IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received November 7, 2017, accepted February 3, 2018, date of publication February 12, 2018, date of current version March 16, 2018.

Digital Object Identifier 10.1109/ACCESS.2018.2804902

Joint Resource Allocation and Content Caching in Virtualized Content-Centric Wireless Networks

THINH DUY TRAN AND LONG BAO LE, (Senior Member, IEEE)
Institut National de la Recherche Scientifique, Université du Québec, Montréal, QC H5A 1K6, Canada
Corresponding author: Long Bao Le (long.le@emt.inrs.ca)

ABSTRACT Efficient content caching plays a crucial role in quality of service enhancement and congestion mitigation of the backhaul and core networks for the fifth-generation (5G) wireless network, which must support a large amount of multimedia and video content. Wireless network virtualization provides a novel paradigm shift in 5G system design which enables to better utilize network resources, rapid development of new services, and reduce the operation cost. Harmonized deployment of a content caching strategy in the virtualized wireless network environment, however, requires a suitable radio resource allocation framework to realize the great benefits of these technologies. In this paper, we study the joint resource allocation and content caching problem which aims to efficiently utilize the radio and content storage resources in the highly congested backhaul scenario. In this design, we minimize the maximum content request rejection rate experienced by users of different mobile virtual network operators in different cells, which results in a mixed-integer non-linear program. We solve this difficult optimization problem by proposing a bisection-search-based algorithm that iteratively optimizes the resource allocation and content caching placement. We further propose a low-complexity heuristic algorithm which achieves moderate performance loss compared to the bisection-search based algorithm. Extensive numerical results confirm the efficacy of our proposed framework which significantly reduces the maximum request outage probability compared with other benchmark algorithms.

INDEX TERMS Wireless network virtualization, content caching, resource allocation, resource slicing, network slicing, OFDMA, multi-cell network, content delivery network, backhaul.

I. INTRODUCTION

Wireless network virtualization (WNV) has been recognized as an essential technology for the 5G wireless network [1], where WNV allows multiple mobile virtual network operators (MVNO) to share the same network infrastructure owned and managed by an infrastructure provider (InP). Moreover, the InP must flexibly allocate network resources such as transmission power, bandwidth, and storage to MVNOs based on their demands and mutual contracts in an efficient manner so that their operations and services can be harmonized on the same infrastructure. The MVNOs can then utilize the resources rent from the InP to provide mobile services to their user equipment (UE) with committed quality of service (QoS). The WNV can potentially help reduce the operation cost (OPEX) and capital expenditure cost (CAPEX) significantly while efficiently utilizing network resources, and guaranteeing the QoS. On the other hand, there has been a cloudification trend in engineering future wireless cellular systems with adaptive function splits where certain network functions and algorithms are deployed in the cloud, which is connected with base stations (also called remote radio heads (RRHs) in the literature) through the backhaul network[1] [2], [3]. When sophisticated network and communication functions such as baseband signal processing and signal detection are performed in the cloud (technically in the baseband units (BBUs)), very large backhaul bandwidth is required to meet the strict latency constraint of the I/Q data exchanges between the BSs and the cloud.

Furthermore, the future wireless network must cope with the explosion of the mobile traffic which has growth rate about 131% per year [3] where the mobile video traffic will account for about 75% of the overall mobile traffic by 2020 [4]. This huge traffic demand will put great pressure on not only the wireless access network but also the backhaul network connecting the BSs and the core network (CN). Therefore, fundamental improvement of the efficiency of radio resource utilization and mitigation of backhaul congestion become very critical research issues, especially in the

[1]This is called a fronthaul network some time in the literature.

virtualized network environment. Toward this end, development of a joint efficient content caching and virtualized resource allocation framework for the 5G wireless network is required to resolve the access and backhaul network congestion while enhancing users' QoS [5]–[7].

Content caching at the network edge has been shown to lead to significant improvement in users' QoS [6], [7]. In particular, by deploying content storage at BSs and prefetching popular contents to these storage facilities, we can reduce access latency and mitigate traffic congestion on the backhaul links during high-traffic hours, thus improving network performance and users' QoS [8]. However, in the virtualized wireless environment where multiple MVNOs operate on the shared infrastructure with limited storage capacity, network performance improvement due to content caching could be less significant if the InP simply partitions the available storage capacity and allocates these storage partitions to MVNOs. Therefore, efficient and shareable content caching among MVNOs jointly with radio resource allocation can enable to boost the network performance.

A. RELATED WORK

Most existing works in the literature treat the WNV, content caching, resource allocation design issues separately. In particular, Poularakis *et al.* [10] focus on improving the caching performance, i.e., increasing the hit rate and reducing access delay, for small-cell wireless networks. Khreishah *et al.* [6], [7] only consider joint content caching with conventional resource allocation in wireless networks without WNV. There are a few works such as [11] and [12] studying the joint caching, resource allocation, and WNV. However, adaptation of cache placement based on the signal-to-noise ratio (SNR) may not be cost efficient, since caching decisions at BS should be made over a long time scale while the SNR typically varies rapidly.

Recently, different content caching frameworks have been introduced to leverage the evolution of network architecture. In [13] and its related work, the authors propose to install storage repository at femtocells, which are deployed in high density and closer to UEs, to assist the macro BS through offloading content requests. Another approach called hierarchical caching is to leverage the hierarchical structure of modern network topology and coding theory for content caching as in [14]. To adapt to C-RAN based network architecture, Tang *et al.* [16] propose to install the storage repository not only at the RRH but also in the cloud, which can be considered as another version of hierarchical caching.

Most of these existing works do not consider the highly congested network scenario due to the lack of radio resource and bandwidth in the wireless access and backhaul links [17]. In fact, the ultra-reliable and critical machine-type communications (cMTC) and massive machine-type communications (mMTC) require a large number of reliable connections [9]. Moreover, a great deal of control signaling data, which would consume valuable radio resources in the wireless access links and result in further network congestion.

For wireless backhaul networks (WBN) [18]–[20], wireless backhaul links are used in lieu of the traditional cable links for connection with the CN. Wireless networks with wireless backhaul may suffer from performance degradation if the radio resources allocated for backhaul links are not sufficient and/or heavy contents such as large video files are transferred over these backhaul links from the CN. Another innovative content caching approach which enables to significantly reduce the backhaul traffic is to leverage device caching and device-to-device (D2D) communication [15], which caches content on mobile device's storage. This approach unfortunately is hindered by the mobility nature of mobile devices and their economical selfishness in providing content caching, which could consume their limited battery and storage capacity.

B. RESEARCH CONTRIBUTIONS

Motivated by the aforementioned issues, we consider the content caching design for wireless networks with highly congested backhaul links. Specifically, we study the joint radio resource allocation and content caching design for the virtualized wireless network where we make the following key contributions.

- We present the problem formulation that minimizes the maximum request outage probability for all MVNOs at different BSs while avoiding content caching redundancy at the storage locations. This design captures dynamic content request arrivals/departures and possible cache misses when a requested content is not cached or radio resources are not sufficient to support the user-BS communications. Our framework considers the network scenario in which the backhaul links are highly congested and wireless access bandwidth is limited. Specifically, the design objective enables to maximize the content accesses from the BSs, which mitigates the long service latency due to content transfer over congested backhaul links. Moreover, the similarity on the content set and content preferences among different MVNOs is exploited where different MVNOs are allowed to share the same cached contents at individual BSs.
- To solve the underlying mixed integer non-linear problem (MINLP), we propose a novel iterative algorithm based on the bisection-search method. In particular, the proposed algorithm exploits special properties of the Erlang-B function and the optimal channel allocation. The algorithm is proved to converge to a local optimal solution. Furthermore, we propose a caching decision rounding solution and a low-complexity algorithm with more affordable computation burden.
- Extensive numerical results are presented to demonstrate the efficacy of our proposed bisection search based algorithm, the caching decision rounding technique and the heuristic algorithm. Specifically, we study different scenarios where the content popularity patterns of different MVNOs are in same and different orders

at each BS. It is confirmed through numerical studies that our proposed design performs well in both scenarios.

The preliminary results of this work were presented at IEEE GLOBECOM 2017 [21]. This journal manuscript, however, makes several major extensions in comparison to this conference version. First, the objective function in the conference version is the request rejection rate while the objective function in this current manuscript is the request outage probability. Second, this journal version considers two important cases regarding the file popularity, namely the same-order and different-order file popularity cases for different MVNOs at each BS in our design. This was not investigated in the conference version. Third, the rounding procedure for the caching decision variables developed in the current manuscript is much more sophisticated than that in the conference version. Finally, compared with the conference version, much more extensive performance evaluation and comparisons of the proposed algorithms with baseline algorithms are presented in this journal version.

TABLE 1. Summary of key notations.

Notation	Description
K	Number of base stations
M	Number of MVNOs
MVNO (m,k)	MVNO m associated with BS k
$W^{\max}$	Number of wireless channels
w_{km}	Number of channels allocated to MVNO (m,k)
$\boldsymbol{w}$	Channel allocation vector
λ_{km}	Average number of arrival request from MVNO (m,k)
C_k	Storage capacity of BS k
F	Number of contents/files
q_{kmf}	Request probability of file f by MVNO (m,k)
$\mathcal{Q}_{km}$	Request probability distribution for MVNO (m,k)
x_{kmf}	Caching decision variable for file f by MVNO (m,k)
$\boldsymbol{x}_{km}$	Caching decision vector for MVNO (m,k)
$\boldsymbol{x}$	Caching decision vector
$h_{kmf}(\boldsymbol{x})$	Cache-hit rate for file f by MVNO (m,k)
$h_{km}(\boldsymbol{x})$	Total cahe-hit rate by MVNO (m,k)
$\bar{h}_{kmf}(\boldsymbol{x})$	Cache-missed rate for file f by MVNO (m,k)
$\bar{h}_{km}(\boldsymbol{x})$	Total cahe-missed rate by MVNO (m,k)
T_{km}	Service time (s) for BS k to serve a cache-hit file request from MVNO m
$P_{km}(\boldsymbol{x},\boldsymbol{w})$	Probability that there are w_{km} ongoing cache-hit file requests from MVNO (m,k)
$\mu_{km}(\boldsymbol{x},\boldsymbol{w})$	Rejection rate for the cache-hit request from MVNO (m,k)
$\Phi_{km}(\boldsymbol{x},\boldsymbol{w})$	Total file request outage probability of MVNO (m,k)
$Z(f,\gamma)$	Zipf distribution of f-th most popular file with skewness γ

The rest of our paper is as follows. We describe the system model and the problem formulation in Sections II and III, respectively. We then present the algorithms to solve the considered problem in Section IV. Section V shows the numerical results and Section VI concludes our paper. The summary of key notations is presented in Table 1.

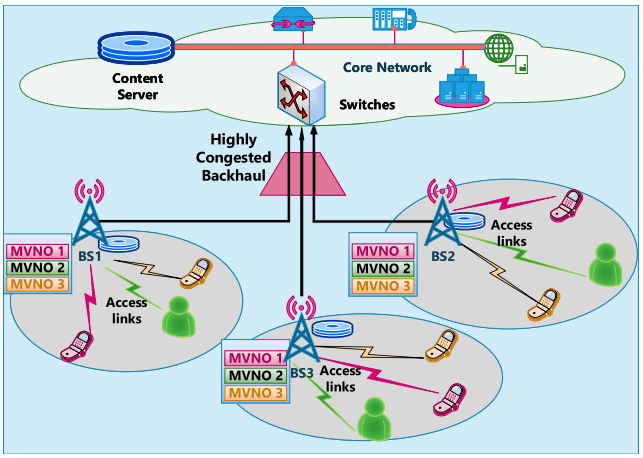

FIGURE 1. System model.

II. SYSTEM MODEL

We consider a downlink virtualized OFDMA multi-cell wireless network with caching repository deployed at each base station (BS). The system consists of K BSs in a set denoted as $\mathcal{K} = \{1,\ldots,K\}$. These BSs are connected to the core network (CN) via *highly congested* backhaul links. It is assumed that the network has $W^{\max}$ wireless channels of equal bandwidth serving all the UEs associated with these BSs. To avoid severe interference in the network, we assume these channels are allocated in the orthogonal manner.[2] This network infrastructure including all BSs, the backhaul and core networks, radio and storage resources are assumed to be owned and managed by an InP. For illustration, our system model is depicted in Figure 1.

In this network, the InP serves M MVNOs in the set $\mathcal{M} = \{1,\ldots,M\}$, which rent resources and infrastructure to serve their UEs. For convenience, we use MVNO (m,k) to denote MVNO m associated with BS k. For the channel allocation, we denote $\boldsymbol{w} = \{w_{11},\ldots,w_{km},\ldots,w_{KM}\}$ as the channel allocation vector, whose elements w_{km} represent the number of wireless channels allocated to MVNO (m,k).

It is assumed that UEs of each MVNO m are interested in accessing contents in a content set $\mathcal{F} = \{f_1,\ldots,f_F\}$ of F files or contents.[3] Note that it is plausible to assume a common file set F for all MVNOs as the common file set can be formed by aggregating all MVNOs' file sets. Content requests from UEs of MVNO m in the coverage of BS k are assumed to follow the Poisson process with an average rate λ_{km} (requests/s). Note that the Poisson process is the popular mathematical tool to model random arrival processes in practical telecommunication and computer systems.

Without loss of generality, we assume that each file in $\mathcal{F}$ has the normalized size of 1 unit and the BSs can cache popular contents in advance for future possible accesses [10], [11]. In practice, a large file, e.g., a movie file,

[2]These channels can represent frequency bands in OFDM systems or sub-channels as in LTE-based systems [22].

[3]We will use the terms file and content interchangeably in the following.

can be split into equal-size chunks of data whose size can then be normalized to 1. Let C_k denote the capacity of the storage repository installed at BS k, which can cache up to C_k files where $C_k \in \mathbb{Z}_+$. Moreover, $\mathcal{Q}_{km} = \{q_{km1}, \ldots, q_{kmF}\}$ denotes the content request probability distribution where q_{kmf} represents the probability that UEs of MVNO (m, k) requests file f. Therefore, we have

$$\sum_{\mathcal{F}} q_{kmf} = 1, \quad \forall k \in \mathcal{K}, \ \forall m \in \mathcal{M}. \tag{1}$$

Note also that UEs must communicate with their associated BSs to download the requested file. We assume that $\mathcal{Q}_{km}$ can be different for different MVNO-BS pairs (m, k) and this allows us to capture the spatial variations of content popularity patterns.

To model the content caching placement, we introduce the caching decision vectors $\boldsymbol{x}_{km} = \{x_{km1}, \ldots, x_{kmF}\}$ for BS k and MVNO m and $\boldsymbol{x} = \{\boldsymbol{x}_{11}, \ldots, \boldsymbol{x}_{km}, \ldots, \boldsymbol{x}_{KM}\}$ to denote the content caching decision vector for all MVNOs (m, k), where $x_{kmf} \in \{0, 1\}$ and $x_{kmf} = 1$ if file f is cached at BS k to serve requests from MVNO m, and $x_{kmf} = 0$, otherwise. Moreover, to ensure some minimum QoS requirement, we assume that one channel (if available) must be allocated to download a requested file from the associated BS for any UE.[4]

In general, a particular UE can download its requested file from the content server (CS) in the CN and such content must be transferred over both backhaul and wireless access networks if a particular file is not cached at the UE's associated BS. With the highly congested backhaul network, the end-to-end content download time from CN can be very large, which severely affects the user's QoS. Therefore, to maintain satisfactory users' QoS, we assume that any particular content request results in a cache hit only if the requested content is cached at the UE's associated BS and there are available channels to be support the UE-BS communications.

To elaborate the content access and transmission, let us consider a particular request from MVNO m to file $f \in \mathcal{F}$ at BS k. If file f is cached at this BS (i.e., cache-hit file request) and there is an available channel in the budget of w_{km} channels, the request of file f is accepted and the file is downloaded to the requesting UE. In contrast, a content request is rejected if all w_{km} channels are allocated for serving other file requests (even if the requested content is cached at the UE's associated BS). As discussed above, due to the highly congested backhaul links between the BSs and CN, we do not account for the case where these cache-missed file requests are redirected to the CS in the CN in our design and optimization. In the next sections, we present the problem formulation and our proposed algorithm.

III. PROBLEM FORMULATION

We now describe the joint content caching and channel allocation problem which aims to minimize the maximum file request outage probability over all MVNOs and BSs. Because different MVNOs could share the cached files at each BS, we do not cache the same file at the same BS's caching repository to serve requests from different MVNOs. Such avoidance of content caching redundancy can be mathematically expressed by the following constraints:

$$\sum_{i \in \mathcal{M}} x_{kif} \leq 1, \quad \forall k \in \mathcal{K}, \ \forall f \in \mathcal{F}. \tag{2}$$

Moreover, the finite storage capacity constraints at different BSs can be written as

$$\sum_{f \in \mathcal{F}} \sum_{i \in \mathcal{M}} x_{kif} \leq C_k, \quad \forall k \in \mathcal{K}. \tag{3}$$

We now study the file rejections due to lack of radio resources (i.e., there is no available channel) for a given caching solution $\boldsymbol{x}$. The request rate for file f from MVNO m at BS k if file f is cached at this BS k can be calculated as

$$h_{kmf}(\boldsymbol{x}) = \lambda_{km} q_{kmf} \left(\sum_{i \in \mathcal{M}} x_{kif} \right). \tag{4}$$

Hence, the total request rate from MVNO m for all files in $\mathcal{F}$, if they are cached at BS k, is

$$h_{km}(\boldsymbol{x}) = \sum_{f \in \mathcal{F}} \lambda_{km} q_{kmf} \left(\sum_{i \in \mathcal{M}} x_{kif} \right). \tag{5}$$

Otherwise, if file f is not cached at BS k (i.e., cache-missed file), the corresponding request rate from MVNO m to this file at this BS is equal to

$$\bar{h}_{kmf}(\boldsymbol{x}) = \lambda_{km} q_{kmf} \left(1 - \sum_{i \in \mathcal{M}} x_{kif} \right), \tag{6}$$

and the total cache-missed file request rate from MVNO m to all files in $\mathcal{F}$ at BS k is

$$\begin{aligned} \bar{h}_{km}(\boldsymbol{x}) &= \sum_{f \in \mathcal{F}} \lambda_{km} q_{kmf} \left(1 - \sum_{i \in \mathcal{M}} x_{kif} \right) \\ &= \lambda_{km} - h_{km}(\boldsymbol{x}). \end{aligned} \tag{7}$$

Note that all these involved arrival processes are Poisson processes because splitting or merging Poisson processes creates Poisson processes. Assume that it takes T_{km} (s) for BS k to serve a cache-hit file request from MVNO m. T_{km} represents the download time from the content cache to the UE of MVNO (m, k). With w_{km} channels allocated by the InP to MVNO (m, k) to serve requests of UEs, at most w_{km} file requests from MVNO m can be simultaneously served by its associated BS. The file requests from MVNO m at BS k can be modeled as an $M/D/w_{km}/w_{km}$ queue with Poisson arrivals, deterministic service time, w_{km} servers, and no waiting buffer [23].

Recall that all cache-missed file requests are rejected due to high delay for downloading content from the CN. Additionally, any cache-hit file request from MVNO m at BS k is only rejected if all w_{km} channels are used to service other

[4]This assumption can be relaxed where more than one channels can be required to support the content download.

ongoing w_{km} requests. From [23], the probability that there are w_{km} ongoing cache-hit file requests from MVNO m being served by BS k can be calculated as

$$P_{km}(\boldsymbol{x}, \boldsymbol{w}) = \frac{(h_{km}(\boldsymbol{x})T_{km})^{w_{km}}}{w_{km}!} \left(\sum_{i=0}^{w_{km}} \frac{(h_{km}(\boldsymbol{x})T_{km})^{i}}{i!} \right)^{-1}. \quad (8)$$

Consequently, the rejection rate for the cache-hit request from MVNO m at BS k due to channel unavailability can be expressed as

$$\mu_{km}(\boldsymbol{x}, \boldsymbol{w}) = h_{km}(\boldsymbol{x}) P_{km}(\boldsymbol{x}, \boldsymbol{w}). \quad (9)$$

From (7) and (9), the total file request outage probability from MVNO m at BS k can be calculated as

$$\Phi_{km}(\boldsymbol{x}, \boldsymbol{w}) = \frac{\mu_{km}(\boldsymbol{x}, \boldsymbol{w}) + \bar{h}_{km}(\boldsymbol{x})}{\lambda_{km}}. \quad (10)$$

To avoid poor QoS and unfair treatment in serving file requests from different MVNOs at different BSs, we consider the joint channel allocation and content caching optimization problem which minimizes the highest outage probability among MVNOs at all BSs while accounting for the file caching redundancy avoidance and other system constraints. This problem can be formulated as follows:

$$\min_{\boldsymbol{x}, \boldsymbol{w}} \max_{k \in \mathcal{K}, m \in \mathcal{M}} \Phi_{km}(\boldsymbol{x}, \boldsymbol{w}) \quad (11a)$$

$$\text{s.t.} \sum_{m \in \mathcal{M}} x_{kmf} \leq 1, \quad \forall k \in \mathcal{K}, \ \forall f \in \mathcal{F} \quad (11b)$$

$$\sum_{m \in \mathcal{M}} \sum_{f \in \mathcal{F}} x_{kmf} \leq C_k, \quad \forall k \in \mathcal{K} \quad (11c)$$

$$w_{km} \geq W_{km}^{\min}, \quad \forall k \in \mathcal{K}, \ \forall m \in \mathcal{M} \quad (11d)$$

$$\sum_{k \in \mathcal{K}} \sum_{m \in \mathcal{M}} w_{km} \leq W^{\max} \quad (11e)$$

$$x_{kmf} \in \{0, 1\} \quad \forall k \in \mathcal{K}, \ \forall m \in \mathcal{M}, \ \forall f \in \mathcal{F}, \quad (11f)$$

where (11b) and (11c) capture the file redundancy avoidance and storage capacity constraints, respectively; (11d) represents the service-level-agreement (SLA) constraints for MVNO m at BS k, which guarantees certain minimum number of allocated channels for each MVNO; (11e) denotes the bandwidth constraint; and (11f) denotes the integer caching decision variables at BSs.

IV. PROPOSED ALGORITHMS

In problem (11), because $\boldsymbol{x}$ and $\boldsymbol{w}$ are vectors of integer optimization variables and the elements of $\boldsymbol{w}$ are in the exponent and factorial parts of $P_{km}(\boldsymbol{x}, \boldsymbol{w})$ in (8), this problem is a mixed-integer non-linear program (MINLP), which is very difficult to solve optimally. Therefore, we propose a two-step iterative algorithm to tackle (11). In iteration i, we propose Algorithm 1 which is used to find the optimal channel allocation $\boldsymbol{w}^*_{(i)}$ based on the caching solution $\boldsymbol{x}^*_{(i-1)}$ obtained in the previous iteration $(i-1)$. Then, the proposed bisection-search based Algorithm 3 is used to determine the caching decision solution $\boldsymbol{x}^*_{(i)}$ based on the newly obtained value $\boldsymbol{w}^*_{(i)}$. With the newly obtained $\boldsymbol{w}^*_{(i)}$ and $\boldsymbol{x}^*_{(i)}$, we compute the maximum request outage probability $\varphi_{(i)}$ for iteration i. The overall procedure can be illustrated as

$$\underbrace{\boldsymbol{x}^*_{(0)} \rightarrow \boldsymbol{w}^*_{(0)}}_{\text{Initialization, } \varphi_{(0)}} \rightarrow \ldots \rightarrow \underbrace{\boldsymbol{x}^*_{(i)} \rightarrow \boldsymbol{w}^*_{(i)}}_{\text{Iteration } i, \varphi_{(0)}} \rightarrow \ldots \rightarrow \underbrace{\boldsymbol{x}^* \rightarrow \boldsymbol{w}^*}_{\text{Optimal } \varphi^*},$$

where the stopping condition is $|\varphi_{(i)} - \varphi_{(i-1)}| < \varepsilon$ with $0 < \varepsilon << 1$. Finally, we propose a heuristic fast algorithm (Section IV-C) for joint resource allocation and content caching based on the properties studied from the bisection search based algorithm.

Algorithm 1 Channel Allocation for a Given Caching Solution

1: **allocate** $W_{km}^{\min}$ channels to MVNO m at BS k to satisfy (11d).
2: **calculate** $W^{\text{free}} = W^{\max} - \sum_{k \in \mathcal{K}} \sum_{m \in \mathcal{M}} W_{km}^{\min}$
3: **while** $W^{\text{free}} > 0$ **do**
4: **find** $(k^*, m^*) = \underset{k,m}{\text{argmax}} \, \Phi_{km}(\boldsymbol{x}, \boldsymbol{w})$
5: $w_{k^*m^*} = w_{k^*m^*} + 1$
6: $W^{\text{free}} = W^{\text{free}} - 1$
7: **end while**
8: **obtain** optimal $\boldsymbol{w}^*$

A. CHANNEL ALLOCATION FOR A GIVEN CACHING POLICY

In this subsection, we propose an algorithm which allocates the optimal number of channels to MVNOs at each BS to minimize the maximum request rejection rate in the network for a given caching solution (i.e., for given $\boldsymbol{x}^*$). First, we characterize the properties of $\Phi_{km}(\boldsymbol{x}^*, \boldsymbol{w})$ in the following Proposition 1.

Proposition 1: (i) For a given $\boldsymbol{x}^*$, $P_{km}(\boldsymbol{x}^*, \boldsymbol{w})$ in (8) is a decreasing function of $\boldsymbol{w}$. (ii) For a given $\boldsymbol{w}^*$, $P_{km}(\boldsymbol{x}, \boldsymbol{w}^*)$ is an increasing function of $\boldsymbol{x}$.

Proof: Please refer to [26] for the proof of (i). Also from [26], $P_{km}(h_{km}(\boldsymbol{x})T_{km}, \boldsymbol{w})$ increases with $h_{km}(\boldsymbol{x})T_{km}$, where $T_{km} > 0, \forall k \in \mathcal{K}, \forall m \in \mathcal{M}$. Further, $h_{km}(\boldsymbol{x})$ in (5) is an increasing function of $\boldsymbol{x}$. Hence, $P_{km}(\boldsymbol{x}, \boldsymbol{w})$ increases with $\boldsymbol{x}$. ■

Proposition 1 suggests that to minimize $\Phi_{km}(\boldsymbol{x}^*, \boldsymbol{w})$, we need to make w_{km} as large as possible (i.e., allocating the largest possible number of channels). These results are leveraged to develop our channel allocation algorithm, which is described in Algorithm 1. Specifically, we initially attempt to satisfy all SLA bandwidth constraints by allocating $W_{km}^{\min}$ channels to MVNO m at BS k. Then, we sequentially allocate one available channel at each iteration to the MVNO m at BS k, whose $\Phi_{km}(\boldsymbol{x}, \boldsymbol{w})$ is highest at each allocation step, until all channels are used up. Lemma 1 stated in the following confirms the optimality of Algorithm 1.

Lemma 1: For a given caching strategy $\boldsymbol{x}^*$, Algorithm 1 optimally allocates channels to individual MVNOs at all BSs

to minimize the largest request outage probability in the network.

Proof: For a given $\boldsymbol{x}^*$, denote $(k^*, m^*) = \underset{k,m}{\text{argmax}}\, \Phi_{km}(\boldsymbol{x}^*, \boldsymbol{w})$ as the MVNO having current largest outage probability φ, i.e., $\varphi = \max_{k,m} \Phi_{km}(\boldsymbol{x}^*, \boldsymbol{w})$. Suppose we allocate a channel for an arbitrary MVNO (k, m) different from MVNO (k^*, m^*). As mentioned above, $\Phi_{km}(\boldsymbol{x}^*, \boldsymbol{w})$ decreases as w_{km} increases. Thus we have

$$\Phi_{km}(\boldsymbol{x}^*, w_{km} + 1) < \Phi_{km}(\boldsymbol{x}^*), w_{km}) < \varphi$$

is true for all MVNOs different from MVNO (k^*, m^*). Obviously, this strategy does not reduce φ. Therefore, allocating one available channel to the MVNO having current largest request outage probability in each step of Algorithm 1 is the optimal strategy. ■

B. CACHING STRATEGY FOR A GIVEN CHANNEL ALLOCATION SOLUTION

We now optimize the caching decision variables $\boldsymbol{x}$ at all BSs to minimize the maximum request outage probability among MVNOs at all BSs, given the channel allocation solution $\boldsymbol{w}^*$ obtained from Algorithm 1. Specifically, problem (11) becomes

$$\min_{\boldsymbol{x}} \max_{k\in\mathcal{K}, m\in\mathcal{M}} \Phi_{km}\left(\boldsymbol{x}, \boldsymbol{w}^*\right) \tag{12a}$$

$$\text{s.t.} \sum_{m\in\mathcal{M}} x_{kmf} \le 1, \quad \forall k \in \mathcal{K},\ \forall f \in \mathcal{F} \tag{12b}$$

$$\sum_{m\in\mathcal{M}} \sum_{f\in\mathcal{F}} x_{kmf} \le C_k, \quad \forall k \in \mathcal{K} \tag{12c}$$

$$x_{kmf} \in \{0, 1\} \quad \forall k \in \mathcal{K},\ \forall m \in \mathcal{M},\ \forall f \in \mathcal{F}. \tag{12d}$$

Note that problem (12) is still an MINLP because $\Phi_{km}\left(\boldsymbol{x}, \boldsymbol{w}^*\right)$ is a non-linear function of integer variable vector $\boldsymbol{x}$. Thus, we propose to tackle this problem indirectly by exploiting the following properties of $\Phi_{km}\left(\boldsymbol{x}, \boldsymbol{w}^*\right)$. First, using the results in (7), (8), and (9), we can rewrite (10) as follows:

$$\begin{aligned}\Phi_{km}\left(\boldsymbol{x}, \boldsymbol{w}^*\right) &= \frac{\mu_{km}\left(\boldsymbol{x}, \boldsymbol{w}^*\right) + \bar{h}_{km}\left(\boldsymbol{x}\right)}{\lambda_{km}} \\ &= \frac{h_{km}(\boldsymbol{x}) P_{km}\left(h_{km}(\boldsymbol{x}), \boldsymbol{w}^*\right) + \lambda_{km} - h_{km}(\boldsymbol{x})}{\lambda_{km}} \\ &= \Phi_{km}\left(h_{km}(\boldsymbol{x}), \boldsymbol{w}^*\right). \end{aligned} \tag{13}$$

Hence, $\Phi_{km}\left(\boldsymbol{x}, \boldsymbol{w}^*\right)$ can be considered as a function of $h_{km}(\boldsymbol{x})$ for a given $\boldsymbol{w}^*$, i.e., $\Phi_{km}\left(h_{km}(\boldsymbol{x}), \boldsymbol{w}^*\right)$.

Its properties, especially its convexity, are stated in the following Proposition 2.

Proposition 2: $\Phi_{km}\left(h_{km}(\boldsymbol{x}), \boldsymbol{w}^*\right)$ *is a convex function of* $h_{km}(\boldsymbol{x})$ *for a given* $\boldsymbol{w}^*$. *Moreover, it is a decreasing function of* $h_{km}(\boldsymbol{x})$.

Proof: From [27], the loss rate $h_{km}(\boldsymbol{x}) P_{km}\left(h_{km}(\boldsymbol{x}), \boldsymbol{w}^*\right)$ is a convex function of $h_{km}(\boldsymbol{x})$ for a given $\boldsymbol{w}^*$. Also from [27], $\frac{\partial\left[h_{km}(\boldsymbol{x}) P_{km}\left(h_{km}(\boldsymbol{x}), \boldsymbol{w}^*\right)\right]}{\partial h_{km}(\boldsymbol{x})} \in [0, 1]$. Therefore, $\frac{\partial \Phi_{km}\left(h_{km}(\boldsymbol{x}), \boldsymbol{w}^*\right)}{\partial h_{km}(\boldsymbol{x})} \le 0$, which means $\Phi_{km}\left(h_{km}(\boldsymbol{x}), \boldsymbol{w}^*\right)$ is decreasing with $h_{km}(\boldsymbol{x})$. ■

Consequently, $\max_{k\in\mathcal{K}, m\in\mathcal{M}} \Phi_{km}\left(h_{km}\left(\boldsymbol{x}\right), \boldsymbol{w}\right)$ can be considered as the pointwise maximum function over $h_{km}(\boldsymbol{x})$, which is convex [24]. We now transform problem (12) to the following convex optimization problem over $\boldsymbol{h}$, where $\boldsymbol{h} = \{h_{km}\left(\boldsymbol{x}\right)\}, \forall k \in \mathcal{K}, \forall m \in \mathcal{M}$.

$$\min_{\boldsymbol{h}, \varphi} \ \varphi \tag{14a}$$

$$\text{s.t. } \Phi_{km}\left(h_{km}(\boldsymbol{x}), \boldsymbol{w}^*\right) \le \varphi, \quad \forall k \in \mathcal{K},\ \forall m \in \mathcal{M} \tag{14b}$$

$$h_{km}\left(\boldsymbol{x}\right) \in \mathcal{H}, \quad \forall k \in \mathcal{K},\ \forall m \in \mathcal{M}. \tag{14c}$$

In (14), $\mathcal{H}$ denotes the set of all feasible values of $h_{km}\left(\boldsymbol{x}\right)$, which is dependent on the feasible set of $\boldsymbol{x}$ according to the constraints of problem (12). Particularly, $\mathcal{H}$ can be determined from the following constraints:

$$h_{km}\left(\boldsymbol{x}\right) = \sum_{f\in\mathcal{F}} \lambda_{km} q_{kmf} \left(\sum_{i\in\mathcal{M}} x_{kif}\right), \quad \forall k \in \mathcal{K},\ \forall m \in \mathcal{M} \tag{15a}$$

$$\sum_{m\in\mathcal{M}} x_{kmf} \le 1, \quad \forall k \in \mathcal{K},\ \forall f \in \mathcal{F} \tag{15b}$$

$$\sum_{m\in\mathcal{M}} \sum_{f\in\mathcal{F}} x_{kmf} \le C_k, \quad \forall k \in \mathcal{K} \tag{15c}$$

$$x_{kmf} \in [0, 1], \quad \forall k \in \mathcal{K},\ \forall m \in \mathcal{M},\ \forall f \in \mathcal{F}. \tag{15d}$$

Constraints (15b) and (15c) are originally from (12b) and (12c), respectively. Meanwhile, (15d) is the relaxed version of (12d) to achieve the actual upper bound for all h_{km} and continuous value for $\mathcal{H}$.

Obviously Φ_{km} must be in [0, 1] for all k and m because it is the probability. Moreover, $\mathcal{H}$ is constrained by (15). Based on the results in Proposition 2, we can solve problem (14) by using the bisection search method [24] to find the optimal value φ^*, i.e., the minimum of maximal request outage probability. From φ^*, we obtain the corresponding optimal solution $h^*_{km} \in \mathcal{H}, \forall k \in \mathcal{K}, \forall m \in \mathcal{M}$ based on (13). However, it is difficult to map φ^* back to h^*_{km} for all k and m due to the unknown inversed function of $\Phi_{km}\left(h_{km}(\boldsymbol{x}), \boldsymbol{w}^*\right)$, which is denoted as Φ^{-1}_{km}. Therefore, in the next subsection, we apply Newton's method [24] to find an approximate output h^*_{km} from Φ^{-1}_{km} taking the outage probability φ^* and channel allocation $\boldsymbol{w}^*$ as the inputs.

1) FINDING h_{km} FROM φ

Given the request outage propability φ_{km} and channel allocation w_{km} for MVNO m at BS k, we need to find

$$h_{km} \text{ s.t. } \Phi_{km}\left(h_{km}, w_{km}\right) = \varphi_{km}. \tag{16}$$

Without loss of generality and for the sake of simplicity, we omit the subscripts of $\Phi_{km}\left(h_{km}, w_{km}\right)$ and the input w_{km}. Thus, (16) can be re-written as

$$h : \Phi\left(h\right) = \varphi. \tag{17}$$

Algorithm 2 Finding Cache-Hit Rate h From Given Outage Probability φ

1: **calculate** $L = a\,(1 - P(a, w))$.
2: **initialize** h_0 according to (24).
3: **update** h_1 according to (18).
4: **repeat** step (3) **until** convergence with small error ε.

According to Newton's search method, with a properly initial guess h_0, we can find a better approximation h_1 for (17) by

$$h_1 = h_0 - \frac{\Phi\,(h_0) - \varphi}{\nabla_h \Phi(h_0)}, \tag{18}$$

where

$$\nabla_h \Phi(h) \triangleq \frac{\partial \Phi}{\partial h} = \frac{P + (w - hT + hTP)\,P - 1}{\lambda}, \tag{19}$$

and $P \triangleq P_{km}\,(h_{km}, w)$, $w \triangleq w_{km}$, and $T \triangleq T_{km}$ for a particular MVNO (m, k) under consideration in (8). We perform the iterative update (18) until convergence to achieve a stable approximation of h.

Now, we present an approach to determine a feasible initial value h_0. From (7) and (10), we have

$$\varphi = \frac{hP\,(a, w) - h + \lambda}{\lambda} = \frac{\lambda - \frac{a(1-P(a,w))}{T}}{\lambda} = \frac{\lambda - \frac{L}{T}}{\lambda}, \tag{20}$$

where

$$a \triangleq hT \tag{21}$$

$$L \triangleq a\,(1 - P(a, w))\,. \tag{22}$$

Hence, we have $L = T\lambda\,(1 - \varphi)$. According to [28, eq. (53)], we have the inequality

$$a_0 < L\left(1 + \frac{L}{w\,(w - L)}\right). \tag{23}$$

Thus, we can choose the initial value h_0 as follows:

$$h_0 = \frac{a_0}{T} = \frac{L\left(1 + \frac{L}{w(w-L)}\right)}{T}. \tag{24}$$

The Newton's search method for calculating the hit rate h given the request outage probability φ and the parameters w, T and λ is summarized in Algorithm 2.

We state the convergence property and the solution uniqueness of Algorithm 2 in the following proposition.

Proposition 3: Algorithm 2 converges to a unique value of h.

Proof: Due to Proposition 2, Φ represents the *one-to-one mapping* function between its output φ and the input cache hit rate h. Therefore, Algorithm 2 returns a *unique value* of h for (16). ∎

Recall that $\Phi_{km}\left(h_{km}\,(\boldsymbol{x}), \boldsymbol{w^*}\right)$ is a decreasing function of $h_{km}\,(\boldsymbol{x})$ for all $h_{km} \in \mathcal{H}$ according to Proposition 2. Therefore, given a request outage probability value φ that constraint (14a) is satisfied, the cache hit rate $h_{km}(\boldsymbol{x})$ must satisfy the following one-to-one relationship

$$\Phi_{km}\left(h_{km}(\boldsymbol{x}), \boldsymbol{w^*}\right) \leq \varphi \iff h_{km}(\boldsymbol{x}) \geq h, \tag{25}$$

where $\Phi_{km}\left(h, \boldsymbol{w^*}\right) = \varphi$. However, all $h_{km}(\boldsymbol{x})$'s are constrained by $\mathcal{H}$ as in (15), we need to verify the feasibility of $\boldsymbol{x}$. This motivates us to study the caching strategy in two different relevant scenarios in which the popularity orders of different files at different BS are the *same* and *different*, in the following two subsections.

2) SAME-ORDER FILE POPULARITY CASE

Suppose that we rank the content request probabilities for different files f of each MVNO (m, k) in the descending order of their values. In the same-order popularity case, all MVNOs at each BS k have the same order of file indices under this ranking (i.e., the most popular file, second most popular file,... of all MVNOs at each BS k are the same). Suppose that MVNO (m^*, k^*) is the one having largest request outage probability for a given channel allocation solution $\boldsymbol{w^*}$, i.e.,

$$\left(k^*, m^*\right) = \underset{k,m}{\operatorname{argmax}}\ \Phi_{k^*m^*}\left(h_{k^*m^*}\,(\boldsymbol{x}), \boldsymbol{w^*}\right). \tag{26}$$

Given the channel allocation solution $\boldsymbol{w^*}$, $\Phi_{k^*m^*}\left(h_{k^*m^*}\,(\boldsymbol{x}), \boldsymbol{w^*}\right)$ decreases as $h_{k^*m^*}\,(\boldsymbol{x})$ increases according to Proposition 2. From (5), $h_{k^*m^*}\,(\boldsymbol{x})$ is a non-negative linear combination of caching decision vector $\boldsymbol{x}$. Consequently, increasing $h_{km}\,(\boldsymbol{x}_{km})$ by caching more content preferred by MVNO (m^*, k^*) is the best strategy for reducing its request outage probability. Moreover, as file popularity ranks of different files for all $m \in \mathcal{M}$ are identical at each BS k, the *most popular caching (MPC)* is the best strategy for each BS for all MVNOs with the same-order file popularity to reduce the outage probability. In particular, this MPC strategy results in the optimal caching solution for the following problem considering the cache redundancy avoidance constraint:

$$\max_{\boldsymbol{x}_{k^*m}} \sum_{m \in \mathcal{M}} h_{k^*m}\,(\boldsymbol{x}_{k^*m}) \tag{27a}$$

$$\text{s.t.} \sum_{m \in \mathcal{M}} x_{k^*mf} \leq 1 \quad \forall f \in \mathcal{F} \tag{27b}$$

$$\sum_{m \in \mathcal{M}} \sum_{f \in \mathcal{F}} x_{k^*mf} \leq C_{k^*} \tag{27c}$$

$$x_{k^*mf} \in [0, 1] \quad \forall m \in \mathcal{M},\ \forall f \in \mathcal{F}. \tag{27d}$$

Note that as each BS has its own storage repository, the caching decisions at different BSs are independent. This means that only $\boldsymbol{x}_{k^*m}$ for all $m \in \mathcal{M}$ is affected by the problem (27). Hence, we can apply (27) for all BSs having the same-order file popularity over all MVNOs. Due to Proposition 2 and the non-negative linear combination of $\boldsymbol{x}$ in (5), the solution of (27) will reduce the request outage probability for all MVNOs at the considered BSs.

3) DIFFERENT-ORDER FILE POPULARITY CASE

In this case, the popularity ranks of different files for different MVNOs at each BS k can be different (e.g., the most popular file for MVNO 1 can be different from the most popular file for MVNO 2). Hence, maximizing $h_{k^*m^*}\,(\boldsymbol{x})$ may cause the decrement of $h_{k^*m}\,(\boldsymbol{x})$ for some $m \in \mathcal{M}/\{m^*\}$.

In fact, caching more content preferred by MVNO (m^*, k^*) may evict other MVNOs' most favorite content due to the storage capacity limitation. This results in the increment of $\Phi_{k^*m}\left(h_{k^*m}(\boldsymbol{x}), \boldsymbol{w}^*\right)$ for some $m \in \mathcal{M}/\{m^*\}$ according to Proposition 2. Denote the pre-caching-decision largest request outage probability by φ. If the post-caching-decision request outage probability Φ_{k^*m} for some $m \neq m^*$ exceeds φ, then the caching decision problem by (27) fails to obtain better outage probability, i.e., $\varphi^{\text{new}} < \varphi$. Therefore, the MPC strategy in (27) may not be the best strategy in the case of different-order file popularity. To this end, we need to guarantee that optimizing $\boldsymbol{x}_{k^*m^*}$ for MVNO (k^*, m^*) does not make post-caching-decision Φ_{k^*m} exceed φ for all $m \in \mathcal{M}/\{m^*\}$, which is mathematically imposed by

$$\Phi_{km}\left(h_{km}(\boldsymbol{x}), \boldsymbol{w}^*\right) \leq \varphi, \quad \forall m \in \mathcal{M},\ \forall k \in \mathcal{K}. \tag{28}$$

Due to Proposition 2, constraint (28) is equivalent to

$$h_{km}(\boldsymbol{x}) \geq h_{km}^{\text{low}}, \quad \forall m \in \mathcal{M},\ \forall k \in \mathcal{K}, \tag{29}$$

where

$$\Phi_{km}\left(h_{km}^{\text{low}}, \boldsymbol{w}^*\right) = \varphi, \quad \forall m \in \mathcal{M},\ \forall k \in \mathcal{K}. \tag{30}$$

In other words, h_{km}^{low} is the output of the inverse function Φ_{km}^{-1} taking φ as the input. We will use Algorithm 2 to find h_{km}^{low}. With constraint (29), the caching decision problem (27) becomes (31) and we state its properties in Proposition 4.

$$\max_{\boldsymbol{x}_{km}} \sum_{m\in\mathcal{M}} h_{km}(\boldsymbol{x}_{km}) \tag{31a}$$

$$\text{s.t. } h_{km}(\boldsymbol{x}) \geq h_{km}^{\text{low}}, \quad \forall m \in \mathcal{M} \tag{31b}$$

$$\sum_{m\in\mathcal{M}} x_{k^*mf} \leq 1 \quad \forall f \in \mathcal{F} \tag{31c}$$

$$\sum_{m\in\mathcal{M}}\sum_{f\in\mathcal{F}} x_{kmf} \leq C_k \tag{31d}$$

$$x_{kmf} \in [0, 1] \quad \forall m \in \mathcal{M},\ \forall f \in \mathcal{F}. \tag{31e}$$

Proposition 4: Problem (32) *covers problem* (27) *in both cases of same-order and different-order file popularity.*

Proof: First, if we remove constraint (31b) from (31), then it is equivalent to (27). Hence, the feasible solution set of (31) is a subset of the feasible solution set of (27). Second, in the case of same-order file popularity at BS k, the constraint (31b) is always satisfied. In fact, all the cache hit rate $h_{km}(\boldsymbol{x})$ for all m will increase by caching any file due to the identical set of file popularity $\mathcal{Q}_{km}$ for all m, i.e., $h_{km}\left(\boldsymbol{x}^*\right) \geq h_{km}^{\text{low}}, \forall m \in \mathcal{M}$. ■

With Proposition 4, solving (31) is sufficient for obtaining caching decisions for the both cases of file popularity at each BS. Recall that each BS is equipped with an independent storage repository. Therefore, by taking summation over $\mathcal{K}$ (all BSs) in the objective function of (31), we obtain the caching decision optimization problem for all BSs. Mathematically, this caching problem can be stated as

$$\max_{\boldsymbol{x}} \sum_{k\in\mathcal{K}}\sum_{m\in\mathcal{M}} h_{km}(\boldsymbol{x}) \tag{32a}$$

Algorithm 3 Iterative Channel Allocation and Content Caching Placement

set $i = 1$ and tolerance $\varepsilon > 0$.
initialize $\boldsymbol{x}_{(i)}^*$ according to most popular caching strategy with equal storage partition.
initialize channel allocation $\mathbf{w}_{(i)}$ using Algorithm 1 given $\boldsymbol{x}_{(i)}^*$.
calculate $\Phi_{km}, \forall k \in \mathcal{K}, \forall m \in \mathcal{M}$.
find largest outage probability $\varphi_{(i)} = \max_{k,m} \Phi_{km}$.
set $\Delta_\varphi = 1$
while $\Delta_\varphi > \varepsilon$ **do**
  $i = i + 1$
  set $\phi^{\text{up}} = 1$
  set $\phi^{\text{low}} = 0$
  while $\phi^{\text{up}} - \phi^{\text{low}} > \varepsilon$ **do**
    $\phi_{(i)} = \left(\phi^{\text{up}} + \phi^{\text{low}}\right)/2$
    find h_{km} from $\phi_{(i)}$ by using Algorithm 2.
    solve problem (32) to find $\boldsymbol{x}^*$.
    if $\boldsymbol{x}^*$ is feasible **then**
      $\phi^{\text{up}} = \phi_{(i)}$
      $\boldsymbol{x}_{(i)}^* = \boldsymbol{x}^*$
    else
      $\phi^{\text{low}} = \phi_{(i)}$
    end if
  end while
  find optimal $\boldsymbol{w}_{(i)}^*$ by using Algorithm 1.
  calculate $\Phi_{km}^{(i)}, \forall k \in \mathcal{K}, \forall m \in \mathcal{M}$.
  find $\varphi_{(i)} = \max_{k,m} \Phi_{km}^{(i)}$
  calculate $\Delta_\varphi = |\varphi_{(i)}^* - \varphi_{(i-1)}^*|$
end while
obtain final $\boldsymbol{w}^*$ and $\boldsymbol{x}^*$ from Algorithm 1 given $\boldsymbol{x}_{(i)}^*$.

$$\text{s.t. } h_{km}(\boldsymbol{x}) \geq h_{km}^{\text{low}}, \quad \forall m \in \mathcal{M},\ \forall k \in \mathcal{K} \tag{32b}$$

$$\sum_{m\in\mathcal{M}} x_{kmf} \leq 1, \quad \forall k \in \mathcal{K},\ \forall f \in \mathcal{F} \tag{32c}$$

$$\sum_{m\in\mathcal{M}}\sum_{f\in\mathcal{F}} x_{kmf} \leq C_k, \quad \forall k \in \mathcal{K} \tag{32d}$$

$$x_{kmf} \in [0, 1] \quad \forall m \in \mathcal{M},\ \forall f \in \mathcal{F}. \tag{32e}$$

We can solve this linear programming problem by using any available solver such as CVX [25]. Finally, Algorithm 3 summarizes our iterative joint channel allocation (Algorithm 1) and content caching placement strategy in (32).

4) ROUNDING CACHING DECISION VARIABLES

After obtaining the result from Algorithm 3, we need to round the caching decision variables due to the underlying constraint relaxation. For the same-order file popularity case, the optimal solution $\boldsymbol{x}^*$ of (32) is actually an integral vector as this is the result of the most popular caching policy. For different-order file popularity case, $\boldsymbol{x}^*$ is a real-value vector in $[0, 1]^{K\times M\times F}$ due to the relaxed con-

Algorithm 4 Rounding Caching Decision Variables

1: **initialize** small $\varepsilon > 0$
2: **obtain** the optimal request outage probability value φ from Algorithm 3.
3: **repeat**
4: **obtain** $h_{km}^{\text{low}}, \forall k \in \mathcal{K}, \forall m \in \mathcal{M}$ with Algorithm 2.
5: **solve** problem (32) with integral constraint.
6: **if** integral solution $\boldsymbol{x}_{\text{INT}}^*$ is not found **then**
7: $\varphi = \varphi + \varepsilon$
8: **end if**
9: **until** integral solution $\boldsymbol{x}_{\text{INT}}^*$ is found.

straint (32e). To efficiently round $\boldsymbol{x}^*$ to a binary vector, we propose an algorithm which is based on the modified version of problem (32). Specifically, we first loosen the constraint (32b) by adding a positively small ε to the optimal φ obtained from the original problem (32), and thus obtaining the corresponding h_{km}^{low} for all MVNO (m, k). Next, we transform (32e) to the corresponding integral constraint. Finally, we solve the modified version of (32) using a solver supporting integer linear programming (ILP) such as Gurobi [29]. We continue loosening the constraint (32b) by adding ε to current φ, then solve the modified (32) iteratively until obtaining a feasible integral solution $\boldsymbol{x}_{\text{INT}}^*$. This procedure is summarized in the Algorithm 4.

5) CONVERGENCE AND COMPLEXITY ANALYSIS

First, the convergence of Algorithm 3 is stated in Lemma 2

Lemma 2: Algorithm 3 converges to a local optimum point of channel allocation and caching decision.

Proof: The bisection search based Algorithm 3 consists of two main steps: channel allocation (Algorithm 1) and relaxed caching decision (problem (32)) through the Newton's search method (Algorithm 2). In each step, a single type of variables is optimized while the remaining variables remain the same as in the previous iteration. Algorithm 1 achieves the optimal channel allocation solution $\boldsymbol{w}^*$ in each iteration according to Lemma 1. Meanwhile, Algorithm 2 returns a unique h_{km} given the outage probability φ due to the one-to-one mapping property induced by Proposition 2. This results in a unique caching decision for problem (32). Consequently, Algorithm 3 implements the block coordinate descent (BCD) search, whose convergence is guaranteed [30]. However, we can only guarantee that Algorithm 3 converges to a local optimal. This is because the caching decision problem (32) is solved through approximation of h_{km} for all k and m. ■

Regarding the complexity, the inner while-loop of Algorithm 3 is upper-bounded by $\lceil \log_2 \left(\frac{\Phi^{\text{up}} - \Phi^{\text{low}}}{\varepsilon} \right) \rceil$ search iterations. Algorithm 1 has $\mathcal{O}(KMW)$ complexity, where K, M and W are the total numbers of BSs, MVNOs and wireless channels, respectively. The Newton's search method in Algorithm 2 converges within tens of iterations, where each iteration has complexity of $\mathcal{O}(KM)$. Solving linear programming problem (32) involves polynomial time complexity. Algorithm 1 thus has polynomial time complexity. Its running time is affordable for the underlying resource provisioning optimization as it is only repeated once over a long time period, e.g., hours or days.

C. PROPOSED HEURISTIC ALGORITHM

For performance evaluation, we now present another heuristic algorithm. In this fast algorithm, we first equally split the storage repository into M partitions at each BS, each partition is then assigned to one MVNO for fairness. However, we allow the contents cached on these partitions to be shared among all MVNOs co-located at each BS. Moreover, the contents are cached in an iterative manner as described in the following. In each iteration and considered storage partition corresponding to a particular MVNO, we cache the MVNO's most favorite content which still not exists in any storage partitions. The procedure is repeated until all the partition capacity is fully cached. Feeding the newly derived caching solution to Algorithm 1, we finally obtain the channel allocation solution.

It can be verified that this algorithm requires about $K \times M \times C_k$ steps for caching files in each BS, thus resulting in the total complexity of $\mathcal{O}(KMC_k)$. Note that this is much faster than the proposed bisection-search algorithm for caching decision. Moreover, Algorithm 1 has $\mathcal{O}(KMW)$ complexity. Hence, the proposed heuristic algorithm has overall complexity of $\mathcal{O}(KM(W + C_k))$.

V. NUMERICAL RESULTS

In this section, we evaluate the performance of our proposed algorithms through computer simulation under the following setting. We consider the network with 5 BSs serving 3 MVNOs, which access a list of 100 files, i.e., $K = 5, M = 3$ and $F = 100$. The average request rates for each MVNO are randomly chosen in the range of [1, 15], which results in the total of request rates from tens to hundreds requests arriving to the considered network in one second. We assume that wireless channels are allocated and accessed in the orthogonal manner to avoid strong interference. File requests (i.e., content popularity) are assumed to follow the Zipf distribution where the probability of requesting the file having rank $f \in \{1, \ldots, F\}$ is given by

$$Z(f, \gamma) = \frac{f^{-\gamma}}{\sum_{n=1}^{F} n^{-\gamma}}, \tag{33}$$

where F is the total number of files in the list and γ is the Zipf parameter. This parameter is chosen as $\gamma \in [0.3, 1.2]$ in the simulation setting, and MVNOs co-located at the same BS are assumed to have the same γ value. However, the rank of each file in the list is set randomly with respect to each MVNO and BS. The obtained numerical results are averaged over 100 realizations of the file ranking in the list. The obtained results, therefore, represent the average performance of the proposed algorithms over different realizations of file popularity including same-order and different-order cases.

We assume the same service time with $T_{km} = 1$ for all MVNO (m, k).

We assume that all BSs share $W^{\max} = 90$ wireless channels in the orthogonal manner to serve file requests from MVNOs. Each SLA requirement is set with $W_{km}^{\min} = 2$, $\forall k \in \mathcal{K}, \forall m \in \mathcal{M}$. All the BSs have equal storage capacity which varies from 3% to 51% of the total file size. Note that the storage capacity can be much smaller than the total size of all files in the list in practice [8]. Finally, we set $\varepsilon = 10^{-4}$ in the stopping condition for all experiments. Note that the outage probability presented in all following figures corresponds to the content access failure from the network edge (i.e., BS caches). In practice, any delay-tolerant content requests can be served (i.e., zero end-to-end outage probability) because requested contents can be always retrieved from their content servers despite potentially large download delay.

For performance evaluation, we compare our proposed algorithms with baseline algorithms based on the most popular caching (MPC) strategy and different channel allocation strategies. The the following baseline algorithms and the proposed heuristic algorithm are considered in the performance evaluation.

- **MPC partitioning + Equal Chanl:** The storage repository and the channel budget are equally divided to all MVNOs at each BS. The MPC strategy is independently applied for each MVNO. The file sharing among MVNOs at the same BS is disabled.
- **MPC partitioning + Opt Chanl:** This setting is similar to the one above for storage repository partitioning. However, the channels are allocated to MVNOs following the proposed channel allocation in Algorithm 1.
- **MPC sharing + Opt Chanl:** This is our proposed fast heuristic algorithm in Section IV-C.

Storage resource strongly impacts the caching performance and thus our proposed algorithms. Figures 2a and 2b show that the proposed bisection-search based algorithm (Algorithm 3) with cache sharing consistently achieves the smallest maximum request outage probability in the cases with same-order and different-order file popularity, respectively. Moreover, the proposed rounding operation for caching decision variables result in negligible performance loss compared to the achieved performance before rounding, which confirms the efficacy of our design (the request outage probability obtained under relaxation from Algorithm 3 is the lower bound of the optimum value). Moreover, the proposed heuristic algorithm achieves performance very close to the proposed bisection-search based algorithm in the different-order popularity case, and both algorithms result in the same solution in the same-order file popularity case.

Moreover, these figures also confirm that cache sharing results in significant performance enhancement in the virtualized wireless network serving multiple MVNOs. In fact, sharing the cache allows efficient coordination among different MVNOs to avoid file caching redundancy, thus leaving more storage space to store more files. This in turns leads to reduction of the request outage probability. In contrast,

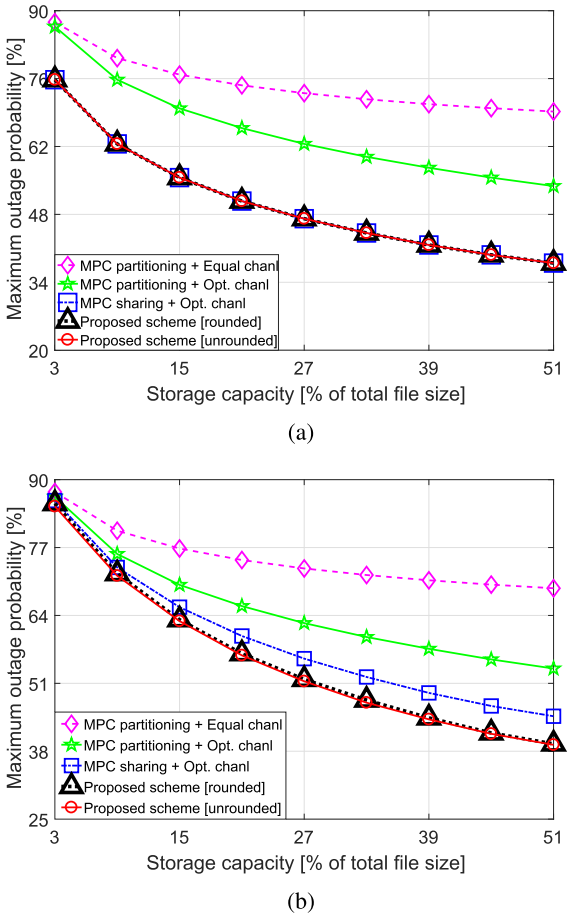

FIGURE 2. Maximum outage probability vs storage capacity. (a) Same-order file popularity. (b) Different-order file popularity.

the baselines schemes perform worse than our proposed algorithms, especially at the high storage capacity regime. This is because the larger number of files cached at BS is, the more flexibility is available for file sharing, especially in the case of different-order file popularity. Further, channel allocation with knowledge about the caching solution also helps to improve performance. This is confirmed by the fact that the baseline schemes with optimal channel allocation result in lower maximum request outage probability in comparison with those employing the equal channel allocation.

We present the maximum request outage probability among MVNOs at all BSs versus the total number of channels in Figures 3a and 3b for the cases of same-order and different-order file popularity, respectively. The results are obtained with the storage capacity equal to 15% of the total size of all files. Similar to Figure 2, Figure 3 confirms the greatest performance of our proposed bisection-search based algorithm as it achieves the lowest request outage probability compared with the remaining baselines. Figures 2 and 3 imply that instead of partitioning the available storage space to individual MVNOs, it is better to share it among MVNOs co-located at the same BS.

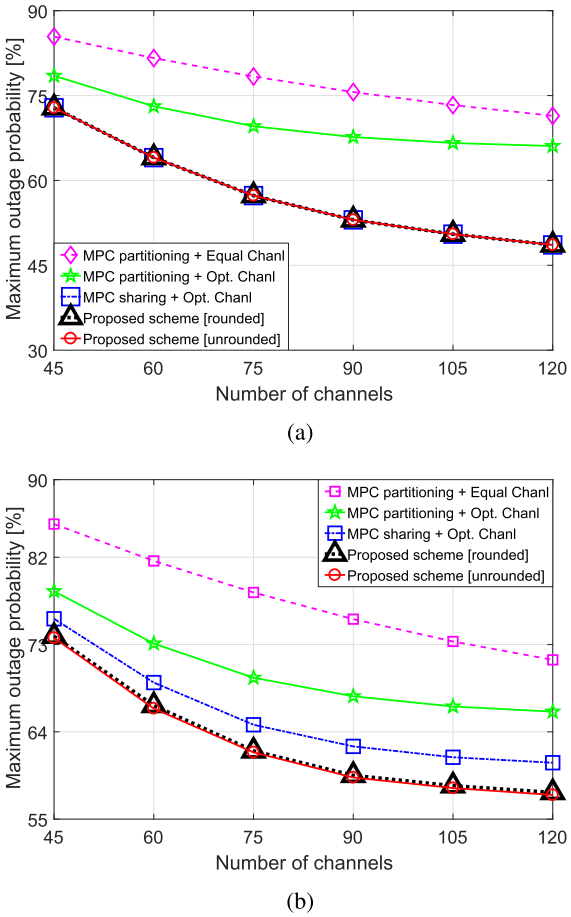

FIGURE 3. Maximum outage probability vs number of channels. (a) Same-order file popularity. (b) Different-order file popularity.

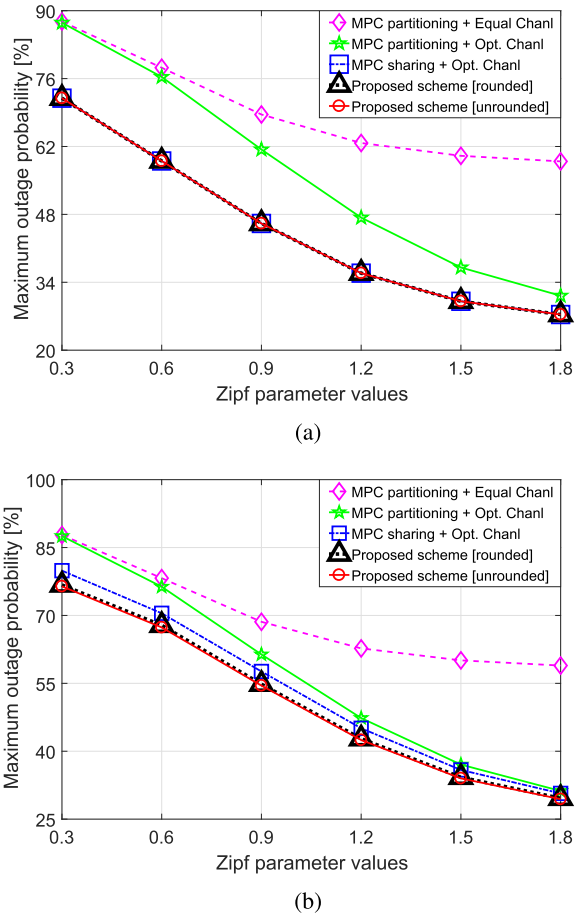

FIGURE 4. Maximum outage probability vs Zipf parameter. (a) Same-order file popularity. (b) Different-order file popularity.

Figure 4 shows the maximum request outage probability as a function of the Zipf parameter γ, which is set equal for all MVNOs and BSs. In this experiment, the storage capacity at each BS is set equal to 18% of the total size of all files in the file list. Meanwhile, the numerical results are averaged over four settings with different number of wireless channels which are equal to 60, 90, 120, and 150. The content request rate is generated randomly in the range of $[1, 15]$. Similar to the results in previous figures, our proposed algorithm achieves the best performance in terms of maximum request outage probability. Moreover, the proposed bisection-search based algorithm and heuristic algorithms result in significant reduction of the request outage probability when γ increases in the range $\gamma \in [0.3, 1.2]$, whereas the request outage probability decreases more slowly with γ when $\gamma > 1.2$. This is due to the property of Zipf distribution, in which its cumulative distribution function (CDF) increases faster with the increment of γ, given the same number of files.

Further, the probability mass function (PMF) of the Zipf distribution becomes long-tailed for large values of γ. Hence, caching the same number of files with larger Zipf parameter γ results in the greater hit rate $h_{km}(\boldsymbol{x})$, which in turn greatly decreases the rejection rate $\mu_{km}(\boldsymbol{x}, \boldsymbol{w})$ according to Proposition 2. However, when γ enters the large-value regime, the additional contributions to the hit rate by the low-rank files are negligible. Note that MVNOs co-located at the same BS would have very similar file ranking in practice (due to the their similar UEs' file preferences). This explains why our proposed heuristic algorithm, i.e., most popular caching strategy with our proposed channel allocation algorithm, can achieve very similar performance with our proposed bisection-search based algorithm.

Figure 5 shows the maximum request outage probability among MVNOs and BSs as we vary the maximum request rate. Specifically, in this experiment, the average request rate is set in the range $[1, 15 + \Delta]$, where $\Delta \in \{0, 5, 10, 15, 20, 25\}$. The numerical results are averaged over four different settings of channel budget with 60, 90, 120 and 150 wireless channels. Meanwhile, we fix the storage capacity at 18% of the total size of all files and the Zipf parameter at $\gamma = 0.6$. As Δ increases, i.e., the maximum file request rate increases, the maximum request outage probabilities of all schemes increase as well. Again, our proposed algorithm significantly outperforms other baselines over all values of Δ.

All the presented figures confirm the great performance of our proposed bisection-search based algorithm and the proposed heuristic algorithm (i.e, joint MPC sharing strategy

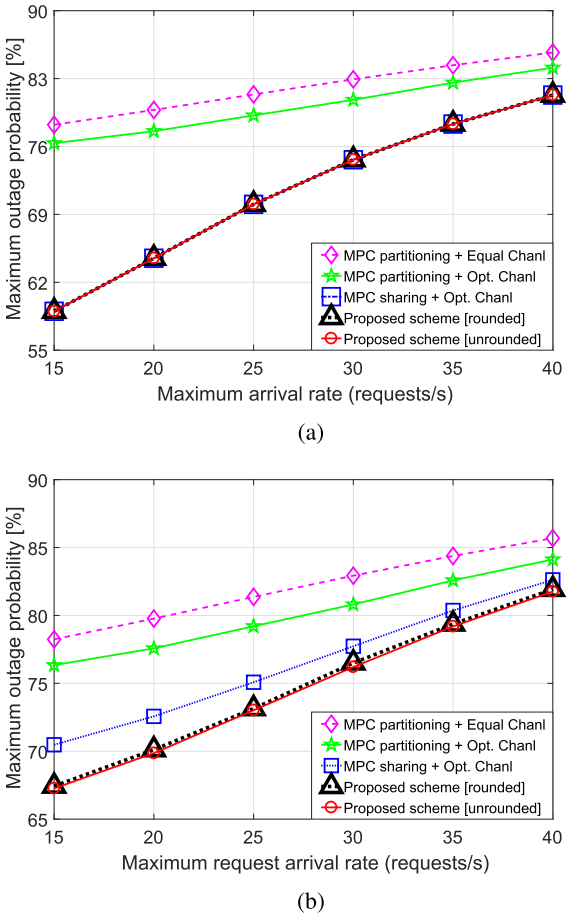

FIGURE 5. **Maximum outage probability vs maximum request arrival rate. (a) Same-order file popularity. (b) Different-order file popularity.**

with optimal channel allocation in Algorithm 1). In the case of same-order file popularity, Figures 2a, 3a, 4a, and 5a show that the proposed bisection-based algorithm and the heuristic algorithm achieve the same solution.

In the case of different-order file popularity, the performance of Algorithm 3 with caching-variable rounding is worse than that of the standalone Algorithm 3 (without rounding of caching decision variables), yet the performance loss due to cache decision variable rounding is negligible, as shown in Figures 2b, 3b, 4b, and 5b. Meanwhile, the proposed heuristic algorithm suffers from less than 5% performance loss in comparison with the standalone Algorithm 3. The performance gap between Algorithm 3 and the proposed heuristic algorithm is only visible in the regimes of large storage capacity, large channel budget, small Zipf parameter, and small maximum request arrival rate.

VI. CONCLUSION

We have studied the joint resource allocation and content caching problem in wireless networks with congested backhaul considering the dynamics of arrival/departure requests and the cache redundancy avoidance. We have proposed a bisection-search based algorithm to tackle the underlying design problem and we have proved that it converges to a local optimal solution in polynomial time. We further proposed a fast heuristic algorithm which attains moderate performance loss in comparison with the bisection-search based algorithm. Numerical results confirms our dominant performance in comparison with baseline schemes in different relevant network settings. Specifically, the results have confirmed that sharing the storage space among different MVNOs can enable to improve the caching performance significantly. Also, joint optimization of content caching and resource allocation is important to achieve the best performance, especially if the content popularity patterns of individual MVNOs at each BS are different.

REFERENCES

[1] J. G. Andrews *et al.*, "What will 5G be?" *IEEE J. Sel. Areas Commun.*, vol. 32, no. 6, pp. 1065–1082, Jun. 2014.
[2] Ericsson. (Sep. 2015). *Cloud Ran—The Benefits of Virtualization, Centralization and Coordination*. [Online]. Available: https://www.ericsson.com/assets/local/publications/white-papers/wp-cloud-ran.pdf
[3] China Mobile Research Institute. (2011). *C-RAN: The Road Towards Green Ran*. [Online]. Available: http://labs.chinamobile.com/cran/
[4] Qualcomm. (Nov. 2013). *The 1000x Mobile Data Challenge*. [Online]. Available: https://www.qualcomm.com/invention/1000x
[5] H. Liu, Z. Chen, and L. Qian, "The three primary colors of mobile systems," *IEEE Commun. Mag.*, vol. 54, no. 9, pp. 15–21, Sep. 2016.
[6] A. Khreishah, J. Chakareski, and A. Gharaibeh, "Joint caching, routing, and channel assignment for collaborative small-cell cellular networks," *IEEE J. Sel. Areas Commun.*, vol. 34, no. 8, pp. 2275–2284, Aug. 2016.
[7] Z. Chen, J. Lee, T. Q. S. Quek, and M. Kountouris, "Cooperative caching and transmission design in cluster-centric small cell networks," *IEEE Trans. Wireless Commun.*, vol. 16, no. 5, pp. 3401–3415, May 2017.
[8] G. Paschos, E. Bastug, I. Land, G. Caire, and M. Debbah, "Wireless caching: Technical misconceptions and business barriers," *IEEE Commun. Mag.*, vol. 54, no. 8, pp. 16–22, Aug. 2016.
[9] A. Maeder *et al.*, "A scalable and flexible radio access network architecture for fifth generation mobile networks," *IEEE Commun. Mag.*, vol. 54, no. 11, pp. 16–23, Nov. 2016.
[10] K. Poularakis, G. Iosifidis, V. Sourlas, and L. Tassiulas, "Exploiting caching and multicast for 5G wireless networks," *IEEE Trans. Wireless Commun.*, vol. 15, no. 4, pp. 2995–3007, Apr. 2016.
[11] C. Liang, F. R. Yu, H. Yao, and Z. Han, "Virtual resource allocation in information-centric wireless networks with virtualization," *IEEE Trans. Veh. Technol.*, vol. 65, no. 12, pp. 9902–9914, Dec. 2016.
[12] Q. Chen, F. R. Yu, T. Huang, R. Xie, J. Liu, and Y. Liu, "Joint resource allocation for software defined networking, caching and computing," in *Proc. IEEE GLOBECOM*, Washington, DC, USA, Dec. 2016, pp. 1–6.
[13] N. Golrezaei, K. Shanmugam, A. G. Dimakis, A. F. Molisch, and G. Caire, "FemtoCaching: Wireless video content delivery through distributed caching helpers," in *Proc. IEEE INFOCOM*, Orlando, FL, USA, Mar. 2012, pp. 1107–1115.
[14] N. Karamchandani, U. Niesen, M. A. Maddah-Ali, and S. N. Diggavi, "Hierarchical coded caching," *IEEE Trans. Inf. Theory*, vol. 62, no. 6, pp. 3212–3229, Jun. 2016.
[15] M. Ji, G. Caire, and A. F. Molisch, "Wireless device-to-device caching networks: Basic principles and system performance," *IEEE J. Sel. Areas Commun.*, vol. 34, no. 1, pp. 176–189, Jan. 2016.
[16] J. Tang and T. Q. S. Quek, "The role of cloud computing in content-centric mobile networking," *IEEE Commun. Mag.*, vol. 54, no. 8, pp. 52–59, Aug. 2016.
[17] M. Peng, C. Wang, V. Lau, and H. V. Poor, "Fronthaul-constrained cloud radio access networks: Insights and challenges," *IEEE Wireless Commun.*, vol. 22, no. 2, pp. 152–160, Apr. 2015.
[18] U. Siddique, H. Tabassum, E. Hossain, and D. I. Kim, "Wireless backhauling of 5G small cells: Challenges and solution approaches," *IEEE Wireless Commun.*, vol. 22, no. 5, pp. 22–31, Oct. 2015.
[19] M. Gregori, J. Gómez-Vilardebó, J. Matamoros, and D. Gündüz, "Wireless content caching for small cell and D2D networks," *IEEE J. Sel. Areas Commun.*, vol. 34, no. 5, pp. 1222–1234, May 2016.

[20] J. Zhao, T. Q. S. Quek, and Z. Lei, "Heterogeneous cellular networks using wireless backhaul: Fast admission control and large system analysis," *IEEE J. Sel. Areas Commun.*, vol. 33, no. 10, pp. 2128–2143, Oct. 2015.
[21] T. D. Tran and L. B. Le, "Joint resource allocation and content caching in virtualized multi-cell wireless networks," in *Proc. IEEE GLOBECOM*, Singapore, Dec. 2017, pp. 1–6.
[22] E. Hossain, M. Rasti, and L. B. Le, *Radio Resource Management in Wireless Networks: An Engineering Approach*. Cambridge, U.K.: Cambridge Univ. Press, 2017.
[23] R. B. Cooper, *Introduction to Queuing Theory*. Amsterdam, The Netherlands: Elsevier, 1981.
[24] S. Boyd and L. Vandenberghe, *Convex Optimization*. New York, NY, USA: Cambridge Univ. Press, 2004.
[25] *CVX: MATLAB Software for Disciplined Convex Programming*. Accessed: Mar. 12, 2017. [Online]. Available: http://cvxr.com/cvx/
[26] G. Zeng, "Two common properties of the Erlang-B function, Erlang-C function, and Engset blocking function," *Math. Comput. Model.*, vol. 37, nos. 12–13, pp. 1287–1296, 2003.
[27] K. R. Krishnan, "The convexity of loss rate in an Erland loss system and sojourn in an Erlang delay system with respect to arrival and service rates," *IEEE Trans. Commun.*, vol. 38, no. 9, pp. 1314–1316, Sep. 1990.
[28] D. L. Jagerman, "Methods in traffic calculations," *AT T Bell Lab. Tech. J.*, vol. 63, no. 7, pp. 1283–1310, Sep. 1984.
[29] *Gurobi Optimizer Reference Manual*, Gurobi Optimization, Houston, TX, USA, 2016. [Online]. Available: http://www.gurobi.com
[30] M. Razaviyayn, M. Hong, and Z.-Q. Luo, "A unified convergence analysis of block successive minimization methods for nonsmooth optimization," *SIAM J. Optim.*, vol. 23, no. 2, pp. 1126–1153, 2013.

THINH DUY TRAN received the B.Eng. degree (Hons.) in computer science and engineering from the Ho Chi Minh City University of Technology, Vietnam, in 2012, and the M.S. degree in computer science and engineering from the Pohang University of Science and Technology, Pohang, South Korea, in 2014. He is currently pursuing the Ph.D. degree in telecommunications with the Institut National de la Recherche Scientifique-Energy, Materials, and Telecommunications Center, Université du Québec, Montréal, QC, Canada. His research interests include management algorithms for communications, caching, and computing resource allocation in next-generation wireless networks with a focus on wireless network virtualization. He has served as a reviewer in several IEEE journals and conferences, including the IEEE TRANSACTIONS ON WIRELESS COMMUNICATIONS, the IEEE ICC, the IEEE GLOBECOM, the IEEE VTC, and the IEEE PIMRC.

LONG BAO LE (S'04–M'07–SM'12) received the B.Eng. degree in electrical engineering from the Ho Chi Minh City University of Technology, Vietnam, in 1999, the M.Eng. degree in telecommunications from the Asian Institute of Technology, Thailand, in 2002, and the Ph.D. degree in electrical engineering from the University of Manitoba, Canada, in 2007. He was a Post-Doctoral Researcher with the Massachusetts Institute of Technology from 2008 to 2010 and the University of Waterloo from 2007 to 2008. Since 2010, he has been with the Institut National de la Recherche Scientifique, Université du Québec, Montréal, QC, Canada, where he is currently an Associate Professor. His current research interests include smartgrids, cognitive radio, radio resource management, network control and optimization, and emerging enabling technologies for 5G wireless systems. He is a co-author of the books *Radio Resource Management in Multi-Tier Cellular Wireless Networks* (Wiley, 2013) and *Radio Resource Management in Wireless Networks: An Engineering Approach* (Cambridge University Press, 2017). He is a member on the Editorial Board of the IEEE TRANSACTIONS ON WIRELESS COMMUNICATIONS and the IEEE COMMUNICATIONS SURVEYS AND TUTORIALS. He has served as the Technical Program Committee Chair/Co-Chair for several IEEE conferences, including the IEEE WCNC, the IEEE VTC, and the IEEE PIMRC.