Efficient Computation Resource Management in Mobile Edge-Cloud Computing

Yongmin Zhang, *Member, IEEE*, Xiaolong Lan, *Student Member, IEEE*, Yue Li, *Student Member, IEEE*, Lin Cai, *Senior Member, IEEE*, and Jianping Pan, *Senior Member, IEEE*

***Abstract*—We study the computation resource management problem in mobile edge-cloud computing networks. Mobile edge servers shall first satisfy the computation requirements of mobile users and Internet of Things (IoT) devices, and then wholesale redundant computation resources to the cloud networks to maximize their profit. Due to the coarse time granularity of wholesales, computation resource buyback may happen occasionally to deal with traffic bursts. Thus, the mobile edge servers need to make a tradeoff between the wholesale profit and the buyback cost. In this paper, the computation resource management problem is modeled as profit maximization. To solve this problem, we first analyze the relationship among the reserved computation resources, the computation tasks of mobile users and IoT devices, and the buyback cost. Then, we design an efficient wholesale scheme to determine the amount of the wholesaled computation resources, by which the total expected profit of the mobile edge server can be maximized. Given the reserved computation resources, we also propose a fast-convergent realtime buyback scheme for mobile edge servers to minimize the buyback cost. Finally, the simulation results show that our proposed efficient wholesale and buyback scheme can increase the total profit while guaranteeing the computation delay of all the computation tasks, especially when the computation workloads are time-varying.**

***Index Terms*—Cloud networks, computation resources, Internet of Things (IoT), mobile edge server, profit maximization.**

I. Introduction

With the development of mobile applications and the Internet of Things (IoT), various new applications have appeared and blossomed in recent years, e.g., e-Health, virtual reality, autonomous driving, natural language processing, interactive gaming, and augmented reality [1]–[5]. The quality of experience (QoE) of these emerging applications heavily relies on the underneath communication and computation platform, which brings a huge challenge to the design of mobile devices, especially for the size-limited and low-power IoT devices. Furthermore, considering the limited network resources and the increasing computation requirements, processing all the computation tasks locally or at a remote server may be costly and inefficient. One promising solution is to introduce mobile edge computing (MEC), which has computing capabilities and provides an IT service environment at the edge of networks to take on some computation tasks. Using MEC, not only the requirements of IoT devices on computation capability and power supply can be reduced, but also the computation latency of these tasks can be shortened [6].

In MEC, mobile users and IoT devices can upload their computation tasks, especially the latency-sensitive ones, to mobile edge servers, and the mobile edge servers will process the received computation tasks locally and then send the results back to the mobile users and IoT devices. This process is referred to as mobile computation offloading [7]. By now, MEC has been extensively studied, including system architecture [8]–[11], energy management [12]–[16], data transmission [17]–[22], computation resources optimization [23]–[27], and operation efficiency [28]–[32]. Most of the existing works focused on the design of compatible mobile edge servers and efficient computation task processing protocols. A well-established MEC system can support more mobile and IoT applications. However, the profitability of the MEC system, which is important for the wide deployment of MEC, still lacks due attention.

Generally, the construction and maintenance costs of an MEC system are high due to its high requirements on both hardware and software and it has to be widely deployed to gain profit. It is difficult for mobile edge servers to generate enough profit in a short term, especially when solely relying on the limited and time-varying computation tasks of the mobile users and IoT devices in their early stage. In this paper, we propose a cost-effective mobile edge-cloud computing system to increase the profit of mobile edge servers. In this system, each mobile edge server can divide its computation resources into two parts: one part is reserved to generate profit by processing the computation tasks of mobile users and IoT devices and the rest is wholesaled to the cloud networks as a flexible profit. Note that, when the reserved computation resources are not sufficient to fulfill the requirements of the local computation tasks, the mobile edge server needs to buy back some computation resources from the cloud networks

Manuscript received September 26, 2018; revised November 6, 2018; accepted November 24, 2018. Date of publication December 6, 2018; date of current version May 8, 2019. This work was supported in part by the Natural Sciences and Engineering Research Council of Canada, in part by the National Natural Science Foundation of China under Grant 61702450 and Grant 61629302, and in part by CSC (China Scholarship Council) and Compute Canada. *(Corresponding author: Lin Cai.)*

Y. Zhang, Y. Li, and L. Cai are with the Department of Electrical and Computer Engineering, University of Victoria, Victoria, BC V8W 3P6, Canada (e-mail: ymzhang@uvic.ca; liyue331@uvic.ca; cai@uvic.ca).

X. Lan is with the Department of Electrical and Computer Engineering, University of Victoria, Victoria, BC V8W 3P6, Canada, and also with the School of Information Science and Technology, Southwest Jiaotong University, Chengdu 611756, China (e-mail: xiaolonglan@my.swjtu.edu.cn).

J. Pan is with the Department of Computer Science, University of Victoria, Victoria, BC V8W 3P6, Canada (e-mail: pan@uvic.ca).

Digital Object Identifier 10.1109/JIOT.2018.2885453

at a higher buyback price. Therefore, there exists a tradeoff between the wholesale profit and the buyback cost. By designing an efficient computation resource management scheme, the total profit of the mobile edge server can be increased, especially when the computation tasks of the mobile users and IoT devices are time-varying.

In this paper, we formulate the computation resource management problem at the mobile edge server as profit maximization. Since the time granularities of the wholesale and the buyback are different, it is difficult to determine the wholesaled and the buyback computation resources simultaneously. To solve this problem, we first derive the minimal expected buyback cost by analyzing the relationship among the reserved computation resources, the distribution of computation workloads, and the buyback cost. Then, we prove that the profit maximization problem is convex with respect to the reserved computation resources at each time slot and design a Bisection method-based wholesale scheme to determine the amount of the wholesaled computation resources efficiently. Given the reserved computation resources, we design a fast-convergent realtime buyback scheme (RBS) to adjust the buyback computation resources according to the realtime computation workloads, such that the total buyback cost can be minimized. Finally, numerical simulations have been conducted to demonstrate the efficiency of our proposed algorithm. The contribution of our works can be summarized in the following.

1) We propose a cost-effective architecture for mobile edge-cloud computing networks, where each mobile edge server not only serves mobile users and IoT devices by taking on their computation tasks but also trades with the cloud networks by wholesaling and buying back computation resources based on the time-varying computation workloads.
2) We formulate the computation resource management problem of the mobile edge server as profit maximization and derive the minimal expected buyback cost given the reserved computation resources.
3) We propose an efficient wholesale and buyback scheme (EWBS) to determine the wholesaled and the buyback computation resources according to the expected and realtime computation workloads, respectively, such that the total profit of the mobile edge server can be maximized.
4) Simulation results show that the total profit of the mobile edge server can be increased by the proposed EWBS, especially when computation workloads are time-varying.

The rest of this paper is organized as follows. Section II presents the operation model of the mobile edge-cloud computing networks, and formulates the computation resource management problem as profit maximization. Section III analyzes the distribution of computation workloads from the mobile users and IoT devices and the relationship between the reserved computation resources and the expected buyback cost. In Section IV, an efficient wholesale scheme (EWS) is designed to determine the wholesaled computation resources for the mobile edge servers and an RBS is designed to determine the optimal buyback computation resources based on the realtime computation workloads. Section V demonstrates the operational performance based on simulation results. Finally, Section VI concludes this paper.

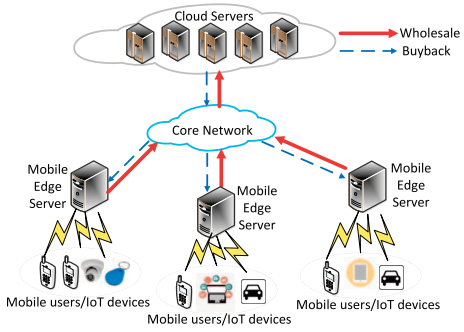


Fig. 1. Architecture of the mobile edge-cloud computing system.

II. System Model and Problem Formulation

Considering a mobile cloud-edge computing network, there are several mobile edge servers to process the computation tasks of mobile users, and IoT devices and cloud networks to collect computation resources from mobile edge servers. Each mobile edge server can serve as both a computation server for mobile users and IoT devices and a flexible computation unit for the cloud network. As a computation server, the computation tasks of mobile users and IoT devices should be satisfied with no compromise, while as a flexible computation unit, the computation resources that wholesaled to the cloud networks can only be adjusted after a given time duration. Given the randomness of computation tasks and the coarse time granularity of wholesales, the reserved computation resources may not be sufficient to complete all the arrived computation tasks in time, so the mobile edge server needs to buy back some computation resources from the cloud networks. Note that, the buyback price is usually higher than the wholesale price. The system model is shown in Fig. 1.

Generally, different mobile edge servers may have different limited computation resources and their computation tasks are various and time-varying. To satisfy all the computation tasks, each mobile edge server needs to make a decision on the reserved computation resources and the buyback computation resources. Our goal in this paper is to manage the computation resources of each mobile edge server to maximize the total profit of the mobile edge server by utilizing the computation resources efficiently. The definitions of the main notations can be found in Table I.

A. Service Model of Mobile Edge Servers

Let t denote the tth time slot, where one time slot is the smallest time duration for mobile edge servers to change the amount of the wholesaled computation resources. Divided one time slot into K time intervals and let k denote the kth time interval during one time slot. Here, one time interval is the smallest time duration for mobile edge servers to change the amount of the buyback computation resources from the

TABLE I
NOTATION DEFINITIONS

Symbol	Definition
a_1	The service price for processing a unit computation workload.
a_2	The profit for wholesaling a unit computation resource to the cloud networks.
c_1, c_2	The parameters for the buyback cost $g(\hat{C}^B_{e,k})$.
e	Mobile edge server e.
k	The k-th time interval.
k'	An upcoming time interval.
K	The total amount of time intervals in one time slot.
t	The t-th time slot.
$\lambda_{e,t}$	The average arrival rate of computation tasks at mobile edge server e during time slot t.
$C_{e,t}$	The total amount of the available computation resources at mobile edge server e during time slot t.
$C^I_{e,t}$	The total amount of the reserved computation resources at mobile edge server e during time slot t.
$C^C_{e,t}$	The total amount of the wholesaled computation resources at mobile edge server e during time slot t.
$\hat{C}_{e,k}$	The total amount of available computation resources at mobile edge server e during time interval k.
$\hat{C}^B_{e,k}$	The amount of the buyback computation resources at mobile edge server e during time interval k.
$\hat{C}^I_{e,k}$	The amount of the reserved computation resources at mobile edge server e during time interval k.
$\hat{D}_{e,k}$	The computation delay for computation workload $\hat{W}_{e,k}$ at mobile edge server e.
$\bar{D}$	The upper bound on the computation delay $\hat{D}_{e,k}$.
$f_X(x)$	The PDF of computation workloads in the queueing system.
$F_X(x)$	The CDF of computation workloads in the queueing system.
$g(\hat{C}^B_{e,k})$	The buyback cost of mobile edge server e for repurchasing $\hat{C}^B_{e,k}$ unit computation resources.
$\hat{Q}_{e,k}$	The unprocessed computation workloads at mobile edge server e at time interval k.
$\hat{R}_t$	The expected computation workload of each computation task.
$\hat{W}_{e,k}$	The amount of computation workloads that are arrived at mobile edge server e during time interval k.
$U^I_{e,t}$	The profit of mobile edge server e for providing servers during time slot t.
$U^S_{e,t}$	The profit of mobile edge server e for wholesaling computation resources during time slot t.
$U^B_{e,t}$	The buyback cost of mobile edge server e during time slot t.
$\bar{U}^B_{e,t}$	The expected value of $U^B_{e,t}$.
$\mathbf{U}_{e,t}$	The total profit of mobile edge server e during time slot t.

cloud networks.[1] At each time slot, each mobile edge server needs to determine the amount of the wholesaled computation resources, and at each time interval, the mobile edge server needs to determine the amount of the buyback computation resources.

At time slot t, let $C_{e,t}$ denote the total amount of the available computation resources at mobile edge server e, $C^I_{e,t}$ denote the amount of the reserved computation resources, and $C^C_{e,t}$ denote the amount of the wholesaled computation resources, respectively. We have

$$C_{e,t} \geq C^I_{e,t} + C^C_{e,t}. \tag{1}$$

It means that the sum of the reserved and the wholesaled computation resources cannot exceed the total available computation resources at each mobile edge server during any time slot. That is, because the mobile edge server may hold some computation resources for other purposes.

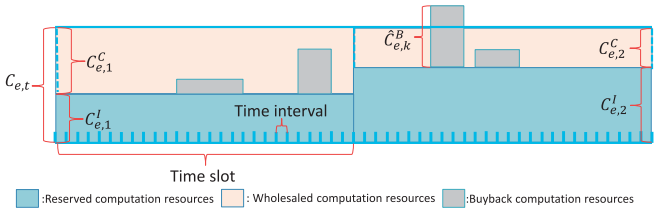


Fig. 2. Computation resource model of the mobile edge server.

The mobile edge server processes the computation tasks of mobile users and IoT devices under the first come first serve (FCFS) policy [33]. If the reserved computation resources $C^I_{e,t}$ are not sufficient to complete all the computation tasks in time, the mobile edge server needs to buy back some computation resources from the cloud networks. Let $\hat{C}^B_{e,k}$ denote the amount of the buyback computation resources at mobile edge server e from the cloud networks during time interval k. The total amount of available computation resources at mobile edge server e during time interval k can be given by

$$\hat{C}_{e,k} = \hat{C}^I_{e,k} + \hat{C}^B_{e,k}. \tag{2}$$

Here, $\hat{C}^I_{e,k} = C^I_{e,t}$ since the reserved computation resources are available during each time interval. The computation resource model of each mobile edge server is shown in Fig. 2. Note that, when one mobile edge server wholesales part of its computation resources to the cloud networks, the wholesaled computation resources will be dedicated to serve the computation tasks from the cloud networks and cannot be used by the mobile edge server without permission. When the mobile edge server buys back some computation resources from the cloud networks, part of the wholesaled computation resources can be released to the mobile edge server or some computation resources from other mobile edge servers or the cloud networks can be allocated to the mobile edge server. Thus, the buyback computation resources maybe are from its wholesaled computation resources or/and the computation resources of other mobile edge servers and the cloud networks. If the buyback computation resources are from its wholesaled computation resources, the computation tasks will be processed locally. Otherwise, part of the computation tasks will be transferred to other mobile edge servers or the cloud networks for processing purposes, such that the buyback process at the mobile edge server is realized.

Let $\hat{W}_{e,k}$ denote the amount of computation workloads that are arrived at mobile edge server e during time interval k and $\hat{Q}_{e,k}$ denote the unprocessed computation workloads at mobile edge server e at time interval k, respectively. We have

$$\hat{Q}_{e,k} = \max\left(0, \hat{Q}_{e,k-1} + \hat{W}_{e,k} - \hat{C}_{e,k}\right). \tag{3}$$

[1]Generally, one time slot is tens of minutes while one time interval is hundreds of milliseconds.

Since the mobile edge server processes the computation workloads under the FCFS policy, given the available computation resources $\{\hat{C}_{e,k'}, k' = k+1, k+2\ldots\}$ in upcoming time slot k', we can derive the computation delay, denoted by $\hat{D}_{e,k}$, by

$$\hat{D}_{e,k} = \begin{cases} m, & \text{if } \sum_{k'=k+1}^{k'+m-1} \hat{C}_{e,k} < \hat{Q}_{e,k} \leq \sum_{k'=k+1}^{k'+m} \hat{C}_{e,k} \\ 1, & \text{if } \hat{Q}_{e,k} \leq \hat{C}_{e,k+1}. \end{cases} \quad (4)$$

It can be found that, if the unprocessed computation workload $\hat{Q}_{e,k}$ is larger than $\sum_{k'=k+1}^{k+m-1} \hat{C}_{e,k}$ and no larger than $\sum_{k'=k+1}^{k+m} \hat{C}_{e,k}$, the unprocessed computation workload $\hat{Q}_{e,k}$ will be finished before/at time interval $k+m$. Thus, the computation delay for the computation tasks that are arrived at the mobile edge server during time interval k is m. Hence, the computation delay $\hat{D}_{e,k}$ depends on the available computation resources and the unprocessed computation workloads at the mobile edge server.

B. Computation Tasks of Mobile Users and IoT Devices

In this paper, we assume that the arrival of computation tasks at mobile edge server e follows a Poisson distribution with an expected value $\lambda_{e,t}$ during time slot t. Thus, the probability for n computation tasks at the mobile edge server is given by

$$P\{n\} = \frac{e^{\lambda_{e,t}} \left(\lambda_{e,t}\right)^n}{n!}, \quad n = 0, 1, 2, \ldots.$$

For each computation task, its computation workload follows an exponential distribution with an expected value of $\hat{R}_t$. In addition, there exists a deadline for completing the computation workloads that are uploaded to mobile edge server e during time slot t, denoted by $\bar{D}_{e,t}$. To guarantee the service quality of computation tasks at mobile edge server e, the following constraint should be satisfied:

$$\bar{D}_{e,t} \geq \hat{D}_{e,k}. \quad (5)$$

Since $\hat{D}_{e,k}$ depends on the available computation resources $\{\hat{C}_{e,k'}, k' \in [k+1, k+m]\}$, the mobile edge server needs to determine the amounts of the wholesaled and the buyback computation resources.

C. Profit Model of Mobile Edge Servers

In the mobile edge-cloud computing networks, the profit of each mobile edge server includes two parts: 1) the profit of processing computation tasks from mobile users and IoT devices and 2) the profit of wholesaling computation resources to the cloud networks. It means that the more computation tasks processed and more computation resources wholesaled, the more profit the mobile edge servers generated. However, due to the limited available computation resources at each mobile edge server, there exists a tradeoff between the amount of the reserved computation resources and the wholesaled ones.

Let a_1 denote the service price of processing a unit computation workload at mobile edge servers. The total expected profit of serving mobile users and IoT devices at mobile edge server e during time slot t, denoted by $U^I_{e,t}$, can be given by

$$U^I_{e,t} = a_1 \sum_{k=1}^{K} \hat{W}_{e,k}. \quad (6)$$

This is because all the computation workloads $\hat{W}_{e,k}$ at mobile edge server e will be processed in time.

Let a_2 denote the profit of wholesaling a unit computation resource to the cloud networks. The total expected profit of the wholesaled computation resources during time slot t, denoted by $U^S_{e,t}$, can be given by

$$U^S_{e,t} = a_2 K C^C_{e,t}. \quad (7)$$

To make sure all the computation tasks at the mobile edge server can be completed before their deadlines, the reserved computation resources may not be sufficient and some computation resources should be bought back from the cloud networks. Thus, the expenses of each mobile edge server include two parts: 1) the cost of maintaining the normal operation of the mobile edge server and 2) the cost of buying back computation resources from the cloud networks. Note that, the first part can be treated as a constant while the second part can be adjusted by managing the computation resources. Thus, we only consider the second part while omitting the first part in this paper for optimization purposes.

Let $g(\hat{C}^B_{e,k})$ denote the buyback cost of mobile edge server e for repurchasing $\hat{C}^B_{e,k}$ unit computation resources from the cloud networks during time interval k. To smooth the buyback computation resources, $g(\hat{C}^B_{e,k})$ usually is an increasing and convex function of $\hat{C}^B_{e,k}$ [34]. Furthermore, to guarantee the priority of the buyback computation resources, the buyback price is much higher than the wholesale price and the service price. Let $U^B_{e,t}$ denote the total buyback cost during time slot t. We have

$$U^B_{e,t} = \sum_{k=1}^{K} g\left(\hat{C}^B_{e,k}\right). \quad (8)$$

Specifically, we set $g(\hat{C}^B_{e,k}) = c_1 \hat{C}^B_{e,k} + c_2 (\hat{C}^B_{e,k})^2$ in this paper.

Let $\mathbf{U}_{e,t}$ denote the total profit of mobile edge server e during time slot t. According to the profit and the cost models, we have

$$\mathbf{U}_{e,t} = U^I_{e,t} + U^S_{e,t} - U^B_{e,t}. \quad (9)$$

It can be found that the total profit $\mathbf{U}_{e,t}$ depends on the profit of serving mobile users and IoT devices, the wholesaled computation resources and the cost of buying back computation resources from the cloud networks. Note that, both $U^I_{e,t}$ and $U^B_{e,t}$ should be estimated when making the decision on $C^I_{e,t}$.

D. Problem Formulation

In this paper, we intend to design an efficient computation resource management scheme for mobile edge servers to maximize their total profit by making a decision on the amounts of the wholesaled and the buyback computation resources at different time granularities, respectively. Since the computation resource management and the processing of

computation tasks at mobile edge servers are independent, we focus on the computation resource management for each mobile edge server.

Generally, mobile edge server e can make a decision on $\{C^I_{e,t}, \forall t\}$ at each time slot and then make a decision on $\{\hat{C}^B_{e,k}, \forall k\}$ at each time interval. The profit maximization problem for mobile edge server e can be formulated as

$$\text{P0}: \max_{C^I_{e,t}, \hat{C}^B_{e,k}} \sum_t \mathbf{U}_{e,t} \tag{10}$$

$$\text{s.t. } C_{e,t} \geq C^I_{e,t} + C^C_{e,t} \qquad \forall t \tag{11}$$

$$\bar{D}_{e,t} \geq \hat{D}_{e,k} \qquad \forall k, t \tag{12}$$

$$\hat{C}^B_{e,k} \geq 0 \qquad \forall k. \tag{13}$$

The objective is to maximize the total profit of the mobile edge server. The first constraint defines the available range of the reserved and the wholesaled computation resources. The second constraint ensures that all the computation tasks should be completed before their deadlines. The third constraint shows that, at each time interval, the mobile edge server can buy back computation resources from the cloud networks. It can be found that both of $C^I_{e,t}$ and $\hat{C}^B_{e,k}$ affect the total profit of the mobile edge server. However, since the time granularities of these variables are different, it is impossible to obtain the optimal solution for both of them simultaneously.

For the reserved computation resources $C^I_{e,t}$, we have the following lemma.

Lemma 1: The optimal reserved computation resources $C^I_{e,t}$ should satisfy $C^I_{e,t} \geq \min\{C_{e,t}, \lambda_{e,t}\hat{R}_t\}$.

Proof: To satisfy all the computation tasks, the mobile edge server needs to buy back some computation resources from the cloud networks when the reserved computation resources are not sufficient. From the definition of computation delay in (4), we have

$$\sum_{k'=k+1}^{k+\bar{D}_{e,t}} \hat{C}_{e,k} \geq \hat{Q}_{e,k} \tag{14}$$

to ensure $\bar{D}_{e,t} \geq \hat{D}_{e,k}$ for all the computation tasks arriving at the mobile edge server during time interval k. Thus, the amount of the buyback computation resources $\hat{C}^B_{e,k}$ should satisfy

$$\sum_{k+1}^{k+\bar{D}_{e,t}} \hat{C}^B_{e,k} \geq \max\left(0, \hat{Q}_{e,k} - \hat{C}^I_{e,k}\bar{D}_{e,t}\right). \tag{15}$$

If $\lambda_{e,t}\hat{R}_t \geq C^I_{e,t}$, it means that the arrival of computation workloads is higher than the service capability of the reserved computation resources. To guarantee that all the computation tasks are completed in time, the mobile edge server needs to buy back some computation resources at a higher price. Thus, the value of $U^S_{e,t} - U^B_{e,t}$ when $C^I_{e,t} > \lambda_{e,t}\hat{R}_t$ is larger than that when $C^I_{e,t} \leq \lambda_{e,t}\hat{R}_t$. However, the total amount of the computation resources at mobile edge server e is limited, $C^I_{e,t} = C_{e,t}$ holds when $\lambda_{e,t}\hat{R}_t \geq C_{e,t}$. Thus, we have $C^I_{e,t} \geq \min\{C_{e,t}, \lambda_{e,t}\hat{R}_t\}$. ■

In the following sections, we only analyze the system performance when $C^I_{e,t} > \lambda_{e,t}\hat{R}_t$. To solve this problem, we first analyze the distribution of computation workloads and the relationship among the computation workloads, the reserved computation resources and the minimal expected buyback cost. Then, we prove that the total profit is a concave function of the amount of the reserved computation resources and a convex function of the amount of the buyback computation resources based on the queueing model. At last, we propose an EWBS to determine the amount of the reserved computation resources based on the minimal expected buyback cost and an RBS given the reserved computation resources to minimize the buyback cost.

III. Queueing Model and Analysis

Generally, the computation task processing at the mobile edge server during one time slot can be treated as an independent queueing system. According to the queueing theorem, we first derive the distribution of the computation workloads at the mobile edge server during one time slot, and then analyze the relationship between the reserved computation resources and the minimal expected buyback cost.

A. Distribution of Computation Workloads

Given the reserved computation resources $C^I_{e,t}$, the computation task processing at mobile edge server e during time slot t can be modeled as an M/M/1 queueing system [35], in which the arrival process of mobile edge server e during time slot t follows a Poisson process with $\lambda_{e,t}$ while the service time of computation tasks at the mobile edge server follows an exponential distribution with $\mu_{e,t} = C^I_{e,t}/\hat{R}_t$. The probability that mobile edge server e has n computation tasks in its queueing system at any time slot t, denoted by $P^e_{n,t}$, can be given by

$$P^e_{n,t} = \left(\frac{\lambda_{e,t}}{\mu_{e,t}}\right)^n P^e_{0,t} \tag{16}$$

where

$$P^e_{0,t} = 1 - \frac{\lambda_{e,t}}{\mu_{e,t}}. \tag{17}$$

If the number of computation tasks is n, the amount of computation workloads follows a Gamma$(n, 1/\hat{R}_t)$ distribution. The corresponding probability density function (PDF) in the shape-rate parametrization is

$$f\left(x; n, 1/\hat{R}_t\right) = \frac{x^{n-1}e^{-\frac{x}{\hat{R}_t}}}{\left(\hat{R}_t\right)^n (n-1)!} \tag{18}$$

where $x \geq 0$.

Let X denote the total amount of computation workloads at mobile edge server e. The cumulative distribution function (CDF), denoted by $F_X(x)$, and the PDF, denoted by $f_X(x)$, of computation workloads in the queueing system is given by

$$\begin{aligned} F_X(x) &= Pr(X \leq x) \\ &= \sum_{n=0}^{\infty} P^e_{n,t} \int_0^x f\left(x; n, 1/\hat{R}_t\right) dx \\ &= P^e_{0,t} u(x) + \int_0^x \sum_{n=1}^{\infty} P^e_{n,t} \frac{x^{n-1}e^{-\frac{x}{\hat{R}_t}}}{\left(\hat{R}_t\right)^n (n-1)!} dx \end{aligned} \tag{19}$$

$$= \left(1 - \frac{\lambda_{e,t}}{\mu_{e,t}}\right)\left[u(x) + \frac{\lambda_{e,t}}{\lambda_{e,t} - \mu_{e,t}} e^{\left(\frac{\lambda_{e,t}}{\mu_{e,t}} - 1\right)\frac{x}{\hat{R}_t}}\right]$$

$$f_X(x) = \frac{\partial F_X(x)}{\partial x} = \left(1 - \frac{\lambda_{e,t}}{\mu_{e,t}}\right)\left[\delta(x) + \frac{\lambda_{e,t}}{\mu_{e,t}\hat{R}_t} e^{\left(\frac{\lambda_{e,t}}{\mu_{e,t}} - 1\right)\frac{x}{\hat{R}_t}}\right] \quad (20)$$

where $u(x)$ and $\delta(x)$ are step function and impulse response function, respectively. Note that, given any amount of computation workloads, we can obtain the corresponding PDF and CDF.

B. Buyback Model

If the computation workload is larger than the service capability of the reserved computation resources, part of the computation tasks cannot be completed in time. To guarantee the system performance, the mobile edge server needs to buy back some computation resources from the cloud networks. Comparing to the reserved computation resources, the amount of the buyback computation resources usually is low. We assume that the buyback computation resources will not affect the status of the queueing system.

Let X denote the amount of computation workloads that should be completed in time. The amount of the buyback computation resources at time interval k, denoted by $\epsilon_{e,k}$, satisfies

$$\epsilon_{e,k} \geq \begin{cases} X - C^I_{e,t}\bar{D}_{e,t}, & \text{if } X \geq C^I_{e,t}\bar{D}_{e,t} \\ 0, & \text{otherwise.} \end{cases} \quad (21)$$

It means that the amount of the buyback computation resources $\epsilon_{e,k}$ is an increasing function of the computation workloads X and a decreasing function of the reserved computation resources $C^I_{e,t}$. Given the distributions of computation workloads in (19) and (20), we can obtain the expected amount of the buyback computation resources during $[k+1, k+\bar{D}_{e,t}]$, denoted by $\bar{\epsilon}_{e,k}$, which can be given by

$$\bar{\epsilon}_{e,k} \geq \int_{C^I_{e,t}\bar{D}_{e,t}}^{\infty} f_X(x)\epsilon_{e,k}dx = \frac{\lambda_{e,t}\hat{R}_t}{\left(C^I_{e,t} - \lambda_{e,t}\hat{R}_t\right)^2} e^{\left(\lambda_{e,t} - \frac{C^I_{e,t}}{\hat{R}_t}\right)\bar{D}_{e,t}}. \quad (22)$$

It can be found that, $\bar{\epsilon}_{e,k}$ is a decreasing function of the reserved computation resources $C^I_{e,t}$. Since each computation task should be completed before its deadline $\bar{D}_{e,t}$, we have

$$\sum_{k'=k+1}^{k+\bar{D}_{e,t}} \hat{C}^B_{e,k} \geq \bar{\epsilon}_{e,k} \quad (23)$$

where k can be any time interval during time slot t. According to the buyback cost in (8), we have

$$U^B_{e,t} = \sum_{k=1}^{K} \int_{C^I_{e,t}\bar{D}_{e,t}}^{\infty} f_X(x) g\left(\hat{C}^B_{e,k}\right) dx. \quad (24)$$

To minimize the buyback cost, we have the following lemma.

Lemma 2: Given the reserved computation resources $C^I_{e,t}$, the minimal buyback cost, denoted by $\bar{U}^B_{e,t}$, is given by (25), shown at the bottom of this page.

Proof: Given the reserved computation resources $C^I_{e,t}$, the amount of the buyback computation resources satisfies (21) and the expected one can be given by (22). To minimize the buyback cost $U^B_{e,t}$, we have the following problem:

$$\min_{\hat{C}^B_{e,k}} \sum_{k=1}^{K} \int_{C^I_{e,t}\bar{D}_{e,t}}^{\infty} f_X(x) g\left(\hat{C}^B_{e,k}\right) dx$$
$$\text{s.t. Constraints (22) and (23)} \quad \forall k$$
$$\hat{C}^B_{e,k} \geq 0 \quad \forall k.$$

Given $C^I_{e,t}$, the value of $f_X(x)$ in (20) for any x is a constant. Since the objective function is an increasing and convex function of $\hat{C}^B_{e,k}$ and all the constraints are linear, the problem is a convex optimization problem. Since constraint (23) should be satisfied during any time duration $[k+1, k+\bar{D}_{e,t}]$, buying computation resources back equally is the best solution to minimize the total buyback cost. Thus, the minimal buyback cost can be given by (25). ■

Note that, for different reserved computation resources, the values of $\bar{U}^B_{e,t}$ are different.

IV. Optimal Reservation and Buyback Scheme

Based on the above analysis, we have derived the relationship between the reserved computation resources and the minimal expected buyback cost. In this section, we first design an EWS for the mobile edge server to manage its reserved and wholesale computation resources at each time slot, such that the total expected profit can be maximized. Then, based on the realtime information of computation workloads, we design an RBS to minimize the buyback cost.

A. Efficient Wholesale Scheme

Given the arrival of computation workloads, the profit of serving the mobile users and IoT devices is a constant according to (6). That is, because all the computation tasks should be completed in time. Thus, $U^I_{e,t}$ can be treated as a constant and Problem P0, shown in (10), can be written as the following:

$$\text{P1}: \max_{C^I_{e,t}, \hat{C}^B_{e,k}} \sum_t U^S_{e,t} - U^B_{e,t} \quad (26)$$

$$\bar{U}^B_{e,t} = \sum_{k=1}^{K} \int_{C^I_{e,t}\bar{D}_{e,t}}^{\infty} f_X(x) g\left(\frac{x - C^I_{e,t}\bar{D}_{e,t}}{\bar{D}_{e,t}}\right) dx = Ke^{\left(\lambda_{e,t} - \frac{C^I_{e,t}}{\hat{R}_t}\right)\bar{D}_{e,t}} \left(\frac{c_1\lambda_{e,t}\hat{R}^2_t}{\left(C^I_{e,t} - \lambda_{e,t}\hat{R}_t\right)\bar{D}_{e,t}} + \frac{2c_2\lambda_{e,t}\hat{R}^3_t C^I_{e,t}}{\left(\left(C^I_{e,t} - \lambda_{e,t}\hat{R}_t\right)\bar{D}_{e,t}\right)^2}\right) \quad (25)$$

$$\text{s.t. } C_{e,t} \geq C^I_{e,t} + C^C_{e,t} \quad \forall t \tag{27}$$
$$\text{Constraints (22) and (23)} \quad \forall k \tag{28}$$
$$\hat{C}^B_{e,k} \geq 0 \quad \forall k. \tag{29}$$

In this problem, the objective is to maximize the profit of wholesaling computation resources and minimize the buyback cost. The first constraint shows the available range for the reserved and the wholesaled computation resources at each time slot. The second and third constraints show the requirements on the amount of the buyback computation resources at each time interval.

According to Lemma 1, the reserved computation resources $C^I_{e,t}$ satisfies $C^I_{e,t} \geq \min\{C_{e,t}, \lambda_{e,t}\hat{R}_t\}$. Furthermore, according to Lemma 2, given the reserved computation resources $C^I_{e,t}$, the minimal expected buyback cost can be given by (25). By now, Problem P1 can be rewritten as

$$\text{P1\_1}: \max_{C^I_{e,t}} \sum_t U^S_{e,t} - \bar{U}^B_{e,t} \tag{30}$$
$$\text{s.t. } \min\left\{C_{e,t}, \lambda_{e,t}\hat{R}_t\right\} \leq C^I_{e,t} \leq C_{e,t} \quad \forall t. \tag{31}$$

It can be found that $C_{e,t} = C^I_{e,t}$ holds when $C_{e,t} \leq \lambda_{e,t}\hat{R}_t$. We only need to solve Problem P1_1 when $C_{e,t} > \lambda_{e,t}\hat{R}_t$.

By analyzing the relationship between the objective function $\sum_t U^S_{e,t} - \bar{U}^B_{e,t}$ and the reserved computation resources $C^I_{e,t}$, we have the following lemma.

Lemma 3: The objective function $\sum_t U^S_{e,t} - \bar{U}^B_{e,t}$ is a concave function of $C^I_{e,t}$.

Proof: Since the reserved computation resources $C^I_{e,t}$ at different time slots are independent, we just need to prove the concavity of $U^S_{e,t} - \bar{U}^B_{e,t}$ with respect to $C^I_{e,t}$.

1) $U^S_{e,t}$ is an increasing and linear function with respect to $C^C_{e,t}$ and $C^C_{e,t} \leq C_{e,t} - C^I_{e,t}$ always holds. To maximize the total profit, $C^C_{e,t} = C_{e,t} - C^I_{e,t}$ should be satisfied. Hence, $U^S_{e,t}$ is a decreasing and linear function of $C^I_{e,t}$.
2) When $C^I_{e,t} > \lambda_{e,t}\hat{R}_t$, we have the first and second derivative of $\bar{U}^B_{e,t}$ with respect to $C^I_{e,t}$, which are given by (32) and (33), shown at the bottom of this page. Since $[(\partial \bar{U}^B_{e,t})/(\partial C^I_{e,t})] < 0$ and $[(\partial^2 \bar{U}^B_{e,t})/(\partial C^{I^2}_{e,t})] > 0$, $\bar{U}^B_{e,t}$ is a decreasing and convex function of $C^I_{e,t}$.

In summary, the objective function $\sum_t U^S_{e,t} - \bar{U}^B_{e,t}$ is a concave function of $C^I_{e,t}$. ■

According to Lemma 3, Problem P1_1 is a convex optimization problem. Thus, there exists a unique optimal solution for Problem P1_1 [36]–[38], which satisfies the following theorem.

Theorem 1: For Problem P1_1, the global optimal solution for $C^I_{e,t}$ satisfies

$$\begin{cases} C^I_{e,t} = \lambda_{e,t}\hat{R}_t, & \text{if } \frac{\partial \bar{U}^B_{e,t}}{\partial C^I_{e,t}}|_{C^I_{e,t}=\lambda_{e,t}\hat{R}_t} > -a_2K \\ C^I_{e,t} = C_{e,t}, & \text{if } \frac{\partial \bar{U}^B_{e,t}}{\partial C^I_{e,t}}|_{C^I_{e,t}=C_{e,t}} < -a_2K \\ -\frac{\partial \bar{U}^B_{e,t}}{\partial C^I_{e,t}} - a_2K = 0, & \text{otherwise.} \end{cases} \tag{34}$$

Proof: Since $[(\partial^2 \bar{U}^B_{e,t})/(\partial C^{I^2}_{e,t})] > 0$, $[(\partial \bar{U}^B_{e,t})/(\partial C^I_{e,t})]$ is an increasing function of $C^I_{e,t}$. According to the extreme value theorem, $[(\partial(U^S_{e,t} - \bar{U}^B_{e,t}))/(\partial C^I_{e,t})] = -a_2K - [(\partial \bar{U}^B_{e,t})/(\partial C^I_{e,t})] = 0$ is a sufficient condition for the optimal solution of Problem P1_1. Thus, according to the value of $[(\partial \bar{U}^B_{e,t})/(\partial C^I_{e,t})]$, we can divide the optimal solution for $C^I_{e,t}$ into three cases.

1) $[(\partial \bar{U}^B_{e,t})/(\partial C^I_{e,t})] > -a_2K$ when $C^I_{e,t} = \lambda_{e,t}\hat{R}_t$.
2) $[(\partial \bar{U}^B_{e,t})/(\partial C^I_{e,t})] < -a_2K$ when $C^I_{e,t} = C_{e,t}$.
3) $[(\partial \bar{U}^B_{e,t})/(\partial C^I_{e,t})] \leq -a_2K$ when $C^I_{e,t} = \lambda_{e,t}\hat{R}_t$ and $[(\partial \bar{U}^B_{e,t})/(\partial C^I_{e,t})] \geq -a_2K$ when $C^I_{e,t} = C_{e,t}$.

For case i), the optimal solution is $\lambda_{e,t}\hat{R}_t$ since $[(\partial(U^S_{e,t} - \bar{U}^B_{e,t}))/(\partial C^I_{e,t})] < 0$ and $U^S_{e,t} - \bar{U}^B_{e,t}$ is a decreasing function of $C^I_{e,t}$.

For case ii), the optimal solution is $C_{e,t}$ since $[(\partial(U^S_{e,t} - \bar{U}^B_{e,t}))/(\partial C^I_{e,t})] > 0$ and $U^S_{e,t} - \bar{U}^B_{e,t}$ is an increasing function of $C^I_{e,t}$.

For case iii), $U^S_{e,t} - \bar{U}^B_{e,t}$ increases first and then decreases. According to the extreme value theorem, the maximal value will be obtained when $-[(\partial \bar{U}^B_{e,t})/(\partial C^I_{e,t})] - a_2K = 0$. ■

By calculating the value of $[(\partial \bar{U}^B_{e,t})/(\partial C^I_{e,t})]$, we can classify the optimal solution into one of these three cases in Theorem 1. For cases i) and ii), the optimal solution can be obtained directly. For case iii), an efficient wholesale algorithm should be designed to find the optimal solution.

Since $[(\partial^2 U^B_{e,t})/(\partial C^{I^2}_{e,t})] > 0$, $[(\partial U^B_{e,t})/(\partial C^I_{e,t})]$ is an increasing function of $C^I_{e,t}$ in $[\lambda_{e,t}\hat{R}_t, C_{e,t}]$. Bisection method is one of most efficient methods to search the optimal solution for a convex problem. Here, we design a Bisection method-based

$$\frac{\partial \bar{U}^B_{e,t}}{\partial C^I_{e,t}} = -\frac{K\lambda_{e,t}\hat{R}^2_t}{\bar{D}_{e,t}}\left[\frac{c_1\bar{D}_{e,t}}{\left(C^I_{e,t} - \lambda_{e,t}\hat{R}_t\right)\hat{R}_t} + \frac{c_1 + 2c_2C^I_{e,t}}{\left(C^I_{e,t} - \lambda_{e,t}\hat{R}_t\right)^2} + \frac{2c_2\left(C^I_{e,t} + \lambda_{e,t}\hat{R}_t\right)\hat{R}_t}{\left(C^I_{e,t} - \lambda_{e,t}\hat{R}_t\right)^3\bar{D}_{e,t}}\right]e^{\left(\lambda_{e,t} - \frac{C^I_{e,t}}{\hat{R}_t}\right)\bar{D}_{e,t}} < 0 \tag{32}$$

$$\frac{\partial^2 \bar{U}^B_{e,t}}{\partial C^{I^2}_{e,t}} = K\lambda_{e,t}\left[\frac{c_1\bar{D}_{e,t}}{C^I_{e,t} - \lambda_{e,t}\hat{R}_t} + \frac{c_1(1+\hat{R}_t) + 2c_2\hat{R}_tC^I_{e,t}}{\left(C^I_{e,t} - \lambda_{e,t}\hat{R}_t\right)^2} + \frac{2c_1 + 2c_2\left(C^I_{e,t} + \lambda_{e,t}\hat{R}_t + C^I_{e,t}\hat{R}^2_t + \lambda_{e,t}\hat{R}^3_t\right)}{\left(C^I_{e,t} - \lambda_{e,t}\hat{R}_t\right)^3\bar{D}_{e,t}} + \frac{4c_2\hat{R}_t\left(2\lambda_{e,t}\hat{R}_t + C^I_{e,t}\right)}{\left(C^I_{e,t} - \lambda_{e,t}\hat{R}_t\right)^4\bar{D}^2_{e,t}}\right]e^{\left(\lambda_{e,t} - \frac{C^I_{e,t}}{\hat{R}_t}\right)\bar{D}_{e,t}} > 0 \tag{33}$$

Algorithm 1: EWS

Input: $(\lambda_{e,t}, \hat{R}_t, \bar{D}_{e,t})$ for computation tasks, $(C_{e,t}, K, a_1, a_2, T)$ for the mobile edge server, and (c_1, c_2) for the cloud networks;
For each time slot t
  Set $l = \lambda_{e,t}\hat{R}_t$ and $r = C_{e,t}$;
  Calculate $L = \frac{\partial \bar{U}^B_{e,t}}{\partial C^I_{e,t}}|_{C^I_{e,t}=l}$ and $R = \frac{\partial \bar{U}^B_{e,t}}{\partial C^I_{e,t}}|_{C^I_{e,t}=r}$;
  If $L > -a_2K$
    Set $C^I_{e,t} = \lambda_{e,t}\hat{R}_t$;
  End if
  If $R < -a_2K$
    Set $C^I_{e,t} = C_{e,t}$;
  End if
  If $L \le -a_2K$ and $R \ge -a_2K$
    1) Calculate $L = \frac{\partial \bar{U}^B_{e,t}}{\partial C^I_{e,t}}|_{C^I_{e,t}=l}$ and $R = \frac{\partial \bar{U}^B_{e,t}}{\partial C^I_{e,t}}|_{C^I_{e,t}=r}$;
    2) Set $m = (l + r)/2$;
    3) Calculate $M = \frac{\partial \bar{U}^B_{e,t}}{\partial C^I_{e,t}}|_{C^I_{e,t}=m}$;
    If $M = -a_2K$
      $C^I_{e,t} = m$ and **Break**;
    End if
    If $M < -a_2K$
      $l = m$ and **return** to Step 12;
    End if
    If $M > -a_2K$
      $r = m$ and **return** to Step 12;
    End if
  End if
End for
Output: $\{C^I_{e,t}, C^C_{e,t} = C_{e,t} - C^I_{e,t} \forall t\}$;

algorithm to solve the problem P1_1 in case iii). The proposed EWS can be described in Algorithm 1.

It can be found that, at each time slot, the designed EWS checks the value of $[(\partial \bar{U}^B_{e,t})/(\partial C^I_{e,t})]$ at the end points of the feasible domain, and then determines the reserved and the wholesaled computation resources when $[(\partial \bar{U}^B_{e,t})/(\partial C^I_{e,t})]|_{C^I_{e,t}=m} = -a_2K$. By Algorithm 1, the optimal reserved and the wholesaled computation resources at each time slot have been obtained.

However, the amount of the buyback computation resources $\hat{C}^B_{e,k}$ is obtained based on the minimal expected buyback cost $\bar{U}^B_{e,t}$, which may not be the optimal solution for the realtime system. In the following part, we design a fast-convergent RBS for the mobile edge server to manage its buyback process at each time interval based on the realtime computation workloads.

B. Realtime Buyback Scheme

At the mobile edge server, the arrival of computation tasks and their computation workloads are time-varying, even if both of them follow the given distributions, respectively. To minimize the buyback cost, the mobile edge server needs to determine the RBS based on the realtime computation workloads.

Given the reserved computation resources $C^I_{e,t}$, Problem P0 can be rewritten as

$$\text{P2}: \min_{\hat{C}^B_{e,k}} \sum_t \sum_k g\left(\hat{C}^B_{e,k}\right) \tag{35}$$

$$\text{s.t. } \bar{D}_{e,t} \ge \hat{D}_{e,k} \quad \forall k, t \tag{36}$$

$$\hat{C}^B_{e,k} \ge 0 \quad \forall k. \tag{37}$$

According to the computation delay defined in (4), the computation delay $\hat{D}_{e,k}$ is a nonincreasing function of the available computation resources $\hat{C}_{e,k}$. Furthermore, constraint (36) can be rewritten as a linear constraint, as

$$\sum_{k'=\tau+1}^{\tau+\bar{D}_{e,t}} \hat{C}^B_{e,k'} \ge \hat{Q}_{e,k'} - \hat{C}^I_{e,k'}\bar{D}_{e,t} \quad \forall \tau \tag{38}$$

where $\tau \in [k - \bar{D}_{e,t}, k]$. Note that the values of $[\hat{C}^B_{e,\tau}, \forall \tau]$ are constants. By now, Problem P2 can be written as

$$\text{P2\_1}: \min_{\hat{C}^B_{e,k}} \sum_t \sum_k g\left(\hat{C}^B_{e,k}\right) \tag{39}$$

$$\text{s.t. } \sum_{k'=\tau+1}^{\tau+\bar{D}_{e,t}} \hat{C}^B_{e,k'} \ge \hat{Q}_{e,k'} - \hat{C}^I_{e,k'}\bar{D}_{e,t} \quad \forall \tau \tag{40}$$

$$\hat{C}^B_{e,k} \ge 0 \quad \forall k. \tag{41}$$

It can be found that Problem P2_1 is a convex optimization problem since the objective function is convex while all the constraints are linear [36]. Due to the convexity of $g(\hat{C}^B_{e,k})$, we have $g(\hat{C}^B_{e,k}) + g(\hat{C}^B_{e,k'}) \ge 2g((\hat{C}^B_{e,k} + \hat{C}^B_{e,k'})/2)$. Using the properties of convexity, we design a heuristic fast-convergent RBS, named as RBS, which is shown in Algorithm 2.

In the proposed RBS, at first, the computation resources are bought back equally to minimize the expected buyback cost, i.e.,

$$\hat{C}^B_{e,k'} = \frac{\hat{Q}_{e,k} - \hat{C}^I_{e,k}\bar{D}_{e,t}}{\bar{D}_{e,t}} \quad \forall k' \in \left[k+1, k+\bar{D}_{e,t}\right]. \tag{42}$$

There may exist a gap between the available computation resources and the required ones. Let $\Delta_{e,k'}$ denote the gap between them at time interval k'. We have

$$\Delta_{e,k'} = \left[\hat{Q}_{e,k'} - \hat{C}^I_{e,k}\bar{D}_{e,t} - \sum_{k''=\tau+1}^{\tau+\bar{D}_{e,t}} \hat{C}^B_{e,k''}\right]^+ \tag{43}$$

where $[\,\cdot\,]^+$ denotes $\max(0, \cdot)$. If $\Delta_{e,k'} > 0$, it means that the mobile edge server needs to increase the buyback computation resources before time interval k' by

$$\hat{C}_{e,k'} = \hat{C}_{e,k'} + \frac{\Delta_{e,k'}}{k' - k} \quad \forall k' \in \left[k+1, k'\right] \tag{44}$$

and reduce the buyback computation resources after time interval k' by

$$\hat{C}_{e,k'} = \hat{C}_{e,k'} - \frac{\Delta_{e,k'}}{k + \bar{D}_{e,t} - k'} \quad \forall k' \in \left[k'+1, k'+\bar{D}_{e,t}\right]. \tag{45}$$

Otherwise, the computation resources are sufficient to completed the unprocessed computation tasks in time. By Algorithm 2, the buyback computation resources at time intervals have been smoothened and the total buyback cost can be minimized.

Algorithm 2: RBS

Input: $(\hat{W}_{e,k}, \bar{D}_{e,t})$ for computation tasks, $(C^I_{e,t})$ for the edge server, and (c_1, c_2) for the cloud networks;
For each time interval k
  Set $\{\hat{C}^B_{e,k'}, \forall k' \in [k+1, k+\bar{D}_{e,t}]\}$ by (42);
  For time interval k'
    Calculate the resource gap $\Delta_{e,k'}$ by (43);
    If $\Delta_{e,k'} > 0$
      Update $\{\hat{C}_{e,k'}, \forall k' \in [k+1, k']\}$ by (44);
      Update $\{\hat{C}_{e,k'}, \forall k' \in [k'+1, k'+\bar{D}_{e,t}]\}$ by (45);
    End if
  End for
End for
Output: $\{\hat{C}^B_{e,k'}, \forall k' \in [k+1, k+\bar{D}_{e,t}]\}$;

Algorithm 3: EWBS

For each time slot t
  Calculate and output optimal $C^I_{e,t}$ and $C^C_{e,t}$ by Algorithm 1;
  For each time interval k
    Calculate and output optimal $\hat{C}^B_{e,k}$ by Algorithm 2;
  End for
End for.

C. Efficient Wholesale and Buyback Scheme

By now, the EWBS can be summarized as Algorithm 3, which has two different time granularities. At each time slot, the reserved and the wholesaled computation resources $C^I_{e,t}$ and $C^C_{e,t}$ can be calculated by Algorithm 1 based on the distribution of computation workloads and the minimal expected buyback cost. Then, given the reserved computation resources $C^I_{e,t}$, the buyback computation resources $\hat{C}^B_{e,k}$ can be updated by Algorithm 2 based on the realtime computation workloads. In this way, the total profit of the mobile edge server can be maximized.

V. Simulations

We evaluate the proposed EWBS in mobile edge-cloud computing networks and show some numerical results in this section. The simulation setting is given as follows. The total amount of the computation resources at the mobile edge server is $C_{e,t} = 3.2$ GHz, the expected computation workloads of each computation task is 100 Kb and each bit needs about 2000 cycles computation resources to be processed [39]. In this paper, one time slot is one hour and one time interval is 100 ms, respectively. Thus, $K = 3.6 \times 10^4$. The maximal computation delay for each computation task is 2 s and $\bar{D}_{e,t} = 2$ s/100 ms $= 20$. The arrivals of computation tasks during one second are $\lambda_{e,t} = \{3, 5, 10\}$ in nonrush, regular, and rush hours, respectively. The price for processing computation tasks is $a_1 = \$0.2764$/Gb, the price for wholesaling computation resources during one time slot is $a_2 = \$0.2487/(\text{GHz} \times \text{hour})$, and the price for buying back computation resources is $c_1 = \$7.6007 \times 10^{-5}/(\text{GHz} \times \text{s})$ and $c_2 = \$2.0729 \times 10^{-4}/(\text{GHz} \times \text{s})^2$, respectively.[2]

[2] a_1 is derived from the price for the pay-as-you-go service at Microsoft Azure, the wholesale price a_2 is $0.5a_1 = 0.5(a_1 \times 3600)/2000$, and the buyback price c_1 is $1.1a_2 = 1.1a_2/3600$ and c_2 is $3c_1$, respectively.

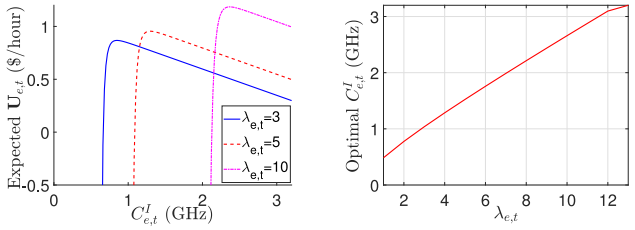


Fig. 3. Concavity of $\mathbf{U}_{e,t}$ with respect to $C^I_{e,t}$ and the optimal reserved computation resources $C^I_{e,t}$ for different arrivals of computation tasks $\lambda_{e,t}$. (a) Concavity of $\mathbf{U}_{e,t}$. (b) Optimal $C^I_{e,t}$.

We show the expected values of the proposed EWBS in Section V-A and compare the proposed EWBS with the following two schemes in Section V-B: 1) SC1, in which the mobile edge server does not wholesale/buy back computation resources to/from the cloud networks and denies the extra computation tasks when its available computation resources are not sufficient and 2) SC2, in which the mobile edge server does not wholesale any computation resources to the cloud networks and just buy back computation resources from the cloud networks to complete all the computation tasks in time.

A. Performance With Expected Computation Workloads

To demonstrate the performance of the proposed EWBS, we first use the numerical simulation to verify the concavity of the total profit $\mathbf{U}_{e,t}$ with respect to the reserved computation resources $C^I_{e,t}$ in Fig. 3(a). It can be found that, given the arrivals of computation tasks, the total profit $\mathbf{U}_{e,t}$ is a concave function of the reserved computation resources $C^I_{e,t}$. Furthermore, with the increase of the arrival rate of computation tasks, the available range for the reserved computation resources $C^I_{e,t}$ will be narrowed and the maximal total profit $\mathbf{U}_{e,t}$ will be increased. The proposed EWBS aims at finding the maximal $\mathbf{U}_{e,t}$ and the corresponding $C^I_{e,t}$. The relationship between the optimal reserved computation resources $C^I_{e,t}$ and the arrival rates of computation tasks $\lambda_{e,t}$ is shown in Fig. 3(b). The simulation results show that, with the increase of the arrival rate of computation tasks, the amount of the reserved computation resources increases. That is, because the buyback price is much higher than the wholesale price and more computation resources should be reserved to deal with the increased buyback cost. Thus, the mobile edge server needs to make a tradeoff between the wholesale profit and the buyback cost.

Beside the arrival of computation tasks, the maximal computation delay affects the total profit of the mobile edge server, since it determines the capability of the mobile edge server to deal with the volatility of computation tasks. With different maximal computation delays, the optimal reserved computation resources $C^I_{e,t}$ and the maximal total profit $\mathbf{U}_{e,t}$ are shown in Fig. 4. It can be found that, with the increase of the maximal computation delay, the maximal profit of the mobile edge server increases while the amount of the reserved computation resources decreases. Generally, the maximal computation delay is the time duration, in which the computation

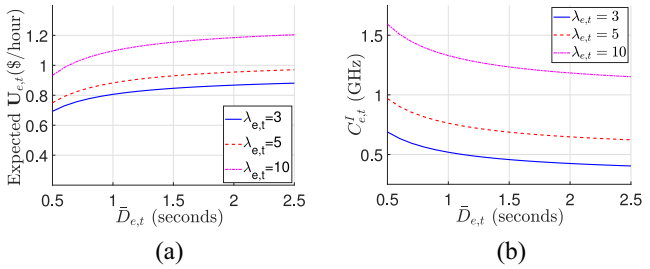


Fig. 4. Expected maximal profit $\mathbf{U}_{e,t}$ and the corresponding reserved computation resources $C^I_{e,t}$ with different maximal computation delay $\bar{D}_{e,t}$. (a) $\mathbf{U}_{e,t}$. (b) $C^I_{e,t}$.

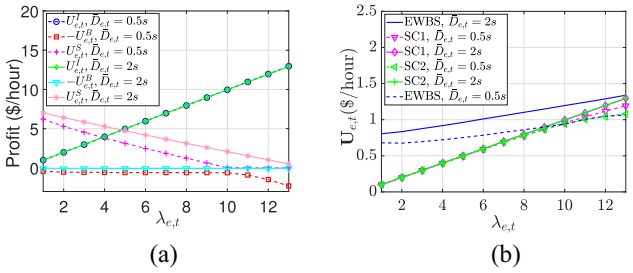


Fig. 5. Distribution of the maximal profit at the mobile edge server and the performance comparison with the SC1 and the SC2 schemes. (a) Profit. (b) Comparison with SC1 and SC2.

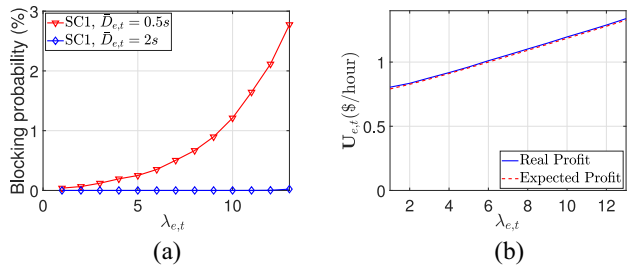


Fig. 6. Blocking probabilities for the SC1 scheme and the relationship between the expected and realtime profit at the mobile edge server. (a) Blocking probabilities for SC1. (b) Expected and realtime profit.

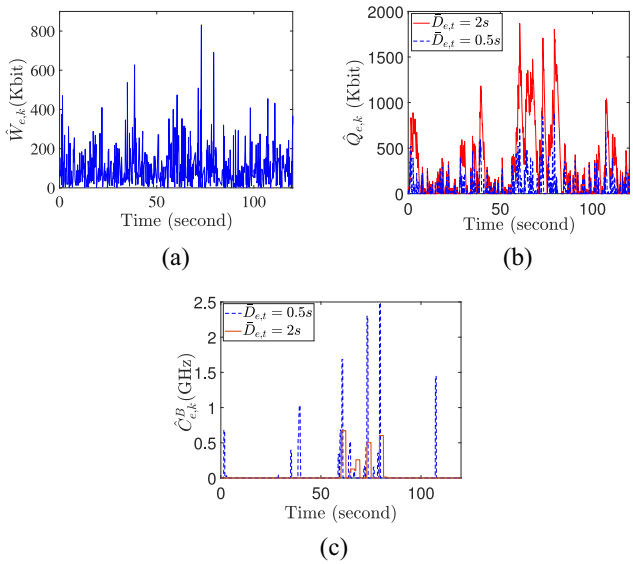


Fig. 7. Two-minute simulation results include the (a) arrival of computation workloads, (b) cumulative computation workloads, and (c) buyback computation resources.

tasks can be processed by the mobile edge server. With the increase of the maximal computation delay, the buyback computation resources can be smoothed and the buyback cost can be reduced. Furthermore, the mobile edge server can reserve fewer computation resources due to the low buyback cost, such that the total profit can be increased.

B. Performance With Realtime Computation Workloads

To evaluate the realtime performance of the proposed EWBS, each part of the maximal profit of the mobile edge server and the comparison with SC1 and SC2 schemes with $\bar{D}_{e,t} = 0.5s$ and $\bar{D}_{e,t} = 2s$ are shown in Fig. 5. It can be found that, with the increase of computation tasks, the profit by serving mobile users and IoT devices $U^I_{e,t}$ increases, but the profits for the wholesaled and the buyback computation resources, i.e., $U^S_{e,t}$ and $-U^B_{e,t}$, decrease. That is, because $U^I_{e,t}$ is a linear function of the completed computation workloads and all the accepted computation tasks have to been completed. However, to satisfy all the computation tasks, less computation resources can be wholesaled and more computation resources should be bought back, such that $U^I_{e,t}$ is an increasing function of $\lambda_{e,t}$ while $U^S_{e,t}$ and $-U^B_{e,t}$ are decreasing ones.

From Fig. 5(b), it can be found that, the proposed EWBS can increase the total profit comparing with the SC1 and the SC2 schemes (except when $\lambda_{e,t} > 9$ with $\bar{D}_{e,t} = 0.5s$), especially when the arrival rate of computation tasks is low. That is, because the proposed EWBS can make a better trade-off between the wholesale profit and the buyback cost than the SC1 scheme or the SC2 scheme. When $\lambda_{e,t} > 9$ with $\bar{D}_{e,t} = 0.5s$, the SC1 scheme generates the highest profit while the proposed EWBS generating similar profit with the SC2 scheme. That is, because the proposed EWBS and the SC2 scheme can guarantee the computation delay of all the computation tasks while the SC1 scheme denying some computation tasks when its computation resources are not sufficient, which may hinder the development of the MEC system.

The blocking probabilities of the SC1 scheme and the expected and the realtime profit at the mobile edge server are shown in Fig. 6. It can be found that, with the increase of computation tasks, the blocking probability for the mobile edge server with the SC1 scheme increases. Furthermore, when the maximal computation delay reduces, the blocking probability increases much faster than before. From Fig. 6(b), the gap between the expected profit and the realtime profit is very small. With the increase of computation tasks, the total profit increases.

The above numerical results show the performance of the proposed EWBS in one time slot. To demonstrate the realtime performance of the proposed EWBS in details, we extract two-minute simulation results when $\lambda_{e,t} = 5$ and show the results in Fig. 7. It can be found that, with the increase of the maximal computation delay $\bar{D}_{e,t}$, the cumulative computation workloads at the mobile edge server $\hat{Q}_{e,k}$ increase, but the amount of the buyback computation resources decreases. That is because,

when the maximal computation delay is large, the mobile edge server has a long time duration to reduce the effects of the computation tasks with large computation workloads. Given the gap between the expected and the realtime computation workloads, the buyback cost can be reduced by smoothing the buyback computation resources into a long time duration.

VI. Conclusion

In this paper, we proposed a cost-effective architecture for the mobile edge-cloud computing networks, in which the mobile edge server can wholesale and buy back computation resources to/from the cloud networks to maximize their profit while ensuring all the computation tasks can be completed in time. The computation resource management problem in mobile edge-cloud computing networks has been formulated as profit maximization at the mobile edge server. To solve this problem, we first analyzed the distribution of computation workloads at the mobile edge server and derived the relationship between the minimal expected buyback cost and the reserved computation resources. Then, we proposed an EWS based on the expected computation workloads and an RBS based on the realtime arrival of the computation workloads to maximize the total profit of the mobile edge server. The numerical simulations have been conducted to demonstrate the efficiency of the proposed EWBS, which can increase the total profit of the mobile edge server, especially when the computation tasks are time-varying.

Our proposed EWBS can improve the profitability of the mobile edge servers in mobile edge-cloud computing networks under FCFS policy. It would be interesting to extend the proposed algorithm to more general mobile edge-cloud computing networks with multiple classes of computation tasks and different service priorities, such that the QoE of mobile edge servers can be further improved. Another interesting extension is to joint consider the computation resource management among the mobile users, the mobile edge servers and the cloud networks with communication costs, such that both the utilization of computation resources and the QoE can be improved.

References

[1] N. Abbas, Y. Zhang, A. Taherkordi, and T. Skeie, "Mobile edge computing: A survey," *IEEE Internet Things J.*, vol. 5, no. 1, pp. 450–465, Feb. 2018.

[2] L. Cai, J. Pan, L. Zhao, and X. Shen, "Networked electric vehicles for green intelligent transportation," *IEEE Commun. Stand. Mag.*, vol. 1, no. 2, pp. 77–83, Jul. 2017.

[3] Y. Li, K. Sun, and L. Cai, "Cooperative device-to-device communication with network coding for machine type communication devices," *IEEE Trans. Wireless Commun.*, vol. 17, no. 1, pp. 296–309, Jan. 2018.

[4] J. Chen *et al.*, "Narrowband Internet of Things: Implementations and applications," *IEEE Internet Things J.*, vol. 4, no. 6, pp. 2309–2314, Dec. 2017.

[5] X. Peng *et al.*, "BOAT: A block-streaming app execution scheme for lightweight IoT devices," *IEEE Internet Things J.*, vol. 5, no. 3, pp. 1816–1829, Jun. 2018.

[6] Y. Mao, C. You, J. Zhang, K. Huang, and K. B. Letaief, "A survey on mobile edge computing: The communication perspective," *IEEE Commun. Surveys Tuts.*, vol. 19, no. 4, pp. 2322–2358, 4th Quart., 2017.

[7] S. E. Mahmoodi, R. N. Uma, and K. P. Subbalakshmi, "Optimal joint scheduling and cloud offloading for mobile applications," *IEEE Trans. Cloud Comput.*, to be published, doi: 10.1109/TCC.2016.2560808.

[8] N. Cheng *et al.*, "Air-ground integrated mobile edge networks: Architecture, challenges, and opportunities," *IEEE Commun. Mag.*, vol. 56, no. 8, pp. 26–32, Aug. 2018.

[9] H. Liu *et al.*, "Mobile edge cloud system: Architectures, challenges, and approaches," *IEEE Syst. J.*, vol. 12, no. 3, pp. 2495–2508, Sep. 2018.

[10] T. G. Rodrigues, K. Suto, H. Nishiyama, N. Kato, and K. Temma, "Cloudlets activation scheme for scalable mobile edge computing with transmission power control and virtual machine migration," *IEEE Trans. Comput.*, vol. 67, no. 9, pp. 1287–1300, Sep. 2018.

[11] J. Ren, H. Guo, C. Xu, and Y. Zhang, "Serving at the edge: A scalable IoT architecture based on transparent computing," *IEEE Netw.*, vol. 31, no. 5, pp. 96–105, Aug. 2017.

[12] W. Zhang *et al.*, "Energy-optimal mobile cloud computing under stochastic wireless channel," *IEEE Trans. Wireless Commun.*, vol. 12, no. 9, pp. 4569–4581, Sep. 2013.

[13] C. You, K. Huang, and H. Chae, "Energy efficient mobile cloud computing powered by wireless energy transfer," *IEEE J. Sel. Areas Commun.*, vol. 34, no. 5, pp. 1757–1771, May 2016.

[14] R. Deng, R. Lu, C. Lai, T. H. Luan, and H. Liang, "Optimal workload allocation in fog-cloud computing toward balanced delay and power consumption," *IEEE Internet Things J.*, vol. 3, no. 6, pp. 1171–1181, Dec. 2016.

[15] X. Lin, Y. Wang, Q. Xie, and M. Pedram, "Task scheduling with dynamic voltage and frequency scaling for energy minimization in the mobile cloud computing environment," *IEEE Trans. Surveys Comput.*, vol. 8, no. 2, pp. 175–186, Mar./Apr. 2015.

[16] J. Zhang *et al.*, "Energy-latency trade-off for energy-aware offloading in mobile edge computing networks," *IEEE Internet Things J.*, vol. 5, no. 4, pp. 2633–2645, Aug. 2018.

[17] M. Li, P. Si, and Y. Zhang, "Delay-tolerant data traffic to software-defined vehicular networks with mobile edge computing in smart city," *IEEE Trans. Veh. Technol.*, vol. 67, no. 10, pp. 9073–9086, Oct. 2018.

[18] Y. Gu, Z. Chang, M. Pan, L. Song, and Z. Han, "Joint radio and computational resource allocation in IoT fog computing," *IEEE Trans. Veh. Technol.*, vol. 67, no. 8, pp. 7475–7484, Aug. 2018.

[19] S. Sardellitti, G. Scutari, and S. Barbarossa, "Joint optimization of radio and computational resources for multicell mobile-edge computing," *IEEE Trans. Signal Process.*, vol. 1, no. 2, pp. 89–103, Jun. 2015.

[20] Q. Yuan *et al.*, "Toward efficient content delivery for automated driving services: An edge computing solution," *IEEE Netw.*, vol. 32, no. 1, pp. 80–86, Jan./Feb. 2018.

[21] T. G. Rodrigues, K. Suto, H. Nishiyama, and N. Kato, "Hybrid method for minimizing service delay in edge cloud computing through VM migration and transmission power control," *IEEE Trans. Comput.*, vol. 66, no. 5, pp. 810–819, May 2017.

[22] J. Ren, Y. Zhang, N. Zhang, D. Zhang, and X. Shen, "Dynamic channel access to improve energy efficiency in cognitive radio sensor networks," *IEEE Trans. Wireless Commun.*, vol. 15, no. 5, pp. 3143–3156, May 2016.

[23] X. Lyu, H. Tian, C. Sengul, and P. Zhang, "Multiuser joint task offloading and resource optimization in proximate clouds," *IEEE Trans. Veh. Technol.*, vol. 66, no. 4, pp. 3435–3447, Apr. 2017.

[24] Y. Wang, M. Sheng, X. Wang, L. Wang, and J. Li, "Mobile-edge computing: Partial computation offloading using dynamic voltage scaling," *IEEE Trans. Commun.*, vol. 64, no. 10, pp. 4268–4282, Oct. 2016.

[25] F. Wang, J. Xu, X. Wang, and S. Cui, "Joint offloading and computing optimization in wireless powered mobile-edge computing systems," *IEEE Trans. Wireless Commun.*, vol. 17, no. 3, pp. 1784–1797, Mar. 2018.

[26] J. Zheng, Y. Cai, Y. Wu, and X. S. Shen, "Dynamic computation offloading for mobile cloud computing: A stochastic game-theoretic approach," *IEEE Trans. Mobile Comput.*, to be published, doi: 10.1109/TMC.2018.2847337.

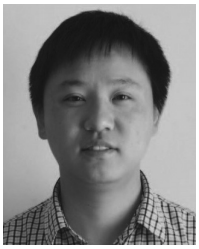
[27] Y. Wu, K. Ni, C. Zhang, L. Qian, and D. H. Tsang, "NOMA assisted multi-access mobile edge computing: A joint optimization of computation offloading and time allocation," *IEEE Trans. Veh. Technol.*, to be published, doi: 10.1109/TVT.2018.2875337.

[28] H. Guo, J. Liu, J. Zhang, W. Sun, and N. Kato, "Mobile-edge computation offloading for ultra-dense IoT networks," *IEEE Internet Things J.*, to be published, doi: 10.1109/JIOT.2018.2838584.

[29] Y. Kim, J. Kwak, and S. Chong, "Dual-side optimization for cost-delay tradeoff in mobile edge computing," *IEEE Trans. Veh. Technol.*, vol. 67, no. 2, pp. 1765–1781, Feb. 2018.

[30] X. Chen, "Decentralized computation offloading game for mobile cloud computing," *IEEE Trans. Parallel Distrib. Syst.*, vol. 26, no. 4, pp. 974–983, Apr. 2015.

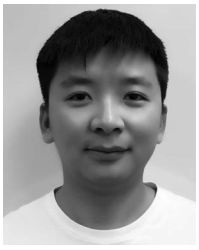
[31] X. Chen, L. Jiao, W. Li, and X. Fu, "Efficient multi-user computation offloading for mobile-edge cloud computing," *IEEE/ACM Trans. Netw.*, vol. 24, no. 5, pp. 2795–2808, Oct. 2016.
[32] J. Ren, Y. Guo, D. Zhang, Q. Liu, and Y. Zhang, "Distributed and efficient object detection in edge computing: Challenges and solutions," *IEEE Netw.*, vol. 32, no. 6, pp. 137–143, Nov./Dec. 2018.
[33] L. F. Bittencourt, J. Diaz-Montes, R. Buyya, O. F. Rana, and M. Parashar, "Mobility-aware application scheduling in fog computing," *IEEE Cloud Comput.*, vol. 4, no. 2, pp. 26–35, Mar./Apr. 2017.
[34] I. Menache, A. Ozdaglar, and N. Shimkin, "Socially optimal pricing of cloud computing resources," in *Proc. 5th Int. ICST Conf. Perform. Eval. Methodol. Tools (ICST)*, 2011, pp. 322–331.
[35] S. Sthapit, J. Thompson, N. M. Robertson, and J. Hopgood, "Computational load balancing on the edge in absence of cloud and fog," *IEEE Trans. Mobile Comput.*, to be published, doi: 10.1109/TMC.2018.2863301.
[36] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge, U.K.: Cambridge Univ. Press, 2004.
[37] Y. Zhang, J. Chen, L. Cai, and J. Pan, "EV charging network design with transportation and power grid constraints," in *Proc. IEEE INFOCOM*, 2018, pp. 2492–2500.
[38] Y. Zhang, S. He, and J. Chen, "Data gathering optimization by dynamic sensing and routing in rechargeable sensor networks," *IEEE/ACM Trans. Netw.*, vol. 24, no. 3, pp. 1632–1646, Jun. 2016.
[39] X. Hu, K.-K. Wong, and K. Yang, "Wireless powered cooperation-assisted mobile edge computing," *IEEE Trans. Wireless Commun*, vol. 17, no. 4, pp. 2375–2388, Apr. 2018.



Yongmin Zhang (S'12–M'15) received the Ph.D. degree in control science and engineering from Zhejiang University, Hangzhou, China, in 2015.

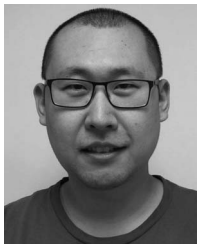
He was a Visiting Student with the California Institute of Technology, Pasadena, CA, USA. He is currently a Post-Doctoral Research Fellow with the Department of Electrical and Computer Engineering, University of Victoria, Victoria, BC, Canada. His current research interests include resource management and optimization in wireless networks, smart grid, and mobile computing.

Dr. Zhang was a recipient of the Best Paper Award of IEEE PIMRC 2012 and the IEEE Asia–Pacific Outstanding Paper Award 2018.



Xiaolong Lan (S'18) received the B.E. degree in mathematics and applied mathematics from the Chengdu University of Technology, Chengdu, China, in 2012. He is currently pursuing the Ph.D. degree at the School of Information Science and Technology, Southwest Jiaotong University, Chengdu.

Since 2017, he has been a visiting Ph.D. Student with the University of Victoria, Victoria, BC, Canada. His current research interests include buffer-aided communication and energy-harvesting wireless communication.



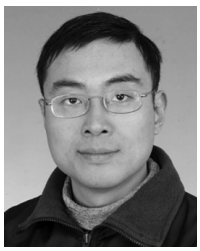
Yue Li (S'18) received the B.E. and M.E. degrees in electrical engineering from the Beijing Institute of Technology, Beijing, China, in 2006 and 2008, respectively, and the Ph.D. degree in electrical and computer engineering from the University of Victoria, Victoria, BC, Canada, in 2018.

From 2008 to 2013, he was a Standards Preresearch Engineer with the Wireless Research Department, Huawei, Shenzhen, China. He has been closely involved in 3GPP standards' evolvement and holds numerous patents in WCMDA, LTE-A, and 5G systems. He is currently a Post-Doctoral Fellow with the Department of Electrical and Computer Engineering, University of Victoria. His current research interests include next-generation cellular systems, wireless network design and optimization, and wireless system modeling and performance analysis.

Lin Cai (S'00–M'06–SM'10) received the M.A.Sc. and Ph.D. degrees in electrical and computer engineering from the University of Waterloo, Waterloo, ON, Canada, in 2002 and 2005, respectively.

Since 2005, she has been with the Department of Electrical and Computer Engineering, University of Victoria, Victoria, BC, Canada, where she is currently a Professor. Her current research interests include communications and networking, with a focus on network protocol and architecture design supporting emerging multimedia traffic over wireless, mobile, ad hoc, and sensor networks.

Dr. Cai was a recipient of the NSERC Discovery Accelerator Supplement Grants in 2010 and 2015, respectively, and the Best Paper Award of IEEE ICC 2008 and IEEE WCNC 2011. She has served as a TPC Symposium Co-Chair for IEEE Globecom'10 and Globecom'13, an Associate Editor for the IEEE TRANSACTIONS ON WIRELESS COMMUNICATIONS, the IEEE TRANSACTIONS ON VEHICULAR TECHNOLOGY, the *EURASIP Journal on Wireless Communications and Networking*, the *International Journal of Sensor Networks*, and the *Journal of Communications and Networks*. She is a Distinguished Lecturer of the IEEE VTS Society.



Jianping Pan (S'96–M'98–SM'08) received the bachelor's and Ph.D. degrees in computer science from Southeast University, Nanjing, China.

He was a Post-Doctoral Researcher with the University of Waterloo, Waterloo, ON, Canada. He is currently a Professor of computer science with the University of Victoria, Victoria, BC, Canada. He was also with the Fujitsu Labs, Tokyo, Japan, and NTT Labs, Tokyo. His area of specialization is computer networks and distributed systems, and his current research interests include protocols for advanced networking, performance analysis of networked systems, and applied network security.

Dr. Pan has been serving on the Technical Program Committee of major computer communications and networking conferences, including IEEE INFOCOM, ICC, Globecom, WCNC, and CCNC. He was the Ad Hoc and Sensor Networking Symposium Co-Chair of IEEE Globecom 2012 and an Associate Editor of the IEEE TRANSACTIONS ON VEHICULAR TECHNOLOGY. He is a Senior Member of the ACM.