Power-Constrained Edge Computing with Maximum Processing Capacity for IoT Networks

Min Qin, Li Chen, Nan Zhao, *Senior Member, IEEE,* Yunfei Chen, *Senior Member, IEEE,*
F. Richard Yu, *Fellow, IEEE,* Guo Wei

***Abstract*—Mobile edge computing (MEC) plays an important role in next-generation networks. It aims to enhance processing capacity and offer low-latency computing services for Internet of Things (IoT). In this paper, we investigate a resource allocation policy to maximize the available processing capacity (APC) for MEC IoT networks with constrained power and unpredictable tasks. First, the APC which describes the computing ability and speed of a served IoT device is defined. Then its expression is derived by analyzing the relationship between task partitioning and resource allocation. Based on this expression, the power allocation solution for the single-user MEC system with a single subcarrier is studied and the factors that affect the APC improvement are considered. For the multiuser MEC system, an optimization problem of APC with a general utility function is formulated and several fundamental criteria for resource allocation are derived. By leveraging these criteria, a binary-search water-filling algorithm is proposed to solve the power allocation between local CPU and multiple subcarriers, and a suboptimal algorithm is proposed to assign the subcarriers among users. Finally, the validity of the proposed algorithms is verified by Monte Carlo simulation.**

***Index Terms*—Available processing capacity, computation offloading, IoT, mobile edge computing, resource allocation.**

I. Introduction

With the wide utilization of Internet of Things (IoT) [1, 2], *e.g.*, mobiles, wearable devices, sensors and vehicles, the demand for high-speed, low-latency and dynamically configurable computing resources at the edge of cellular networks is exploding exponentially. Although more and more powerful CPUs are developed for these mobile things, the computing demand required by the new applications increases even more. Moreover, due to the death of Moore's Law [3] and insurmountable batteries [4], it is almost impossible to break the computation bottleneck on the terminal side. This motivates the development of mobile edge computing (MEC) [5–9], which provides abundant and low-latency computing services in the proximity of users as an important part of the IoT architecture [10, 11]. This kind of computing systems inherit most of the advantages of mobile cloud computing (M-CC) [12] but avoid the problems of long latency and overload in the core networks. The concept of MEC is gaining more interest in recent years. Lots of works have been studied in the literature, including MEC server platform, system architecture, mobility management (virtual machine (VM) migration) and resource management. In this paper, we propose a novel resource allocation policy to maximize the processing capacity of an MEC system.

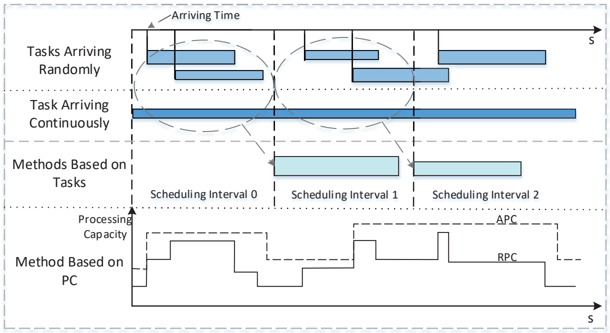


Fig. 1. Unpredictable task model. (The abscissa stands for time and each rectangle represents a task, where the length of the rectangle denotes the execution latency limit and the width denotes the amount of computation per second. Thus, the area of the rectangle denotes the total computation required by the task. Since the tasks are unpredictable, an alternative choice for the methods based on tasks is buffering the tasks and dealing with them in the next scheduling interval. But it causes extra latency, which is intolerant for real-time applications.)

Most of the previous works assume that the tasks to be conducted are known to the resource manager. Thus, their policies, named as methods based on tasks, can be carried out based on the information of computation demand and input data in the next scheduling interval. However, practical scenarios are much more complicated in IoT networks:

1) The tasks of IoT devices aren't predictable. Most tasks arrive randomly (*e.g.*, the interactive instructions for vehicle control) or continuously (*e.g.*, the data stream from sensors and the real-time video on surveillance equipment) and vary with time, as shown in Fig. 1.
2) The IoT networks can't afford the huge signaling cost incurred by the feedback of all task information, which is

This research was supported by National Science and Technology Major Project of China MIIT (Grant No. 2017ZX03001003-003), National Natural Science Foundation of China (Grant No. 61601432), and the Fundamental Research Funds for the Central Universities. This research was supported in part by the open research fund of National Mobile Communications Research Laboratory, Southeast University (No. 2018D03), the National Natural Science Foundation of China (NSFC) under Grant 61871065, and the Fundamental Research Funds for the Central Universities under DUT17JC43. (*Corresponding author: Li Chen*)

M. Qin, L. Chen and G. Wei are with the Department of Electronic Engineering and Information Science, University of Science and Technology of China, Hefei, China (email: qinminss@mail.ustc.edu.cn, {chenli87, wei}@ustc.edu.cn).

N. Zhao is with the School of Info. and Commun. Eng., Dalian University of Technology, Dalian, China (email: zhaonan@dlut.edu.cn).

Y. Chen is with the School of Engineering, University of Warwick, Coventry CV4 7AL, U.K. (e-mail: Yunfei.Chen@warwick.ac.uk).

F.R. Yu is with the Department of Systems and Computer Engineering, Carleton University, Ottawa, ON, K1S 5B6, Canada (email: richard.yu@carleton.ca).

required by the conventional allocation policy for MEC.

3) For IoT devices, the available power is strictly constrained due to the battery life [1].

Therefore, it is worth investigating a method that adapts the unpredictable computation demand, avoids the feedback overhead, and takes account of the power limitation.

Observe that the allocation policies in communication systems are developed by maximizing the throughput capacity [13] while the communication demand also varies randomly. Motivated by this observation, we propose available processing capacity (APC) to describe the computing ability and speed obtained by a served user in MEC. Then, a resource allocation policy to maximize the APC of users under power constraint is developed to satisfy user's required processing capacity (RPC) with the best effort. Since the policy is based on the instantaneous variable of APC, it is not necessary for the resource manager to know the future tasks. Also, since the device sends feedback to the resource manager only when the demand is not satisfied, the overhead of this policy is much less than that of the method based on tasks.

The main contributions of this paper are summarized as following:

- The definition and expression of APC are given. The APC is expressed as a concave function of power and subcarriers by analyzing the relationship between the optimal task partitioning schedule and the resource allocation policy.
- For the single-user MEC system, a closed-form solution for power allocation is derived. We analyze the optimal solutions for four typical cases with different channel and power conditions. The analysis confirms that MEC server can improve the user's APC significantly.
- For the multiuser MEC system, some assignment criteria are derived from the optimization problem of APC with a general utility function first. Based on these criteria, a binary-search water-filling algorithm and a suboptimal algorithm are presented to solve the power and subcarrier allocation problems, respectively.

The remainder of this paper is organized as follows. First, related works are introduced in Section II. Then, Section III presents the system model and the definition and expression of APC. In Section IV, the single-user MEC system is investigated. The power and subcarrier allocation algorithms for the multiuser MEC system are proposed in Section V. The performance of the proposed algorithms is evaluated in Section VI. Finally, we conclude this paper.

II. Related Work

A resource allocation policy for MEC consists of two parts: the computation offloading part and the resource allocation part. The former concentrates on the problem of task partitioning [14–19] in terms of energy consumption and/or execution delay. The latter focuses on regulating the communication rates between the users and MEC server, due to the limited spectrum and power in such a system. In many cases, computation offloading is studied in single-user scenarios, while in multiuser scenarios [20–27], both parts are considered since the users share the limited communication resource. Both single-user and multiuser scenarios have been extensively studied in previous work.

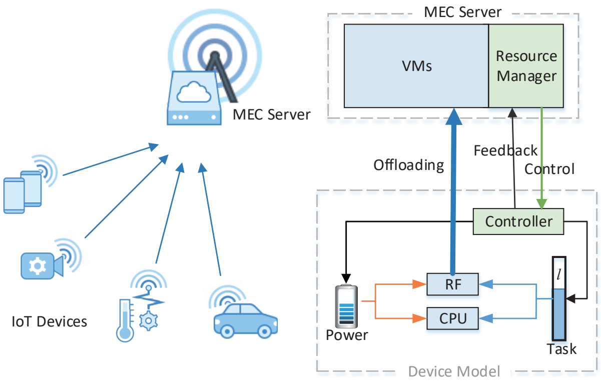


Fig. 2. The system model for multiuser mobile edge computing in the IoT network.

For single-user scenarios, Yang *et al.* [14] initiated the first work to study the task partitioning problem for mobile data stream applications to achieve high processing throughput in MEC. Liu *et al.* [15] proposed an optimal allocation policy by minimizing execution delay via a one-dimensional search algorithm considering the application buffer queuing state, available processing power and channel state information between the user and the MEC server. Mao *et al.* [16] adopted both the execution delay and task failure as the performance metric with dynamic voltage and frequency scaling (DVFS) [28] and energy harvesting techniques [29]. For a similar single-user framework, You *et al.* [17] considered low-complexity users harvesting energy with microwave power transfer. The optimal objective of this work was translated into minimizing the energy consumption of the users under offloading latency constraints, which was also adopted by Wang *et al.* in [18] considering single MEC server and multiple MEC servers respectively. For a single user to multiple MEC servers, an offloading policy for multiple tasks was proposed by Dinh *et al.* in [19], with minimizing the maximum execution latency of tasks.

For multiuser scenarios, a multiuser and multi-cell MEC system was considered by Sardellitti *et al.* in [20], where the communication and computation resources were jointly optimized to minimize the energy consumption with latency constraints. Chen *et al.* proposed a decentralized solution in [21], which was based on a game-theoretic formulation of the problem. A trade-off between the energy consumption and the execution delay for multiuser systems was discussed in [22, 23]. Recently, You *et al.* [24] developed an energy-efficient allocation policies for a multiuser MEC system, in which TDMA and OFDMA were considered respectively. Computation offloading and resource allocation in wireless cellular networks with a single MEC server were investigated for indivisible tasks by Wang *et al.* in [25, 26]. A virtual full-duplex MEC framework, where users are served by small cell base stations equipped with edge computing and caching, was proposed by Tan *et al.* in [27].

III. System Model

Consider a single-cell scenario, where an access point (AP) equipped with one MEC server (cloudlet) serves K active IoT devices/users, denoted as a set $\mathcal{K} = \{1, 2, ..., K\}$, as shown in Fig. 2. For each user, the computation may be processed at local CPU or offloaded to the MEC server. The available power of the user is assigned to a local computing module (CPU module) and an RF transmitter (RF module), correspondingly. In our system, all users aim to maximize their processing capacity. Therefore, they are expected to take full advantage of the CPU and RF modules under the power and subcarrier constraints.

In this section, we introduce the communication and computation models, since both important in the MEC system. Then, we derive a specific expression for the user's APC based on its definition.

A. Communication Model

We assume that the MEC system adopts OFDMA [24, 30] with N orthogonal subcarriers, denoted as a set $\mathcal{N} = \{1, 2, ..., N\}$. Each subcarrier has a bandwidth of $\bar{B}$. Both the users and the AP are equipped with a single antenna. Therefore, there are K independent communication links from the K users to the AP and each link may utilize several subcarriers. The uplink transmission rate $R_{\text{Tx},i}$ of user i's link is given by

$$R_{\text{Tx},i} = \sum_{n \in \mathcal{N}} \rho_{i,n} \bar{B} \log_2 (1 + p_{i,n} g_{i,n}), \tag{1}$$

where $\rho_{i,n} \in \{0, 1\}$ is an indictor variable, $p_{i,n}$ denotes the power allocated to subcarrier n by user i, $g_{i,n} = \|h_{i,n}\|^2 / N_0$ denotes the channel gain of user i on subcarrier n, N_0 is the power spectral density of the additive white Gaussian noise (fixed to 1 in this paper) and $h_{i,n}$ is the channel response. $\rho_{i,n} = 1$ while subcarrier n is assigned to user i and $\rho_{i,n} = 0$ while otherwise. To avoid interference among users, each subcarrier can be assigned to one user at most. Thus, the indictor variable satisfies

$$\sum_{i=1}^{K} \rho_{i,n} \leq 1. \tag{2}$$

This model can be extended to multiple transmit and receive antennas by modifying the transmission rate expression.

B. Computation Model

Assume that the user equipment can dynamically adjust the CPU's computational frequency to adopt the power consumption and execution latency with the DVFS technique [28]. For user i, the computational power $p_{i,0}$ can be modeled as

$$p_{i,0} = f_i^{\kappa} \zeta_i, \tag{3}$$

where f_i, in unit of Hz, is the CPU's computational frequency of user i and $\zeta_i > 0$ is the effective capacitance coefficient depending on chip architecture. The value κ ($\kappa \geq 2$) is a constant [31]. For simplicity, we set $\kappa = 2$ and assume that the local CPU is a single core architecture with a frequency upper bound of $f_{\max,i}$. Thus, the computational power satisfies $0 \leq p_{i,0} \leq f_{\max,i}^2 \zeta_i$. Constrained by the energy harvesting ability or the battery power, the available power of user i is fixed to P_i. Therefore, considering both the CPU and RF modules, the constraint for the sum of all the powers is written as

$$\underbrace{p_{i,0}}_{\text{CPU module}} + \underbrace{\sum_{n=1}^{N} p_{i,n}}_{\text{RF module}} \leq P_i. \tag{4}$$

Note that the receive power of the RF module is ignored since the feedback from the MEC server to users is negligible in most IoT networks.

As for the MEC server, we have the following three assumptions:

1) The MEC server has a multi-core architecture with much higher frequency and draws energy from power grid straightforwardly since it is located on AP.
2) The multiple-VM technique [32] is leveraged to make the server able to serve multiple users simultaneously.
3) The live prefetching strategy [33] is applied, in which the MEC server is fetching the input information of the next task while the current offloaded task is computed.

Based on these assumptions, the latency of the offloaded part is mainly determined by the transmitting latency. Therefore, the execution latency in the MEC server can also be ignored.

C. Available Processing Capacity

Consider a user with unpredictable tasks as shown in Fig. 1. The RPC is the sum of computation demands per second from all tasks. As the tasks arrive and terminate, the user's RPC varies randomly. To complete the tasks without extra latency, a sufficient condition is making the user's APC larger than the RPC all the time. The APC, which describes the instantaneous computing ability and speed obtained by a served user, is defined as follows.

Definition 1 (APC). If ω_i is the maximum available computation obtained by user i between t to $t + \Delta t$ in time, user i's APC at instant t is written as

$$C_i = \lim_{\Delta t \to 0} \frac{\omega_i}{\Delta t}. \tag{5}$$

It is evident that a user's APC is related to task partitioning, power allocation, and wireless channel state. According to the definition, we need to find the amount of computation during a fixed period for a user. It is hard to find that, while finding the execution latency for given computation is much easier, which is also suitable for the above definition.

Model the computation of user i as a divisible task (α_i, ω_i) [34], where α_i, in unit of bit, denotes the input data of the task, and ω_i denotes the required computation, *i.e.*, the number of CPU cycles. Assume that the task can be arbitrarily divided into any two parts in bits and the amount of computation corresponding to 1-bit input data can be written as $\eta_i = \omega_i / \alpha_i$, which is named as the computation-input ratio (CIR). To deal with the task, l_i bits of input data α_i are supposed to be offloaded to the MEC server via the wireless communication link and the remainder of the task is executed locally. Combining the task model with the aforementioned communication and computation models gives the minimum time cost for the computation as

$$t_{\text{cost},i} = \min_{\boldsymbol{P}_i,\boldsymbol{\rho}_i,l_i} \max\left(\frac{(\alpha_i - l_i)\,\eta_i}{\sqrt{p_{i,0}{\zeta_i}^{-1}}}, \frac{l_i}{R_{\text{Tx},i}} \right), \tag{6}$$

where $\boldsymbol{P}_i = [p_{i,0}, p_{i,1}, ..., p_{i,N}]^{\text{T}}$, $\boldsymbol{\rho}_i = [\rho_{i,1}, ..., \rho_{i,N}]^{\text{T}}$ and $R_{\text{Tx},i}$ is the transmission rate in (1). Equation (6) shows that the minimum time cost can be determined by task partitioning and resource allocation policies. When power $\boldsymbol{P}_i$ and subcarrier $\boldsymbol{\rho}_i$ are given and the task is fully divisible, we have the following lemma.

Lemma 1 (Execution Latency for Given Computation)**.** For fully divisible computation ω_i, the execution latency can be written as

$$\Delta t_i(\boldsymbol{P}_i, \boldsymbol{\rho}_i) = \frac{\omega_i}{\sqrt{p_{i,0}{\zeta_i}^{-1}} + \eta_i R_{\text{Tx},i}}, \tag{7}$$

and the optimal task partitioning strategy is

$$l_i = \frac{\alpha_i \eta_i R_{\text{Tx},i}}{\sqrt{p_{i,0}{\zeta_i}^{-1}} + \eta_i R_{\text{Tx},i}}. \tag{8}$$

Proof. See Appendix A. □

Lemma 1 reveals that both the execution latency and the optimal task partitioning are determined by the power and subcarrier allocation policies. By combining Lemma 1 and Definition 1, we have the following proposition.

Proposition 1 (Expression of APC)**.** For an arbitrary power and subcarrier allocation, the expression of APC can be written as

$$C_i(\boldsymbol{P}_i, \boldsymbol{\rho}_i) = \lim_{\Delta t_i \to 0} \frac{\omega_i}{\Delta t_i} = \underbrace{\sqrt{p_{i,0}{\zeta_i}^{-1}}}_{C_{\text{local}}} + \underbrace{\eta_i R_{\text{Tx},i}}_{C_{\text{remote}}}. \tag{9}$$

Proof. Take (7) in Lemma 1 into Definition 1 leading to the proposition. □

Proposition 1 presents the fact that, in the MEC system, a user's APC is determined by the local computing capacity and the transmission rate heading to the MEC server together. The former is denoted as C_{local} and the latter is denoted as C_{remote} which is the product of the transmission rate and CIR.

Remark 1 (RPC Constraint)**.** For the proposed system, the target of the resource allocation is to make the APC satisfy the RPC for each user as

$$C_i \geq C_{\text{req},i}, \forall i \in \mathcal{K}. \tag{10}$$

Remark 2 (Server Capacity Constraint)**.** Although the execution latency in the server is negligible according to the computation model, the total computation offloaded by all the users must be less than the processing capacity of the MEC server as

$$\sum_{i \in \mathcal{K}} \eta_i R_{\text{Tx},i} \leq C_{\text{server}}, \tag{11}$$

where C_{server} denotes the processing capacity of the MEC server.

IV. APC Improvement for Single User

To reveal how the APC is improved by the MEC server, a single-user MEC system is analyzed in this section. We first formulate an optimization problem to maximize the APC of the system and derive a closed-form solution. Then, we investigate some special points in the feasible sets of several typical solutions for the optimization problem.

To simplify the analysis, we consider a special case of the proposed MEC system, with a single user and a single available subcarrier, *i.e.*, $K = 1, N = 1$. The user's power is limited by P and the server knows the uplink channel state information. Using the above definition and constraints, we can obtain the following optimization problem:

$$\begin{aligned} \mathcal{P}_0 : &\max_{p_0, p_1} C \\ &\text{s.t.} \\ &\quad \text{C1} : p_0 + p_1 \leq P \\ &\quad \text{C2} : C \geq C_{\text{req}} \\ &\quad \text{C3} : C_{\text{local}}(p_0) \leq f_{\text{max}} \\ &\quad \text{C4} : C_{\text{remote}}(p_1) \leq C_{\text{server}} \\ &\quad \text{C5} : p_0, p_1 \geq 0, \end{aligned} \tag{12}$$

where $C = C_{\text{local}}(p_0) + C_{\text{remote}}(p_1)$, p_0 denotes the power allocated to the CPU module and p_1 denotes the power allocated to the RF module. To solve $\mathcal{P}_0$, we have the following proposition.

Proposition 2. $\mathcal{P}_0$ is a convex problem. If constraints C2-C4 are ignored, the optimal solution can be written as

$$p_0^o = \min\left(\frac{1}{4\zeta v^2}, P\right), \quad p_1^o = \left[\frac{\eta \bar{B}/\ln 2}{v} - \frac{1}{g}\right]^+, \tag{13}$$

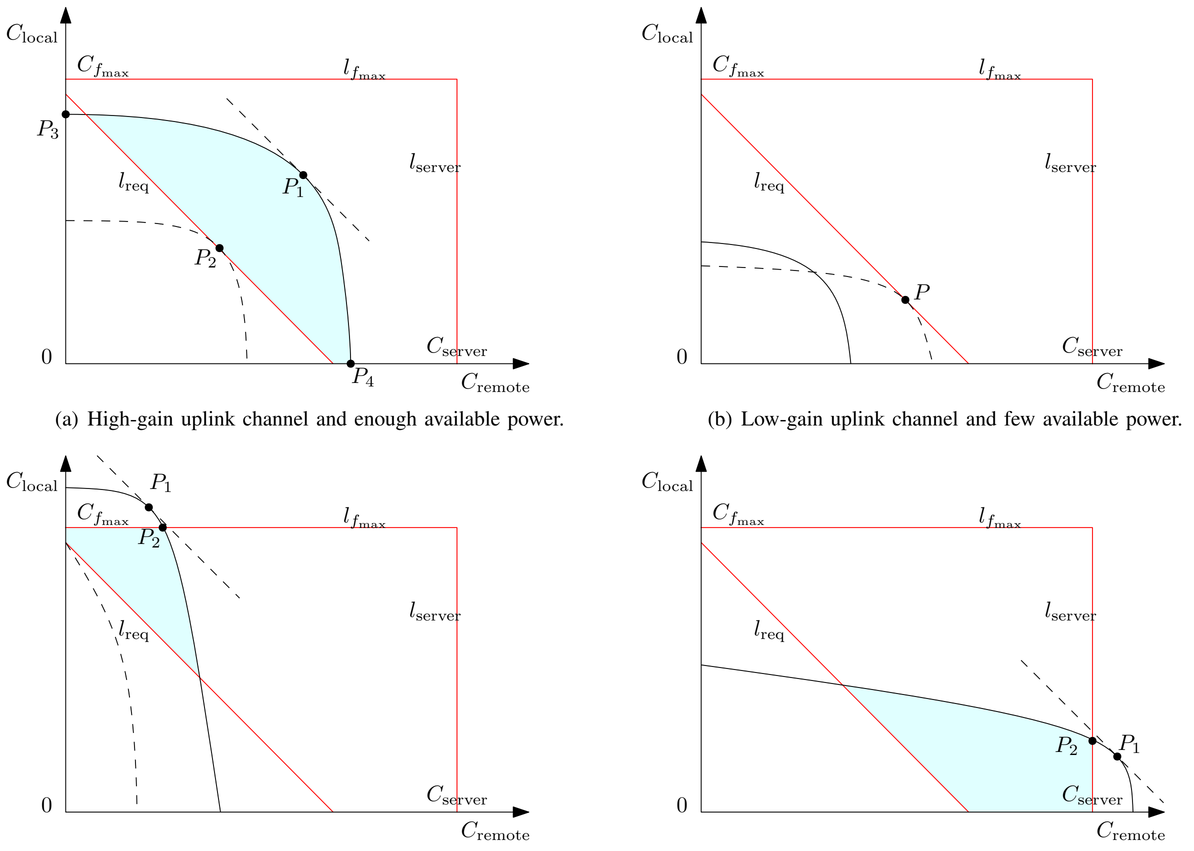
where $\frac{1}{v} = 2\zeta(\sqrt{(\eta\bar{B}/\ln 2)^2 + \frac{1}{\zeta}(P + \frac{1}{g})} - \eta\bar{B}/\ln 2)$, $[x]^+ \triangleq \max[0, x]$.

Proof. See Appendix B. □

Since $C_{\text{local}}(p_0)$ and $C_{\text{remote}}(p_1)$ are concave in p_0 and p_1 respectively, the optimization problem can be illustrated as Fig. 3. In the figure, the curve represents the power constraint, line l_{req} represents the user's RPC, line $l_{f_{\text{max}}}$ represents the upper frequency bound of the CPU and line l_{server} represents the server capacity constraint. When the effective capacitance coefficient ζ and CIR η are fixed, the uplink channel state is the only factor that affects the power constraint curve according to the definition of APC. Fig. 3 (a-d) show several typical solutions for the optimization problem with different uplink channel states and available powers. Note that p_0^* and p_1^* are the optimal powers with all constraints.

Case 1 (High-gain channel and enough power)**:** When

$$\begin{aligned} &C(p_0^o, p_1^o) \geq C_{\text{req}} \\ &C_{\text{local}}(p_0^o) \leq f_{\text{max}} \\ &C_{\text{remote}}(p_1^o) \leq C_{\text{server}}, \end{aligned} \tag{14}$$



(a) High-gain uplink channel and enough available power.

(b) Low-gain uplink channel and few available power.

(c) Low-gain uplink channel and enough available power.

(d) High-gain uplink channel and few available power.

Fig. 3. The APC improvement provided by the MEC server. (The horizontal axis shows C_{remote} and the vertical axis shows C_{local}.)

then

$$\begin{aligned} p_0^* &= p_0^o \\ p_1^* &= p_0^o. \end{aligned} \tag{15}$$

The uplink channel state is proper so that the optimal point is obtained on the curve as shown in Fig. 3(a). Point P_1 is the point of tangency of the power constraint curve and the straight line $C_{\text{local}} + C_{\text{remote}} = c$ where c is a constant. This point, named as the best effort point, is achieved when the user fully utilizes its power and reaches the maximum APC. Point P_3 represents that all the power is allocated to the CPU module and point P_4 is obtained when all the power is allocated to the RF module. Obviously, by allocating power to the CPU and RF module properly, we can obtain much more APC for the user than computing locally only.

Especially, while all the points in the feasible area satisfy the RPC constraint, there is a point P_2 which is with minimum power consumption. This point is actually *the energy-efficient point* in previous work [17, 24], where the energy consumption is minimized on the premise of matching the RPC demand.

Case 2 (Low-gain channel and few power)**:** When

$$C(p_0^o, p_1^o) < C_{\text{req}}, \tag{16}$$

the solution doesn't exist. Fig. 3(b) shows the case when the uplink channel suffers from deep fading and the user's power is not enough to reach the required APC. Therefore, the feasible area does not exist. To avoid this, the resource manager should assign more communication resource to the user and improve the communication rate from the user to the server shown as the dashed curve in the figure.

Case 3 (Low-gain channel and enough power)**:** When

$$\begin{aligned} &C(p_0^o, p_1^o) \geq C_{\text{req}} \\ &C_{\text{local}}(p_0^o) > f_{\max} \\ &C_{\text{remote}}(p_1^o) \leq C_{\text{server}}, \end{aligned} \tag{17}$$

then

$$\begin{aligned} p_0^* &= f_{\max}^2 \zeta \\ p_1^* &= P - p_0^*. \end{aligned} \tag{18}$$

The case in Fig. 3(c) happens while the uplink channel fades deeply but the available power is enough. It is easy to find that the tangency point P_1 is out of the feasible area and the optimal point is P_2 where the CPU works with its maximum frequency.

What's more interesting is that the tangency point is sometimes obtained at the intersection of the power constraints curve and the C_{local} axis, shown as the dashed curve in Fig. 3(c). It happens when the uplink channel fades deeply and $\frac{\eta \bar{B}/\ln 2}{\upsilon} < \frac{1}{g}$. In this case, all the power should be allocated to the CPU module. Hence, $p_1^* = 0$.

Case 4 (High-gain channel and few power)**:** When

$$\begin{aligned} &C(p_0^o, p_1^o) \ge C_{\text{req}} \\ &C_{\text{local}}(p_0^o) \le f_{\text{max}} \\ &C_{\text{remote}}(p_1^o) > C_{\text{server}}, \end{aligned} \tag{19}$$

then

$$\begin{aligned} p_1^* &= g^{-1}(2^{C_{\text{server}}\eta^{-1}\bar{B}^{-1}} - 1) \\ p_0^* &= P - p_1^*. \end{aligned} \tag{20}$$

Fig. 3(d) indicates the case that the tangency point P_1 is out of the server capacity constraint and P_2 is the optimal one. It happens when ζ is very small and the uplink channel state is comparatively good. But the tangency point can never appear at the C_{remote} axis, which will be explained in the power allocation part of the next section.

Although the optimal points in Case 1, 3 and 4 are different, the user's APC is significantly enhanced by the MEC server for all three cases. In Case 2, the RPC is unreachable due to the limited power and deeply faded channel. In summary, the APC of an MEC system is determined by the communication resource and the available power, meanwhile constrained by the frequency upper bound of local CPU and the server capacity.

V. Multiuser Resource Allocation

For a multiuser MEC system, to enhance the users' APC, not only the available power is allocated between the CPU module and the subcarriers, but also the subcarriers should be assigned among users. Before investigating the allocation policy, we propose a general utility function as the performance metric to meet the practical allocation principles first. Then, by solving the optimization problem, we derive several criteria for the power and subcarrier allocation and propose some efficient algorithms based on these criteria. Furthermore, the subcarrier assignment criteria for three specific utility functions, i.e., sum APC maximization, proportional fairness and max-min fairness, are analyzed respectively.

A. Problem Formulation

Definition 2 (Utility Function)**.** Define $U_i(C_i)$ as the utility function of user i, if U_i is a concave and twice differentiable function in the domain $\mathbb{R}^+$, and $\tilde{U}_i$ is strictly increasing in $\mathbb{R}$, where $\tilde{U}_i$ denotes the extend-value extension of the function U_i which assigns the value $-\infty$ to points not in the domain of U_i. Define the utility function of the whole MEC system as

$$U = \sum_{i \in \mathcal{K}} U_i(C_i). \tag{21}$$

For instance, the utility function for maximizing the APC of the system can be written as $U = \sum_{i\in\mathcal{K}} w_i C_i$, where w_i denotes the weighting coefficient of user i. When there are no priorities among the users, $w_i = 1$ for $i \in \mathcal{K}$. Considering the utility function and all the constraints in the system model, we can formulate the resource allocation problem as

$$\begin{aligned} \mathcal{P}_{\text{origin}} : \max_{\boldsymbol{P},\boldsymbol{\rho}} \quad & \sum_{i\in\mathcal{K}} U_i\left(C_i\right) \\ \text{s.t.} \quad & \\ & (2), (4), (10), (11) \\ & 0 \le p_{i,0} \le f_{\max,i}^2 \zeta_i, && \forall i \in \mathcal{K} \\ & p_{i,n} \ge 0, && \forall i \in \mathcal{K}, n \in \{0\} \cup \mathcal{N} \\ & \rho_{i,n} \in \{0,1\}, && \forall i \in \mathcal{K}, n \in \mathcal{N} \end{aligned} \tag{22}$$

where $\boldsymbol{P} = \{p_{i,n}\}_{i\in\mathcal{K}, n\in\{0\}\cup\mathcal{N}}^{K\times(N+1)}$ and $\boldsymbol{\rho} = \{\rho_{i,n}\}_{i\in\mathcal{K},n\in\mathcal{N}}^{K\times N}$.

Since the subcarrier indictor $\rho_{i,n}$ belongs to a set of integers, $\mathcal{P}_{\text{origin}}$ is a mixed integer optimization problem, which is hard to solve. However, if $\rho_{i,n}$ is relaxed to a real value in $[0,1]$, the problem becomes more tractable [30]. Define $\tilde{p}_{i,n} = p_{i,n}\rho_{i,n}$, where $\rho_{i,0} \equiv 1$, $i \in \mathcal{K}$ and $n \in \{0\} \cup \mathcal{N}$. Thus, user i's APC can be rewritten as

$$C_i(\tilde{\boldsymbol{P}}, \boldsymbol{\rho}) = \sqrt{\tilde{p}_{i,0}\zeta_i^{-1}} + \eta_i \sum_{n=1}^{N} \rho_{i,n} \bar{B} \log_2 \left(1 + \frac{\tilde{p}_{i,n}}{\rho_{i,n}} g_{i,n}\right), \tag{23}$$

where $\tilde{\boldsymbol{P}} = \{\tilde{p}_{i,j}\}_{i\in\mathcal{K},n\in\{0\}\cup\mathcal{N}}^{K\times(N+1)}$.

Proposition 3. If $\rho_{i,n}$ is relaxed to $[0,1]$, both C_i and U are jointly concave in $(\tilde{\boldsymbol{P}}, \boldsymbol{\rho})$.

Proof. See Appendix C. □

This proposition indicates that the objective function of the original problem can be relaxed to a concave function. $\mathcal{P}_{\text{origin}}$ can be rewritten as

$$\begin{aligned} \mathcal{P}_1 : \ \max_{\tilde{\boldsymbol{P}},\boldsymbol{\rho}} \quad & \sum_{i\in\mathcal{K}} U_i\left(C_i\right) \\ \text{s.t.} \quad & \\ \text{C1} : & \sum_{i=1}^{K}\sum_{n=1}^{N} \eta_i \rho_{i,n} \bar{B} \log_2 \left(1 + \frac{\tilde{p}_{i,n}}{\rho_{i,n}} g_{i,n}\right) \le C_{\text{server}} \\ \text{C2} : & \sum_{n=0}^{N} \tilde{p}_{i,n} \le P_i, && \forall i \in \mathcal{K} \\ \text{C3} : & \sum_{i=1}^{K} \rho_{i,n} \le 1, && \forall n \in \mathcal{N} \\ \text{C4} : & C_i \ge C_{\text{req},i}, && \forall i \in \mathcal{K} \\ \text{C5} : & 0 \le \tilde{p}_{i,0} \le f_{\max}^2 \zeta_i, && \forall i \in \mathcal{K} \\ \text{C6} : & \tilde{p}_{i,n} \ge 0, && \forall i \in \mathcal{K}, n \in \{0\} \cup \mathcal{N} \\ \text{C7} : & 0 \le \rho_{i,n} \le 1, && \forall i \in \mathcal{K}, n \in \mathcal{N} \end{aligned} \tag{24}$$

The objective function is concave as stated in Proposition 3. Since C_i is concave as stated in Proposition 3, constraint C4 is convex. The rest of the constraints are all linear. However, the problem $\mathcal{P}_1$ is non-convex because the left side of '$\le$' in constraint C1 is a concave function. In general, a non-zero duality gap exists if we solve a non-convex problem by solving its dual. However, it has been proved that the duality gap is always zero when this kind of non-convex optimization problem satisfies certain conditions.

Lemma 2 (Condition for Zero Duality Gap)**.** Let P and D denote the optimal values of the primal and the dual problem in (24), respectively. If the number of subcarriers is sufficiently large, then strong duality holds and the duality gap is always zero, *i.e.*, $P = D$.

Proof. Please refer to the proof of Lemma 2 in [35]. □

By leveraging Lemma 2, it is possible to solve the optimization problem (24) by solving its dual. For simplicity, we assume that the number of subcarriers is large enough[1]. Hence the duality gap can be ignored and the Karush-Kuhn-Tucker (KKT) conditions are necessary and sufficient for the optimal value of this problem. Since relaxation is used, the optimal value of $\mathcal{P}_1$ is suboptimal for $\mathcal{P}_{org}$. By analyzing the KKT conditions, we can derive some criteria for resource allocation and obtain a suboptimal solution for the original problem. To obtain the KKT conditions, we first obtain the Lagrangian function of $\mathcal{P}_1$, which is written as

$$\begin{aligned}
&L(\tilde{\boldsymbol{P}}, \boldsymbol{\rho}, \boldsymbol{\beta}, \gamma, \boldsymbol{\upsilon}, \boldsymbol{\lambda}) \\
&= \sum_{i\in\mathcal{K}} U_i\left(C_i\right) + \sum_{i\in\mathcal{K}} \beta_i \left[C_i - C_{\text{req},i}\right] \\
&+ \gamma \left[C_{\text{server}} - \sum_{i\in\mathcal{K}} \sum_{n=1}^{N} \eta_i \rho_{i,n} \bar{B} \log_2 \left(1 + \frac{\tilde{p}_{i,n}}{\rho_{i,n}} g_{i,n}\right)\right] \qquad (25) \\
&+ \sum_{i\in\mathcal{K}} \upsilon_i \left[P_i - \sum_{n=0}^{N} \tilde{p}_{i,n}\right] + \sum_{n=1}^{N} \lambda_n \left[1 - \sum_{i\in\mathcal{K}} \rho_{i,n}\right] \\
&\text{s.t.} \quad \text{C5}, \text{C6}, \text{C7},
\end{aligned}$$

where $\boldsymbol{\beta} \succeq 0$ is the Lagrange multiplier vector related to the RPC constraint C4, $\gamma \geq 0$ is the Lagrange multiplier associated with the maximum server capacity constraint C1, and $\boldsymbol{\upsilon} \succeq 0$ and $\boldsymbol{\lambda} \succeq 0$ is the Lagrange multiplier vector corresponding to the power constraint C2 and the subcarrier indicator constraint C3, respectively.

The necessary and sufficient conditions for the optimal value of $\mathcal{P}_1$ are obtained as

$$\begin{aligned}
\frac{\partial L}{\partial \tilde{p}^*_{i,n}} =& (U'_i\left(C_i\right) + \beta_i - \gamma) \left[\frac{\eta_i \bar{B} g_{i,n} / \ln 2}{\rho_{i,n} + \tilde{p}_{i,n} g_{i,n}}\right] - \upsilon_i \\
& \begin{cases} < 0, & \tilde{p}^*_{i,n} = 0 \\ = 0, & \tilde{p}^*_{i,n} > 0 \end{cases}, \forall i \in \mathcal{K}, n \in \mathcal{N}
\end{aligned} \qquad (26)$$

$$\begin{aligned}
\frac{\partial L}{\partial \tilde{p}^*_{i,0}} =& \frac{U'_i\left(C_i\right) + \beta_i}{\sqrt{4\tilde{p}_{i,0}\zeta_i}} - \upsilon_i \\
& \begin{cases} < 0, \; \tilde{p}^*_{i,0} = 0 \\ = 0, \; 0 < \tilde{p}^*_{i,0} < f^2_{\max,i}\zeta_i \\ > 0, \; \tilde{p}^*_{i,n} = f^2_{\max,i}\zeta_i \end{cases}, \forall i \in \mathcal{K}
\end{aligned} \qquad (27)$$

$$\begin{aligned}
\frac{\partial L}{\partial \rho^*_{i,n}} =& (U'_i\left(C_i\right) + \beta_i - \gamma) \eta_i F_{i,n} - \lambda_n \\
& \begin{cases} < 0, & \rho^*_{i,n} = 0 \\ = 0, & 0 < \rho^*_{i,n} < 1 \\ > 0, & \rho^*_{i,n} = 1 \end{cases}, \forall i \in \mathcal{K}, n \in \mathcal{N},
\end{aligned} \qquad (28)$$

where $F_{i,n} = \bar{B}\left[\log_2\left(1 + p^*_{i,n} g_{i,n}\right) - \frac{p^*_{i,n} g_{i,n} / \ln 2}{1 + p^*_{i,n} g_{i,n}}\right]$. Based on (26), (27) and (28), several criteria are discussed in the next sub-sections.

B. Power Allocation

Proposition 4. For a given subcarrier assignment, the optimal solution for power allocation is

$$p^*_{i,0} = \frac{\tilde{p}^*_{i,0}}{\rho_{i,0}} = \min\left(\frac{(1+\tilde{\beta}_i)^2}{4\tilde{\upsilon}_i^2 \zeta_i}, P_i, f^2_{\max}\zeta_i\right), \forall i \in \mathcal{K}, \qquad (29)$$

$$p^*_{i,n} = \frac{\tilde{p}^*_{i,n}}{\rho_{i,n}} = \left[\frac{(1+\tilde{\beta}_i - \tilde{\gamma}_i)\eta_i \bar{B}}{\tilde{\upsilon}_i \ln 2} - \frac{1}{g_{i,n}}\right]^+, \forall i \in \mathcal{K}, n \in \mathcal{N}, \qquad (30)$$

where $\tilde{\gamma}_i = \gamma / U'_i(C_i)$, $\tilde{\beta}_i = \beta_i / U'_i(C_i)$ and $\tilde{\upsilon}_i = \upsilon_i / U'_i(C_i)$.

Proof. As the utility function $U_i(C_i)$ is monotonically increasing in C_i, $U'_i(C_i) > 0$. Hence, dividing the Lagrange multipliers by the derivative of utility function leads to new multipliers $\tilde{\gamma}_i$, $\tilde{\beta}_i$ and $\tilde{\upsilon}_i$. By solving the KKT conditions (26) and (27) straightly, we can obtain the proposed results with the new multipliers. □

Notice that Equation (30), which allocates the power to the subcarriers, is the standard water-filling algorithm. While the other one, which calculates the power allocated to the CPU module, is another story. Before reaching the up bound of the local computation power, $p_{i,0}$ is proportional to the square of the water level during the water-filling process. Assume that there are j subcarriers, denoted as a set $\mathcal{N}_j$, that have been allocated power. In other words, the reciprocals of channel gains of these j subcarriers are below the water level. To maximize the APC, user i tends to take full advantage of the available power[2]. Therefore, the available power P_i equals the sum of transmitting powers and local computation power. By rearranging terms, we have a following equation

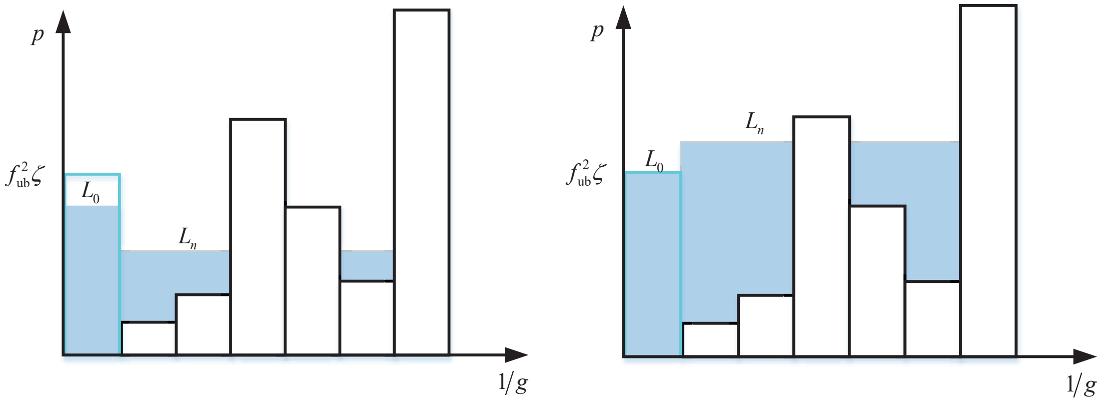
$$P_i + \sum_{n\in\mathcal{N}_j} \frac{1}{g_{i,n}} = \frac{(1+\tilde{\beta}_i - \tilde{\gamma}_i)\eta_i \bar{B}}{\tilde{\upsilon}_i \ln 2} j + \frac{(1+\tilde{\beta}_i)^2}{4\tilde{\upsilon}_i^2 \zeta_i}. \qquad (31)$$

If the Lagrange multipliers $\tilde{\beta}_i$ and $\tilde{\gamma}_i$ are given, Equation (31) is a quadratic polynomial in $1/\tilde{\upsilon}_i$ for a given set of subcarriers $\mathcal{N}_j$. Since the discriminant is positive, the polynomial has two distinct real roots and $1/\tilde{\upsilon}_i$ is the positive one. Because the transmitting power must be positive, the channel gains of the subcarriers in $\mathcal{N}_j$ must satisfy $\frac{(1+\tilde{\beta}_i - \tilde{\gamma})\eta_i \bar{B}}{\tilde{\upsilon}_i \ln 2} > \frac{1}{g_{i,n}}$. To solve the power allocation problem, we only need to find $\mathcal{N}_c$ including all the subcarriers, of which the channel gains satisfy the above inequality. Meanwhile, taking into account the upper bound of the local computation power, we fix $p^*_{i,0}$ to $f^2_{\max,i}\zeta_i$, while $\frac{(1+\tilde{\beta}_i)^2}{4\tilde{\upsilon}_i^2 \zeta} > f^2_{\max,i}\zeta_i$. Based on the above analysis, Algorithm 1, named as the binary-search water-filling algorithm, is proposed.

Algorithm 1 gives the power allocation policy to maximize the APC of users. This algorithm is distinguished from the

[1] The simulation in [35] shows that the duality gap is nearly zero for 10 subcarrier and small enough for even less subcarrires in an OFDMA system.

[2] Actually, they may not. Due to the server capacity limit and the upper bound of the local computation power, users maybe not able to make full use of the available power, which is further discussed in Remark 5.



(a) L_0 is below the upper bound.

(b) L_0 reaches the upper bound.

Fig. 4. The water-filling process for power allocation in MEC system.

Algorithm 1 Binary-search water-filling algorithm

```
Initialize:
    Set $l = 1$, $u = N$.
Sort user $i$'s channel gains on all $N$ subcarriers and $g_{i,n}^{(k)}$ denotes the $k$th largest one.
repeat
    $j = \lfloor (l+u)/2 \rfloor$.
    Solve the quadratic polynomial
        $$P_i + \sum_{k=1}^{j} \frac{1}{g_{i,n}^{(k)}} = \frac{(1+\tilde{\beta}_i - \tilde{\gamma}_i)\eta_i \bar{B}}{\tilde{v}_i \ln 2} j + \frac{(1+\tilde{\beta}_i)^2}{4\tilde{v}_i^2 \zeta_i}$$
    to obtain $\tilde{v}_i(j)$.
    if $\frac{(1+\tilde{\beta}_i-\tilde{\gamma}_i)\eta_i\bar{B}}{\tilde{v}_i(j)\ln 2} > \frac{1}{g_{i,n}^{(j)}}$ then
        $l = j$.
    else
        $u = j$.
    end if
until $u - l = 1$;
$\tilde{v}_i = \tilde{v}_i(l)$.
if $\frac{(1+\tilde{\beta}_i)^2}{4\tilde{v}_i^2\zeta_i} > f_{\max,i}^2 \zeta_i$ then
    $p_{i,0}^* = f_{\max,i}^2 \zeta_i$.
    Allocate the transmitting power $p_{i,n}^*$ utilizing standard water-filling algorithm.
    Close.
else
    Obtain $p_{i,0}^*$ and $p_{i,n}^*$ from (29) and (30) respectively.
end if
```

traditional water-filling algorithm, since the water level is searched by binary-search process due to the existence of the local computation power. If the power of the local CPU is below the upper bound, we need to solve the quadratic polynomial to obtain the water level when searching the active subcarrier set $\mathcal{N}_j$, shown as Fig. 4(a). If the power of the local CPU reaches its upper bound, the remainder of the available power is allocated to the subcarriers by the standard water-filling algorithm, as shown in Fig. 4(b).

Define the water level of the local computation power as L_0 and the water level of the communication power as L_n. In Fig. 4(a), where the local computation power is below the upper bound, L_0 can be written as

$$L_0 = \frac{1}{4\zeta_i} \left[\frac{(1+\tilde{\beta}_i)\ln 2}{(1+\tilde{\beta}_i - \tilde{\gamma}_i)\eta_i \bar{B}} \right]^2 L_n^2. \tag{32}$$

Equation (32) indicates that L_0 is proportional to the square of L_n.

Remark 3 (Impact of ζ_i and η_i)**.** The ratio between L_0 and the square of L_n is determined by the effective capacitance coefficient ζ_i and the CIR η_i. If ζ_i is large, which means the chip architecture of user i can't transform the power into the computation capacity effectively, the power allocation policy will allocate less power to the local computation. If η_i is large, which means the tasks have much computation and little input data, the policy will decrease the local computation power and allocate more power to the RF module to offload the computation to the MEC server. Obviously, the power allocation policy agrees with our intuition to the MEC system.

Remark 4 (Impact of Wireless Channel)**.** Wireless channel has a profound effect on the power allocation policy. If the channel of a subcarrier fades severely, therefore the channel gain is very small so that the reciprocal of channel gain is below the water level L_n (see Fig. 4(a)), the algorithm won't allocate any power to this subcarrier. Furthermore, if the channel gains of all the subcarriers are small enough as $\min_i(\frac{1}{g_{i,n}}) \le L_n$, we will allocate no power to the RF module and all the power is allocated to the CPU module. However, it is impossible that all the power is allocated to the RF module theoretically because the water-filling step for the local CPU is always zero.

C. Subcarrier Assignment

Proposition 5. Subcarrier should be allocated to the user who satisfies the following condition

$$i^* = \arg\max_{i \in \mathcal{K}} \left[(U'_i(C_i) + \beta_i - \gamma)\eta_i F_{i,n} \right]. \tag{33}$$

Proof. Whether the problem $\mathcal{P}_{org}$ is convex or not, Equation (28) is a necessary condition for the subcarrier indicator parameter $\boldsymbol{\rho}$. Since the value of $\rho_{i,n}$ can only be 1 or 0 in practice, exploiting Equation (28), we can obtain

$$\rho_{i,n}^{*} = \begin{cases} 1, & \lambda_n < (U'_i\left(C_i\right) + \beta_i - \gamma)\eta_i F_{i,n} \\ 0, & \lambda_n > (U'_i\left(C_i\right) + \beta_i - \gamma)\eta_i F_{i,n} \end{cases} \tag{34}$$

Since one subcarrier can only be assigned to at most one user, we should assign a proper value to the Lagrange multiplier λ_n, so that there is only one user whose $(U'_i\left(C_i\right) + \beta_i - \gamma)\eta_i F_{i,n}$ value is larger than λ_n. In other words, subcarrier n should be assigned to the user who have the largest $(U'_i\left(C_i\right) + \beta_i - \gamma)\eta_i F_{i,n}$, as stated in Proposition 5. □

Note that the second term of $F_{i,n}$ is small and can be ignored. Hence, $\eta_i F_{i,n}$ approximately equals the APC obtained through subcarrier n by user i.

The proposed subcarrier assignment criterion can be adopted for the MEC system with different utility functions. To further understand the subcarrier assignment criterion, we consider the following three utilities.

1) Sum APC Maximization: The utility function is $U = \sum_{i\in\mathcal{K}} w_i R_i$ where w_i represents user i's priority. $U'_i\left(R_i\right) = w_i, \forall i \in \mathcal{K}$. Thus, the subcarrier assignment criterion is recast as

$$i^{*} = \arg \max_{i\in\mathcal{K}} \left\{(w_i + \beta_i - \gamma)\eta_i F_{i,n}\right\}. \tag{35}$$

If the users have the same priority, Equation (35) implies that the system should assign the subcarrier to the user who can obtain the largest APC through it.

2) Proportional Fairness: Proportional fairness originated from Kelly's work [36, 37]. Users' APC $\mathbf{C}$ is proportionally fair when any change in $\mathbf{C}$ results in the sum of proportional changes being non-positive, *i.e.*,

$$\sum_{i\in\mathcal{K}} \frac{\tilde{C}_i - C_i}{C_i} \tag{36}$$

where $\tilde{C}_i$ is any other feasible value and C_i is the proportionally fair value for user i. Define the utility function $U_i\left(C_i\right) = \ln\left(C_i\right)$. When the subcarriers are assigned by maximizing $\sum_{i\in\mathcal{K}} \ln\left(C_i\right)$, we will obtain a proportionally fair result, which is proved in [36]. With this utility, the subcarrier assignment criterion is transformed to

$$i^{*} = \arg \max_{i\in\mathcal{K}} \left\{(\frac{1}{C_i} + \beta_i - \gamma)\eta_i F_{i,n}\right\}. \tag{37}$$

Ignoring the Lagrange multiplier γ, we can conclude that the system tends to assign the subcarrier to the user who has the largest ratio of the APC obtained through the subcarrier to the user's whole APC for now.

3) Max-Min Fairness: Also in [36, 37], to reach a max-min fairness among users, the author proposed the following utility function

$$U_i\left(C_i\right) = -\left[\ln\left(\frac{A}{C_i}\right)\right]^{a}, \tag{38}$$

where $a \to \infty$ and A is constant, which is large enough to make $C_i/A \in (0,1), \forall i \in \mathcal{K}$. Thus, the subcarrier assignment criterion is recast as

$$i^{*} = \arg \max_{i\in\mathcal{K}} \left\{(\frac{a}{C_i}\left[\ln\left(\frac{A}{C_i}\right)\right]^{a-1} + \beta_i - \gamma)\eta_i F_{i,n}\right\}. \tag{39}$$

Since $a \to \infty$, the large term $\left[\ln\left(A/C_i\right)\right]^{a-1}$ dominates the value of the formula. Thus, the equation can be simplified to $i^{*} = \arg\min_{i\in\mathcal{K}} C_i$, which implies that the system always assigns a subcarrier to the user with lower APC for now.

According to the criterion in Proposition 5, Algorithm 2 is obtained as follow.

Algorithm 2 Suboptimal Subcarrier Assignment Algorithm

```
Initialize:
    Set $\mathcal{M}_i = \varnothing \ \forall i \in \mathcal{K}, n \in \mathcal{N}$.
    Set $\mathcal{N}_{\text{remain}} = \mathcal{N}$.
repeat
    Set $c_{i,n} = 0$, $p_{i,0} = 0$ and $p_{i,n} = 0 \ \forall i \in \mathcal{K}, n \in \mathcal{N}$.
    For each user, allocate the power $P_i$ to the subcarriers
    in $\mathcal{M}_i \cup \mathcal{N}_{\text{remain}}$ utilizing Algorithm 1 and obtain $p_{i,0}$,
    $p_{i,n}$.
    $c_{i,n} = (U'_i(C_i) + \beta_i - \gamma)\eta_i F_{i,n}, \forall n \in \mathcal{N}_{\text{remain}}$.
    For each $n$, find $i^*$ with $c_{i^*,n} \geq c_{i,n}, \forall i \in \mathcal{K}$.
    For all $(n, i^*)$, find $n^*$ with $c_{i^*,n^*} \geq c_{i^*,n}$.
    Update $\mathcal{M}_{i^*} = \mathcal{M}_{i^*} \cup \{n^*\}$, $\mathcal{N}_{\text{remain}} = \mathcal{N}_{\text{remain}} - \{n^*\}$.
    $\rho_{i^*,n^*} = 1$.
until $\mathcal{N}_{\text{remain}} = \varnothing$ or $p_{i,n} = 0, \forall n \in \mathcal{N}_{\text{remain}}$.
```

Algorithm 2 stipulates that each user can only obtain at most one subcarrier in a loop. That is because, once a user gets a subcarrier, its indicator $c_{i,n}$ in the criterion changes. However, the indicators are not recalculated until the next power allocation process. Hence, Algorithm 2 finds the best user for each unassigned subcarrier first. If there are multiple subcarriers paired to user i, user i only choose the one with largest $c_{i,n}$.

D. Solution to the Dual Problem

There are two constraints affecting the APC that a user can obtain, the RPC constraint and the server capacity constraint. These two constraints are reflected by the Lagrange multipliers β_i and γ in the KKT conditions. In our previous analysis, these two Lagrange multipliers are regarded as constants. However, they affect the APC of every user during the resource allocation. Take γ as an example. When the sum of the computation offloaded by all the users is more than the server capacity, the policy should turn γ up to decrease the power allocated to the RF module, thereby lowering the offloaded computation.

These Lagrange multipliers need to be updated according to the dual problem. From (25), the dual problem is written as

$$\begin{aligned} &\mathcal{D}_1: \\ &\min \ \ D(\gamma, \boldsymbol{\beta}, \boldsymbol{\upsilon}, \boldsymbol{\lambda}) \\ &\text{s.t.} \quad \gamma \geq 0, \boldsymbol{\beta} \succeq 0, \boldsymbol{\upsilon} \succeq 0, \boldsymbol{\lambda} \succeq 0, \end{aligned} \tag{40}$$

where $D(\gamma, \boldsymbol{\beta}, \boldsymbol{\upsilon}, \boldsymbol{\lambda}) = \max\limits_{\tilde{\boldsymbol{P}}, \boldsymbol{\rho}} L(\tilde{\boldsymbol{P}}, \boldsymbol{\rho}, \boldsymbol{\beta}, \gamma, \boldsymbol{\upsilon}, \boldsymbol{\lambda})$. As the objective function of the dual problem is linear in the Lagrange multipliers, the dual problem is convex, and the Lagrange multipliers can be solved by subgradient projection method. In the previous subsection, we have solved $\boldsymbol{\upsilon}$ and $\boldsymbol{\lambda}$ by binary-search water-filling algorithm and analyzing the subcarrier assignment criterion respectively. Therefore, we only need to deal with γ and $\boldsymbol{\beta}$.

Proposition 6. For the dual problem $\mathcal{D}_1$, the subgradients and iteration methods of $D(\gamma, \boldsymbol{\beta}, \boldsymbol{\upsilon}, \boldsymbol{\lambda})$ give

$$\Delta\gamma = C_{\text{server}} - \sum_{i\in\mathcal{K}} \sum_{n\in\mathcal{N}} \eta_i \rho^*_{i,n} \bar{B} \log_2 \left(1 + p^*_{i,n} g_{i,n}\right), \quad (41)$$

$$\Delta\beta_i = C^*_i - C_{\text{req},i}, \quad (42)$$

$$\gamma(t+1) = [\gamma(t) - \tau_1(t)\Delta\gamma(t)]^+, \quad (43)$$

$$\beta_i(t+1) = [\beta_i(t) - \tau_2(t)\Delta\beta_i(t)]^+, \quad (44)$$

where t is the iteration index, $\tau_1(t)$, $\tau_2(t)$ are step sizes (positive and sufficiently small).

Proof. See Appendix D. □

By updating γ and $\boldsymbol{\beta}$ with above equations, we summarize the whole resource allocation procedure as Algorithm 3.

Algorithm 3 Iterative Resource Allocation Algorithm

```
Initialize:
    Set $t = 0$ and maximum iteration times $t_{\max}$.
    Set $\gamma(t)$, $\boldsymbol{\beta}(t)$ and allowable error $\delta$.
repeat
    Calculate computing power $p^*_{i,0}(t)$, transmitting power $p^*_{i,n}(t)$ and subcarrier indicator $\rho^*_{i,n}(t)$ utilizing Algorithm 2, $\forall i \in \mathcal{K}, n \in \mathcal{N}$.
    Update $\gamma(t+1)$ and $\boldsymbol{\beta}(t+1)$ from (43) and (44).
    if $\|\gamma(t+1) - \gamma(t)\|_2 < \delta$ and $\|\boldsymbol{\beta}(t+1) - \boldsymbol{\beta}(t)\|_2 < \delta$ then
        Close.
    end if
    if $\Delta\gamma < 0$ and $p_{i,0} = \min(P_i, f^2_{\max,i}\zeta)$ then
        Reach the system upper bound.
        Close.
    end if
    $t = t + 1$.
until $t > t_{\max}$.
```

The main loop in Algorithm 3 consists of the power and subcarrier allocation algorithms, which are described in Algorithm 1 and 2 respectively. Note that the Lagrange multiplier β_i can be ignored in Algorithm 1. Since the RPCs of users are independent and the power is allocated for each user respectively, $(1 + \tilde{\beta}_i)$ in (29) and (30) can be regarded as a part of the Lagrange multiplier $\tilde{\upsilon}_i$ and thus can be ignored in the water-filling process. Note that the step size function is supposed to make $(U'_i(C_i) + \beta_i - \gamma)$ maintain non-negative during the iteration process.

Remark 5 (Upper Bound of the Sum APC)**.** In the proposed MEC system, the sum APC can't grow unboundedly as the available power grows. The upper bound relies on the amount of all available computation resource in the system including all the mobile devices and the MEC server, which is given as

$$C_{\text{ub}} = \sum_{i\in\mathcal{K}} \min(f_{\max,i}, \sqrt{(P_i\zeta^{-1}}) + C_{\text{server}}. \quad (45)$$

To maximize the APC of the system, we have assumed that the users take full advantage of its available power by allocating the power to the local computation or RF modules in Algorithm 1. However, if the system doesn't have enough computation resource, the upper bound will be reached before all the power is consumed. In this case, the local CPU works with its available maximum frequency $(\min(f_{\max,i}, \sqrt{P_i\zeta^{-1}}))$ and the remaining power is all allocated to the RF module for the user with extra power. The Lagrange multiplier γ can't converge to a constant since $\Delta\gamma$ keeps negative due to the over-offloaded computation to the server. To solve that, we can decrease the transmitting power with Equation (30) to satisfy the system capacity constraint C1.

VI. Simulation Results and Discussion

In this section, we evaluate the performance of the proposed resource allocation policy utilizing Monte Carlo simulation of 2000 channel realizations. A single-user MEC system and a multiuser MEC system are considered. The parameters are referred in [38] and [24]. The channel gain of each subcarrier is written as $g_{i,n} = D_i ||a_{i,n}||^2$, where D_i is the large-scale fading path loss parameter and $||a_{i,n}||^2$ satisfies the standard Rayleigh distribution, with average bandwidth $\bar{B} = 1$MHz. The effective capacitance coefficient ζ is fixed to $[1, 10] \times 10^{-17}$, calculated from Table 1 in [38], and the CPU frequency of the users varies in $(0, 500]$MHz. The CIR belongs to $[500, 1500]$ followed in [24]. In the simulation, the default settings are $\eta = 1000$, $\zeta = 10^{-17}$, $D_i = 1$ and $f_{\max} = 500$MHz.

For the single-user MEC system with a single subcarrier, we assume that the available power of the equipment is set as $P = 1$W and the server capacity C_{server} is set to 2GHz. While the wireless channel varies, the power allocation solution changes as shown in Fig. 5. When the channel gain g is small (Case 3 in the figure), all the power is allocated to the local CPU module ($f^2_{\max}\zeta >$1W) and the RF module stays quiet. Then, Case 1 becomes true where part of the computation is offloaded to the MEC server. In this case, the power for CPU decreases and the power for RF module increases as the channel gain g increases. When the server capacity is exhausted, Case 4 becomes true where the power for RF module decreases and the remaining power is allocated to CPU to obtain extra income as g further increases.

Fig. 6 depicts the sum processing capacity of the multiuser MEC system versus the number of subcarriers with different server capacities. The available power for each user is fixed to 0.1W. The resource allocation policy with $U_i(C_i) = C_i$ is carried out under $K = 5$ and $C_{\text{server}} = 1.0, 1.2, 1.4, 2.0$GHz, respectively. As we can see, when the number of subcarriers increases, the sum APC increases rapidly and reaches the server capacity limit very soon. Furthermore, the figure indicates

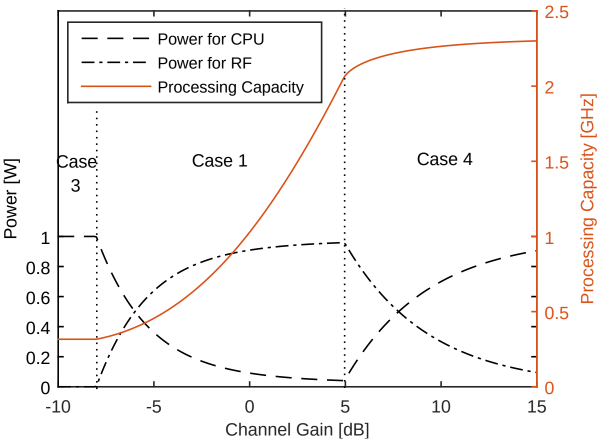


Fig. 5. The processing capacity and the power allocation versus channel gain for the single-user MEC system.

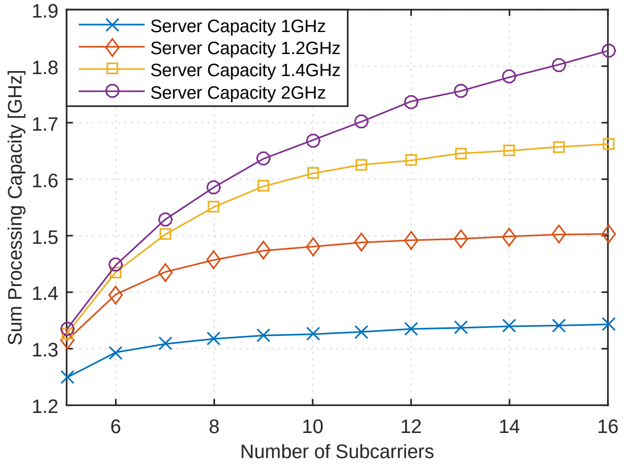


Fig. 6. The sum processing capacity versus the number of subcarriers for the multiuser MEC system.

that the MEC system with a higher server capacity can obtain a better performance.

As shown in Fig. 7, the available power, the CIR η and effective capacitance coefficient ζ also have significant affects on the sum processing capacity. We set $P_i = 0.1$W, $N = 10$, $K = 5$ and $C_{\text{server}} = 1.5$ and 2.0GHz here. It can be observed that the sum APCs with different server capacity and ζ have similar curves when the power is low and $\eta = 1000$. However, the system with higher server capacity has a higher upper bound and with a lower ζ can obtain more processing capacity from local CPU. Meanwhile, the CIR η determines the rate of the processing capacity rise when the power is low.

Fig. 8 evaluates the performances of different utility functions. In this case, the distances between users and the AP are assumed to be random and the large-scale fading path loss parameter D_i is distributed evenly over $(0, 5]$. The available power for each user is set to $P_i = 0.15$W to give better observations. It's evident that the algorithm with the APC maximization utility achieves the highest processing capacity for the MEC system, while the performance of the proportional fairness utility is closed to the former. The algorithm with the max-min fairness utility doesn't perform well on the sum APC, but it pays more attention to the users with bad power and communication conditions.

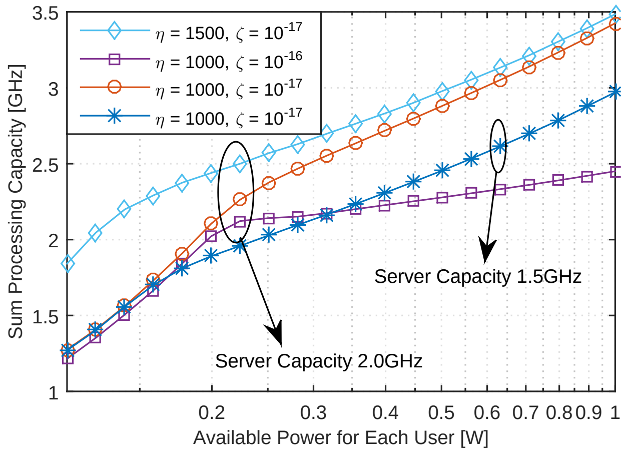


Fig. 7. The sum processing capacity versus the available power of each user for the multiuser MEC system.

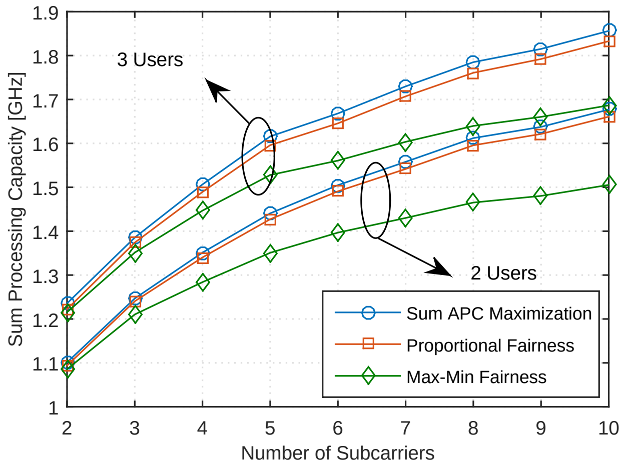


Fig. 8. The sum processing capacity versus the number of subcarriers with different utility functions.

VII. Conclusion

This paper has investigated a resource allocation policy based on the processing capacity for MEC IoT networks. For fully divisible tasks, we demonstrated that the optimal task partitioning is determined by the resource allocation policy, thus a user's APC can be represented as a concave function of power and subcarriers. Then, the factors that affect the APC improvement in the MEC system was revealed by analyzing the optimization problem of a single-user MEC system. After that, we proposed the power and subcarrier allocation algorithms for the multiuser MEC system, which satisfy users' RPC demands and meanwhile address concerns on the APC utility functions. Note that this paper aims to find the best-effort point where the APC is maximized for the MEC system, while the energy-efficient point, where the power consumption is minimized, can also be found easily. The APC avoids the complicated analysis on the arrival and termination of tasks. Therefore, the policy based on it is appropriate for IoT networks with unpredictable tasks and constrained power, which has been verified by Monte Carlo simulation.

For most of this paper, we have only discussed the resource allocation for a centralized MEC network with OFDMA. APC-based scheduling policies can be further investigated for other networks with other access modes in the future. For example, it is necessary to reallocate unbalanced processing capacities of the devices in wireless ad-hoc networks, in which each device may be used as an MEC server.

Appendix A

According to [39], we can always minimize a function by first minimizing over some of the variables and then minimizing over the remaining ones. Thus, Equation (6) can be rewritten as

$$t_{\text{cost},i} = \min_{\boldsymbol{P}_i} f(\boldsymbol{P}_i), \tag{46}$$

where

$$f(\boldsymbol{P}_i) = \min_{l_i} \ \max\left(\frac{(\alpha_i - l_i)\,\eta_i}{\sqrt{p_{i,0}{\zeta_i}^{-1}}}, \frac{l_i}{R_{\text{Tx},i}}\right). \tag{47}$$

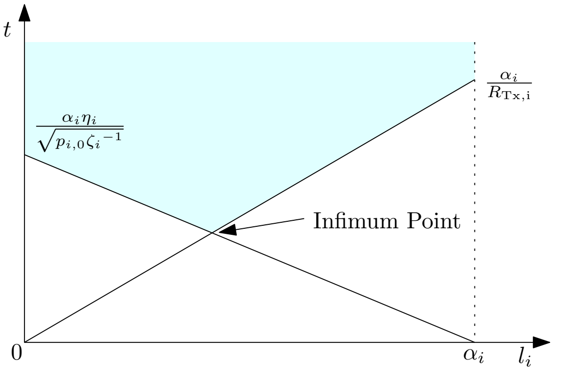


Fig. 9. The solution for the linear programming problem.

Since the task is infinitely divisible, the offloaded data l_i is a continuous variable over range $[0, \alpha_i]$. Equation (47) can be transformed to

$$\begin{aligned} &\min_{l_i} \ t \\ &\text{s.t.} \ \ \tfrac{(\alpha_i - l_i)\eta_i}{\sqrt{p_{i,0}{\zeta_i}^{-1}}} \le t, \ \ \tfrac{l_i}{R_{\text{Tx},i}} \le t, \end{aligned} \tag{48}$$

which is an equivalent linear programming problem of l_i. As shown in Fig. 9, the optimal point is always obtained at the cross of the two constraint lines, where l_i is the proposed value. By taking l_i into Equation (6), $t_{\text{cost},i}$ is rewritten as

$$t_{\text{cost},i} = \min_{\boldsymbol{P}_i,\boldsymbol{\rho}_i} \frac{\omega_i}{\sqrt{p_{i,0}{\zeta_i}^{-1}} + \eta_i R_{\text{Tx},i}}. \tag{49}$$

Equation (49) implies that we can always obtain the minimum execution latency by allocating the power of the user to the CPU module and the RF module properly for assigned wireless channels. Expanding it to an arbitrary power and subcarrier allocation, we have the lemma.

Appendix B

According to Section II, we can figure that $C_{\text{local}}(p_0) = \sqrt{p_0\zeta^{-1}}$ and $C_{\text{remote}}(p_1) = \eta\bar{B}\log_2(1 + p_1 g)$ are concave. Thus, the objective function $C = C_{\text{local}}(p_0) + C_{\text{remote}}(p_1)$ is concave and the required APC constraint $C_{\text{req}} - C$ is convex. In addition, $C_{\text{local}}(p_0) \le C_{f_{\max}}$ and $C_{\text{remote}}(p_1) \le C_{\text{server}}$ can be equivalently recast as $p_0 - \zeta C_{f_{\max}}^2 \le 0$ and $p_1 - g^{-1}(2^{C_{\text{server}}\eta^{-1}\bar{B}^{-1}} - 1) \le 0$. Hence, the rest constraints are all linear. Therefore, $\mathcal{P}_0$ is a convex optimization problem. Its partial Lagrangian function can be written as

$$L(p_0, p_1, \upsilon) = \sqrt{p_0\zeta^{-1}} + \eta\bar{B}\log_2(1 + p_1 g) + \upsilon(P - p_0 - p_1), \tag{50}$$

where $\upsilon \ge 0$ is the Lagrange multiplier for the maximum power constraint. Applying KKT conditions leads to the following results

$$\frac{\partial L}{\partial p_0^*} = \frac{1}{\sqrt{4p_0\zeta}} - \upsilon, \quad \frac{\partial L}{\partial p_1^*} = \frac{\eta\bar{B}/\ln 2}{1 + p_1 g} - \upsilon. \tag{51}$$

Make the partial derivatives equal 0, then carry them into the maximum power constraint. We can obtain

$$P + \frac{1}{g} = \frac{1}{4\zeta\upsilon^2} + \frac{\eta\bar{B}/\ln 2}{\upsilon}. \tag{52}$$

Solving the quadratic equation, we have $\frac{1}{\upsilon} = 2\zeta(\sqrt{(\eta\bar{B}/\ln 2)^2 + \frac{1}{\zeta}(P + \frac{1}{g})} - \eta\bar{B}/\ln 2)$. Thus, we have the proposition.

Appendix C

The perspective operation preserves convexity [39], that is, if $f(x)$ is a convex function, then so its perspective function $tf(x/t)$ where $t > 0$. Therefore, $\rho_{i,n}\bar{B}\log_2\left(1 + \frac{\tilde{p}_{i,n}}{\rho_{i,n}} g_{i,n}\right)$ is concave since it is the perspective function of $\bar{B}\log_2(1 + \tilde{p}_{i,n} g_{i,n})$. Meanwhile, $\sqrt{\tilde{p}_{i,0}{\zeta_i}^{-1}}$ is concave on $\mathbb{R}^+$. Since the sum of concave functions also preserves convexity, user i's APC (23) is concave. Note that $h(g(x))$ is concave, if $h(x)$ is concave, its extended-value extension function $\tilde{h}(x)$, which assigns the value $-\infty$ to points not in the domain of $h(x)$, is nondecreasing, and g(x) is concave according to Section 3.2.4 in [39]. Hence, user i's utility function $U_i(C_i)$ is concave. Because $U = \sum_{i\in\mathcal{K}} U_i$, U is concave.

Appendix D

Since $D(\gamma, \boldsymbol{\beta}, \boldsymbol{\upsilon}, \boldsymbol{\lambda}) = \max_{\tilde{\boldsymbol{P}},\boldsymbol{\rho}} L(\tilde{\boldsymbol{P}}, \boldsymbol{\rho}, \boldsymbol{\beta}, \gamma, \boldsymbol{\upsilon}, \boldsymbol{\lambda})$, we have

$$\begin{aligned} D(\gamma, \boldsymbol{\beta}', \boldsymbol{\upsilon}, \boldsymbol{\lambda}') \ge &\sum_{i\in\mathcal{K}} U_i\left(C_i^*\right) + \sum_{i\in\mathcal{K}} \beta_i\left[C_i^* - C_{\text{req},i}\right] \\ &+ \gamma\left[C_{\text{server}} - \sum_{i\in\mathcal{K}}\sum_{n=1}^{N} \eta_i \rho_{i,n}^* \bar{B}\log_2\left(1 + \frac{\tilde{p}_{i,n}^*}{\rho_{i,n}^*} g_{i,n}\right)\right] \\ &+ \sum_{i\in\mathcal{K}} \upsilon_i\left[P_i - \sum_{n=0}^{N} \tilde{p}_{i,n}^*\right] + \sum_{n=1}^{N} \lambda_n\left[1 - \sum_{i\in\mathcal{K}} \rho_{i,n}^*\right], \end{aligned} \tag{53}$$

where $\rho^*_{i,n}$ and $\tilde{p}^*_{i,n}$ are the optimal solutions corresponding to $\gamma, \boldsymbol{\beta}, \boldsymbol{\upsilon}, \boldsymbol{\lambda}$. (53) can be rearranged as

$$\begin{aligned} &D(\gamma, \boldsymbol{\beta}', \boldsymbol{\upsilon}, \boldsymbol{\lambda}') \geq D(\gamma, \boldsymbol{\beta}, \boldsymbol{\upsilon}, \boldsymbol{\lambda}) + \\ &\sum_{i \in \mathcal{K}} \left(\beta_i{}' - \beta_i\right) \left[C_i^* - C_{\text{req},i}\right] \\ &+ (\gamma' - \gamma) \left[C_{\text{server}} - \sum_{i \in \mathcal{K}} \sum_{n=1}^{N} \eta_i \rho^*_{i,n} \bar{B} \log_2 \left(1 + \frac{\tilde{p}^*_{i,n}}{\rho^*_{i,n}} g_{i,n}\right)\right]. \end{aligned} \tag{54}$$

Note that δ is defined as a subgradient of a convex function $f(\cdot)$ if $f(x') \geq f(x) + \delta(x' - x)$ holds for all x' and x in the domain. Hence, the proposition holds.

References

[1] L. Atzori, A. Iera, and G. Morabito, "The internet of things: A survey," *Comput. Networks*, vol. 54, no. 15, pp. 2787–2805, Oct. 2010.

[2] A. Zanella, N. Bui, A. Castellani, L. Vangelista, and M. Zorzi, "Internet of things for smart cities," *IEEE IoT J.*, vol. 1, no. 1, pp. 22–32, Feb. 2014.

[3] M. M. Waldrop, "The chips are down for moore's law," *Nature News*, vol. 530, no. 7589, pp. 144–147, 2016.

[4] H. D. Yoo, E. Markevich, G. Salitra, D. Sharon, and D. Aurbach, "On the challenge of developing advanced technologies for electrochemical energy storage and conversion," *Mater. Today*, vol. 17, no. 3, pp. 110–121, Apr. 2014.

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

[6] Q. Yuan, H. Zhou, J. Li, Z. Liu, F. Yang, and X. S. Shen, "Toward efficient content delivery for automated driving services: An edge computing solution," *IEEE Network*, vol. 32, no. 1, pp. 80–86, Jan. 2018.

[7] Y. Wu, L. P. Qian, H. Mao, X. Yang, H. Zhou, X. Tan, and D. H. K. Tsang, "Secrecy-driven resource management for vehicular computation offloading networks," *IEEE Network*, vol. 32, no. 3, pp. 84–91, May. 2018.

[8] J. Cao, A. Castiglione, G. Motta, F. Pop, Y. Yang, and W. Zhou, "Human-driven edge computing and communication: Part 1," *IEEE Commun. Mag.*, vol. 55, no. 11, pp. 70–71, Nov. 2017.

[9] ——, "Human-driven edge computing and communication: Part 2," *IEEE Commun. Mag.*, vol. 56, no. 2, pp. 134–135, Feb. 2018.

[10] H. Guo, J. Ren, D. Zhang, Y. Zhang, and J. Hu, "A scalable and manageable IoT architecture based on transparent computing," *J. Parallel Distrib. Comput.*, Jul. 2017.

[11] G. Premsankar, M. D. Francesco, and T. Taleb, "Edge computing for the internet of things: A case study," *IEEE IoT J.*, vol. 5, no. 2, pp. 1275–1284, Apr. 2018.

[12] H. T. Dinh, C. Lee, D. Niyato, and P. Wang, "A survey of mobile cloud computing: architecture, applications, and approaches," *Wireless Commun. Mob. Comput.*, vol. 13, no. 18, pp. 1587–1611, Oct. 2011.

[13] P. Wan and Z.-G. Wan, "Maximizing networking capacity in multi-channel multi-radio wireless networks," *J. Comput. Sci. Technol.*, vol. 29, no. 5, pp. 901–909, Sep. 2014.

[14] L. Yang, J. Cao, Y. Yuan, T. Li, A. Han, and A. Chan, "A framework for partitioning and execution of data stream applications in mobile cloud computing," *ACM SIGMETRICS Perform. Eval. Rev.*, vol. 40, no. 4, pp. 23–32, Mar. 2013.

[15] J. Liu, Y. Mao, J. Zhang, and K. B. Letaief, "Delay-optimal computation task scheduling for mobile-edge computing systems," in *Proc. ISIT*. IEEE, Jul. 2016, pp. 1451–1455.

[16] 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.

[17] 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.

[18] 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.

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

[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. Networks*, vol. 1, no. 2, pp. 89–103, Jun. 2015.

[21] X. Chen, L. Jiao, W. Li, and X. Fu, "Efficient multi-user computation offloading for mobile-edge cloud computing," *IEEE/ACM Trans. Networking*, vol. 24, no. 5, pp. 2795–2808, Oct. 2016.

[22] O. Munoz, A. Pascual-Iserte, and J. Vidal, "Optimization of radio and computational resources for energy efficiency in latency-constrained application offloading," *IEEE Trans. Veh. Technol.*, vol. 64, no. 10, pp. 4738–4755, Oct. 2015.

[23] Y. Mao, J. Zhang, S. Song, and K. B. Letaief, "Power-delay tradeoff in multi-user mobile-edge computing systems," in *Proc. GLOBECOM*. IEEE, Dec. 2016, pp. 1–6.

[24] C. You, K. Huang, H. Chae, and B.-H. Kim, "Energy-efficient resource allocation for mobile-edge computation offloading," *IEEE Trans. Wireless Commun.*, vol. 16, no. 3, pp. 1397–1411, Mar. 2017.

[25] C. Wang, C. Liang, F. R. Yu, Q. Chen, and L. Tang, "Computation offloading and resource allocation in wireless cellular networks with mobile edge computing," *IEEE Trans. Wireless Commun.*, vol. 16, no. 8, pp. 4924–4938, Aug. 2017.

[26] 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.

[27] Z. Tan, F. R. Yu, X. Li, H. Ji, and V. C. M. Leung, "Virtual resource allocation for heterogeneous services in full duplex-enabled scns with mobile edge computing and caching," *IEEE Trans. Veh. Technol.*, vol. 67, no. 2, pp. 1794–1808, Feb. 2018.

[28] W. Zhang, Y. Wen, K. Guan, D. Kilper, H. Luo, and D. O. Wu, "Energy-optimal mobile cloud computing under stochastic wireless channel," *IEEE Trans. Wireless Commun.*, vol. 12, no. 9, pp. 4569–4581, Sep. 2013.

[29] S. Ulukus, A. Yener, E. Erkip, O. Simeone, M. Zorzi, P. Grover, and K. Huang, "Energy harvesting wireless communications: A review of recent advances," *IEEE J. Sel. Areas Commun.*, vol. 33, no. 3, pp. 360–381, Mar. 2015.

[30] K. Kim, Y. Han, and S.-L. Kim, "Joint subcarrier and power allocation in uplink OFDMA systems," *IEEE Commun. Lett.*, vol. 9, no. 6, pp. 526–528, Jun. 2005.

[31] M. E. Gerards, J. L. Hurink, and J. Kuper, "On the interplay between global DVFS and scheduling tasks with precedence constraints," *IEEE Trans. Comput.*, vol. 64, no. 6, pp. 1742–1754, Jun. 2015.

[32] M. Patel, B. Naughton, C. Chan, N. Sprecher, S. Abeta, A. Neal *et al.*, "Mobile-edge computing introductory technical white paper," *White Paper, Mobile-edge Computing (MEC) industry initiative*, 2014.

[33] S.-W. Ko, K. Huang, S.-L. Kim, and H. Chae, "Live prefetching for mobile computation offloading," *IEEE Trans. Wireless Commun.*, vol. 16, no. 5, pp. 3057–3071, May. 2017.

[34] Y. Mao, C. You, J. Zhang, K. Huang, and K. B. Letaief, "Mobile edge computing: Survey and research outlook," *arXiv preprint arXiv:1701.01090*, 2017.

[35] D. W. K. Ng, E. S. Lo, and R. Schober, "Energy-efficient resource allocation in multi-cell OFDMA systems with limited backhaul capacity," *IEEE Trans. Wireless Commun.*, vol. 11,

no. 10, pp. 3618–3631, Oct. 2012.

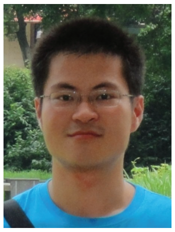
[36] F. Kelly, “Charging and rate control for elastic traffic,” *Trans. Emerging Telecommun. Technol.*, vol. 8, no. 1, pp. 33–37, Jan. 1997.

[37] F. P. Kelly, A. K. Maulloo, and D. K. Tan, “Rate control for communication networks: shadow prices, proportional fairness and stability,” *J. Oper. Res. Soc.*, vol. 49, no. 3, pp. 237–252, Apr. 1998.

[38] A. P. Miettinen and J. K. Nurminen, “Energy efficiency of mobile clients in cloud computing,” *HotCloud*, vol. 10, pp. 4–4, 2010.

[39] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge, U.K.: Cambridge Univ. Press, 2003, pp. 133–134.

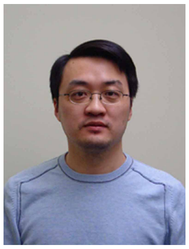
Min Qin received the B.S. from the Department of Electronic Engineering and Information Science (EEIS), University of Science and Technology of China (USTC), Hefei, China, in 2014. He is currently pursuing for his Ph.D. degree in Communication and Information Engineering at USTC. His current research interests include wireless channel estimation, mobile edge computing and optimization theory.



Li Chen received the B.E. in electrical and information engineering from Harbin Institute of Technology, Harbin, China, in 2009 and the Ph.D. degree in electrical engineering from the University of Science and Technology of China, Hefei, China, in 2014. He is currently a faculty member with the Department of Electronic Engineering and Information Science, University of Science and Technology of China. His research interests include wireless IoT communications and wireless optical communications.

Nan Zhao (S'08-M'11-SM'16) is currently an Associate Professor at Dalian University of Technology, China. He received the B.S. degree in electronics and information engineering in 2005, the M.E. degree in signal and information processing in 2007, and the Ph.D. degree in information and communication engineering in 2011, from Harbin Institute of Technology, Harbin, China. His recent research interests include UAV Communications, Caching and Computing, and Physical Layer Security.

Dr. Zhao is serving or served on the editorial boards of 7 SCI-indexed journals, including IEEE Transactions on Green Communications and Networking. He received Top Reviewer Award from IEEE Transactions on Vehicular Technology in 2016, and was nominated as an Exemplary Reviewer by IEEE Communications Letters in 2016. He won the best paper awards in IEEE VTC 2017 Spring, MLICOM 2017, ICNC 2018 and CSPS 2018. He also received the IEEE Communications Society Asia Pacific Board Outstanding Young Researcher Award in 2018.



Yunfei Chen (S'02-M'06-SM'10) received his B.E. and M.E. degrees in electronics engineering from Shanghai Jiaotong University, Shanghai, P.R.China, in 1998 and 2001, respectively. He received his Ph.D. degree from the University of Alberta in 2006. He is currently working as an Associate Professor at the University of Warwick, U.K. His research interests include wireless communications, cognitive radios, wireless relaying and energy harvesting.

F. Richard Yu (S'00-M'04-SM'08-F'18) received the PhD degree in electrical engineering from the University of British Columbia (UBC) in 2003. From 2002 to 2006, he was with Ericsson (in Lund, Sweden) and a start-up in California, USA. He joined Carleton University in 2007, where he is currently a Professor. He received the IEEE Outstanding Service Award in 2016, IEEE Outstanding Leadership Award in 2013, Carleton Research Achievement Award in 2012, the Ontario Early Researcher Award (formerly Premiers Research Excellence Award) in 2011, the Excellent Contribution Award at IEEE/IFIP TrustCom 2010, the Leadership Opportunity Fund Award from Canada Foundation of Innovation in 2009 and the Best Paper Awards at IEEE ICNC 2018, VTC 2017 Spring, ICC 2014, Globecom 2012, IEEE/IFIP TrustCom 2009 and Int'l Conference on Networking 2005. His research interests include wireless cyber-physical systems, connected/autonomous vehicles, security, distributed ledger technology, and deep learning.

He serves on the editorial boards of several journals, including Co-Editor-in-Chief for Ad Hoc & Sensor Wireless Networks, Lead Series Editor for IEEE Transactions on Vehicular Technology, IEEE Transactions on Green Communications and Networking, and IEEE Communications Surveys & Tutorials. He has served as the Technical Program Committee (TPC) Co-Chair of numerous conferences. Dr. Yu is a registered Professional Engineer in the province of Ontario, Canada, a Fellow of the Institution of Engineering and Technology (IET), and a Fellow of the IEEE. He is a Distinguished Lecturer, the Vice President (Membership), and an elected member of the Board of Governors (BoG) of the IEEE Vehicular Technology Society.

Guo Wei received the B.S. degree in electronic engineering from the University of Science and Technology of China (USTC), Hefei, China, in 1983 and the M.S. and Ph.D. degrees in electronic engineering from the Chinese Academy of Sciences, Beijing, China, in 1986 and 1991, respectively. He is currently a Professor with the School of Information Science and Technology, USTC. His current research interests include wireless and mobile communications, wireless multimedia communications, and wireless information networks.