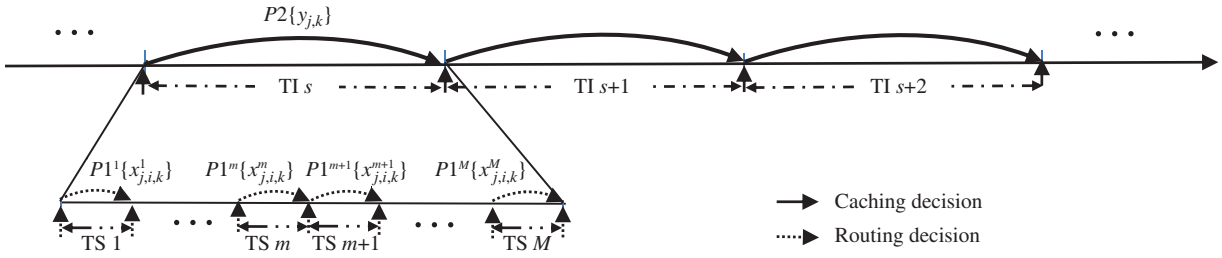

Figure 7 Different time scales for making caching and scheduling decisions.

5.3 Optimal solutions for sub-problems

The optimization problem $\mathcal{P}2$ becomes a typical linear planning problem and the optimal solution can be obtained by applying one of the efficient optimization algorithms, such as the simplex method in [24].

The optimization problem $\mathcal{P}1^m$ is a convex optimization problem.

Proof. It can be seen in $\mathcal{P}1^m$ that with relaxed $\boldsymbol{x}^m$ and fixed $\boldsymbol{\mu}^m = \{\mu^m_{j,i,k}\}$, the objective function is the sum of a set of quadric functions in terms of $x^m_{j,i,k}$. The proof is similar with (12), except that each quadric function of $\mathcal{P}1^m$ contains a product term $x^m_{j,i,k} \cdot \mu^m_{j,i,k}$, which is a linear factor and has no effect on the convexity of the function. Therefore, with convex objective function and linear constraints (convex functions), problem $\mathcal{P}1^m$ is a convex optimization problem and the optimal solution can be obtained by solving its dual problem.

Let $\boldsymbol{\alpha}^m = \{\alpha^m_k, \forall k \in \mathcal{K}\}$, $\boldsymbol{\beta}^m = \left\{\beta^m_j, \forall j \in \mathcal{I}\right\}$ and $\boldsymbol{\chi}^m = \{\chi^m_i, \forall i \in \mathcal{I}\}$ be the dual multipliers associated with constraints (5), (7) and (8), respectively. The dual function of $\mathcal{P}1^m$ is expressed as

$$\begin{aligned}\mathcal{D}^m = &\sum_{j\in\mathcal{I}}\sum_{i\in\mathcal{I}}\sum_{k\in\mathcal{K}} x^m_{j,i,k} \cdot \left[\left(d^m_{i,k} + \hat{d}^m_{j,i,k}\right) p^m_{i,k} + \mu^m_{j,i,k}\right] + \sum_{k\in\mathcal{K}} \alpha^m_k \left(\sum_{j\in\mathcal{I}}\sum_{i\in\mathcal{I}} x^m_{j,i,k} - 1\right) \\ &+ \sum_{j\in\mathcal{I}} \beta^m_j \left(\hat{N}_j - \hat{N}^{\text{rec}}_{j,m} - \sum_{i\in\mathcal{I}}\sum_{k\in\mathcal{K}} x^m_{j,i,k}\right) + \sum_{i\in\mathcal{I}} \chi^m_i \left(N_i - N^{\text{rec}}_{i,m} - \sum_{j\in\mathcal{I}}\sum_{k\in\mathcal{K}} x^m_{j,i,k}\right). \end{aligned} \tag{16}$$

Therefore, the optimal solution to the relaxed problem $\mathcal{P}1^m$ is given by

$$\max_{(\boldsymbol{\alpha}^m, \boldsymbol{\beta}^m, \boldsymbol{\chi}^m) > 0} \min_{0 \leqslant \boldsymbol{x}^m \leqslant 1} \mathcal{D}^m \left(\boldsymbol{x}^m, \boldsymbol{\alpha}^m, \boldsymbol{\beta}^m, \boldsymbol{\chi}^m\right). \tag{17}$$

Then, the relaxed problem can be solved by one of the efficient optimization algorithms, such as the interior-point method in [24].

5.4 Binary solutions based on hastening branch and bound (B&B)

To efficiently find the binary solutions of sub-problems $\mathcal{P}1^m$ and $\mathcal{P}2$, we further propose the JCWBS algorithm based on hastening B&B, the principle behind which is to perform a systematic enumeration of candidate solutions by means of state space search. In the following, problem $\mathcal{P}1^m$ is taken as an example to explain the main procedure of JCWBS.

Constructing the search space which is represented by a tree structure with the problem $\mathcal{P}1^m$ as its root problem $\mathcal{P}1^{m,(0)}$. Denote the optimal variables and the optimal solution of $\mathcal{P}1^{m,(0)}$ as $\boldsymbol{x}^{m*,(0)}$ and $\mathcal{O}1^{m*,(0)}$, respectively. Then, if all elements $x^{m*,(0)}_{j,i,k} \in \boldsymbol{x}^{m*,(0)}$ are binary, the search process is terminated and the optimal solution of $\mathcal{P}1^m$ is found. If not, the root problem on the first non-integer $x^{m*,(0)}_{j',i',k'}$, named as the branching variable, will be split into two sub-problems $\mathcal{P}1^{m,(0)}_{(1)}$ and $\mathcal{P}1^{m,(0)}_{(2)}$ by adding upper bound and lower bound constraints which can be generally expressed as

$$\begin{aligned} \mathcal{P}1^{m,(0)}_{(1)} \quad & \min \ \ \mathcal{O}1^m \left(\boldsymbol{x}^m\right) \\ & \text{s.t.} \ \ (5),\ (7),\ (8),\ x^m_{j',i',k'} = 0, \\ & \qquad x^m_{j,i,k} \geqslant 0, \ \ \forall \left(j,i,k\right) \setminus \left(j',i',k'\right), \end{aligned} \tag{18}$$

Algorithm 1 JCWBR

Require: $\delta \to 0^+$, $m \Leftarrow 1$, *index*$\Leftarrow$1;
input: M;
while *index* **do**
  if $m > M$ **then**
    index$\Leftarrow$0
  end if
  if $\tau = 0$ **then**
    $\left\{\mu^{\tau}_{j,i,k}\right\}_{|\mathcal{I}'|\times|\mathcal{I}|\times|\mathcal{K}|} \Leftarrow \mathbf{I}$
  end if
  $\hat{\mathcal{O}} \Leftarrow \mathcal{O}1$, $\hat{\mathcal{P}} \Leftarrow \mathcal{P}1^m(\mathcal{O}1^m)$, *flag* $\Leftarrow 1$
  while *flag* **do**
    $\mathcal{S} \Leftarrow \hat{\mathcal{P}}^{(0)}$ and call Algorithm 2
    $\boldsymbol{x}^{m,\tau} \Leftarrow \boldsymbol{z}^*$
    if $m == M$ **then**
    $\hat{\mathcal{O}} \Leftarrow \mathcal{O}2$, $\hat{\mathcal{P}} \Leftarrow \mathcal{P}2(\mathcal{O}2)$, *flag* $\Leftarrow 1$
    $\mathcal{S} \Leftarrow \hat{\mathcal{P}}^{(1)}$ and call Algorithm 2
    $\boldsymbol{y}^{m,\tau} \Leftarrow \boldsymbol{z}^*$
    end if
    $\mu^{m,\tau+1}_{j,i,k} \Leftarrow \left[\mu^{m,\tau}_{j,i,k} + \varepsilon^{\tau}\left(y^{m,\tau}_{j,k} - x^{m,\tau}_{j,i,k}\right)\right]^+$
    $\tau \Leftarrow \tau + 1$
    if $|\mathcal{O}^{\tau} - \mathcal{O}^{\tau-1}| \leqslant \delta$ **then**
      $\{\boldsymbol{x}^{\boldsymbol{m}*}, \boldsymbol{y}^{\boldsymbol{m}*}\} \Leftarrow \{\boldsymbol{x}^{m,\tau}, \boldsymbol{y}^{m,\tau}\}$
      $m \Leftarrow m + 1$, *flag* $\Leftarrow$0
    end if
  end while
end while

Algorithm 2 Hastening B&B

Require: $\hat{\mathcal{O}}^{(\mathrm{up})} \Leftarrow \infty$, $\boldsymbol{z}^* \Leftarrow \emptyset$;
input: $\mathcal{S}$;
while $\mathcal{S} \neq$null **do**
  pop $\hat{\mathcal{P}}^{(j)}$ from $\mathcal{S}$
  solve $\hat{\mathcal{P}}^{(j)}$ and obtain its optimal variables $\boldsymbol{z}^{(j)}$ and $\hat{\mathcal{O}}^{(j)}$
  if $\hat{\mathcal{O}}^{*(j)} < \hat{\mathcal{O}}^{(\mathrm{up})}$ **then**
    if $\boldsymbol{z}^{(j)}$ are all integers **then**
      $\hat{\mathcal{O}}^{(\mathrm{up})} \Leftarrow \hat{\mathcal{O}}^{(j)}$
      $\boldsymbol{z}^* \Leftarrow \boldsymbol{z}^{(j)}$
      $\mathcal{S} \Leftarrow \mathcal{S}\backslash\{\hat{\mathcal{P}}^{(j)}\}$
    else
      split $\hat{\mathcal{P}}^{(j)}$ into $\hat{\mathcal{P}}^{(j)}_{(1)}$ and $\hat{\mathcal{P}}^{(j)}_{(2)}$
      if *hastening condition* is met
        delete the corresponding sub-problem
        push another into $\mathcal{S}$
      else
        $\mathcal{S} \Leftarrow \mathcal{S} \cup \{\hat{\mathcal{P}}^{(j)}_{(1)}\}$
        $\mathcal{S} \Leftarrow \mathcal{S} \cup \{\hat{\mathcal{P}}^{(j)}_{(2)}\}$
      end if
    end if
  else
    $\mathcal{S} \Leftarrow \mathcal{S}\backslash\{\hat{\mathcal{P}}^{*(j)}\}$
  end if
end while

$$\begin{aligned} \mathcal{P}1^{m,(0)}_{(2)} \quad & \min \quad \mathcal{O}1^m(\boldsymbol{x}^m) \\ & \text{s.t.} \quad (5),\ (7),\ (8),\ x^m_{j',i',k'} = 1, \\ & \qquad x^m_{j,i,k} \geqslant 0,\ \ \forall (j,i,k) \setminus (j',i',k'). \end{aligned} \tag{19}$$

To find the optimal binary solution, the depth-first search can be adopted, which starts at the root and explores as far as possible along each branch before backtracking the non-visited nodes recorded by a last-in-first-out stack. Therefore, we first go down to sub-problem $\mathcal{P}^{m,(0)}_{(1)}$. The optimal solution and the optimal value of its objective function can be denoted as $\boldsymbol{x}^{m*,(1)}$ and $\mathcal{O}1^{m*,(1)}$, respectively. Again if any value of $\boldsymbol{x}^{m*,(1)}$ is not binary, the problem will be further split into two more sub-problems.

While splitting or branching the problem at any level tree node, the branch can be pruned with the *hastening condition* before solving the problem, which is expressed as the violation of the constraint $\sum_{j\in\mathcal{I}, i\in\mathcal{I}} x^m_{j,i,k} = 1$, i.e., MU k is not served.

The proposed JCWBS for finding binary solution of $\mathcal{P}$, is summarized in Algorithm 1. Note that the caching variables only need to be determined at integral multiple of M TSs, thus the computation complexity of proposed algorithm is only $1/M$ of the algorithm where content object caching and scheduling decisions are determined at the same time scale.

6 Performance evaluation and analysis

In this section, on the basis of real-world dataset used in Section 3, a statistical simulation framework is developed using MATLAB to evaluate the performance of the proposed algorithm, in terms of the reductions of transmission delay and the outgoing backhaul traffic.

6.1 Evaluation setup and comparing algorithms

Following SCN simulation setups in [25], $|\mathcal{I}|$ SBSs with coverage radius R are randomly and uniformly dropped in S measured in m^2, where $|\mathcal{I}|\pi R^2/S > 2.5$ to assure ultra-dense SBS distribution. Half of

MUs are uniformly placed in random independent positions within the circle of radius d around the SBSs, and d is small enough such that the MUs with the corresponding distances would always be associated with their central SBS. The rest of MUs are uniformly distributed outside all cluster circles. All SBSs are connected to their closest aggregation gateway (AG), and then connected to the CDN via the CN and Internet by which the topology of the backhaul network is constructed. SBS j connects SBS i with probability $\rho_{j,i} = \zeta \mathrm{e}^{-d_{j,i}/(d_{\max}-d_{j,i})}$ [26], where ζ is the parameter of the connection model, $d_{j,i}$ is the Euclidean distance between SBS j and SBS i and $d_{\max} = 50$ m is the maximum distance between any two SBSs. The parameter ζ is adjusted between 0.4 and 0.8 to adapt to different simulation settings. The total bandwidth of SBS i, $c_i = 100$ Mbps. The average transmission delay between directly connected SBSs, AGs, and SBSs and AGs are distributed uniformly [1/100 1/40] μs/bit.

The performance of JCWBS is compared with that of the joint caching and wireless scheduling (JCWS) algorithm, the joint caching and backhaul scheduling (JCBS) algorithm and the upper bound (UB).

JCWS: With randomly selected cooperative cache or CDN strategy on the backhaul side, the JCWS only considers the joint optimization of caching and wireless scheduling by exploring re-association of MUs and SBSs, e.g., the content-aware user clustering and content caching algorithm in [14].

JCBS: With initially associated SBSs strategy on the wireless side, JCBS only considers the joint optimization of caching and backhaul scheduling by bringing the requested content object over more light traffic-load backhaul links, e.g., the cooperative cache-router caching algorithm in [22].

UB: UB provides the upper bound of the solution of problem (11) which is obtained by solving the relaxed problem of (11) without binary variable regression. As the UB is not feasible, it is only used as a benchmark for measuring the efficiency of the proposed algorithm.

6.2 Performance analysis

(1) Transmission delay reduction: Denote the average transmission delays (ATDs) obtained in UB, JCWBS, JCBS and JCWS as d_{UB}, d_{JCWBS}, d_{JCBS} and d_{JCWS}, respectively. Further, it is assumed that the ATD of the selfish algorithm without any cache cooperation is d_{sel}. The relative ATD reduction ratio of UB, JCWBS, JCBS and JCWS over the selfish algorithm are defined as $(d_{\mathrm{sel}} - d_{\mathrm{UB}})/d_{\mathrm{sel}}$, $(d_{\mathrm{sel}} - d_{\mathrm{JCWBS}})/d_{\mathrm{sel}}$, $(d_{\mathrm{sel}} - d_{\mathrm{JCBS}})/d_{\mathrm{sel}}$ and $(d_{\mathrm{sel}} - d_{\mathrm{JCWS}})/d_{\mathrm{sel}}$, respectively. Figure 8 illustrates cache capacity versus the ATD reduction ratio. The results with different connection probability among SBSs are also provided, by setting $\zeta = 0.4, 0.6, 0.8$, and denoted as low probability $\rho = \rho_{\mathrm{L}}$, medium probability $\rho = \rho_{\mathrm{M}}$ and high probability $\rho = \rho_{\mathrm{H}}$, respectively. The cache capacity of each SBS is varied from 5% to 85% of the entire unique content object size that requested in its coverage, denoted as $\Omega = \{5\%, 25\%, 45\%, 65\%, 85\%\}$. As expected, increasing the cache capacity and the connection probability reduce the ATD for all algorithms as more content requests can be satisfied locally or by cooperation among caches. In addition to UB, JCWBS results in the largest ATD reduction ratio compared to the rest algorithms where the maximum difference is up to 42% when $\Omega = 85\%$ and $\rho = \rho_{\mathrm{L}}$. Meanwhile, the performance of JCWBS is very close to that of UB (less than 5% worse).

In Figure 9, the cache capacity of each SBS is fixed with $\Omega = 65\%$, it can be observed that the ATD reduction of all algorithms increase as the number of SBSs increases. For example, the ATD reduction ratio of JCWBS increases to 65% with 150 SBSs from 50% with 30 SBSs, when $\rho = \rho_{\mathrm{M}}$. Moreover, for 150 SBSs and $\rho = \rho_{\mathrm{H}}$, the ATD reduction ratio of JCWBS achieves about 70%, which is 15% and 30% more than that of JCBS and JCWS, respectively. This is not only because that increasing the number of SBSs increases the total cache capacity of the system, but also due to the fact that more SBSs brings more degree of freedom to implement scheduling.

(2) Outgoing backhaul traffic reduction: Next we evaluate the performance of JCWBS in terms of backhaul traffic reduction ratio, the definition of which is expressed as $(\varpi_{\mathrm{sel}} - \varpi_{\mathrm{JCWBS}})/\varpi_{\mathrm{sel}}$, where ϖ_{sel} and ϖ_{JCWBS} are the generated backhaul traffic owing to cache misses or link congestions of the selfish algorithm and the JCWBS, respectively. Figure 10 depicts the backhaul traffic reduction ratio versus the SBSs' connection probability and the number of MUs, when $\Omega = 65\%$ and $|\mathcal{I}| = 120$. It is shown that the backhaul traffic reduction ratio decreases as the number of MUs increases, and increases

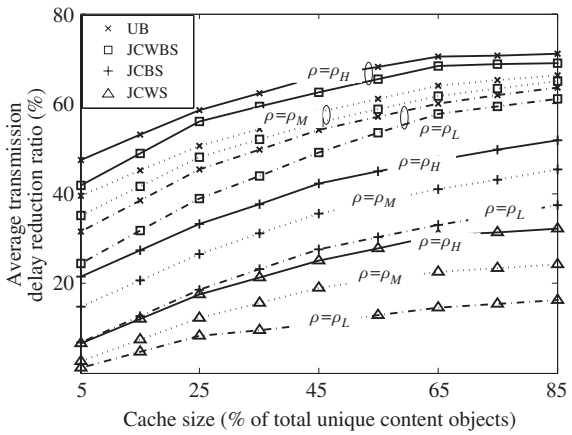

Figure 8 Delay reduction versus cache capacity.

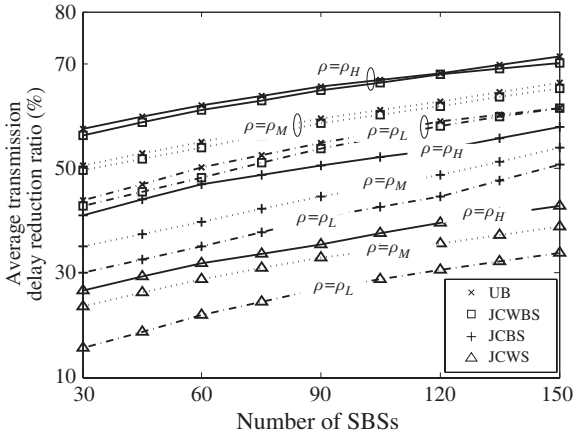

Figure 9 Delay reduction versus number of SBSs.

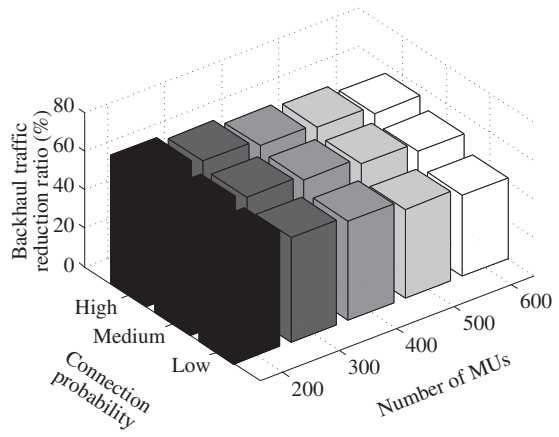

Figure 10 Traffic reduction versus number of MUs and connection probability.

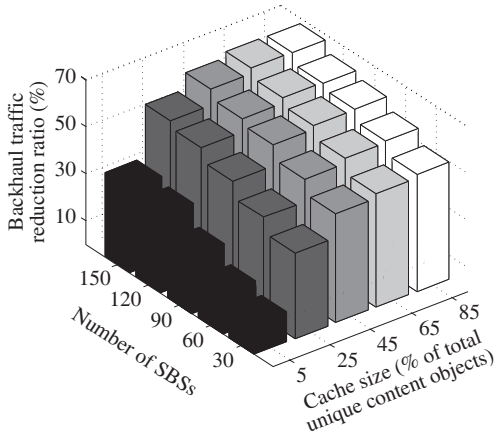

Figure 11 Traffic reduction versus cache capacity and number of SBSs.

as the connection probability increases. With $\rho = \rho_{\mathrm{H}}$ and 200 MUs, the backhaul traffic reduction ratio is about 60%. This is because that the high connection probability among SBSs improves the opportunity of cooperation, and in turn improves the cache hit ratio and thus reduces the outgoing backhaul traffic. Seen from Figure 11, where $\rho = \rho_{\mathrm{M}}$ and the number of MUs equals to 500, the outgoing backhaul traffic reduction ratio also increases as either the number of SBSs or the cache capacity increases.

7 Conclusion and future work

Corresponding to three motivation problems in Section 1, the following conclusions can be drawn. (1) By utilizing the real-world dataset, the feasibility of cooperative caching is proved. (2) Further, it is shown that the problems of content object caching and scheduling should be resolved at different time scales, for being triggered by different events with different time scales. (3) Based on the findings, the joint content object caching and scheduling problem is constructed, and the JCWBS is proposed based on Lagrangian relaxation and decomposition, and hastening B&B. The performance of the proposed algorithm is evaluated, which shows that the proposed algorithm has the advantage on the objective performance by exploiting the degrees of freedom when selecting wireless and backhaul links. The proposed JCWBS is a centralized algorithm which needs data convergence, a distributed algorithm will be studied in our future work.

Acknowledgements This work was supported by National Nature Science Foundation of China (Grant No. 61302108) and National Science and Technology Major Project (Grant No. 2015ZX03003004).

Conflict of interest The authors declare that they have no conflict of interest.

References

1 Ericsson. On the pulse of networked society. http://www.ericsson.com/mobility-report. 2016
2 Andrews J G, Claussen H, Dohler M, et al. Femtocells: past, present and future. IEEE J Sel Area Commun, 2012, 30: 497–508
3 Altobridge debuts Intel-based network edge small cells caching solution. http://www.mobileeurope.co.uk/press-wire/altobridge-debuts-intel-based-hierarchical-network-edge-caching-solution. 2013
4 Saguna Networks. Saguna open-RAN. http://www.saguna.net/products/saguna-cods-open-ran. 2015
5 Imbrenda C, Muscariello L, Rossi D. Analyzing cacheable traffic in ISP access networks for micro CDN applications via content-centric networking. In: Proceedings of the 1st ACM Conference on Information-Centric Networking, Paris, 2014. 57–66
6 Saroiu S, Gummadi K P, Dunn R J, et al. An analysis of Internet content delivery systems. In: Proceedings of the 5th Symposium on Operating Systems Design and Implementation. New York: ACM, 2002. 36: 315–327
7 Pathan A, Buyya R. A taxonomy and survey of content delivery networks. Technical Report, GRIDS-TR-2007-4. Grid Computing and Distributed Systems Laboratory, The University of Melbourne. 2007
8 Zhang G Q, Tang M D, Cheng S Q, et al. P2P traffic optimization. Sci China Inf Sci, 2012, 55: 1475–1492
9 Erman J, Gerber A, Hajiaghayi M, et al. To cache or not to cache: the 3G case. IEEE Internet Comput, 2011, 15: 27–34
10 Woo S, Jeong E, Park S, et al. Comparison of caching strategies in modern cellular backhaul networks. In: Proceedings of ACM International Conference on Mobile Systems, Applications, and Services, Taipei, 2013. 319–332
11 Li H J, Yang C, Huang X Q, et al. Cooperative RAN caching based on local altruistic game for single and joint transmissions. IEEE Commun Lett, 2016, doi: 10.1109/LCOMM.2016.2635637
12 Ahlehagh H, Dey S. Video caching in radio access network: impact on delay and capacity. In: Proceedings of IEEE Wireless Communications and Networking Conference, Shanghai, 2012. 2276–2281
13 Li H J, Hu D, Ci S. iCacheOS: In-RAN caches orchestration strategy through content joint wireless and backhaul routing in small-cell networks. In: Proceedings of IEEE Global Communications Conference, San Diego, 2015. 1–7
14 Huang X Q, Ansari N. Content caching and distribution in smart grid enabled wireless networks. IEEE Internet Things J, 2016, doi: 10.1109/JIOT.2016.2577701
15 Huang X Q, Ansari N. Content caching and user scheduling in heterogeneous wireless networks. In: Proceedings of IEEE Global Communications Conference, Washington DC, 2016
16 Dehghan M, Seetharam A, Jiang B, et al. On the complexity of optimal routing and content caching in heterogeneous networks. IEEE Comput Commun, 2015, 75: 11–15
17 Arvidsson A, Mihly A, Westberg L. Optimised local caching in cellular mobile networks. Computer Netw: Int J Comput Telecommun Netw, 2011, 55: 4101–4111
18 Wei Q, Choi C, Biermann T, et al. Optical mobile network. NTT DOCOMO Tech J, 2012, 14: 43–53
19 3GPP. Evolved universal terrestrial radio access (E-UTRA); radio resource control (RRC) protocol specification. TR 36.331. http://www.3gpp.org/DynaReport/36331.htm. 2014
20 Li H J, Xu X D, Hu D, et al. Clustering strategy based on graph method and power control for frequency resource management in femtocell and macrocell overlaid system. IEEE J Commun Netw, 2011, 13: 664–677
21 Lai K, Baker M. Measuring link bandwidths using a deterministic model of packet delay. ACM SIGCOMM Comput Commun Rev, 2010, 30: 283–294
22 Liu R, Yin H, Cai X J, et al. Cooperative caching scheme for content oriented networking. IEEE Commun Lett, 2013, 17: 781–784
23 Sahni S, Gonzalez T. P-complete approximation problems. J ACM, 1976, 23: 555–565
24 Bertsekas D. Convex Optimization Theory. Belmont: Athena Scientific, 2009. 347–364
25 Li H J, Wang Z J, Hu D. Joint wireless and backhaul load balancing in cooperative caches enabled small-cell networks. In: Proceedings of IEEE International Symposium on Personal, Indoor and Mobile Radio Communications, Hong Kong, 2015. 1889–1894
26 Zegura E W, Calvert K L, Bhattacharjee S. How to model an internetwork. In: Proceedings of International Conference on Computer Communications, San Francisco, 1996. 594–602