Resource Allocation and Computation Offloading for Multi-Access Edge Computing With Fronthaul and Backhaul Constraints

Jun Chen, Zheng Chang, *Senior Member, IEEE*, Xijuan Guo, Renchuan Li, Zhu Han, *Fellow, IEEE*, and Timo Hämäläinen, *Senior Member, IEEE*

***Abstract*—Edge computing is able to provide proximity solutions for the future wireless network to accommodate different types of devices with various computing service demands. Meanwhile, in order to provide ubiquitous connectivities to massive devices over a relatively large area, densely deploying remote radio head (RRH) is considered as a cost-efficient solution. In this work, we consider a vertical and heterogeneous multi-access edge computing system. In the system, the RRHs are deployed for providing wireless access for the users and the edge node with computing capability can process the computation requests from the users. With the objective to minimize the total energy consumption for processing the computation task, a joint radio resource allocation and offloading decision optimization problem is presented under the explicit consideration of capacity constraints of fronthaul and backhaul links. Due to the non-convexity of the formulated problem, we divide the original problem into several sub-problems and address them accordingly to find the optimal solution. Extensive simulation studies are conducted and illustrated to evaluate the advantages of the proposed scheme.**

***Index Terms*—Multi-access edge computing, fronthaul/backhaul link, offloading, resource allocation.**

I. Introduction

A. Background and Motivation

IT IS estimated that the data generated by mobile devices will grow exponentially every year [1], which challenges the current architecture of cellular network. At the same time, how to provide more convenient, fast and efficient computing services for mobile devices has also become a major challenge for the wireless network evolution. Due to the insufficient computing capability and limited battery capacity of mobile devices, the processing of computing tasks often results in a large amount of delay and reduce the lifetime of devices, which inevitably affects the Quality of Experience (QoE). Over the past decade, cloud computing has emerged as a novel computing paradigm, where different computational and storage resources are provided to resource-constrained mobile devices in a centralized manner [2].

Nevertheless, the cloud computing servers are usually far away from mobile devices, which leads to the fact that the provided service may not be able to meet the stringent latency requirements of emerging mission-critical applications. Moreover, a large amount of frequency bandwidth will be consumed for the data delivery from mobile devices to the centralized cloud server. In order to overcome these limitations, multi-access edge computing (MEC) emerges as a promising solution that provides computational resource at the network edge by deploying distributed edge nodes (ENs) to access points or cellular base stations (BSs). In the MEC, mobile devices can offload computation-intensive and latency-critical tasks to the ENs for remote execution. The so-called computation offloading is able to provide ubiquitous computing services to the users by executing the offloaded tasks and returning the obtained results. By such, radio resources can be saved together with the reduced latency, in addition to the decreased cost and power consumption of devices [3], [4].

MEC has shown great potential to boost the development of future wireless networks. The service applications of MEC include computation offloading, collaborative computing, memory replication and content distribution, etc. With the emergence of MEC, the ability of transferring computation tasks from resource-constrained mobile devices to ENs is expected to support numerous new services and applications, such as augmented reality, IoT, autonomous vehicles and video processing. Meanwhile, in order to meet the requirement for massive wireless connections, the wireless network is experiencing densification and becoming heterogeneous. Different types of BSs and remote radio heads (RRHs) are deployed so that the wireless access services are becoming proximate to end users. The integration of heterogeneous networks and MEC are foreseeable to provide users both wireless access and computing services with improved Quality of Service (QoS). However, increasing the

Manuscript received February 4, 2021; revised May 5, 2021; accepted June 11, 2021. Date of publication June 17, 2021; date of current version August 13, 2021. This work was supported in part by NSFC under Grant 62071105, in part by NSF of Hebei under Grant E2017203351, and in part by the Key Research and Development Project of Hebei under Grants 19252106D, NSF EARS-1839818, CNS1717454, CNS-1731424, and CNS-1702850. The review of this article was coordinated by Prof. M. Peng. *(Corresponding author: Zheng Chang.)*

Jun Chen and Xijuan Guo are with the College of Information Science and Engineering, Yanshan University, Qinhuangdao 066004, China (e-mail: 2418507806@qq.com; xjguo@ysu.edu.cn).

Renchuan Li is with the Department of Joint Service, National Defence University, Beijing, China (e-mail: 3175529658@qq.com).

Zheng Chang and Timo Hämäläinen are with the Faculty of Information Technology, University of Jyväskylä, FIN-40014 Jyväskylä, Finland (e-mail: zheng.chang@jyu.fi; timo.t.hamalainen@jyu.fi).

Zhu Han is with the Department of Electrical and Computer Engineering, the University of Houston, Houston, TX 77004 USA, and also with the Department of Computer Science and Engineering, Kyung Hee University, Seoul 446-701, South Korea (e-mail: zhan2@uh.edu).

Digital Object Identifier 10.1109/TVT.2021.3090246

density of the BSs along with ENs may be a cost yet practical solution. On the other hand, integrating the computational resources, such EN, at BSs while the RRHs are used for wireless access seems an effective and efficient alternative. In such an architecture, BSs/ENs and RRHs are connected via the fronthaul, and the BS/ENs can connect to the cloud server via backhaul transmission for further task offloading if necessary.

While being considered as a potential solution for massive connections and ubiquitous computing, there are quite a few challenges ahead for realizing such a MEC architecture. How to associate the massive users with a number of RRHs need to be investigated. In addition, the computation offloading in the dense networks can result in unexpected transmission delay. The capacity constraint on the fronthaul and backhaul links are also seen as a bottleneck. All these limitations require careful design of computation offloading schemes about what and how to offload from the users to RRHs, and resource optimization schemes that coordinate different radio resources, such as power and frequency resources, among different layers.

B. Related Works

MEC introduces computing and storage resources to the networks by which the computation demand of the users can meet with required QoS [5]. Currently, many works have been dedicated to the computation offloading design and radio resource allocation [5], which aims to reduce processing delays and energy consumption of mobile devices [6]. The majority of research works have been done to study different problems on computation offloading [7]–[10], including energy minimization, delay minimization, resource allocation, and throughput maximization, so as to improve the user's QoS and the transmission capacity.

Some works have studied the trade-off between energy consumption and execution delay of computation offloading. In [11], the author has studied to minimize the execution delay of total tasks and energy consumption of mobile devices by jointly optimizing the offloading decision tasks and the CPU cycle frequency of mobile devices. In [8], the authors have proposed to optimize the offloading decision, and resource allocation so as to minimize the overall energy consumption of the whole system. The authors of [12] have developed a decision offloading framework to reduce response time and energy consumption in vehicular network. In [13], the authors have proposed an adaptive method for making offloading decisions, in which the objective is the weighted sum of energy and computation time. In order to improve the energy efficiency of latency-critical application, the authors of [14] have presented a user cooperation scheme in both computation and communication domains, which takes into account partial offloading and binary offloading models.

In addition to the investigation of MEC, there are increasing interest on the interplay of cloud computing and edge computing, in which the tasks can be handled by local devices, ENs or remote cloud servers. In [15], the authors have introduced a collaborative three-tier computing network which can utilize the vertical collaboration between mobile devices, edge nodes and cloud servers, as well as the horizontal collaboration between edge nodes. In order to improve the energy utilization in MEC and save energy consumption, there are a number of works investigating the energy harvesting effect [16]–[18]. The authors of [16] have proposed a wireless energy harvesting design for the multi-user MEC system, where the edge node is integrated with the multi-antenna access point (AP). The AP broadcasts wireless power which the users can utilize to execute the computation tasks locally or offload the tasks to the edge node. In this system, the authors propose to jointly optimize the energy beamforming, the central processing unit frequencies and the amount of offloaded task. The authors of [17] have studied a energy efficient system with wireless power transfer. Based on the Lyapunov optimization, a low complexity dynamic offloading algorithm is proposed to reduce the execution cost of delay and task failure. In [18], the authors have proposed a wireless power transfer solution that can enable computing to be performed at low-powered devices such as sensors and wearable computing devices. In terms of computing resource allocation, the authors of [19] have proposed a distributed optimization scheme for joint computation offloading and resource allocation in a MEC system. The optimal computation offloading strategy, uplink subchannel and transmit power allocation schemes are developed to jointly minimize the latency and energy consumption. In [20], the authors have presented a joint optimization problem of radio resources and computational resources, so as to minimize the energy consumption of all the users with delay constraint. In [21], the authors study the NOMA-based multi-access MEC system, and optimize the task offloading and time allocation. In [22], the authors study joint communication and computing resource allocation to minimize the delay of all the devices in a cloud-edge collaborative system. The authors of [23] propose to minimize the end-to-end delay required to complete the task by solving the problem of computing resource allocation and the joint design of C-RAN signal processing strategies.

Increasing the density of the BSs to meet stringent requirements of massive users lead to the research of exploring the potential of wireless backhaul and fronthaul links. To improve the system throughput, the authors propose to jointly optimized the wireless power and fronthaul transmission rate [24]. In [25], the authors consider the fronthaul and backhaul links when designing the computation offloading scheme, formulate an energy cost minimization problem with resource and latency constraints. In [26], the author optimizes the radio resource, computing resource and user scheduling in the uplink and downlink based on the constraints of limited backhaul capacity. The authors of [27] propose a computation offloading scheme for edge computing system, with explicit consideration of backhaul link capacity constraint.

C. Contribution

As we can observe, there are increasing interests investigating the joint optimization of radio resources and computation offloading in MEC. However, most of the works do not consider the coexistence of RRHs and ENs when providing wireless access, and ignore the capacity limitation of the associated fronthaul and backhaul links. Therefore, in this work, we tempt to explore this practical yet under-investigated MEC architecture and minimize the total energy consumption for offloading tasks when the

users have computation requests. Our major contributions are summarized as follows.

- In this work, we consider a vertical and heterogeneous MEC, where the users have computation service requests. In the considered system, the RRHs are deployed for providing wireless access for the users, and BS with EN can process the computation requests from the users. In addition, the tasks can be further offloaded to the cloud center if needed. The considered scenario is able to be widely applied to provide computing and wireless connection services in a dense network with massive devices.
- With the objective to minimize the total energy consumption for processing the computation task, a joint resource allocation and offloading decision optimization problem is presented under the explicit consideration of capacity constraints of fronthaul and backhaul links. The formulated problem concerns the power allocations of the users, RRHs and ENs, and the offloading decisions of users and ENs, and the computational resource allocation at ENs.
- Due to the non-convexity of the formulated problem, we divide the original problem into several sub-problems and address them accordingly to find the optimal solution. Extensive simulation studies are conducted and illustrated to evaluate the advantages of the proposed scheme. It is revealed that the users prefer to offload more data when the computation latency constraint becomes more stringent. It is also shown that the careful design of radio and computational resource allocation is needed for the MEC with the capacity constraints of the fronthaul and backhaul links.

D. Organization

The rest of this paper is organized as follows. Section II describes the system model. In Section III, we formulate the optimization problem, and introduce resource allocation and computation offloading solutions in Section IV. We demonstrate the benefits of our proposed algorithm in Section V through simulation study, and finally conclude this work in Section VI.

II. System Model

In this paper, we consider a heterogeneous MEC system as shown in Fig. 1. In the system, there are N RRHs deployed as the access points serving for a total number of U mobile users. When the users have computing tasks to be offloaded, the requests will be transmitted to the RRH. Typically, the RRH is only with access functions, and needs to forward the received requests to the ENs via wireless fronthaul. We consider there are in total M ENs. In addition to the functions for wireless access, the edge nodes (ENs), which have relatively rich computational resources, are integrated into the BSs and can process the computation requests from the users. If the requests cannot be processed at the EN, they will be delivered to the cloud for execution via the wireless backhaul.

We define the set of RRHs as $\mathcal{N} = \{1, \ldots, n, \ldots, N\}$, the set of users as $\mathcal{U} = \{1, \ldots, i, \ldots, U\}$, the set of ENs as $\mathcal{M} = \{1, \ldots, m, \ldots, M\}$. We assume that each user has computing tasks that can be described as $T_i = \{d_i, w_i, t_i^{max}\}$, $i \in \mathbb{U}$. d_i represents the data size of each computing task, w_i represents the total computational resources required to complete the computing task, and t_i^{max} is the maximum delay that the user can tolerate. For ease of reference, we also list the key notations of our system model in Table I.

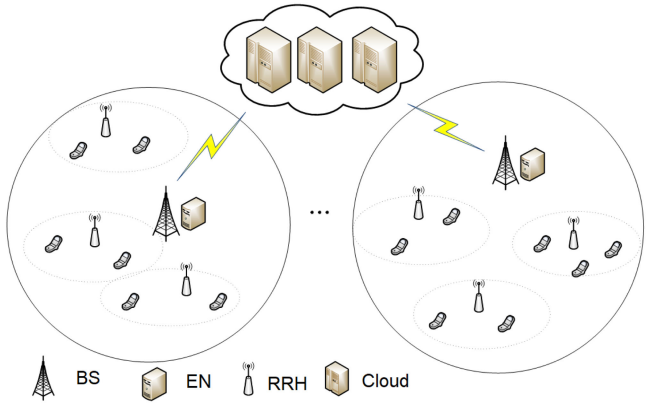


Fig. 1. Considered MEC system model.

TABLE I
Key Notations

Notations	Description
φ_i	the amount of energy consumed per second of users
V_i^l	the number of CPU cycles required to complete the task locally
V_i^f	the number of CPU cycles required to complete the task at EN
C_i^l	local computing capacity (CPU cycles per second) of user
p_i	the transmit power of user
θ_i	the portion of offloaded task
d_i	the data size of user's computation task
p_{i_n}	the transmit power of RRH n allocated for task of user i
p_{i_m}	the transmit power of BS m to the cloud for task of user i
χ_{i_m}	the offloading computing decision indicator of BS m
a_{i_n}	the user RRH association indicator
ω_m	the amount of energy consumed per second of servers
f_{i_m}	the portion of computational resources that EN assigns to user
F_m^M	the total CPU computing capacity of the EN
t_i^{max}	the maximum delay that the user can tolerate
w_i	the computational resources required to complete the computing task

A. Local Execution Model

To model the execution model, we first use $\theta_i d_i$ to represent the data size that has been offloaded to the EN to be processed, where $(0 \leq \theta_i \leq 1)$. Then $(1-\theta_i)d_i$ is the data size that can be executed locally. To successfully obtain the computed results of d_i, the amount of needed computational resource is w_i. Then, $V_i^l = (1-\theta_i)w_i$ is the number of CPU cycles required to complete the application locally. For user i, we assume local computing capacity is C_i^l (e.g. CPU cycles per second), and then we have the local execution time is

$$t_i^l = \frac{V_i^l}{C_i^l}. \tag{1}$$

We define φ_i as the energy unit of user, and the energy consumption for local computing is expressed as:

$$E_i^l = \frac{\varphi_i V_i^l}{C_i^l}. \tag{2}$$

B. Communication Model

We consider the wireless access between the users and RRHs, fronthaul and backhaul links are all OFDM-based. We first present the model of wireless link between the user and RRH, and then introduce the models of fronthaul and backhaul links.

1) Wireless Access Link: When an user has computing tasks to offload, it will choose one of RRHs to connect. The data rate of wireless access link between the user i and RRH can be expressed as:

$$r_i = \sum_{n=1}^{N} \alpha_{i_n} B_u \log_2 \left(1 + \frac{p_i g_{i,n}}{\sigma^2}\right), \tag{3}$$

where $a_{i_n} \in \{0, 1\}$ is defined an indicator for user-RRH association. $a_{i_n} = 1$ indicates that user i has chosen RRH n to offload tasks, otherwise $a_{i_n} = 0$. B_u is the channel bandwidth, p_i is the transmit power of user i, $g_{i,n}$ is the channel gain between i and RRH n, and σ^2 is noise variance. The transmission time can be expressed as:

$$t_i = \frac{\theta_i d_i}{r_i}. \tag{4}$$

The transmit energy consumption of user i for wireless access is $p_i \theta_i d_i / r_i$.

2) Fronthaul Link: After receiving the data from the users, RRH can forward the data to EN. The data rate from RRH n to EN m for delivering task of user i is expressed as:

$$r_{i_n} = B_r \log_2 \left(1 + \frac{p_{i_n} g_{n,m}}{\sigma^2}\right), \tag{5}$$

where p_{i_n} is the transmit power of RRH n for delivering task of user i, $g_{n,m}$ represents the channel gain between RRH n and EN m. Then the transmission time on fronthaul is

$$t_{i_n} = \frac{\theta_i d_i}{r_{i_n}}. \tag{6}$$

The transmit energy consumption of RRH n for delivering the task of user i is $p_{i_n} \theta_i d_i / r_{i_n}$.

3) Backhaul Link: In this work, we assume that computing tasks can be executed at ENs or cloud computing center. If the tasks are not executed in the EN, they will be transmitted via a backhaul link to the cloud center. We can express the data rate of backhaul link between ENs and cloud center as follows:

$$r_{i_m,c} = B_f \log_2 \left(1 + \frac{p_{i_m} g_{m,c}}{\sigma^2}\right), \tag{7}$$

where p_{i_m} is the transmit power of EN m to cloud, for transmitting task of user i and $g_{m,c}$ represents the channel gain between EN m and cloud. It should be noticed that the data rate on the fronthaul link should be limited by its capacity. We define $\chi_{i_m} \in \{0, 1\}$ as an offloading decision indicator for choosing EN or cloud computing for execution. $\chi_{i_m} = 1$ indicates that computing task of user i will be offloaded to EN m for computing. $\chi_{i_m} = 0$ indicates that the computing tasks of user i will be offloaded to the cloud for computing via EN m. Then, the transmission time of the backhaul link can be expressed as:

$$t_{i_m,c} = (1 - \chi_{i_m}) \frac{\theta_i d_i}{r_{i_m,c}}. \tag{8}$$

Correspondingly, the transmit energy consumption of EN m for delivering the task of user i is $p_{i_m} t_{i_m,c}$.

C. EN Execution Model

After computation tasks are offloaded, the EN allocates computational resources for task execution. We define F_m^M as the maximum computing capacity of the EN m, $f_{i_m} \geq 0$ represents the portion of computing capacity that EN m assigns to user i, and $\sum_{i=1}^{U} f_{i_m} \leq 1$. When (part of) task is executed at the EN, $V_i^f = \theta_i w_i$ represents the number of CPU cycles required for completing the task of user i, and the execution time is given as follow:

$$t_{i_m}^e = \chi_{i_m} \frac{V_i^f}{f_{i_m} F_m^M}. \tag{9}$$

Correspondingly, the energy consumption of task execution can be expressed as:

$$E_i^e = \chi_{i_m} \frac{\omega_m V_i^f}{f_{i_m} F_m^M}, \tag{10}$$

We define ω_m as the energy unit of EN servers. At cloud side, due to the rich computing resources of the cloud center, the execution time is quite short. we do not consider the execution time and the energy consumption on cloud if the task is offloaded to cloud [22], [23].

III. Problem Formulation

With the above analysis on the system model, the main objective of this work is to minimize the total energy consumption under the constraints of time delay, link capacities of fronthaul and backhaul and transmit power. The optimization variables include transmitting power of users $\mathbf{p}_u = \{p_i\}$, transmit power of RRH $\mathbf{p}_{RRH} = \{p_{i_n}\}$ and EN $\mathbf{p}_{EN} = \{p_{i_m}\}$, user association factor $\boldsymbol{\alpha} = \{\alpha_{i_n}\}$, task offloading decisions $\boldsymbol{\theta} = \{\theta_i\}$ and $\boldsymbol{\chi} = \{\chi_{i_m}\}$, and computational resource allocation $\mathbf{f} = \{f_{i_m}\}$. Accordingly, $\forall n \in \mathcal{N}$, $\forall i \in \mathcal{U}$, and $\forall m \in \mathcal{M}$, the problem can be formulated in **P1** as follows,

$$\begin{aligned}
\mathbf{P1}: \min_{\substack{\{\mathbf{p}_u, \mathbf{p}_{RRH}, \mathbf{p}_{EN}, \\ \boldsymbol{\alpha}, \boldsymbol{\theta}, \boldsymbol{\chi}, \mathbf{f}\}}} & \sum_{i=1}^{U} \left(\frac{\varphi_i V_i^l}{C_i^l} + \frac{p_i \theta_i d_i}{r_i}\right) + \sum_{i=1}^{U} \sum_{n=1}^{N} \frac{p_{i_n} \theta_i d_i}{r_{i_n}} \\
& + \sum_{i=1}^{U} \sum_{m=1}^{M} \left[\frac{(1 - \chi_{i_m}) \omega_m V_i^f}{f_{i_m} F_m^M} + \frac{\chi_{i_m} p_{i_m} \theta_i d_i}{r_{i_m,c}}\right], \\
s.t. \quad & C1: 0 \leq p_i \leq p_i^{max}, \\
& C2: 0 \leq \sum_{\mathrm{i}=1}^{N} a_{i_n} \leq 1, a_{i_n} \in \{0, 1\}, \\
& C3: 0 \leq \sum_{m=1}^{M} \chi_{i_m} \leq 1, \chi_{i_m} \in \{0, 1\},
\end{aligned}$$

$$
\begin{aligned}
&C4: 0 \le \sum_{i=1}^{U} f_{i_m} \le 1, 0 \le f_{i_m} \le 1,\\
&C5: 0 < \sum_{i=1}^{U} \alpha_{i,n} B_r \log_2\left(1+\frac{p_{i_n} g_{n,m}}{\sigma^2}\right) < C_n^F,\\
&C6: 0 < \sum_{i=1}^{U} \chi_{i_m} B_f \log_2\left(1+\frac{p_{i_m} g_{m,c}}{\sigma^2}\right) < C_m^B,\\
&C7: \max\{t_i^l, t_i + t_{i_n} + t_{i_m,c} + t_{i_m}^e\} \le t_i^{\max},\\
&C8: \sum_{i=1}^{U} p_{i_n} \le p_n^{\max}, p_{i_n} < p_n^{\max},\\
&C9: \sum_{i=1}^{U} p_{i_m} \le p_m^{\max}, p_{i_m} < p_m^{\max},\\
&C10: 0 \le \theta_i \le 1.
\end{aligned} \tag{11}
$$

C1 can ensure that the transmit power of user cannot exceed the maximum transmit power p_i^{max}. **C2** is the constraint on the user-RRH association indicator meaning that one user can only connect with one RRH when the user has task to be offloaded. **C3** ensures that the task of user can be executed by one EN or by the cloud center. **C4** indicates the constraints of computational resource allocation of EN. C_n^F in **C5** and C_m^B in **C6** are the maximum fronthaul and backhaul link capacity, respectively. **C7** guarantees the maximum transmission and computation delay for the offloading part. **C8** and **C9** ensure that the power allocation of RRH n and EN m can not exceed the maximum transmit power $p_n^{\max}$ and $p_m^{\max}$, respectively. **C10** is the portion of offloaded task of user i.

It can be found that **P1** is a non-convex problem. More preciously, it is a mixed integer programming problem because of the non-convexity of the objective function and constraints. Addressing such a problem is recognized as NP-hard. An exhaustive search is needed to obtain the global optimum, which leads to a high computational cost. In order to make the problem tractable and to simplify the problem, we divided the original problem into several sub-problems and solved them accordingly.

IV. Proposed Solution

In the following, we divide the original problem into several sub-problems, including task offloading, user association, resource allocation at RRHs and ENs, and address them accordingly.

A. Task Offloading Decision

First, we investigate the task offloading decision-making process, i.e. how to determine and select the optimal offloading decision θ_i to minimize the overall energy consumption. Based on the given variable $\{\mathbf{p}_u, \mathbf{p}_{RRH}, \mathbf{p}_{EN}, \boldsymbol{\alpha}, \mathbf{f}, \boldsymbol{\chi}\}$, the optimal $\boldsymbol{\theta}$ can be obtained by solving the following problem.

$$
\begin{aligned}
\mathbf{P2}: \min_{\{\boldsymbol{\theta}\}} &\sum_{i=1}^{U}\left(\frac{\varphi_i V_i^l}{C_i^l} + \frac{p_i \theta_i d_i}{r_i}\right) + \sum_{i=1}^{U}\sum_{n=1}^{N}\frac{p_{i_n}\theta_i d_i}{r_{i_n}} +\\
&\sum_{i=1}^{U}\sum_{m=1}^{M}\left[\frac{(1-\chi_{i_m})\omega_m V_i^f}{f_{i_m} F_m^M} + \frac{\chi_{i_m} p_{i_m}\theta_i d_i}{r_{i_m,c}},\right]\\
s.t.\quad & C7, C10.
\end{aligned} \tag{12}
$$

We can rewrite **C7** as

$$
C7': \ t_i^l \le t_i^{\max}, t_i + t_{i_n} + t_{i_m,c} + t_{i_m}^e \le t_i^{\max}. \tag{13}
$$

Then we have

$$
\theta_i \ge 1 - \frac{C_i^l t_i^{\max}}{V_i}, \tag{14}
$$

and

$$
\theta_i \le t_i^{\max} / \left(\frac{d_i}{r_i} + \frac{d_i}{r_{i_n}} + \frac{\chi_{i_m} d_i}{r_{i_m,c}} + \frac{(1-\chi_{i_m})V_i}{f_{i_m} F_m^M}\right). \tag{15}
$$

Thus, the domain of θ_i is given as following

$$
\theta_i(min) = \max\left(0, 1 - \frac{C_i^l t_i^{\max}}{V_i}\right), \tag{16}
$$

$$
\theta_i(max) = \min\left(\frac{t_i^{\max}}{\frac{d_i}{r_i} + \frac{d_i}{r_{i_n}} + \frac{\chi_{i_m} d_i}{r_{i_m,c}} + \frac{(1-\chi_{i_m})V_i}{f_{i_m} F_m^M}}, 1\right). \tag{17}
$$

Problem **P2** can be rewritten as following form:

$$
\begin{aligned}
\mathbf{P2}': \min_{\boldsymbol{\theta}} &\sum_{i=1}^{U}\left(\frac{\varphi V_i^l}{C_i^l} + \frac{p_i\theta_i d_i}{r_i}\right) + \sum_{i=1}^{U}\sum_{n=1}^{N}\frac{p_{i_n}\theta_i d_i}{r_{i_n}} +\\
&\sum_{i=1}^{U}\sum_{m=1}^{M}\left[\frac{(1-\chi_{i_m})\omega_m V_i^f}{f_{i_m} F_m^M} + \frac{\chi_{i_m} p_{i_m}\theta_i d_i}{r_{i_m,c}}\right]\\
s.t.\quad & C11: \theta_i(min) \le \theta_i \le \theta_i(max)
\end{aligned} \tag{18}
$$

$\mathbf{P2}'$ is a linear function about θ_i, so we can use some classical schemes, such as bisection method, to find the optimal value of θ_i.

B. User-RRH Association

In this subsection, we investigate the problem of User-RRH association and the optimal power allocation of the user. Accordingly, we need to address the following subproblem of **P1** given offloading decisions and the resource allocation at ENs and RRHs, i.e.,

$$
\begin{aligned}
\mathbf{P3}: \min_{\{\boldsymbol{\alpha}, \mathbf{p}_u\}} &\sum_{i=1}^{U} p_i \frac{\theta_i d_i}{\alpha_{i_n} B_u \log_2(1 + \frac{p_i g_{i,n}}{\sigma^2})},\\
s.t.\quad & C1, C2, C7'.
\end{aligned} \tag{19}
$$

When (part of) the tasks are offloaded for execution, the user can select a certain RRH for wireless connection. Since the objective function is an integer nonlinear programming problem with non-convex architecture, **P3** is a non-convex optimization

Algorithm 1: Propose Algorithm for Achieving q_i^*.

Set maximum tolerance δ;
while (!Convergence) **do**
  Solve **P4** for a given q_i and obtain user association and power allocation $\{\boldsymbol{\alpha}', \boldsymbol{p}_u'\}$;
  if $U(\boldsymbol{p}_u') - q_i R(\boldsymbol{\alpha}', \boldsymbol{p}_u') \leq \delta$ **then**
    Convergence = true;
    return$\{\boldsymbol{\alpha}^*, \boldsymbol{p}_u^*\} = \{\boldsymbol{\alpha}', \boldsymbol{p}_u'\}$ and obtain q_i^* by (20);
  else
    Convergence = false;
    returnObtain $q_i = U(\boldsymbol{p}_u')/R(\boldsymbol{\alpha}', \boldsymbol{p}_u')$;
  end if
end while

problem. First, the binary variable is relaxed and let $\alpha_{i_n} = [0, 1]$, which can be interpreted at resource sharing factor among multiple RRHs. Therefore, according to the fractional linear programming [30], we can transform it into a linear form to solve this problem.

First, we assume q_i is one solution of **P3**, and q_i^* is the optimal solution, as shown in the following:

$$q_i^* = \min_{\{\boldsymbol{\alpha}, \mathbf{p}_u\}} \frac{p_i^* \theta_i d_i}{\alpha_{i_n}^* B_u \log_2(1 + \frac{p_i^* g_{i,n}}{\sigma^2})}, \tag{20}$$

where p_i^* is the optimal power allocation, $\alpha_{i_n}^*$ is optimal association between the user and RRH. Then, we can obtain the following theorem.

Theorem 1: q_i can reach its optimal value if and only if the following conditions are satisfied.

$$\min_{\{\boldsymbol{\alpha}, \mathbf{p}_u\}} U(\mathbf{p}_u) - q_i R(\boldsymbol{\alpha}, \mathbf{p}_u) = 0, \tag{21}$$

where $U(\mathbf{p}_u) = p_i \theta_i d_i$ and $R(\boldsymbol{\alpha}, \mathbf{p}_u) = \alpha_{i_n} B_u \log_2(1 + \frac{p_i g_{i,n}}{\sigma^2})$.

Proof: Similar proof process can refer to Theorem 2 in [32].■

Theorem 1 gives a necessary and sufficient condition with respect to (w.r.t.) optimal power allocation and user association. Particularly, for the considered problem with an objective function in fractional form, there exists an equivalent optimization problem with an objective function in subtractive form, and both formulations result in the same solution. To achieve the optimal q_i^*, the iterative algorithm with guaranteed convergence can be applied [32] which is shown in Algorithm 1.

When we get the optimal solutions p_i^*, $\alpha_{i_n}^*$ and the optimal value q_i^*, it means that

$$p_i^* \theta_i d_i - q_i^* a_{i_n}^* B_u \log_2\left(1 + \frac{{p_i}^* g_{i,n}}{\sigma^2}\right) = 0. \tag{22}$$

From the constraints of **P3**, we can get

$$B_u \log_2\left(1 + \frac{p_i g_{i,n}}{\sigma^2}\right) \geq \frac{\varphi_i \theta_i d_i}{t_i^{\max} - \Delta T_i}, \tag{23}$$

where for simplicity, we have defined

$$\Delta T_i = \frac{\theta_i d_i}{r_{i_n}} + \chi_{im} \frac{\theta_i d_i}{r_{i_m}(p_{i_m})} + (1 - \chi_{im}) \frac{V_i^f}{f_{i_m} F_m^M}. \tag{24}$$

Therefore, $\mathbf{P3}$ can be transformed to the following problem:

$$\mathbf{P4}: \min_{\{\boldsymbol{\alpha}, \mathbf{p}_u\}} \sum_{i=1}^{U} \left(p_i \theta_i d_i - q_i a_{i_n} B_u \log_2\left(1 + \frac{p_i g_{i,n}}{\sigma^2}\right)\right),$$
$$s.t. \quad C1, C2, \tag{25}$$
$$C7'': B_u \log_2(1 + \frac{p_i g_{i,n}}{\sigma^2}) \geq \frac{\varphi_i \theta_i d_i}{t_i^{\max} - \Delta T_i}.$$

The objective function is a concave function w.r.t. p_i, as its Heisenberg matrix is semi-positive definite, and it is a linear function w.r.t. α_{i_n}. Therefore, the transformed problem **P4** is a convex optimization problem and satisfies the Slaters condition. Thus, it can be solved by using the Lagrange dual decomposition and subgradient method. Denoting $\boldsymbol{\Phi} = \{\rho_i, \gamma_i, \tau_i\}$, the Lagrange function can be expressed as:

$$\begin{aligned} &L(\mathbf{p}_u, \boldsymbol{\alpha}, \boldsymbol{\Phi}) \\ &= \sum_{i=1}^{U} (p_i \theta_i d_i - q_i a_{i_n} B_u \log_2(1 + p_i h_i)) + \sum_{i=1}^{U} \rho_i (p_i - p_i^{\max}) \\ &\quad + \sum_{i=1}^{U} \gamma_i [k_i - B_u \log_2(1 + p_i h_i)] + \sum_{i=1}^{U} \tau_i (a_{i_n} - 1), \end{aligned} \tag{26}$$

where $h_i = \frac{g_{i,n}}{\sigma^2}$ and $k_i = \frac{\varphi_i \theta_i d_i}{t_i^{\max} - \Delta T_i}$.

The optimal power allocation and the user association factor depend on dual variables $\boldsymbol{\Phi}$, which can be updated by solving the dual function of **P4**. Then we have

$$\begin{aligned} &\mathbf{P4}': \max_{\{\boldsymbol{\Phi}\}} \min_{\{\mathbf{p}_u, \boldsymbol{\alpha}\}} L(\mathbf{p}_u, \boldsymbol{\alpha}, \boldsymbol{\Phi}), \\ &s.t. \ \ \rho_i \geq 0, \gamma_i \geq 0, \tau_i \geq 0. \end{aligned} \tag{27}$$

From (26), we can minimize $L(\mathbf{p}_u, \boldsymbol{\alpha})$ for a given set of dual variables $\{\rho_i\}$, $\{\gamma_i\}$, $\{\tau_i\}$. Then, we can take the first derivation of p_i and a_{i_n} respectively.

$$\frac{\partial L}{\partial p_i} = \theta_i d_i - \frac{q_i a_{i_n} B_u h_i}{(1 + p_i h_i) \ln 2} + \rho_i - \gamma_i \frac{B_u h_i}{(1 + p_i h_i) \ln 2}, \tag{28}$$

$$\frac{\partial L}{\partial a_{i_n}} = \tau_i - q_i B_u \log_2(1 + p_i h_i). \tag{29}$$

Let $\frac{\partial L}{\partial p_i} = 0$ and $\frac{\partial L}{\partial a_{i_n}} = 0$, the optimal power allocation is obtained as follows:

$$p_i^* = \frac{2^{\frac{\tau_i}{q B_u}} - 1}{h_i}. \tag{30}$$

Substituting p_i^* into (29), and we can get the optimal a_{i_n}:

$$a_{i_n}^* = \frac{[(\theta_i d_i + \rho_i) \ln 2] 2^{\frac{\tau_i}{q_i B_u}} - B_u \gamma_i}{q B_u}. \tag{31}$$

We can use subgradient method to find the optimal dual variables, i.e.

$$\nabla \rho_i = {p_i}^* - p_i^{\max}, \tag{32}$$

$$\nabla \gamma_i = \frac{\varphi_i \theta_i d_i}{t_i^{\max} - \Delta T_i} - B_u \log_2\left(1 + \frac{p_i g_{i,n}}{\sigma^2}\right), \tag{33}$$

$$\nabla \tau_i = a_{i_n} - 1. \tag{34}$$

Algorithm 2: User Association and Power Allocation Algorithm.

Initialization: dual variables, $p_i^{\max}$, q_i, and maximum tolerance δ.
while !Convergence **do**
  power allocation p_i and association according (30).
  Update dual variables according to (35), (36), (37).
  if $U(\boldsymbol{p}'_u) - q_i R(\boldsymbol{\alpha}', \boldsymbol{p}'_u) \leq \delta$ **then**
    Convergence=true;
    $\{\boldsymbol{\alpha}^*, \boldsymbol{p}_u^*\} = \{\boldsymbol{\alpha}', \boldsymbol{p}'_u\}$, $a_{i_n} = {a_{i_n}}'$, and obtain q_i^*.
  else
    Convergence=false;
    $q_i = \frac{U(\boldsymbol{p}'_u)}{R(\boldsymbol{\alpha}', \boldsymbol{p}'_u)}$.
  end if
end while
returnObtain user association and power allocation policies.

The Lagrangian multiplication is updated with the subgradient method in the following way,

$$\rho_i(l+1) = [\rho_i(l) - \varepsilon_{\rho_i}(l)\nabla\rho_i(l)]^+, \tag{35}$$

$$\gamma_i(l+1) = [\gamma_i(l) - \varepsilon_{\gamma_i}(l)\nabla\gamma_i(l)]^+, \tag{36}$$

$$\tau_i(l+1) = [\tau_i(t) - \varepsilon_{\tau_i}(l)\nabla\tau_i(l)]^+, \tag{37}$$

where l is the number of iterations, $\varepsilon_{\rho_i}(l), \varepsilon_{\gamma_i}(l), \varepsilon_{\tau_i}(l)$ are the corresponding step sizes. The Lagrange multiplier is iteratively updated until the condition is satisfied. The proposed user association and power allocation algorithm is summarized in Algorithm 1. In this algorithm, we first define a sufficiently small positive real number. If convergence is not achieved, the loop continues until the optimal solution is reached. At each iteration, the dual variables are updated according to (35), (36), and (37). The power allocation is derived from (30). If ${p_i}'\theta_i d_i - qB_u\log_2(1+\frac{{p_i}' g_{i,n}}{\sigma^2}) \leqslant \delta$, then the convergence has been reached and the optimal value can be obtained [33]. The overall process is summarized in Algorithm 2.

The obtained $a_{i_n}^*$ is a relaxed value, and we can now recover the binary variable $a_{i_n}^*$. Substituting p_i and a_{i_n} into (29), when the partial derivative of a_{i_n} tends to zero, then $a_{i_n} = 1$. When $\frac{\partial L}{\partial a_{i_n}} \to \infty$, that is, the partial derivative of a_{i_n} tends to infinity, and then $a_{i_n} = 0$.

C. RRH Power Allocation

In this part, we will present the subproblem about power allocation of the RRH for delivering the task to the EN. Then, the problem can be shown as follows:

$$\mathbf{P5}: \min_{\{\mathbf{p}_{RRH}\}} \sum_{i=1}^{U}\sum_{n=1}^{N} p_{i_n}\frac{\theta_i d_i}{r_{i_n}(p_{i_n})}$$

$$s.t. C7''': \frac{\theta_i d_i}{r_i} + \frac{\theta_i d_i}{r_{i_n}(p_{i_n})} + \frac{(1-\chi_{i_m})\theta_i d_i}{r_{i_m,c}} + \frac{\chi_{i_m} V_i^f}{f_{i_m} F_m^M} < t_i^{\max},$$

$$C8: \sum_{i=1}^{U} p_{i_n} \leq p_n^{\max}, p_{i_n} < p_n^{\max}. \tag{38}$$

In **P5**, $r_{i_n}(p_{i_n}) = B_r\log_2(1+\frac{p_{i_n} g_{n,m}}{\sigma^2})$, and we assume that $f(p_{i_n}) = p_{i_n}\frac{\theta_i d_i}{B_r\log_2(1+\frac{p_{i_n} g_{n,m}}{\sigma^2})}$. For $f(p_{i_n})$, we have following lemma.

Lemma 1: $f(p_{i_n})$ is unimodal in its domain [34].

Proof: First, we can see that $f(x) = x(\theta_i d_i \frac{2^{\frac{1}{B_r x}}-1}{h})$ is convex, $x \geq 0$, $h = \frac{g_{i_n,m}}{\sigma^2}$, Because the Heisenberg matrix of function $f(x) = x(\theta_i d_i \frac{2^{\frac{1}{Bx}}-1}{h})$ is semi-positive definite, therefore, if function $f(x)$ has a minimum value, the minimum value is unique. In addition, let $p_{i_n} = \frac{2^{\frac{1}{B_r x}}-1}{h}$, $f(x)$ can be expressed as an inverse function of $f(\frac{1}{B_r\log_2(1+hp_{i_n})})$ with respect to p_{i_n}, i.e, $f(p_{i_n})$. In this process, what we do is to apply variable substitution, which is monotonous. Therefore, if $f(p_{i_n})$ has a minimum value, this minimum value is unique. In other words, $f(p_{i_n})$ is unimodal. ■

From Lemma 1, it can be found that the optimal value will be taken from three points, namely, two end points and the peak point of function $f(p_{i_n})$. From C8, we can obtain the following results.

$$p_{i_n} \geq \frac{2^{\frac{\theta_i d_i}{B_r(t_i^{\max}-\Delta t_i)}}}{h} = p_{i_n}^{\min}, \tag{39}$$

where $\Delta t_i = t_i^{\max} - [\frac{\theta_i d_i}{r_i} + (1-\chi_{i_m})\frac{\theta_i d_i}{r_{i_m,c}} + \chi_{i_m}\frac{V_i^f}{f_{i_m} F_m^M}]$, and the optimal closed form of p_{i_n} is:

$$p_{i_n}^* = \begin{cases} p_{i_n}^{\min}, p_{i_n}^{\min} > {p'}_{i_n}, \\ {p'}_{i_n}, p_{i_n}^{\min} \leq {p'}_{i_n} \leq p_{i_n}^{\max}, \\ p_{i_n}^{\max}, {p'}_{i_n} > p_{i_n}^{\max} \end{cases}$$

where p'_{i_n} represents the optimal transmit power when $f(p_{i_n})$ is minimized, i.e., $\nabla f(p_{i_n})|_{p'_{i_n}} = 0$.

D. Resource Allocation of EN

In this subsection, we address the problem at the EN, i.e, resource allocation of the EN and whether the task can be further offloaded to the cloud center. Based on the given resource allocation $\{\mathbf{p}_u, \boldsymbol{\theta}, \mathbf{p}_{RRH}, \boldsymbol{\alpha}\}$, the optimal resource allocation problem can be simplified to the following problem:

$$\mathbf{P6}: \min_{\{\mathbf{f}, \mathbf{p}_{EN}, \boldsymbol{\chi}\}} \sum_{i=1}^{U}\sum_{m=1}^{M}\left[\frac{(1-\chi_{i_m})\omega_m V_i^f}{f_{i_m} F_m^M} + \frac{\chi_{i_m} p_{i_m}\theta_i d_i}{r_{i_m,c}(p_{i_m})}\right],$$

$$s.t. \quad C3, C4, C6, C7,' C9. \tag{40}$$

As defined, χ_{i_m} is a binary variable. To address this problem, we relax χ_{i_m} to be [0,1] instead of a Boolean, which can be interpreted as the computing sharing factor for task execution. First, we solve the problem of computational resource allocation f_{i_m}, which means to address the following problem.

$$\mathbf{P7}: \min_{\{\mathbf{f}\}} \sum_{i=1}^{U} \frac{(1-\chi_{i_m})\omega_m V_i^f}{f_{i_m} F_m^M},$$

$$s.t. \quad C4, C7'. \tag{41}$$

It can be found that **P7** is a convex optimization problem, the optimal allocation $f_{i_m}^*$ can be obtained by using the Lagrange dual decomposition. The Lagrange function of **P7** is shown as follow:

$$L(\mathbf{f},\mathbf{\Psi}) = \sum_{i=1}^{U}\frac{(1-\chi_{i_m})\omega_m V_i^f}{f_{i_m}F_m^M} + \beta\left(\sum_{i=1}^{U} f_{i_m} - 1\right) + \sum_{i=1}^{U}\mu_i\left(t_{i,delay} + \frac{(1-\chi_{i_m})V_i^f}{f_{i_m}F_m^M} - t_i^{\max}\right), \tag{42}$$

where $\mathbf{\Psi} = \{\beta, \mu_i\}$. $\beta \geq 0$ and $\mu_i \geq 0$ are Lagrange multipliers associated with different constraints, and $t_{i,delay} = t_i^l + t_i + t_{i_n} + t_{i_m,c}$. Accordingly, the dual problem is

$$\max_{\mathbf{\Psi}} \min_{\mathbf{f}} L(\mathbf{f},\mathbf{\Psi}). \tag{43}$$

Similar to the procedure of addressing **P4**, the dual problem (43) is decomposed into two layers, where minimization of (42) is the inner problem and maximization of (43) is the outer problem. The dual problem can be addressed by solving these two problems iteratively. In each iteration, the inner problem about optimal computational allocation is solved by using the Karush-Kuhn-Tucker (KKT) conditions for a set of Lagrange multipliers with the fixed value, and the outer problem is solved using the subgradient method [31].

Using convex optimization schemes and applying the KKT conditions, the closed-form optimal computational allocation of EN can be obtained as

$$f_{i_m}^* = \left[\left(\frac{\beta F_m^M}{\omega_m V_i^f + \mu_i V_i^f}\right)^{\frac{1}{2}}\right]^+. \tag{44}$$

Since the optimal resource allocation depends on dual variables β and $\boldsymbol{\mu}$, the subgradient method with guaranteed convergence can be used to address the Lagrange multiplier, which leads to

$$\begin{aligned}\beta(l+1) &= [\beta(l) - \varepsilon_\beta(l)\nabla\beta]^+,\\ \mu_i(l+1) &= [\mu_i(l) + \varepsilon_{\mu_i}(l)\nabla\mu_i]^+,\end{aligned} \tag{45}$$

where

$$\begin{aligned}\nabla\beta &= \sum_{i=1}^{U} f_{i_m} - 1,\\ \nabla\mu_i &= t_{i,delay} + \frac{(1-\chi_{i_m})V_i^f}{f_{i_m}F_m^M} - t_i^{\max}.\end{aligned} \tag{46}$$

In (45), l represents the number of iteration. $\varepsilon_\beta(l)$ and $\varepsilon_{\mu_i}(l)$ represent the corresponding step sizes. Since (41) is a convex optimization problem, it is guaranteed that the iteration between the outer and inner problems converges to the primal optimal solution.

After solving the optimal resource allocation $f_{i_m}^*$, **P6** is transformed into the following form:

$$\begin{aligned}\mathbf{P8}:\ &\min_{\{\boldsymbol{\chi},\mathbf{p}_{EN}\}} \sum_{i=1}^{U}\sum_{m=1}^{M}\left[(1-\chi_{i_m})\frac{\omega_m V_i^f}{f_{i_m}^* F_m^M} + \frac{\chi_{i_m}p_{i_m}\theta_i d_i}{r_{i_m,c}(p_{i_m})}\right],\\ s.t.\quad &C3, C6, C7,' C9.\end{aligned} \tag{47}$$

In the following, we use $\tilde{p}_{i_m}$ instead of p_{i_m} in **C10**, i.e $\tilde{p}_{i_m} = \chi_{i_m}p_{i_m}$. **P8** can be transformed to the following problem **P9**.

$$\begin{aligned}\mathbf{P9}:\ &\min_{\{\chi_{i_m},p_{i_m}\}} \sum_{i=1}^{U}\sum_{m=1}^{M} \times \left[\frac{(1-\chi_{i_m})\omega_m V_i^f}{f_{i_m}^* F_m^M} + \frac{\tilde{p}_{i_m}\theta_i d_i}{B_f\log_2(1+\frac{\tilde{p}_{i_m}g_{m,c}}{\sigma^2})}\right],\\ s.t.\quad &C1': 0 \leq \chi_{i_m} \leq 1,\\ &C2': 0 < \sum_{i=1}^{U} B_f\log_2\left(1+\frac{\tilde{p}_{i_m}g_{m,c}}{\sigma^2}\right) < C_m^B,\\ &C3': t_{i,delay}^b + \frac{\theta_i d_i}{B_f\log_2\left(1+\frac{\tilde{p}_{i_m}g_{m,c}}{\sigma^2}\right)} + \frac{(1-\chi_{i_m})V_i^f}{f_{i_m}^* F_m^M} < t_i^{\max},\\ &C4': \sum_{i=1}^{U}\tilde{p}_{i_m} \leq p_m^{\max}, 0 \leq \tilde{p}_{i_m} \leq \chi_{i_m}p_m^{\max}.\end{aligned} \tag{48}$$

where $t_{i,delay}^b = t_i^l + t_i + t_{i_n}$. In order to solve the above **P9**, we introduce Lagrange method to solve it.

$$L(\tilde{p}_{i_m},\chi_{i_m}\lambda) = \sum_{i=1}^{U}\sum_{m=1}^{M}\left[\frac{\tilde{p}_{i_m}\theta_i d_i}{B_f\log_2(1+\frac{\tilde{p}_{i_m}g_{m,c}}{\sigma^2})}\right] + \sum_{i=1}^{U}\sum_{m=1}^{M}(1-\chi_{i_m})\frac{\omega_m V_i^f}{f_{i_m}^* F_m^M} + \lambda\sum_{i=1}^{U}\sum_{m=1}^{M}\chi_{i_m}(1-\chi_{i_m}), \tag{49}$$

where the λ is a penalty factor, if λ is large enough, problem **P9** and $L(\tilde{p}_{i_m},\chi_{i_m},\lambda)$ have same optimal values. The minimum values are obtained iteratively by using the interior penalty function method. The optimization of problem **P9** can be transformed into the following **P10**.

$$\begin{aligned}\mathbf{P10}:\ &\min_{\{\chi_{i_m},\tilde{p}_{i_m}\}}[f_1(\chi_{i_m},\tilde{p}_{i_m}) - f_2(\chi_{i_m})]\\ s.t.\quad &C1,' C2,' C3,' C4'.\end{aligned} \tag{50}$$

$$f_1(\chi_{i_m},\tilde{p}_{i_m}) = \sum_{i=1}^{U}\sum_{m=1}^{M}\left[\frac{\tilde{p}_{i_m}\theta_i d_i}{B_f\log_2(1+\frac{\tilde{p}_{i_m}g_{m,c}}{\sigma^2})}\right] + \sum_{i=1}^{U}\sum_{m=1}^{M}\left[\frac{(1-\chi_{i_m})\omega_m V_i^f}{f_{i_m}^* F_m^M} + \lambda\chi_{i_m}\right], \tag{51}$$

$$f_2(\chi_{i_m}) = \lambda\sum_{i=1}^{U}\sum_{m=1}^{M}\chi_{i_m}^2. \tag{52}$$

However, $f_1(\chi_{i_m},\tilde{p}_{i_m})$ is still non-convex. In order to solve this problem, we will use the Frank-Wolfe algorithm for non-convex objective functions to find the optimal solution of **P10** [28]. **P10** can be written in the following form,

$$\begin{aligned}\mathbf{P11}:\ &\min\left[f_1(\chi_{i_m},\tilde{p}_{i_m}) - f_2(\chi_{i_m}^{(l)}) - \left\langle\nabla f_2(\chi_{i_m}^{(l)}), \chi_{i_m} - \chi_{i_m}^{(l)}\right\rangle\right],\\ s.t.\quad &C1,' C2,' C3,' C4'.\end{aligned} \tag{53}$$

Algorithm 3: Joint Resource Allocation and Offloading Decision at EN.

Input: $p_{i_m}, p_{i_m}^{\max}, f_{i_m}^{\max}, V_i^f, \theta_i d_i, \forall i \in \mathbb{U}, n \in \mathbb{N}, m \in \mathbb{M}$

Initialization:$p_{i_m}^{(0)}$, $\chi_{i_m}^{(0)}$, λ;
for all $\forall i \in$ **do**
  for all $\forall m \in$ **do**
    calculate $f_{i_m}^*$ by (44)
  end for
end for
calculate $L(\tilde{p}_{i_m}^{(0)}, \chi_{i_m}^{(0)}, \lambda)$ by (49)
while $|L(\tilde{p}_{i_m}^{(l)}, \chi_{i_m}^{(l)}, \lambda) - L(\tilde{p}_{i_m}^{(l-1)}, \chi_{i_m}^{(l-1)}, \lambda)| > \zeta$ **do**
  calculate $\tilde{p}_{i_m}^{(l)}$ and $\chi_{i_m}^{(l)}$ by (58)
  update p_{i_m}, χ_{i_m} and λ;
  $l = l + 1$
end while
returnall optimal offloading decisions $\chi_{i_m}^*$, computing resource allocation$f_{i_m}^*$, power allocation of BS $p_{i_m}^*$

where $\chi_{i_m}^{(l)}$ is the result of χ_{i_m} at l iteration. Since the constraint in **P11** is non-concave, we use the Difference of Convex functions (DC) programming scheme to solve the non-concave constraint [29]. $C2'$ can be rewritten as:

$$C2'': \sum_{i=1}^{U} [g_1(\tilde{p}_{i_m}) - \widetilde{g}(\tilde{p}_{i_m})] < C_m^B \tag{54}$$

where

$$g_1(\tilde{p}_{i_m}) = \mathrm{Blog}_2(\tilde{p}_{i_m} g_{m,c} + \sigma^2) \tag{55}$$

$\tilde{g}_2(\tilde{p}_{i_m})$ is the first order Taylor approximation of $g_2(\tilde{p}_{i_m})$.

$$g_2(\tilde{p}_{i_m}) = \log_2(\sigma^2) \tag{56}$$

$$\tilde{g}_2(\tilde{p}_{i_m}) = g_2(\tilde{p}_{i_m}) + \left\langle \nabla g_2(\hat{P}_{i_m}^{(l)}), \tilde{p}_{i_m} - \hat{P}_{i_m}^{(l)} \right\rangle \tag{57}$$

$\hat{P}_{i_m}^{(l)}$ is the result of $\tilde{p}_{i_m}$ iterations, substituting (54) into **P10**, we can arrive at:

$$\mathbf{P11}': \min \left[f_1(\chi_{i_m}, \tilde{p}_{i_m}) - f_2(\chi_{i_m}^l) - \left\langle \nabla f_2(\chi_{i_m}^l), \chi_{i_m} - \chi_{i_m}^l \right\rangle \right]$$
$$s.t. \quad C1,' C2,'' C3,' C4'. \tag{58}$$

To this end, **P8** can be solved by iterative algorithm, and procedure is shown in Algorithm 3.

E. Summary of the Proposed Scheme

In the proposed solution, we first address the task offloading decision by adopting the method of fractional linear programming, and corresponding computational complexity is $O(N)$. Then the transmit power allocation p_i of users and the association factor α_{i_n} between users and RRH are addressed according to Algorithm 2. The computational complexity of Algorithm 2 is $O(N) = O(\zeta N)$, where ζ is the number of basic steps required for calculation in (35-37). Algorithm 3 uses Frank-Wolfe algorithm to solve the problem computational resource allocation f_{i_m}, power allocation p_{i_m} and offloading decision χ_{i_m}. The total computational complexity of Algorithm 3 is

Algorithm 4: Iteration Algorithm Design.

Initialization: Initialize values of θ_i, p_i, a_{i_n}, p_{i_n}, f_{i_m}, p_{i_m}, χ_{i_m};
Input: $t_i^{\max}$, V_i, d_i, ω_m, F_m^M, φ_i;
Iteration: Set $t = 0$
while $\left| \begin{matrix} G(\theta_i{}^{(t+1)}, p_i^{(t+1)}, a_{i_n}^{(t+1)}, f_{i_m}^{(t+1)}, p_{i_n}^{(t+1)}, p_{i_m}^{(t+1)}, \chi_{i_m}^{(t+1)}) \\ -G(\theta_i{}^{(t)}, p_i^{(t)}, a_{i_n}^{(t)}, f_{i_m}^{(t)}, p_{i_n}^{(t)}, p_{i_m}^{(t)}, \chi_{i_m}^{(t)}) \end{matrix} \right| \leq \varepsilon$ **do**
  given $a_{i_n}^{(t)}$, $p_i^{(t)}$, $p_{i_n}^{(t)}$, $f_{i_m}^{(t)}$, $p_{i_m}^{(t)}$, $\chi_{i_m}^{(t)}$
  solve **P2**;
  obtain the optimal $\theta_i{}^{*(t)}$;
  given $\theta_i{}^{*(t)}$, $p_{i_n}^{(t)}$, $f_{i_m}^{(t)}$, $p_{i_m}^{(t)}$, $\chi_{i_m}^{(t)}$
  solve **P3**;
  obtain the optimal $a_{i_n}^{*(t)}$, $p_i^{*(t)}$;
  substitute $\theta_i{}^{*(t)}$, $a_{i_n}^{*(t)}$, $p_i^{*(t)}$, $f_{i_m}^{(t)}$, $p_{i_m}^{(t)}$, $\chi_{i_m}^{(t)}$, to solve **P5**;
  obtain the optimal $p_{i_n}^{*(t)}$;
  substitute $\theta_i{}^{*(t)}$, $a_{i_n}^{*(t)}$, $p_i^{*(t)}$, $p_{i_n}^{*(t)}$ to solve **P6**;
  obtain the optimal $f_{i_m}^{*(t)}$, $p_{i_m}^{*(t)}$, $\chi_{i_m}^{*(t)}$;
  set $p_i^{(t+1)}$=$p_i^{*(t)}$, $a_{i_n}^{(t+1)}$=$a_{i_n}^{*(t)}$, $f_{i_m}^{(t+1)}$=$f_{i_m}^{*(t)}$, $p_{i_n}^{(t+1)}$=$p_{i_n}^{*(t)}$, $p_{i_m}^{(t+1)}$=$p_{i_m}^{*(t)}$, $\chi_{i_m}^{(t+1)}$=$\chi_{i_m}^{*(t)}$;
  $t = t + 1$;
end while
end
Output p_i^*=$p_i^{*(t)}$, $a_{i_n}^*$=$a_{i_n}^{*(t)}$, $f_{i_m}^*$=$f_{i_m}^{*(t)}$, $p_{i_n}^*$=$p_{i_n}^{*(t)}$, $p_{i_m}^*$=$p_{i_m}^{*(t)}$, $\chi_{i_m}^*$=$\chi_{i_m}^{*(t)}$

$O(N^2) = O(\xi N^2 + N)$, where ξ is the number of basic steps required for Lagrangian multiplier calculation in (45).

Algorithm 4 uses iterative algorithm design, and provide the overall solution of the original problem. At first, θ_i, p_i, a_{i_n}, p_{i_n}, f_{i_m}, p_{i_m} and χ_{i_m} are initialized. Bring these initial values into the described algorithms in the previous subsections, get the initial iteration value, and then iterate until the convergence condition is met. The final optimal value can be obtained accordingly. The computational complexity of each iteration of loop is related to Algorithm 1, Algorithm 2 and Algorithm 3. The algorithm diagram is given in Fig. 2. The computational complexity of the inner iteration is $O(N + \zeta N + N^2\xi + N)$, so the overall computational complexity of Algorithm 4 is $O(N^3) = O(N(N + \zeta N + N^2\xi + N))$. Note that in Algorithm 4, $G(p_i^{(t)}, a_{i_n}^{(t)}, p_{i_n}^{(t)}, f_{i_m}^{(t)}, p_{i_m}^{(t)}, \chi_{i_m}^{(t)})$ is

$$G(p_i^{(t)}, a_{i_n}^{(t)}, f_{i_m}^{(t)}, p_{i_n}^{(t)}, p_{i_m}^{(t)}, \chi_{i_m}^{(t)}) = \sum_{i=1}^{U} \left(\frac{\varphi_i V_i^l}{C_i^l} + \frac{p_i^{(t)} \theta_i d_i}{r_i} \right) + \sum_{i=1}^{U} \sum_{n=1}^{N} \frac{p_{i_n}^{(t)} \theta_i d_i}{r_{i_n}(a_{i_n}^{(t)}, p_{i_n}^{(t)})} + \sum_{i=1}^{U} \sum_{m=1}^{M} \times \left[\frac{(1 - \chi_{i_m}^{(t)}) \omega_m V_i^f}{f_{i_m}^{(t)} F_m^M} + \frac{\chi_{i_m}^{(t)} p_{i_m}^{(t)} \theta_i d_i}{r_{i_m,c}(p_{i_m}^{(t)})} \right]. \tag{59}$$

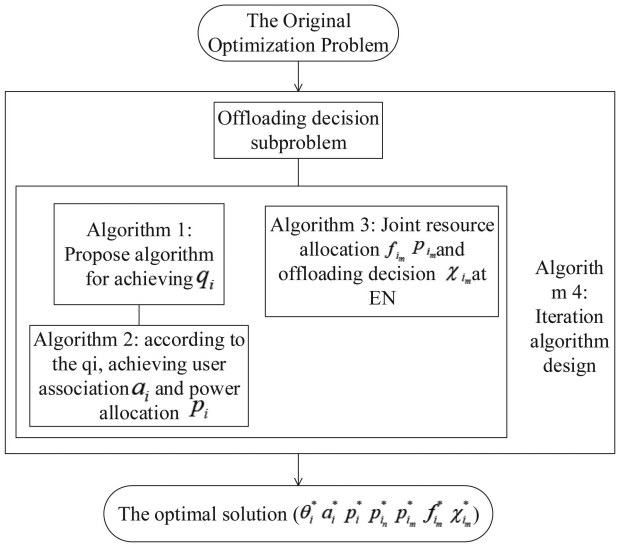


Fig. 2. Algorithm diagram.

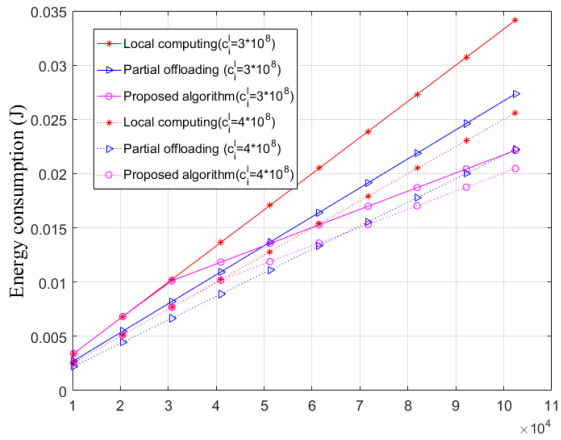


Fig. 3. The energy consumption v.s. different task sizes, offloading schemes and computing capacities of user, where $\chi_{i_m} = 1$.

V. Numerical Results

In this section, we verify the feasibility and superiority of the proposed method through simulation experiments. In the simulations, we set the number of users as 10, the number of RRHs as 5, and the number of ENs as 2. We set the input data for any task varies between 10 KB and 210 KB, the maximum transmit power of users to be 30dbm, the maximum transmit power of RRHs to be 40dbm and the maximum power of ENs to be 46dbm. The bandwidth between users and RRHs is 10 MHz, the bandwidth of fronthaul link between RRHs and ENs is 20 MHz, the bandwidth of backhaul link between ENs and the cloud set to be 40 MHz. The average distance between users and RRHs is 50 m, and the average distance between RRHs and ENs is 500 m. The maximum tolerance delays for completing the task are set to be between 0.5 s and 2 s.

In Figs. 3 and 4, we present the energy consumption of the user's task data size under different offloading strategies. Fig. 3 shows the EN processes the computing task, i.e., $\chi_{i_m} = 1$. In

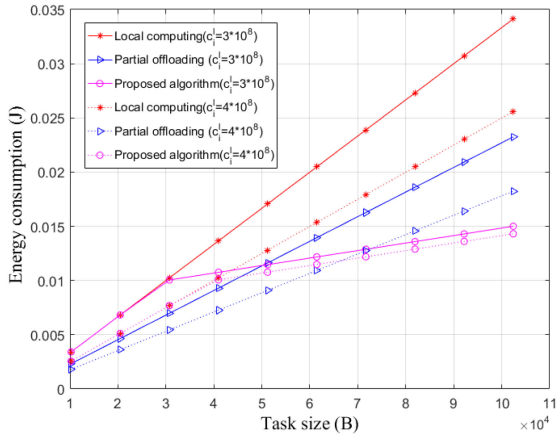


Fig. 4. The energy consumption v.s. different task sizes, offloading schemes and computing capacities of user, where $\chi_{i_m} = 0$.

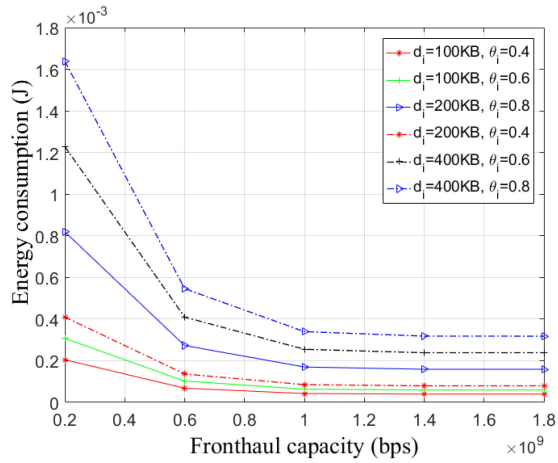


Fig. 5. The energy consumption on the fronthaul link v.s. fronthaul capacity.

Fig. 4, we consider all the tasks are offloaded to the cloud for execution, i.e. $\chi_{i_m} = 0$. In these figures, it is assumed that the computing processing capacity of users is 3×10^8 cycle/s and 4×10^8 cycle/s, respectively. As the computing capacity of users increases, the corresponding energy consumption decreases. It can be seen from Figs. 3 and 4 that local computing has a higher energy consumption than partial offloading, because the computing of server and cloud has a stronger computing capacity. Partial offloading scheme can reduce energy consumption by offloading tasks to the EN or cloud computing. It can be seen from Figs. 3 and 4 that the proposed algorithm can further reduce the energy consumption of task processing by selecting the appropriate offloading ratio according to the task size. Meanwhile, due to the limited local computing capacity of users, local computing can lead to more energy consumption. At the same time, if the task is total offloaded for computing, the energy consumption of the transmission process will be too high due to the long distance of task transmission and the energy consumption of task execution.

Figs. 5 and 6 investigate the energy consumption on the fronthaul link and backhaul link, respectively. Here, we assume that the offloading ratio of users is 0.4, 0.6 and 0.8, respectively. In

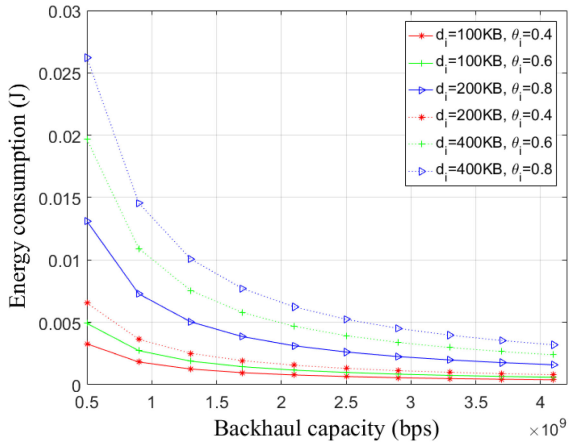


Fig. 6. The energy consumption on the backhaul link v.s. backhaul capacity.

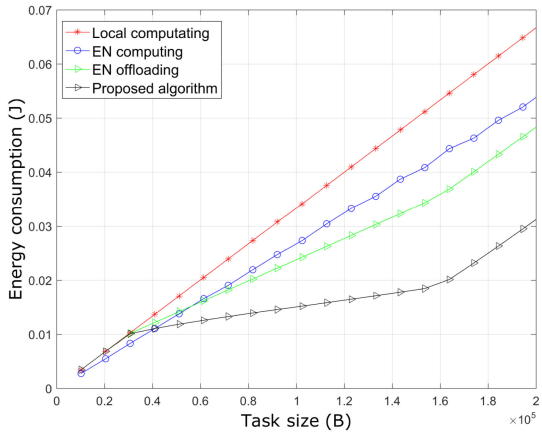


Fig. 9. Energy consumption v.s. different task sizes .

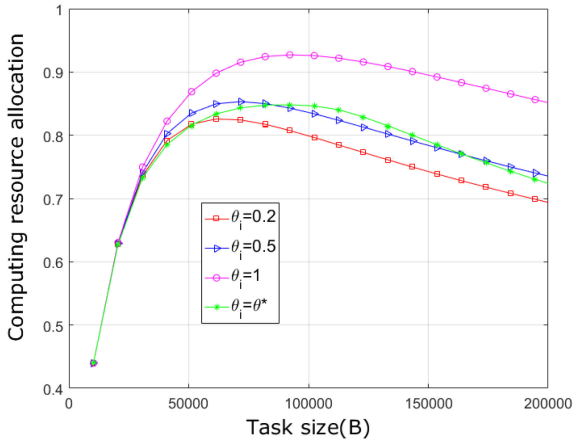


Fig. 7. Computing resource allocation on the EN v.s. different task sizes.

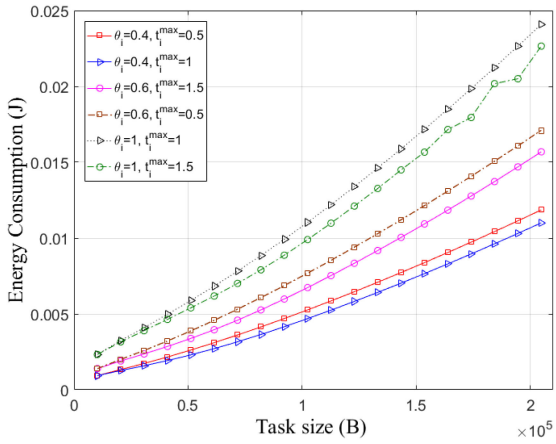


Fig. 8. Energy consumption on the EN v.s. different task sizes .

both figures, it can be seen that when $\theta_i = 8$ and $d_i = 400\ KB$, the energy consumption is the highest. From these two figures, we can also find that with the increase of the capacity of fronthaul/backhaul link, the transmit energy consumption of fronthaul/backhaul link is also increasing.

Fig. 7 plots the computing resource allocation of the FN by varying the task size and offloading ratio and Fig. 8 plots the corresponding energy consumption. The assumed offloading ratio θ_i are 0.2, 0.5, 1 and the optimal offloading ratio θ_i^*.

From Fig. 7, we can see that when the user's offloading ratio is larger, the EN allocates more computational resource to task processing, so as to ensure that the user's task can be completed within the maximum tolerance delay. In Fig. 7, the amount of used computational resource of the proposed optimal offloading scheme is lower than when the users choose total offloading. This is because when the amount of data offloading is small, we will use the local computing method to process tasks. In addition, when the amount of data is small, the energy consumption of local computing is lower than that of offloading. Meanwhile, in Fig. 8, we can see that although the computational resource amount allocated by our proposed optimal task offloading ratio method is lower than the others, which shows our proposed scheme can reduce the energy consumption. With the same amount of task sizes, the more users offload, the more energy EN consumes. The energy consumption also influenced by the maximum tolerance delay of users $t_i = t$. In order to meet the task processing delay and improve the processing efficiency, the EN will allocate more computing resource to the users who have a smaller maximum tolerance delay, and the corresponding energy consumption will increase.

In Fig. 9, we present the energy consumption performance of different computation offloading schemes. In this figure, we compare our proposed algorithm with the local computing where all the tasks are executing locally, the EN computing where all the tasks are performed on EN, and the EN offloading scheme presented in [35]. It can be found that as the task data size gradually increases, the total energy consumption also increases. In the local computing scheme, because the user's computing power is limited by the CPU processor, battery capacity, and other factors, it shows the worst energy consumption performance. In the EN computing, since all the tasks are executed at the EN, higher transmit power is induced. Although the EN offloading introduces computing offloading scheme, the cloud computing platform is absent. In the proposed computation offloading algorithm strategy, we jointly optimize the computation offloading and radio resource including the fronthaul/backhaul

link resources. Therefore, our proposed scheme outperforms the others in terms of energy consumption performance.

VI. Conclusion

In this work, we consider a vertical and heterogeneous multi-access edge computing system, where the users have computation service demand. In the considered system, the RRHs are deployed for providing wireless access for the users and EN with edge node can process the computation request from the user. Wireless backhaul and fronthaul are assumed where the capacity may be the bottleneck for service provisioning. With the objective to minimize the total energy consumption for processing the computation task, a joint radio resource allocation and offloading decision optimization problem is presented under the explicit consideration of capacity constraints of fronthaul and backhaul links. Due to the non-convexity of the formulated problem, we divide the original problem into several sub-problems and address them accordingly to find the optimal solution. Extensive simulation studies are conducted and performance evaluations demonstrate the advantages of the proposed scheme. In the future, we will further study the problem of how to solve computation offloading and resources allocation problem in the dynamic environment, where the user mobility can be explicitly considered.

References

[1] Cisco visual networking index: Global mobile data traffic forecast update, 2015–2020, Cisco White paper, 2016.

[2] Y. Mao, C. You, J. Zhang, K. Huang, and K. B. Letaief, "A survey on mobile edge computing: The communication perspective," *IEEE Commun. Surv. Tut.*, vol. 19, no. 4, pp. 2322–2358, Oct.-Dec. 2017.

[3] L. Liu, Z. Chang, and X. Guo, "Socially-aware dynamic computation offloading scheme for fog computing system with energy harvesting devices," *IEEE Internet Things J.*, vol. 5, no. 3, pp. 1869–1879, Jun. 2018.

[4] J. Pan and J. McElhannon, "Future edge cloud and edge computing for internet of things applications," *IEEE Internet Things J.*, vol. 5, no. 1, pp. 439–449, Feb. 2018.

[5] P. Mach and Z. Becvar, "Mobile edge computing: A survey on architecture and computation offloading," *IEEE Commun. Surv. Tut.*, vol. 19, no. 3, pp. 1628–1656, Jul.-Sep. 2017.

[6] Z. Liang, Y. Liu, T. Lok, and K. Huang, "Multiuser computation offloading and downloading for edge computing with virtualization," *IEEE Trans. Wireless Commun.*, vol. 18, no. 9, pp. 4298–4311, Sep. 2019.

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

[8] C. Wang, F. R. Yu, C. Liang, Q. Chen, and L. Tang, "Joint computation offloading and interference management in wireless cellular networks with mobile edge computing," *IEEE Trans. Veh. Technol.*, vol. 66, no. 8, pp. 7432–7445, Aug. 2017.

[9] Z. Ning *et al.*, "Mobile edge computing enabled 5G health monitoring for internet of medical things: A decentralized game theoretic approach," *IEEE J. Sel. Areas Commun.*, vol. 39, no. 2, pp. 463–478, Feb. 2021.

[10] X. Wang, Z. Ning, S. Guo, M. Wen, and V. Poor, "Minimizing the age-of-critical-information: An imitation learning-based scheduling approach under partial observations," *IEEE Trans. Mobile Comput.*, to be published doi: 10.1109/TMC.2021.3053136.

[11] T. Q. Dinh, J. Tang, Q. D. La, and T. Q. S. Quek, "Offloading in mobile edge computing: Task allocation and computational frequency scaling," *IEEE Trans. Commun.*, vol. 65, no. 8, pp. 3571–3584, Aug. 2017.

[12] Z. Ning *et al.*, "Partial computation offloading and adaptive task scheduling for 5G-enabled vehicular networks," *IEEE Trans. Mobile Comput.*, to be published, doi: 10.1109/TMC.2020.3025116.

[13] A. Samanta and Z. Chang, "Adaptive service offloading for revenue maximization in mobile edge computing with delay-constraint," *IEEE Internet Things J.*, vol. 6, no. 2, pp. 3864–3872, Apr. 2019.

[14] X. Cao, F. Wang, J. Xu, R. Zhang, and S. Cui, "Joint computation and communication cooperation for energy-efficient mobile edge computing," *IEEE Internet Things J.*, vol. 6, no. 3, pp. 4188–4200, Mar. 2019.

[15] Y. Wang, X. Tao, X. Zhang, P. Zhang, and Y. T. Hou, "Cooperative task offloading in three-tier mobile computing networks: An ADMM framework," *IEEE Trans. Veh. Technol.*, vol. 68, no. 3, pp. 2763–2776, Mar. 2019.

[16] 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.

[17] Y. Mao, J. Zhang, and K. B. Letaief, "Dynamic computation offloading for mobile-edge computing with energy harvesting devices," *IEEE J. Sel. Areas Commun.*, vol. 34, no. 12, pp. 3590–3605, Dec. 2016.

[18] Z. Chang, L. Liu, X. Guo, and T. Ristaniemi, "Dynamic resource allocation and computation offloading for IoT fog computing system," *IEEE Trans. Ind. Informat.*, vol. 17, no. 5, pp. 3348–3357, May 2021.

[19] L. Liu, Z. Chang, X. Guo, S. Mao, and T. Ristaniemi, "Multi-objective optimization for computation offloading in fog computing," *IEEE Internet Things J.*, vol. 5, no. 1, pp. 283–294, Feb. 2018.

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

[21] Y. Wu, K. Ni, C. Zhang, L. P. Qian, and D. H. K. Tsang, "NOMA-Assisted multi-access mobile edge computing: A. joint optimization of computation offloading and time allocation," *IEEE Trans. Veh. Technol.*, vol. 67, no. 12, pp. 12244–12258, Dec. 2018.

[22] J. Ren, G. Yu, Y. He, and G. Y. Li, "Collaborative cloud and edge computing for latency minimization," *IEEE Trans. Veh. Technol., vol*. 68, no. 5, pp. 5031–5044, May 2019.

[23] S. -H, Park, S. Jeong, J. Na, O. Simeone, and S., Shamai, "Collaborative cloud and edge mobile computing in C-RAN systems with minimal end-to-end latency," *IEEE Trans. Signal Inf. Process. Netw.*, vol. 7, pp. 259–274, Apr. 2021.

[24] L. Liu, S. Bi, and R. Zhang, "Joint power control and fronthaul rate allocation for throughput maximization in OFDMA-Based cloud radio access network," *IEEE Trans. Commun.*, vol. 63, no. 11, pp. 4097–4110, Nov. 2015.

[25] L. Yang, H. Zhang, M. Li, J. Guo, and H. Ji, "Mobile edge computing empowered energy efficient task offloading in 5 G," *IEEE Trans. Veh. Technol.*, vol. 67, no. 7, pp. 6398–6409, Jul. 2018.

[26] A. Al-Shuwaili, O. Simeone, A. Bagheri, and G. Scutari, "Joint uplink/downlink optimization for backhaul-limited mobile cloud computing with user scheduling," *IEEE Trans. Signal Inf. Process. Netw.*, vol. 3, no. 4, pp. 787–802, Dec. 2017.

[27] Q. Pham, L. B. Le, S. Chung, and W. Hwang, "Mobile edge computing with wireless backhaul: Joint task offloading and resource allocation," *IEEE Access*, vol. 7, pp. 16444–16459, 2019.

[28] S. Lacoste-Julien, "Convergence rate of frank-wolfe for non-convex objectives," 2016, *arXiv:1607.00345*, [Online]. Available: https://arxiv.org/pdf/1607.00345.pdf

[29] H. H. Kha, H. D. Tuan, and H. H. Nguyen, "Fast global optimal power allocation in wireless networks by local D. C. programming," *IEEE Trans. Wireless Commun.*, vol. 11, no. 2, pp. 510–515, Feb. 2012.

[30] W. Dinkelbach, "On nonlinear fractional programming," *Manage. Sci.*, vol. 13, pp. 492–498, Mar. 1967.

[31] S. Boyd and L. Vandenberghe, *Convex Optimization*, Cambridge, U.K., Cambridge Univ. Press, 2004.

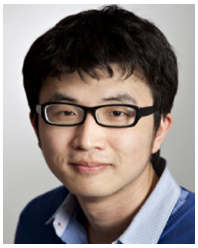
[32] Z. Chang, J. Gong, T. Ristaniemi, and Z. Niu, "Energy efficient resource allocation and user scheduling for collaborative mobile clouds with hybrid receivers," *IEEE Trans. Veh. Technol.*, vol. 65, no. 12, pp. 9834–9846, Dec. 2016.

[33] Z. Chang, Z. Wang, X. Guo, C. Yang, Z. Han, and T. Ristaniemi, "Distributed resource allocation for energy efficiency in OFDMA multicell networks with wireless power transfer," *IEEE J. Sel. Areas Commun.*, vol. 37, no. 2, pp. 345–356, Feb. 2019.

[34] 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.

[35] 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.

Jun Chen is currently working toward the Ph.D. degree with the College of Information Science and Engineering, Yanshan University, Qinhuangdao, China. His research interests include resource allocation, edge computing, and mobile communications.



Zheng Chang (Senior Member, IEEE) received the B.Eng. degree from Jilin University, Changchun, China, in 2007, the M.Sc. (Tech.) degree from the Helsinki University of Technology (currently Aalto University), Espoo, Finland, in 2009, and the Ph.D. degree from the University of Jyväskylä, Jyväskylä, Finland, in 2013. Since 2008, he has been in various research positions with the Helsinki University of Technology, University of Jyväskylä, and Magister Solutions Ltd, Finland. From June to August in 2013, he was a Visiting Researcher with Tsinghua University, Beijing, China, and from April to May 2015, with the University of Houston, Houston, TX, USA.

He has authored or coauthored more than 100 papers in journals and conferences. His research interests include IoT, cloud/edge computing, security and privacy, vehicular networks, and green communications. He is the Editor of the IEEE ACCESS, *Springer Wireless Networks,* and *International Journal of Distributed Sensor Networks*, and the Guest Editor of the IEEE NETWORK, IEEE WIRELESS COMMUNICATIONS, *IEEE Communications Magazine*, IEEE INTERNET OF THINGS JOURNAL, IEEE TRANSACTIONS ON INDUSTRIAL INFORMATICS, *Physical Communications*, *EURASIP Journal on Wireless Communications and Networking*, and *Wireless Communications and Mobile Computing*. In 2018, he was the exemplary Reviewer of the IEEE WIRELESS COMMUNICATION LETTERS. He has participated in organizing workshop and special session in Globecom' 19, WCNC'18-20, SPAWC'19, and ISWCS'18. He is also a TPC Member for many IEEE major conferences, including INFOCOM, ICC, and Globecom. He has been awarded by the Ulla Tuominen Foundation, the Nokia Foundation and the Riitta and Jorma J. Takanen Foundation for his research excellence. He has been awarded as 2018 IEEE Communications Society best young Researcher for Europe, Middle East and Africa Region. He was the recipient of Best Paper awards from IEEE TCGCC and APCC in 2017.

Xijuan Guo received the Ph.D. degree from Yanshan University, Qinhuangdao, China. She is currently a Professor with the College of Information Science and Engineering, Yanshan University. Her research interests include high performance computing, cloud computing, image processing, and wireless communications.

Renchuan Li, photograph and biography not available at the time of publication.

Zhu Han (Fellow, IEEE) received the B.S. degree in electronic engineering from Tsinghua University, Beijing, China, in 1997, and the M.S. and Ph.D. degrees in electrical and computer engineering from the University of Maryland, College Park, MD, USA, in 1999 and 2003, respectively.

From 2000 to 2002, he was an R&D Engineer of JDSU, Germantown, MD, USA. From 2003 to 2006, he was a Research Associate with the University of Maryland. From 2006 to 2008, he was an Assistant Professor with Boise State University, Boise, ID, USA. He is currently a John and Rebecca Moores Professor with the Electrical and Computer Engineering Department and the Computer Science Department, University of Houston, Houston, TX, USA. His research interests include wireless resource allocation and management, wireless communications and networking, game theory, big data analysis, security, and smart grid. He was the recipient of the NSF Career Award in 2010, the Fred W. Ellersick Prize of the IEEE Communication Society in 2011, the EURASIP Best Paper Award for the Journal on Advances in Signal Processing in 2015, the IEEE Leonard G. Abraham Prize in the field of Communications Systems (Best Paper Award in IEEE JSAC) in 2016, and various Best Paper awards in IEEE conferences. He was an IEEE Communications Society Distinguished Lecturer from 2015 to 2018, has been a AAAS Fellow since 2019 and ACM distinguished Member since 2019. Since 2017, he has been 1% highly cited Researcher according to Web of Science. He was also the winner of 2021 IEEE Kiyo Tomiyasu Award, for outstanding early to mid-career contributions to technologies holding the promise of innovative applications, with the following citation for contributions to game theory and distributed management of autonomous communication networks.

Timo Hämäläinen (Senior Member, IEEE) received the Ph.D. degree in telecommunication from the University of Jyväskylä, Jyväskylä, Finland, in 2002. In 1997, he joined the University of Jyväskylä, where he is currently a Professor of computer networks. He has more than 25 years research and teaching experience of computer networks. He has led many external funded network management related projects. He has launched and leads master programs with the University of Jyväskylä (SW & Comm. Eng.), and teaches network management related courses. He has more than 200 internationally peer reviewed publications and he has supervised almost 40 Ph.D. dissertation. His research interests include network resource management, IoT, and networking security.