Research Article

Joint Congestion Control and Resource Allocation for Delay-Aware Tasks in Mobile Edge Computing

Shichao Li $^{1,2}$, Qiuyun Wang $^{2}$, Yunfeng Wang $^{2}$, Jianli Xie $^{3}$, Cuiran Li $^{3}$, Dengtai Tan $^{2}$, Weigang Kou $^{2}$ and Wenjie Li $^{4}$

$^{1}$Guangxi Key Laboratory of Wireless Wideband Communication and Signal Processing, Guilin University of Electronic Technology, Guilin 541004, China
$^{2}$School of Evidence Law and Forensic Science, Gansu University of Political Science and Law, Lanzhou 730070, China
$^{3}$School of Electronic and Information Engineering, Lanzhou Jiaotong University, Lanzhou 730070, China
$^{4}$State Key Laboratory of Rail Traffic Control and Safety, Beijing Jiaotong University, Beijing 100044, China

Correspondence should be addressed to Qiuyun Wang; wqy6212@gsli.edu.cn

Received 27 June 2020; Revised 18 October 2020; Accepted 20 November 2020; Published 9 January 2021

Copyright © 2021 Shichao Li et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Recently, in order to extend the computation capability of smart mobile devices (SMDs) and reduce the task execution delay, mobile edge computing (MEC) has attracted considerable attention. In this paper, a stochastic optimization problem is formulated to maximize the system utility and ensure the queue stability, which subjects to the power, subcarrier, SMDs, and MEC server computation resource constraints by jointly optimizing congestion control and resource allocation. With the help of the Lyapunov optimization method, the primal problem is transformed into five subproblems including the system utility maximization subproblem, SMD congestion control subproblem, SMD computation resource allocation subproblem, joint power and subcarrier allocation subproblem, and MEC server scheduling subproblem. Since the first three subproblems are all single variable problems, the solutions can be obtained directly. The joint power and subcarrier allocation subproblem can be efficiently solved by utilizing alternating and time-sharing methods. For the MEC server scheduling subproblem, an efficient algorithm is proposed to solve it. By solving the five subproblems at each slot, we propose a delay-aware task congestion control and resource allocation (DTCCRA) algorithm to solve the primal problem. Theoretical analysis shows that the proposed DTCCRA algorithm can achieve the system utility and execution delay trade-off. Compared with the intelligent heuristic (IH) algorithm, when the control parameter $V$ increases from $10^6$ to $10^7$, the total backlogs are decreased by 5.03% and the system utility is increased by 3.9% on average for the extensive performance by using the proposed DTCCRA algorithm.

1. Introduction

With the recent advancement of wireless communication technology and the Internet of Things (IoT), smart mobile devices (SMDs) gain enormous popularity, which can support many intelligent applications, such as map navigation, interactive gaming, and virtual reality [1, 2]. These novel services and applications require intensive computation resources and strict delay. However, since the limitation of the resources (e.g., CPU-cycle frequency, storage capability, and energy), these applications cannot be executed by the SMDs. Therefore, the novel solutions extending the computation capability and reducing the task execution delay are emerging [3, 4]. Among them, mobile edge computing (MEC) has been proposed as a novel approach by offloading part or all of the data from the SMDs to the computation server on the edge of the network [5–7]. By utilizing the MEC framework, resource utilization can be improved, and the computation capabilities of SMDs can be enhanced. Compared with the traditional cloud servers, MEC architecture has the advantages of low overhead and low latency [8]. However, considering the scalable deployment and economy, the MEC server has limited computation capability [9, 10]. In order to improve the system utility and resource utilization, novel resource allocation schemes should be carefully designed.
Recently, task computation offloading and resource allocation in MEC have become a hot topic. There are mainly two different aspects to study task offloading and resource allocation. The first aspect is to consider the static task offloading strategies. The new tasks cannot be randomly generated before the old tasks are executed. A joint offloading selection, radio, and computation resource allocation scheme was proposed to minimize the SMDs energy consumption, where a suboptimal result was obtained by utilizing the reformulation-linearization-technique-based branch-and-bound method [11]. In order to achieve the trade-off between energy consumption and latency, an energy-aware offloading scheme was proposed by jointly optimizing communication and computation resources with the energy and sensitive latency constraints [12]. Wang et al. presented an alternating direction method of multiplier (ADMM) algorithm for task offloading, resource allocation, and Internet caching optimization to maximize the MEC system utility [13]. The second aspect focuses on the dynamic task offloading strategies. The tasks are randomly generated at each slot. Lyu et al. proposed the optimal offloading schedules for MEC under partial network knowledge, where a perturbed Lyapunov function was formulated to maximize the system utility function [14]. In order to minimize the mobile device power consumption, a Lyapunov optimization-based scheduling scheme for energy-efficient offloading for multicore mobile devices was proposed [15]. When considering the different timescales in the task execution process and the channel fading process, a Markov decision process-based task scheduling scheme was proposed [16]. All the previous works assumed that the randomly generated tasks are inside the network capacity region. However, when the randomly generated tasks exceed the network capacity region, the network will be congested. Therefore, considering the stability of the MEC system, congestion control is necessary. Moreover, for a lot of 5G services, such as the online game and virtual reality, both task execution delay and system utility are important performance metrics. Large delay and low system utility will reduce the quality of experience (QoE) of users [17, 18]. Therefore, how to make full use of the resources and achieve the trade-off between system utility and execution delay are challenging issues.

This paper focuses on the joint congestion control and resource allocation with power, subcarrier, and computation resource constraints of SMDs and MEC server. With the help of the Lyapunov optimization method, the primal problem is transformed into five subproblems. Since the first three subproblems are all single variable problems, the solutions can be obtained directly. For the joint power and subcarrier subproblem, the alternating and time-sharing methods are utilized to solve it efficiently. For the MEC server scheduling subproblem, we propose an efficient algorithm. By utilizing the results of five separate subproblems, a delay-aware task congestion control and resource allocation (DTCCRA) algorithm is proposed. The performance bound of the proposal is analyzed, indicating that the system utility and task execution delay can achieve an \( \theta(1/V), \theta(V) \) trade-off. Compared with the intelligent heuristic (IH) algorithm, when the control parameter \( V \) increases from \( 10^6 \) to \( 10^7 \), the total backlogs are decreased by 5.03% and the system utility is increased by 3.9% on average for the extensive performance by using the proposed DTCCRA algorithm.

The rest of this paper is organized as follows. The system model is introduced in Section 2. The system utility maximization problem formulation and transformation are provided in Section 3. In Section 4, five subproblems are transformed by utilizing the Lyapunov optimization method. By solving the five subproblems, the DTCCRA algorithm is proposed in Section 5. The performance bound of the proposed DTCCRA algorithm is analyzed in Section 6. Simulation results are given in Section 7, and the paper is concluded in Section 8. The notations used in this paper are summarized in Table 1.

2. System Model

We first describe the system model of MEC, including the transmission model, SMD computation model, MEC server scheduling model, and queue models in this section.

2.1. Transmission Model. Figure 1 shows the proposed MEC offloading scenario. The system consists of one eNodeB which is connected to the MEC server via a fiber-wired link, and \( I \) SMDs; each SMD has one task to be executed. Let \( \mathcal{I} = \{1, \ldots, I\} \) denote the set of SMD indices, which can also be viewed as the set of task indices.

For the task offloading, we consider the conventional OFDMA system. \( W \) is the available bandwidth of the system, and the whole system bandwidth is divided into \( K \) subcarriers. Therefore, each subcarrier bandwidth is \( W/K \). The set of subcarrier is denoted by \( \mathcal{K} = \{1, \ldots, K\} \). Since the subcarriers are orthogonally allocated to the SMDs, there is no interference between them. We denote a binary variable \( \alpha_{ik}(t) \in \{0, 1\} \) to indicate that subcarrier \( k \) is allocated to SMD \( i \) at slot \( t \) with \( \alpha_{ik}(t) = 1 \), and \( \alpha_{ik}(t) = 0 \) otherwise. Each slot duration is \( \tau \). We denote \( g_{ik}(t) \) as the channel state information (CSI) of SMD \( i \) on subcarrier \( k \) at slot \( t \). Note that the CSI accounts for fast fading, path loss, shadow fading, antenna gain, and noise. \( g_{ik}(t) \) represents the transmission power of SMD \( i \) on subcarrier \( k \) at slot \( t \). Therefore, the transmission bits of task in SMD \( i \) to the MEC server at slot \( t \) can be given by

\[
d_{\text{trans},i}(t) = \sum_{k \in \mathcal{K}} \alpha_{ik}(t) \frac{W}{K} \tau \log_2(1 + g_{ik}(t)p_{ik}(t)).
\] (1)

2.2. SMD Computation Model. Each task can be partially executed on the SMD and MEC server. We denote \( s_i \) as the number of required CPU-cycle frequency to accomplish one bit of task in SMD \( i \).

When the tasks are partially executed by the local SMDs, let \( f_{\text{loc},i}(t) \) be the CPU-cycle frequency allocated to task in SMD \( i \) at slot \( t \), and the maximal CPU-cycle frequency of SMD \( i \) is \( f_{\text{loc},i}^{\max} \). Therefore, the bits of task executed by the local SMD \( i \) at slot \( t \) is given by [19]

\[
d_{\text{loc},i}(t) = \tau f_{\text{loc},i}(t)s_i^{-1}. \tag{2}
\]
The power consumption for the local SMD $i$ at slot $t$ is given by
\[ p_{\text{loc},i}(t) = \kappa_{\text{loc}} f_{\text{loc},i}(t), \]  
where $\kappa_{\text{loc}}$ is the conversion coefficient of SMDs, and it is determined by the CPU chip architecture [20].

2.3. MEC Server Scheduling Model. There are $M$-core CPU in the MEC server, let $\mathcal{M} = \{1, \cdots, M\}$ denote the set of CPU
cores. Let $f_{\text{ser},m}(t)$ represent the CPU-cycle frequency of CPU core $m$ allocated to tasks at slot $t$, and the maximal CPU-cycle frequency of CPU core $m$ be $f_{\text{ser}}^{\text{max}}$. Each CPU core can generate bits from different SMDs. Therefore, the bits of task in SMD $i$ executed by the CPU cores in the MEC server at slot $t$ should satisfy

$$
\sum_{i \in \mathcal{F}} d_{\text{ser},i}(t) s_i \leq \sum_{m \in \mathcal{M}} f_{\text{ser},m}(t) r_i,
$$

which means that the allocated CPU cycles should be no larger than the total CPU cycles in the MEC server.

The power consumption for CPU cores in the MEC server at slot $t$ is given by

$$
P_{\text{ser}}(t) = \sum_{m \in \mathcal{M}} \kappa_{\text{ser}} f_{\text{ser},m}(t)^3,
$$

where $\kappa_{\text{ser}}$ is the conversion coefficient of the CPU core, and it is determined by the CPU chip architecture [20].

2.4. Queue Models. In order to describe the task offloading process from the SMDs to the MEC server, we formulate two dynamic queue models in this system. The task offloading process is shown in Figure 2; firstly, the task is executed in the SMDs, including task computation and task transmission. And then, the task is executed in the MEC server [21]. For these two phases, we can formulate task execution queues in the SMDs and task execution queues in the MEC server, respectively.

For the task execution queues in the SMDs. We denote $H(t) = [H_1(t), \cdots, H_i(t)]$ as the backlogs of the tasks executed by the SMDs at slot $t$. $A(t) = [A_1(t), \cdots, A_i(t)]$ is the generated bits vector in each SMD at slot $t$. Generally speaking, the generated bits of each task normally follows a truncated Poisson distribution with the average arrival rate $\lambda_i = E[A_i(t)]$. We denote $B_i$ as the maximum generated bits of task in SMD $i$. At each slot, the SMDs need to decide the number of bits from the newly generated bits to be stored in the buffers. Let $r_i(t) \in [0, A_i(t)]$ denote the congestion control. Therefore, the task in the SMD $i$ execution queue model can be expressed as

$$
H_i(t + 1) = \max \left[ H_i(t) - d_{\text{sum},i}(t), 0 \right] + r_i(t),
$$

where $d_{\text{sum},i}(t) = d_{\text{trans},i}(t) + d_{\text{loc},i}(t)$.

For the task execution queues in the MEC server. We denote $Q(t) = [Q_1(t), \cdots, Q_i(t)]$ as the backlogs of the tasks executed by the MEC server at slot $t$. According to Equation (1), we can obtain the task execution queues in the MEC server for the task in SMD $i$ at slot $t + 1$ as

$$
Q_i(t + 1) = \max \left[ Q_i(t) - d_{\text{ser},i}(t), 0 \right] + d_{\text{trans},i}(t).
$$

3. Problem Formulation and Transformation

In this section, we first formally formulate the stochastic problem with the objective function maximizing the system utility function with power, subcarrier, SMDs, and MEC server computation resource constraints. Furthermore, with the help of the Lyapunov optimization method, we transform the problem into an equivalent problem.

3.1. Queue Stability and Problem Formulation. The objective function is to maximize the average system utility in this paper. $U_i(r_i)$ denotes the system utility function, which means executed bits benefit for the task in SMD $i$. $U_i(r_i)$ can be regarded as a function of nondecreasing concave continuous with the task in SMD $i$. A typical example of $U(\cdot)$ is $\log(\cdot)$. By using it, the proportional fairness among SMDs can be achieved.

In this work, for the network stability, we first define the long-term time-average expectation of any quantity $z$ as

$$
\bar{z} := \lim_{T \to \infty} \frac{1}{T} \sum_{t=0}^{T-1} E[z(t)].
$$

For example, $\overline{H_i}$ can be viewed as the average execution queue backlogs for task in SMD $i$.

In order to model the impacts of joint congestion control and resource allocation on the system utility, the definitions of queue stability are given [22, 23].

Definition 1. When $\overline{H_i} < \infty$, we define the single queue $H(t)$ is strongly stable.

Definition 2. When $\lim_{T \to \infty} \frac{\sum t \in [0, T]}{T} = 0$, we define the single queue $H(t)$ is mean rate stable.
Let \( r = \{ r_i(t), i \in \mathcal{I} \} \), \( p = \{ p_{ik}(t), i \in \mathcal{I}, k \in \mathcal{K} \} \), \( a = \{ a_{ik}(t), i \in \mathcal{I}, k \in \mathcal{K} \} \), and \( f_{loc} = \{ f_{loc}(t), i \in \mathcal{I} \} \) denote the vectors of congestion control, SMD power, subcarrier, and CPU-cycle frequency allocation, respectively. Let \( d_{ser} = \{ d_{ser}(t), i \in \mathcal{I} \} \), \( f_{ser} = \{ f_{ser,m}(t), m \in \mathcal{M} \} \) denote the vectors of bits of task execution and CPU-cycle frequency allocation in the MEC server, respectively. We denote the system operation as \( \rho = \{ r, p, a, f_{loc}, d_{ser}, f_{ser} \} \).

In order to maximize the system utility and ensure the queue stability by jointly optimizing congestion control and resource allocation for delay-aware tasks, the problem can be formulated as

\[
\text{(P1)} \quad \max \rho \sum_{i \in \mathcal{I}} U_i(\tau_i) \tag{9a}
\]

subject to

\[
\sum_{k \in \mathcal{K}} \alpha_{ik}(t)p_{ik}(t) \leq P_{\text{max}_{\text{trans,i}}}, \quad \forall i \in \mathcal{I}, \tag{9b}
\]

\[
P_{\text{loc,i}}(t) \leq P_{\text{max}_{\text{loc,i}}}, \quad \forall i \in \mathcal{I}, \tag{9d}
\]

\[
r_i(t) \in [0, A_i(t)], \quad \forall i \in \mathcal{I}, \tag{9e}
\]

\[
f_{loc,i}(t) \in [0, f_{\text{loc,i}}^{\text{max}}], \quad \forall i \in \mathcal{I}, \tag{9f}
\]

\[
\sum_{i \in \mathcal{I}} \alpha_{ik}(t) \leq 1, \quad \forall k \in \mathcal{K}, \tag{9g}
\]

\[
\alpha_{ik}(t) = \{ 0, 1 \}, \quad \forall i \in \mathcal{I}, \forall k \in \mathcal{K}, \tag{9h}
\]

\[
d_{ser,i}(t)s_i \leq \sum_{m \in \mathcal{M}} f_{ser,m}(t) - T_i, \quad \forall i \in \mathcal{I}, \tag{9i}
\]

\[
d_{ser,i}(t) \geq 0, \quad \forall i \in \mathcal{I}, \tag{9j}
\]

\[
f_{ser,m}(t) \in [0, f_{\text{ser,m}}^{\text{max}}], \quad \forall m \in \mathcal{M}, \tag{9k}
\]

where \( P_{\text{max}_{\text{trans,i}}} \) and \( P_{\text{max}_{\text{loc,i}}} \) are the maximum transmission power and task execution power of the SMD \( i \), respectively. \( P_{\text{ser}} \) is the average execution power of the MEC server. Constraint (9b) corresponds to queue stability. Constraints (9c) and (9d) are the transmission power and task execution power constraints, respectively. Constraint (9e) ensures that the amount of admitted data cannot exceed the amount of arrived data. Constraint (9f) corresponds to SMD computation resources. Constraints (9g) and (9h) guarantee that one subcarrier cannot be allocated to more than one SMD. Constraints (9i) and (9j) are the MEC server executed bits of task constraints. Constraint (9k) is the MEC server computation resources. Constraint (9l) is the power constraint in the MEC server. Considering the long-term average expectation and binary variables [24], we transform the problem in the following subsection.

3.2. Problem Transformation. It is difficult to handle problem (P1) since it involves maximizing the long-term average system utility. In this subsection, the Lyapunov optimization method is applied to transform problem (P1) as an equivalent problem [22]. The transformation can be achieved by introducing auxiliary variables \( \gamma_i(t) \). Then, the corresponding virtual queues \( Z_i(t) \) are given by

\[
Z_i(t + 1) = \max \left[ Z_i(t) - r_i(t), 0 \right] + \gamma_i(t), \quad \forall i \in \mathcal{I}, \tag{10}
\]

where the initial value \( Z_i(0) = 0, r_i(t) \) can be regarded as the executed bits of virtual queue \( Z_i(t) \), and \( \gamma_i(t) \) can be regarded as the generated bits of virtual queue \( Z_i(t) \), respectively.

For the average power constraint (9l), we also define the virtual queues \( X_i(t) \), which can be expressed as

\[
X_i(t + 1) = \max \left[ X_i(t) - P_{\text{ser}}^{\text{avg}}, 0 \right] + P_{\text{ser}}(t), \quad \forall i \in \mathcal{I}, \tag{11}
\]

where \( P_{\text{ser}}^{\text{avg}} \) and \( P_{\text{ser}}(t) \) can be regarded as the executed and generated bits of virtual queue \( X_i(t) \), respectively.

We introduce \( \gamma = \{ \gamma_i(t), i \in \mathcal{I} \} \) as the vector of auxiliary variables. Then problem (P1) can be transformed into the following equivalent problem

\[
\text{(P2)} \quad \max \rho \sum_{i \in \mathcal{I}} U_i(\gamma_i) \tag{12a}
\]

subject to

\[
\sum_{i \in \mathcal{I}} \gamma_i \leq \tau_i, \quad \forall i \in \mathcal{I}, \tag{12b}
\]

\[
0 \leq \gamma_i(t) \leq B_i, \quad \forall i \in \mathcal{I}, \tag{12c}
\]

\[
H_i(t), Q_i(t), Z_i(t), \text{ and } X_i(t) \text{ are strongly stable}, \tag{12d}
\]

\[
\gamma_i = \gamma_i(t) \tag{12e}
\]

**Proposition 1.** The optimal solution of problem (P1) can be turned into the optimal solution of problem (P2).

**Proof.** See Appendix A.

Since problem (P2) has a long-term average expectation and binary variables, it is hard to handle [24]. In the following section, we propose an efficient algorithm to solve problem (P2).

4. Problem Transformation

We take advantage of the Lyapunov optimization method to transform problem (P2) into five subproblems in this section.

4.1. Lyapunov Drift. We define \( Z(t) \) and \( X(t) \) as the vector of virtual queues \( Z_i(t) \) and \( X_i(t) \), respectively. \( \Theta(t) \) is the combined vector of virtual queues and actual queues, which is given by

\[
\Theta(t) = [H^T(t), Q^T(t), Z^T(t), X^T(t)]^T. \tag{13}
\]

The quadratic Lyapunov function can be written as [25]

\[
L(\Theta(t)) = \frac{1}{2} \sum_{i \in \mathcal{I}} (H_i(t))^2 + (Q_i(t))^2 + (Z_i(t))^2 + (X_i(t))^2. \tag{14}
\]

In order to show the stability property of the queuing systems, we employ the well-developed stability theory in
Markov chains using the Lyapunov drift. Based on the Lyapunov function, we define the Lyapunov drift as the expected change in the Lyapunov function from one slot to the next, which is given by [25]
\[
\Delta(\Theta(t)) = E[L(\Theta(t+1)) - L(\Theta(t))|\Theta(t)].
\] (15)

Based on the virtual queues and actual queues above, the upper bound of \(\Delta(\Theta(t))\) is derived in the following proposition [25].

**Proposition 2.** According to the Lyapunov optimization method, the upper bound of \(\Delta(\Theta(t))\) can be obtained from the following inequality:
\[
\Delta(\Theta(t)) \leq D + E[G(t)|\Theta(t)],
\] (16)
where
\[
G(t) = \sum_{i \in J} H_i(t)[r_i(t) - d_{sum,i}(t)] + \sum_{i \in J} Q_i(t)[d_{trans,i}(t) - d_{ser,i}(t)]
\nonumber
+ \sum_{i \in J} Z_i(t)[y_i(t) - r_i(t)] + \sum_{i \in J} X_i(t)p_{ser}(t) - P_{ser}^m
\] (17)
and \(D\) is a constant.

**Proof.** See Appendix B.

4.2. Problem Transformation. Instead of minimizing the upper bound of \(\Delta(\Theta(t))\) directly, the drift-plus-penalty theory can be utilized to minimize the following “drift-plus-penalty” problem [26].

\[
\min_{\mathcal{P}} \mathbb{E}\left[G(t) - V \sum_{i \in J} U_i(y_i(t)y_i(t))\right] \quad (18a)
\nonumber
s.t. 0 \leq y_i(t) \leq B_i, \quad \forall i \in \mathcal{J}, \quad (18b)
0 \leq r_i(t) \leq A_i(t), \quad \forall i \in \mathcal{J}, \quad (18c)
0 \leq f_{loc,i}(t) \leq f_{i max}^{loc}, \quad \forall i \in \mathcal{J}, \quad (18d)
\kappa f_{loc,i}(t) \leq f_{i max}^{loc}, \quad \forall i \in \mathcal{J}, \quad (18e)
\sum_{k \in \mathcal{K}} \alpha_k(t) p_{ik}(t) \leq p_{i max}^{trans,i}, \quad \forall i \in \mathcal{J}, \quad (18f)
\sum_{i \in \mathcal{J}} \alpha_k(t) \leq 1, \quad \forall k \in \mathcal{K}, \quad (18g)
\alpha_k(t) = \{0, 1\}, \quad \forall k \in \mathcal{K}, \quad (18h)
\sum_{i \in J} d_{ser,i}(t) s_i \leq \sum_{m \in \mathcal{M}} f_{ser,m}(t) \tau, \quad (18i)
d_{ser,i}(t) \geq 0, \quad \forall i \in \mathcal{J}, \quad (18j)
f_{ser,m}(t) \in [0, f_{i max}^{ser,m}], \quad \forall m \in \mathcal{M}, \quad (18k)
\]
where \(V \geq 0\) represents an arbitrary control parameter, which can be used to control the system utility and delay trade-off. A larger \(V\) means that more emphasis is put on the system utility during the optimization. On the other hand, when \(V\) is small, queue stability carries more weight during the optimization [27].

Considering Equations (18a), (18b), (18c), (18d), (18e), (18f), (18g), (18h), (18i), (18j), and (18k) has a separable structure, we can transform it into five separate subproblems. The system utility maximization subproblem can be obtained by splitting \(y_i(t)\) as
\[
\max \sum_{i \in J} [V U_i(y_i(t)) - Z_i(t)y_i(t)] \quad (19a)
\nonumber
s.t. 0 \leq y_i(t) \leq B_i, \quad \forall i \in \mathcal{J}. \quad (19b)
\]

Similarly, the congestion control subproblem can be obtained by splitting \(r_i(t)\) as
\[
\max \sum_{i \in J} [Z_i(t) - H_i(t)]r_i(t) \quad (20a)
\nonumber
s.t. 0 \leq r_i(t) \leq A_i(t), \quad \forall i \in \mathcal{J}. \quad (20b)
\]

The SMD computation resource allocation subproblem can be obtained by splitting \(f_{loc,i}(t)\) as
\[
\max \sum_{i \in J} H_i(t) f_{loc,i}(t)s_i^{-1} \quad (21a)
\nonumber
s.t. 0 \leq f_{loc,i}(t) \leq f_{i max}^{loc}, \quad \forall i \in \mathcal{J}, \quad (21b)
\kappa f_{loc,i}(t) \leq f_{i max}^{loc}, \quad \forall i \in \mathcal{J}. \quad (21c)
\]

The joint power and subcarrier allocation subproblem can be obtained by splitting \(p_{ik}(t)\) and \(\alpha_k(t)\) as
\[
\max \sum_{i \in J} (H_i(t) - Q_i(t))d_{trans,i}(t) \quad (22a)
\nonumber
s.t. \sum_{k \in \mathcal{K}} \alpha_k(t) p_{ik}(t) \leq p_{i max}^{trans,i}, \quad \forall i \in \mathcal{J}, \quad (22b)
\sum_{i \in \mathcal{J}} \alpha_k(t) \leq 1, \quad \forall k \in \mathcal{K}, \quad (22c)
\alpha_k(t) = \{0, 1\}, \quad \forall i \in \mathcal{J}, \quad \forall k \in \mathcal{K}. \quad (22d)
\]

The MEC server scheduling subproblem can be obtained by splitting \(d_{ser,i}(t)\) and \(f_{ser,m}(t)\) as
\[
\max \sum_{i \in J} [Q_i(t)d_{ser,i}(t) - X_i(t)p_{ser}(t)] \quad (23a)
\nonumber
s.t. \sum_{i \in J} d_{ser,i}(t)s_i \leq \sum_{m \in \mathcal{M}} f_{ser,m}(t) \tau, \quad (23b)
d_{ser,i}(t) \geq 0, \quad \forall i \in \mathcal{J}, \quad (23c)
f_{ser,m}(t) \in [0, f_{i max}^{ser,m}], \quad \forall m \in \mathcal{M}. \quad (23d)
\]
5. The Proposed Delay-Aware Task Congestion Control and Resource Allocation Algorithm

By solving the five separate subproblems in the last section, the DTCCRA algorithm is proposed.

5.1. System Utility Maximization Subproblem. For the system utility maximization subproblem, we can transform Equations (19a) and (19b) as

\[
\max_{y_i(t)} VU_i(y_i(t)) - Z_i(t)\gamma_i(t) \tag{24a}
\]

\[
s.t. 0 \leq \gamma_i(t) \leq B_i. \tag{24b}
\]

In order to solve Equations (24a) and (24b), the format of the system utility function is the key point. Aiming at obtaining the system utility function as

\[U_i(y_i) = \omega_i \min \{y_i, x_i/\lambda_i\}, \tag{25}\]

where \(\omega_i\) is the weight of task in SMD \(i\) and \(x_i\) is the maximum desired offloading ratio of task in SMD \(i\) [23]. In general, \(0 \leq x_i \leq 1\) and \(0 \leq \lambda_i \leq B_i\). Therefore, the solution of Equations (24a) and (24b) is

\[
\gamma_i(t) = \begin{cases} \frac{x_i}{\lambda_i}, & \text{if } Z_i(t) \leq V \omega_i, \\ 0, & \text{otherwise}. \end{cases} \tag{26}\]

5.2. SMD Congestion Control Subproblem. For the SMD congestion control subproblem, we can transform Equations (20a) and (20b) as

\[
\max_{r_i(t)} [Z_i(t) - H_i(t)]r_i(t) \tag{27a}
\]

\[
s.t. 0 \leq r_i(t) \leq A_i(t). \tag{27b}
\]

The optimal solution of Equations (27a) and (27b) is related to the queue backlogs of \(H_i(t)\) and \(Z_i(t)\), which can be expressed as

\[
r_i(t) = \begin{cases} A_i(t), & \text{if } Z_i(t) \geq H_i(t), \\ 0, & \text{otherwise}. \end{cases} \tag{28}\]

From Equation (28), we can find that if \(Z_i(t) \geq H_i(t)\), the newly generated bits can be admitted into the SMD \(i\). On the contrary, if \(Z_i(t) < H_i(t)\), the newly generated bits are dropped.

5.3. SMD Computation Resource Allocation Subproblem. For the SMD computation resource allocation subproblem, we can transform Equations (21a), (21b), and (21b) as

\[
\max_{f_{loc,i}(t)} H_i(t) f_{loc,i}(t) s_i^{-1} \tag{29a}
\]

\[
s.t. 0 \leq f_{loc,i}(t) \leq \frac{P_{\text{max,i}}}{K_i}. \tag{29b}
\]

For Equations (29a), (29b), and (29c), when \(H_i(t)\), \(r\), and \(s_i\) are constant, the optimal SMD computation resource allocation is

\[f_{loc,i}(t) = \min \left\{ \frac{P_{\text{max,i}}}{K_i}, \frac{P_{\text{max}}}{N_i} \right\}. \tag{30}\]

5.4. Joint Power and Subcarrier Allocation Subproblem. The joint power and subcarrier allocation subproblem can be expressed as

\[
(P3) \max_{\alpha_{ik}, \beta_{ik}} \sum_{i \in \mathcal{I}} (H_i(t) - Q_i(t))d_{\text{trans},i}d_{\text{trans},i}(t) \tag{31a}
\]

\[
s.t. \sum_{i \in \mathcal{I}} \alpha_{ik}(t)p_{ik}(t) \leq P_{\text{max,trans}}, \quad \forall i \in \mathcal{I}, \tag{31b}
\]

\[
\sum_{i \in \mathcal{I}} \alpha_{ik}(t) \leq 1, \quad \forall k \in \mathcal{K}, \tag{31c}
\]

\[
\alpha_{ik}(t) = \{0, 1\}, \quad \forall i \in \mathcal{I}, \forall k \in \mathcal{K}. \tag{31d}
\]

For problem (P3), the optimal solution is related to the queue backlogs of \(H_i(t)\) and \(Q_i(t)\). If \(H_i(t) \leq Q_i(t)\), the optimal solution is \(p_{ik}(t) = 0\) and \(\alpha_{ik}(t) = 0\). If \(H_i(t) > Q_i(t)\), we analyze the subproblem as follows.

Problem (P3) is a mixed-integer nonlinear programming (MINLP) problem, in this subsection, we apply the time-sharing and alternating methods to solve it [28, 29].

Firstly, by utilizing the time-sharing method, we relax constraint (31d) as

\[
\alpha_{ik}(t) \in [0, 1], \quad \forall i \in \mathcal{I}, \forall k \in \mathcal{K}. \tag{32}\]

Secondly, problem (P3) can be addressed by alternately solving the following two subproblems. The first subproblem is to update the allocated power with fixed time-share values. The second subproblem is to update time-share values with fixed allocated power.

Proposition 3. With determined time-share values, the power allocation of problem (P3) can be expressed as

\[
p_{ik}(t) = \max \left( \frac{(H_i(t) - Q_i(t))W}{\xi_i(t) K_i \ln 2} - \frac{1}{g_{ik}(t)}, 0 \right), \tag{33}\]

where \(\xi_i(t)\) is the Lagrange multiplier at slot \(t\).

Proof. See Appendix C.

\(p_{ik}(t)\) is instantaneous power; we denote \(q_{ik}(t) = \alpha_{ik}(t) p_{ik}(t)\) as the average power allocated by SMD \(i\) on subcarrier \(k\) at slot \(t\). With determined average power allocation values, the time-share variables of problem (P3) can be solved by the
values can be obtained when Appendix D.

Lagrange multiplier at slot $t$ where $u_i(t)$ and $q_{ik}(t)$ are determined, the marginal utility of SMDs can be balanced by the optimal time-share values of Equation (34). Based on the above analysis, an iterative algorithm is proposed to solve Equation (35).

Firstly, we give feasible initial time-share values set when the subcarrier $k$ is determined. Secondly, the time-share values can be obtained when $q_{ik}(t)$ is determined.

$$
\alpha_{ik}(t) = \arg \max_{a_{ik}(t)} \sum_{t \in T} \sum_{k \in K} (H_i(t) - Q_i(t)) \times \frac{a_{ik}(t)}{W} \log_2(1 + p_{ik}(t)g_{ik}(t)), \tag{34}
$$

which is subject to the constraint (32).

**Proposition 4.** With determined average power allocation values, the time-share values of Equation (34) can be solved by

$$
u_i(y(t)) = (H_i(t) - Q_i(t)) \frac{W}{K} \left( \log_2(1 + y(t)) - \frac{y(t)}{\ln 2(1 + y(t))} \right) = \beta_k(t), y(t) = \frac{q_{ik}(t)g_{ik}(t)}{\alpha_{ik}(t)}, \tag{35}
$$

where $u_i(y(t))$ is the marginal utility of SMD $i$ and $\beta_k(t)$ is the Lagrange multiplier at slot $t$.

**Proof.** See Appendix D.

Proposition 3 indicates that when the subcarrier is determined, the marginal utility of SMDs can be balanced by the optimal time-share values. Based on the above analysis, an iterative algorithm is proposed to solve Equation (35).

Firstly, we give feasible initial time-share values set when the subcarrier $k$ is determined. Secondly, the time-share values can be obtained when $q_{ik}(t)$ is determined.

$$
\alpha_{ik} = \frac{u_{ik}g_{ik}}{\sum_{i=1}^{I} u_{ik}g_{ik}} \tag{36}
$$

According to the results of two subproblems, a joint power and subcarrier allocation algorithm is proposed to solve problem (P3), which is summarized in Algorithm 1.

**Algorithm 1: Joint power and subcarrier allocation algorithm.**

5.5. MEC Server Scheduling Subproblem. For the MEC server scheduling subproblem, it can be represented as

$$
\max \sum_{d_{m,t}} \left[ Q_i(t)d_{ser,i}(t) - X_i(t) \sum_{m \in \mathcal{M}} \kappa_{m}f_{ser,m}^3(t) \right] \tag{37a}
$$

s.t. $\sum_{m \in \mathcal{M}} f_{ser,m}(t) \leq \sum_{m \in \mathcal{M}} f_{ser,m}(t) \tau$, \tag{37b}

$d_{ser,i}(t) \geq 0$, \quad $\forall i \in \mathcal{I}$, \tag{37c}

$f_{ser,m}(t) \in \{0, f_{ser,m}^{max}\}$, \quad $\forall m \in \mathcal{M}$. \tag{37d}

For problem (37a), (37b), (37c), and (37d), we can utilize the following proposition to solve it in closed form.

**Proposition 5.** For any feasible $f_{ser,m}(t)$, the optimal solution for problem (37) is that at most, one SMD is scheduled at one slot, and the SMD being scheduled has the largest value of $Q_i(t)$. We denote the SMD index as $i_{max}$.

**Proof.** See Appendix E.

According to Proposition 5, problem (37a), (37b), (37c), and (37d) can be reformulated as

$$
\max \sum_{d_{m,t}} \left[ Q_i^{max}(t)d_{ser,i}^{max}(t) - X_i^{max}(t) \sum_{m \in \mathcal{M}} \kappa_{m}f_{ser,m}^3(t) \right] \tag{38a}
$$

s.t. $d_{ser,i}^{max}(t) \leq \sum_{m \in \mathcal{M}} f_{ser,m}(t) \tau$, \tag{38b}

$d_{ser,i}^{max}(t) \geq 0$, \quad $\forall i \in \mathcal{I}$, \tag{38c}

$f_{ser,m}(t) \in \{0, f_{ser,m}^{max}\}$, \quad $\forall m \in \mathcal{M}$. \tag{38d}

Since $Q_i^{max}(t)$, (38a), (38b), (38c), and (38d) such that $d_{ser,i}^{max}(t)$ $s_{max} = \sum_{m \in \mathcal{M}} f_{ser,m}(t) \tau$. By substituting it into the objective
1: Initialization
2: while $t \in [0, T]$ do
3: $\forall r = 1$ to $R$ do
4: Computing $y_r(t)$ according to Equation (26);
5: Computing $r_i(t)$ according to Equation (28);
6: Computing $f_{\text{ser},i}(t)$ according to Equation (30);
7: Computing $p_{\text{res}}(t)$ and $a_{\text{res}}(t)$ according to Algorithm 1;
8: Computing $f_{\text{ser},i}(t)$ and $d_{\text{ser},i}(t)$ according to Equations (40) and (41), respectively;
9: Updating $H_i(t)$, $Q_i(t)$, $Z_i(t)$, and $X_i(t)$ according to Equations (6), (7), (10), and (11), respectively;
10: end for
11: end while

Algorithm 2: Proposed DTCCRA algorithm.

function, we can get the optimal solution of $f_{\text{ser},i}^*(t)$ from

$$
\max_{0 \leq f_{\text{ser},i}(t) \leq f_{\text{ser},i}^*} Q_{\text{res}}(t) f_{\text{ser},i}(t) \tau + X_{\text{res}}(t) k_{\text{ser},f_{\text{ser},i}^*}(t).
$$

(39)

Therefore, the optimal solution of $f_{\text{ser},i}^*(t)$ is

$$
f_{\text{ser},i}^*(t) = \begin{cases} 
\min \left\{ f_{\text{ser},i}^* \sqrt[\epsilon]{\frac{Q_{\text{res}}(t) \tau}{X_{\text{res}}(t) k_{\text{ser},f_{\text{ser},i}^*}(t)}} \right\}, & X_{\text{ser}}^* > 0, \\
\min \left\{ f_{\text{ser},i}^* \right\}, & X_{\text{ser}}^* = 0.
\end{cases}
$$

(40)

Based on Equations (4) and (40), the optimal solution for bits of task execution in the MEC server can be calculated by

$$
d_{\text{ser},i}^*(t) = S_{\text{ser}} \sum_{m \in \mathcal{I}} f_{\text{ser},i}^*(t) \tau,
$$

(41)

and $d_{\text{ser},i}^*(t) = 0, i \neq \hat{i}$. 

5.6. Delay-Aware Task Congestion Control and Resource Allocation Algorithm. By solving the five subproblems, the DTCCRA algorithm is proposed in Algorithm 2.

In the proposed DTCCRA algorithm, we first solve the system utility maximization subproblem, the solution $y_r(t)$ can be obtained. And then, we solve the SMD congestion control subproblem and SMD computation resource allocation subproblem to obtain the solution of $r_i(t)$ and $f_{\text{loc},i}(t)$. Furthermore, according to Algorithm 1, $p_{\text{res}}(t)$ and $a_{\text{res}}(t)$ can be obtained. At last, solving the MEC server scheduling subproblem, $f_{\text{ser},i}(t)$ and $d_{\text{ser},i}(t)$ can be obtained.

6. Performance Analysis

The performance bound of the proposed DTCCRA algorithm is analyzed in this section.

**Proposition 6.** The proposed DTCCRA algorithm can achieve the bound of system utility as follows:

$$
U_i^* - \sum_{i \in \mathcal{I}} U_i(y_i) \leq \Theta(1/V),
$$

(42)

where $U_i^*$ is the maximal system utility of problem (P1). Equation (42) means that the system utility of the proposed DTCCRA algorithm can arbitrarily close to $U_i^*$ by increasing the control parameter $V$.

**Proof.** Please refer to Appendix F.

**Proposition 7.** The bound of the actual queue backlogs can be expressed as

$$
\sum_{i \in \mathcal{I}} H_i + \sum_{i \in \mathcal{I}} Q_i \leq \frac{D}{\epsilon} + \Theta(V),
$$

(43)

where $\epsilon$ is a parameter and $D$ can be found in Equation (B.9).

**Proof.** See Appendix G.

Based on Proposition 6 and Proposition 7, we can conclude that the system utility and execution delay can achieve an $[\Theta(1/V), \Theta(V)]$ trade-off.

7. Simulation

The performance of the proposed DTCCRA algorithm is evaluated in this section. There are 10 SMDs uniformly distributed over the cell. We utilize the path loss model as $31.5 + 40 \log 10(d)$, where $d$ is the distance between the SMDs and MEC server. The noise power spectral density is $4 \times 10^{-8}$ W/Hz. The slot duration is 1 ms. Both the maximum transmission power and task execution power of each SMD are set as 0.2 W. The maximal CPU-cycle frequency of each SMD is 100 MHz, $s_i = 737.5$ cycle/bit, $\kappa_{\text{loc}} = 10^{-27}$ [19]. The system includes 10 subcarriers, and each subcarrier bandwidth is 1 kHz. In addition, there are 8 CPU cores in the MEC server, and the maximal CPU-cycle frequency of each CPU core is 8 GHz. The average execution power of the MEC server is 5 W. $\kappa_{\text{ser}} = 10^{-27}$. The task average generated.
bits rate $\lambda = 400 \text{kbits/slot}$. The simulation results are obtained by averaging 500 randomly generated network topologies over 1000 slots.

Figure 3 shows the total system utility versus different control parameter $V$. The total system utility increases with respect to the control parameter $V$ at first. Because the larger $V$ means that system utility is given more weight. However, with the increase of $V$, the improvement of total system utility is diminishing. Because the resources are limited, the system utility cannot increase all the time.

Figure 4 shows the total backlogs versus different $V$. It can be observed that the total backlogs are linearly increased with $V$. Because more tasks wait in the queues for transmission, Proposition 6 and Proposition 7 can verify the observations. Combining Figures 3 and 4, we can achieve the trade-off between the system utility and task backlogs. If we want to obtain higher system utility, the task backlogs will be higher and vice versa.

Figure 5 shows the total system utility versus data arrival rate. As can be seen, the system utility is higher with a larger number of SMDs, which can be explained as multiuser diversity. Meanwhile, when the data arrival rate is 400 kbits/slot, the system utility is close to stability.

In order to verify the effectiveness of the proposed DTCCRA algorithm, the numerical results of the other four different algorithms are given. For the four different algorithms, we utilize the same algorithm in this paper to solve the system utility maximization subproblem, SMD congestion control subproblem, and SMD computation resource allocation subproblem. However, for the joint power and subcarrier allocation subproblem, and MEC server scheduling subproblem, the methods are different.

(i) Intelligent heuristic (IH) algorithm: based on the greedy principle, a fast and intelligent heuristic greedy algorithm is proposed to solve the joint

(ii) Average power allocation algorithm: in this algorithm, for the joint power and subcarrier allocation subproblem, the subcarrier allocation can be obtained by Equation (35), the power is allocated averagely in each subcarrier at one slot. For the MEC server scheduling subproblem, the MEC server resources are averagely allocated to each task at one slot

(iii) Average subcarrier allocation algorithm: in this algorithm, for the joint power and subcarrier allocation subproblem, the power allocation can be obtained
by Equation (33), the subcarrier is allocated aver-
agely at one slot. For the MEC server scheduling sub-
problem, the MEC server resources are averagely
allocated to each task at one slot.

(iv) Exhaustive search algorithm: in this algorithm, the
joint power and subcarrier allocation subproblem
can be obtained by the exhaustive search method.
The MEC server scheduling subproblem utilized
the same algorithm in this paper.

Figures 6 and 7 show the total system utility and backlog
comparison of five different algorithms, which have the same
trend as Figures 3 and 4; we can explain them for the same
reasons. Compared with the IH algorithm, the backlogs of
the proposed DTCCRA algorithm are decreased by 5.03%
on average, and the system utility is increased by 3.9% on
average. Meanwhile, compared with the exhaustive search
algorithm, the backlogs of the proposed DTCCRA algorithm
are only increased by 1.76% on average, and the system utility
is decreased by 1.72% on average. Therefore, the proposed
DTCCRA algorithm is effective.

Figure 8 illustrates the total system utility versus SMD
number. Due to the multiuser diversity, the system utility is
higher with a larger SMD number. Compared with the
exhaustive search algorithm, the system utility of the pro-
posed DTCCRA algorithm only decreases by 1.97% on aver-
age. However, compared with the IH algorithm, the system
utility increases by 4.97% on average when the SMD number
grows from 8 to 12.

Figure 9 illustrates the total system utility versus the max-
imal CPU-cycle frequency of each CPU core. The system
utility is higher with a higher CPU-cycle frequency. With
the increasing of CPU-cycle frequency, more tasks can be
executed. Compared with the exhaustive search algorithm,
the system utility of the proposed DTCCRA algorithm only
decreases by 2.03% on average. However, compared with
the IH algorithm, the system utility increases by 3.93% on
average when the maximal CPU-cycle frequency of each
CPU core grows from 6 GHz to 10 GHz.

8. Conclusion

In this paper, we formulated a joint congestion control and
resource allocation problem with power, subcarrier, SMDs,
we need to prove that problem (P1) and problem (P2), respectively. For this proposition, we need to prove that problem (P1) and problem (P2) have the same optimal solution. If we can prove $U_1^* \geq U_2^*$, and $U_1^* \leq U_2^*$, Proposition 1 can be proved.

We first prove $U_1^* \geq U_2^*$. We denote the optimal solution to obtain $U_2^*$ as $\rho_2^*(t) = [y^*(t), r^*(t), p^*(t), \alpha^*(t), f_{loc}^*(t), d_{ser}^*(t), f_{set}^*(t)]$. Considering $U(\cdot)$ is a concave function, by utilizing Jensen’s inequality, we have

$$\sum_{i \in I} U(\bar{r}_i) \geq \sum_{i \in I} U(\bar{r}_i^*) = U_2^*. \quad (A.1)$$

Besides, since $U(\cdot)$ is a nondecreasing function and $r_i^*$ \geq y_i^*$, we have

$$\sum_{i \in I} U(\bar{r}_i) \geq \sum_{i \in I} U(\bar{r}_i^*). \quad (A.2)$$

Considering the constraints of problem (P1) are also the constraints of problem (P2), $\rho_2^*(t)$ is a feasible solution for problem (P1), and the system utility is no larger than $U_1^*$. Thus, we have

$$U_1^* \geq \sum_{i \in I} U(\bar{r}_i) \geq U_2^*. \quad (A.3)$$

And then, we need to prove $U_1^* \leq U_2^*$. We denote the optimal solution to obtain $U_1^*$ as $\rho_1^*(t) = [y^*(t), r^*(t), p^*(t), \alpha^*(t), f_{loc}^*(t), d_{ser}^*(t), f_{set}^*(t)]$. Considering $\rho_1^*(t)$ can satisfy the constraints (12b) and (12c) in problem (P2). Thus, $y_i^*(t)$ and $r_i^*(t)$ are the feasible solutions of problem (P2). According to the definition of long-term time-average expectation, we have $U(y_i^*(t)) = \lim_{T \to \infty} (1/T) \sum_{t=0}^{T-1} U(y_i^*(t)) = \lim_{T \to \infty} (1/T) \sum_{t=0}^{T-1} U(r_i^*) = U(r_i^*)$. Therefore, we can obtain

$$U_1^* \geq \sum_{i \in I} U(\bar{r}_i) = U_2^*. \quad (A.4)$$

According to the above analysis, we have $U_1^* = U_2^*$. Thus, Proposition 1 is proved.

### B. Proof of Proposition 2

According to the equation

$$(\max [a - b, 0] + c)^2 \leq a^2 + b^2 + c^2 - 2a(b - c), \quad \forall a, b, c \geq 0. \quad (B.1)$$

For Equation (6), we can obtain

$$H_i(t + 1)^2 - H_i(t)^2 = \max [H_i(t) - d_{sum_i}(t), 0] + r_i(t)^2 - H_i(t)^2 \leq [d_{sum_i}(t)^2 + r_i(t)^2] + 2H_i(t)[r_i(t) - d_{sum_i}(t)]. \quad (B.2)$$

By utilizing the same method for actual queue $Q_i(t)$, virtual queues $Z_i(t)$ and $X_i(t)$, we have

$$Q_i(t + 1)^2 - Q_i(t)^2 \leq [d_{ser_i}(t)^2 + d_{trans_i}(t)^2] + 2Q_i(t)[d_{trans_i}(t) - d_{ser_i}(t)], \quad (B.3)$$

$$Z_i(t + 1)^2 - Z_i(t)^2 \leq [y_i(t)^2 + r_i(t)^2] + 2Z_i(t)[y_i(t) - r_i(t)]. \quad (B.4)$$
\[ X_i(t+1)^2 - X_i(t)^2 \leq \left[ p_{ser}(t)^2 + (p_{ser}^{av})^2 \right] + 2X_i(t)[p_{ser}(t) - P_{ser}^{av}]. \]

(B.5)

According to Equation (15), (B.2), (B.3), (B.4), and (B.5), we can obtain \( \Delta(\Theta(t)) \) as the following

\[
\Delta(\Theta(t)) = E \left[ \frac{1}{2} \sum_{i \in \mathcal{I}} \left[ H_i(t+1)^2 - H_i(t)^2 + Q_i(t+1)^2 - Q_i(t)^2 \right] + Z_i(t+1)^2 - Z_i(t)^2 + X_i(t+1)^2 - X_i(t)^2 \right] + \gamma_i(t)^2 + r_i(t)^2 + p_{ser}(t)^2 + (p_{ser}^{av})^2 \right].
\]

(B.6)

For Equation (B.6), based on \( \log_2(1+x) \leq x/\ln 2 \) and \( \log_2(1+x) \leq 2x/(\ln 2)^2 \), for \( x \geq 0 \), we can get

\[ E[d_{trans,i}(t)] \leq \frac{W \eta_i}{\ln 2}, \]

(B.7)

\[ E[d_{sum,i}(t)] \leq E \left[ \left( \tau_{max} f_{loc,i}^{-1} + d_{trans,i} \right)^2 \right] \leq \tau^2 \left( \tau_{max} f_{loc,i}^{-1} \right)^2 + \frac{W}{\ln 2}. \]

(B.8)

where \( \eta_i = 2\tau \lambda_i g_{i,k} p_{trans,i}^{max} / \ln 2 \)

By utilizing Equations (B.7) and (B.8) and \( r_i(t) \leq B_i \), we can get

\[ D = \frac{1}{2} \sum_{i \in \mathcal{I}} \left[ 3B_i^2 \left( \sum_{m \in \mathcal{M}} \kappa_{ser} \left( \tau_{max} f_{ser,m}^{-1} \right)^3 \right)^2 + (p_{ser}^{av})^2 \right] + \left( \sum_{m \in \mathcal{M}} f_{max}^{-1} \right)^2 \left( \tau_{max} f_{loc,i}^{-1} \right)^2 + \eta_i \left( \tau_{max} f_{loc,i}^{-1} + \frac{W}{\ln 2} \right). \]

(B.9)

Thus, Proposition 2 is proved.

\[ L(p_{ik}(t), \xi_i(t)) = -\sum_{i \in \mathcal{I}} \sum_{k \in \mathcal{K}} (H_i(t) - Q_i(t)) \left( \frac{W}{K} \log_2(1 + p_{ik}(t) g_{i,k}(t)) + \xi_i(t) \left( \sum_{k \in \mathcal{K}} p_{ik}(t) - P_{trans,i}^{max} \right) \right). \]

(C.1)

We can maximize the Lagrange function pointwise to maximize Equation (C.1), which can be expressed as

\[ L(p_{ik}(t), \xi_i(t)) = -\sum_{i \in \mathcal{I}} \sum_{k \in \mathcal{K}} (H_i(t) - Q_i(t)) \left( \frac{W}{K} \log_2(1 + p_{ik}(t) g_{i,k}(t)) + \xi_i(t) \left( \sum_{k \in \mathcal{K}} p_{ik}(t) - P_{trans,i}^{max} \right) \right). \]

(C.2)

By utilizing the KKT conditions, we have

\[ \frac{\partial L(p_{ik}(t), \xi_i(t))}{\partial p_{ik}(t)} = 0, \]

(C.3)

Form Equation (C.3), we can get \( p_{ik}(t) \)

\[ p_{ik}(t) = \frac{(H_i(t) - Q_i(t)) \left( \frac{W}{K} \log_2(1 + p_{ik}(t) g_{i,k}(t)) + \xi_i(t) \left( \sum_{k \in \mathcal{K}} p_{ik}(t) - P_{trans,i}^{max} \right) \right)}{\xi_i(t) \ln 2} - \frac{1}{g_{i,k}(t)}. \]

(C.4)

Considering the value of power cannot be negative, the power allocation can be expressed as

\[ p_{ik}(t) = \left( \frac{(H_i(t) - Q_i(t)) \left( \frac{W}{K} \log_2(1 + p_{ik}(t) g_{i,k}(t)) + \xi_i(t) \left( \sum_{k \in \mathcal{K}} p_{ik}(t) - P_{trans,i}^{max} \right) \right)}{\xi_i(t) \ln 2} - \frac{1}{g_{i,k}(t)} \right)^+, \]

(C.5)

where \( x^+ = \max(x, 0) \).

Thus, Proposition 3 is proved.

D. Proof of Proposition 4

When the allocated power is determined, problem (P3) is a convex optimization problem. Thus, some standard convex optimization algorithms can solve this problem. In this paper, we utilize the Lagrange dual method. The Lagrange function is
where

\[ \sum_{i \in \mathcal{K}} (H_i(t) - Q_i(t)) \]

and

\[ \frac{W}{K} \log_2 \left( 1 + p_{ik}(t) g_{ik}(t) \right) \]

Thus, Proposition 4 is proved.

By utilizing the KKT conditions, we have

\[ \frac{\partial L(\alpha_{ik}(t), \beta_k(t))}{\partial \alpha_{ik}(t)} = 0, \quad (D.3) \]

\[ -(H_i(t) - Q_i(t)) \frac{W}{K} \log_2 \left( 1 + p_{ik}(t) g_{ik}(t) \right) \]

\[ \frac{\partial L(\alpha_{ik}(t), \beta_k(t))}{\partial \beta_k(t)} = 0, \quad (D.4) \]

Therefore,

\[ u_i(y(t)) = (H_i(t) - Q_i(t)) \frac{W}{K} \log_2 \left( 1 + p_{ik}(t) g_{ik}(t) \right) \]

\[ \frac{y(t)}{\ln 2(1 + y(t))} \]

\[ = \beta_k(t), \quad y(t) = \frac{q_{ik}(t) g_{ik}(t)}{\alpha_{ik}(t)}. \]

Thus, Proposition 4 is proved.

**E. Proof of Proposition 5**

Because \( Q_i(t) \geq 0, \quad i \in \mathcal{K} \), there is an optimal solution such that \( \sum_{i \in \mathcal{K}} d_{\text{ser},i}(t) s_i = \sum_{i \in \mathcal{K}} E_{\text{ser},i}(t) r_i \). For \( \sum_{i \in \mathcal{K}} E_{\text{ser},i}(t) r_i = 0 \), the optimal solution is \( d_{\text{ser}} = 0 \). For \( \sum_{i \in \mathcal{K}} E_{\text{ser},i}(t) r_i > 0 \), we assume that the optimal solution is \( d_{\text{ser}} \) with \( |d_{\text{ser}}| \geq 2 \). We denote \( \mathcal{N} = \{ i | d_{\text{ser},i}(t) > 0, \quad i \in \mathcal{K} \} \), where \( |\mathcal{N}| = |d_{\text{ser}}| \). We formulate a new feasible solution \( \tilde{d}_{\text{ser}}(t) = d_{\text{ser}}(t) + \sum_{i \in \mathcal{K}} d_{\text{ser},i}(t) s_i \) and \( \tilde{d}_{\text{ser}}(t) = 0, \quad i \neq \text{max} \), where \( \text{max} \) is the SMD which has the largest value of \( Q_i(t) \). For the new feasible solution, the objective function will decrease by

\[ \sum_{i \in \mathcal{K}} Q_i(t) \left( d_{\text{ser},i}(t) - \tilde{d}_{\text{ser},i}(t) \right) \]

\[ = \sum_{i \in \mathcal{K}} d_{\text{ser},i}(t) s_i - \sum_{i \in \mathcal{K}} d_{\text{ser},i}(t) s_i \geq 0, \quad (E.1) \]

This means that there is an optimal solution for problem (37a), (37b), (37c), and (37d) for a given \( f_{\text{ser}} \) such that \( d_{\text{ser},\text{max}}(t) \geq 0 \) and \( d_{\text{ser},i}(t) \geq 0, \quad i \neq \text{max} \).

Thus, Proposition 5 is proved.

**F. Proof of Proposition 6**

According to Equation (16), we have

\[ \Delta(\Theta(t)) = \mathbf{E} \left[ \sum_{i \in \mathcal{K}} U_i(y_i(t)) | \Theta(t) \right] \]

\[ \leq D - \mathbf{E} \left[ \sum_{i \in \mathcal{K}} U_i(y_i(t)) | \Theta(t) \right] \]

\[ + \mathbf{E} \left[ \sum_{i \in \mathcal{K}} H_i(t) [r_i(t) - d_{\text{sum},i}(t)] \right] \]

\[ + \mathbf{E} \left[ \sum_{i \in \mathcal{K}} Z_i(t) [p_{\text{trans},i}(t) - d_{\text{ser},i}(t)] \right] \]

\[ + \mathbf{E} \left[ \sum_{i \in \mathcal{K}} X_i(t) [p_{\text{trans},i}(t) - p_{\text{ser},i}(t)] \right]. \]

When \( \Theta(t) \) is determined, Equation (16) is to minimize the right hand of Equation (F.1). Thus, for any feasible solution \( r_i^*(t), y_i^*(t), f_{\text{loc},i}(t), p_{i\text{sk}}(t), \alpha_{ik}(t), f_{\text{ser},i}(t), d_{\text{ser},i}(t) \), and \( d_{\text{ser},i}(t) \) should hold

\[ \Delta(\Theta(t)) = \mathbf{E} \left[ \sum_{i \in \mathcal{K}} U_i(y_i(t)) | \Theta(t) \right] \]

\[ \leq D - \mathbf{E} \left[ \sum_{i \in \mathcal{K}} U_i(y_i(t)) | \Theta(t) \right] \]

\[ + \mathbf{E} \left[ \sum_{i \in \mathcal{K}} H_i(t) [r_i^*(t) - d_{\text{ser},i}(t)] | \Theta(t) \right] \]

\[ + \mathbf{E} \left[ \sum_{i \in \mathcal{K}} Z_i(t) [d_{\text{trans},i}(t) - d_{\text{ser},i}(t)] | \Theta(t) \right] \]

\[ + \mathbf{E} \left[ \sum_{i \in \mathcal{K}} X_i(t) [p_{\text{trans},i}(t) - p_{\text{ser},i}(t)] | \Theta(t) \right]. \]

where \( d_{\text{ser},i}(t), d_{\text{trans},i}(t), \) and \( p_{\text{ser},i}(t) \) can be obtained, when \( f_{\text{loc},i}(t), p_{i\text{sk}}(t), \alpha_{ik}(t), f_{\text{ser},i}(t) \) are fixed. For problem (P1), if it can be solved, it must have a feasible resource allocation.
scheme to hold the following inequality for any $\delta > 0$ [22].

\[
E \left[ \sum_{i \in \mathcal{I}} U_i(y_i^*(t)) \right] \geq U_1^* - \delta, \\
E[r_i^*(t) - d_{\text{sum},i}(t)] \leq \delta, \\
E[d_{\text{trans},i}(t) - d_{\text{ser},i}(t)] \leq \delta, \\
E[y_i^*(t) - r_i^*(t)] \leq \delta, \\
E[p_{\text{ser}}(t) - P_{\text{ser}}^\ast] \leq \delta,
\]  

where $U_1^*$ is the optimal system utility value of problem (P1). We take $\delta \to 0$, and substitute (F.2) into (F.1), we have

\[
\Delta(\Theta(t)) - V E \left[ \sum_{i \in \mathcal{I}} U_i(y_i(t)) | \Theta(t) \right] \leq D - VU_1^*. 
\]  

According to $L(\Theta(t)) \geq 0$, we have

\[
U_1^* - \frac{1}{T} \sum_{t=0}^{T-1} E \left[ \sum_{i \in \mathcal{I}} U_i(y_i(t)) \right] \leq \frac{D}{V} + \frac{E[L(\Theta(0))]}{VT}. 
\]  

Let $T \to \infty$ for Equation (F.5), we have

\[
U_1^* - \sum_{i \in \mathcal{I}} U_i(y_i(t)) \leq O \left( \frac{1}{V} \right).
\]  

Thus, Proposition 6 is proved.

G. Proof of Proposition 7

According to Equation (F.4), we have

\[
\Delta(\Theta(t)) \leq D + V \left[ E \left[ \sum_{i \in \mathcal{I}} U_i(y_i(t)) | \Theta(t) \right] - U_1^* \right].
\]  

By utilizing the Lyapunov optimization, Equation (G.1) can be viewed as all the virtual queues and actual queues are rate stable and $(1/T)\sum_{t=0}^{T-1} E[\gamma_i^*(t) - r_i^*(t)] \leq 0$. According to the Slater condition, if there exists a feasible resource allocation scheme $v$, which should hold $E[\gamma_i^*(t) - d^\ast_{\text{sum},i}(t)] \leq -\varepsilon$, $E[d^\ast_{\text{trans},i}(t) - d^\ast_{\text{ser},i}(t)] \leq -\varepsilon$, $E[p_{\text{ser}}^\ast(t) - P_{\text{ser}}^\ast(t)] \leq -\varepsilon$, and $E \left[ \sum_{i \in \mathcal{I}} U_i(y_i^*(t)) \right] = U_1^*$. According to Equation (F.2), we have

\[
\Delta(\Theta(t)) \leq D + V \left[ E \left[ \sum_{i \in \mathcal{I}} U_i(y_i(t)) | \Theta(t) \right] - U_1^* \right] - \varepsilon \left( \sum_{i \in \mathcal{I}} H_i(t) + \sum_{i \in \mathcal{I}} Q_i(t) \right). 
\]  

Taking the expectation of Equation (G.2), we have

\[
\frac{1}{T} \sum_{t=0}^{T-1} \sum_{i \in \mathcal{I}} (E[H_i(t)] + E[Q_i(t)]) \\
\leq \frac{D}{\varepsilon} + V \left[ \frac{1}{T} \sum_{t=0}^{T-1} E \left[ \sum_{i \in \mathcal{I}} U_i(y_i(t)) \right] - U_1^* \right] + \frac{E[L(\Theta(0))]}{\varepsilon T}. 
\]  

Thus, Proposition 7 is proved.

Data Availability

The data in this paper is generated from the simulation in Matlab, and the pseudocode can refer to Algorithm 1 and Algorithm 2 in Section 5. Therefore, the data of this study are included in the paper.

Disclosure

This paper has been presented in part at the IEEE 30th International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC) [31], Istanbul, Turkey, 8-11 Sep. 2019.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This work was partly supported by the National Natural Science Foundation of China (Nos. 61861002, 61661025, and 61661026), Guangxi Key Laboratory of Wireless Wideband Communication and Signal Processing (No. GXKL06200121), Natural Science Foundation of Gansu Province (Nos. 61861002, 61861003, 61661025, and 61661026), Guangxi Key Laboratory of Wireless Wideband Communication and Signal Processing (No. GXKL06200121), Natural Science Foundation of Gansu Province (Nos. 18YF1FA083 and 20CX4ZA079), Educational Commission of Gansu Province (Nos. 2019B-119 and 2019B-120), Science and Technology Project of Lanzhou University (No. 2018-3-9), Innovation and Entrepreneurship Project for College Students in Gansu Province (No. [2017]62), and Sci- ence and Technology Project of Lanzhou University (Nos. 18YF1FA083 and 20CX4ZA079), Educational Commission of Gansu Province (Nos. 2019B-119 and 2019B-120), Science and Technology Project of Lanzhou University (No. 2018-3-9), Innovation and Entrepreneurship Project for College Students in Gansu Province (No. [2017]62), and Scientific Research Projects of Gansu Institute of Political Science and Law (No. GZF2018XQNLW10).

References

[1] L. Atzori, A. Iera, and G. Morabito, "The Internet of things: a survey," Computer Networks, vol. 54, no. 15, pp. 2787–2805, 2010.
[2] R. Jiang, K. Xiong, P. Fan, Y. Zhang, and Z. Zhong, "Power minimization in SWIPT networks with coexisting power-
splitting and time-switching users under nonlinear EH model,” *IEEE Internet of Things Journal*, vol. 6, no. 5, pp. 8853–8869, 2019.

[3] S. Li, S. Lin, L. Cai, W. Li, and G. Zhu, “Joint resource allocation and computation offloading with time-varying fading channel in vehicular edge computing,” *IEEE Transactions on Vehicular Technology*, vol. 69, no. 3, pp. 3384–3398, 2020.

[4] S. Li, G. Zhu, and S. Lin, “Joint radio and computation resource allocation with predictable channel in vehicular edge computing,” in *2018 21st International Conference on Intelligent Transportation Systems (ITSC)*, pp. 3736–3741, Maui, HI, USA, November 2018.

[5] Y. Mao, C. You, J. Zhang, K. Huang, and K. B. Letaief, “A survey on mobile edge computing the communication perspective,” *IEEE Communications Surveys and Tutorials*, vol. 19, no. 4, pp. 2322–2358, 2017.

[6] Y. Pei, Z. Peng, Z. Wang, and H. Wang, “Energy-efficient mobile edge computing: three-tier computing under heterogeneous networks,” *Wireless Communications and Mobile Computing*, vol. 2020, Article ID 6098786, 17 pages, 2020.

[7] Z. Qin, X. Qiu, J. Ye, and L. Wang, “User-edge collaborative resource allocation and offloading strategy in edge computing,” *Wireless Communications and Mobile Computing*, vol. 2020, Article ID 8867157, 12 pages, 2020.

[8] Y. Chen, N. Zhang, Y. Zhang, X. Chen, W. Wu, and X. S. Shen, “Energy efficient dynamic offloading in mobile edge computing for internet of things,” *IEEE Transactions on Cloud Computing*, p. 1, 2019.

[9] S. Li, N. Zhang, S. Lin et al., “Joint admission control and resource allocation in edge computing for Internet of things,” *IEEE Network*, vol. 32, no. 1, pp. 72–79, 2018.

[10] Y. Chen, N. Zhang, Y. Zhang, X. Chen, W. Wu, and X. S. Shen, “TOFFEE: task offloading and frequency scaling for energy efficiency of mobile devices in mobile edge computing,” *IEEE Transactions on Cloud Computing*, p. 1, 2019.

[11] P. Zhao, H. Tian, C. Qin, and G. Nie, “Energy-saving offloading by jointly allocating radio and computational resources for mobile edge computing,” *IEEE Access*, vol. 5, pp. 11255–11268, 2017.

[12] J. Zhang, X. Hu, Z. Ning et al., “Energy-latency tradeoff for energy-aware offloading in mobile edge computing networks,” *IEEE Internet of Things Journal*, vol. 5, no. 4, pp. 2633–2645, 2018.

[13] C. Wang, C. Liang, F. R. Yu, Q. Chen, and L. Tang, “Computational offloading and resource allocation in wireless cellular networks with mobile edge computing,” *IEEE Transactions on Wireless Communications*, vol. 16, no. 8, pp. 4924–4938, 2017.

[14] X. Lyu, W. Ni, H. Tian et al., “Optimal schedule of mobile edge computing for Internet of things using partial information,” *IEEE Journal on Selected Areas in Communications*, vol. 35, no. 11, pp. 2606–2615, 2017.

[15] Z. Jiang and S. Mao, “Energy delay tradeoff in cloud offloading for multi-core mobile devices,” *IEEE Access*, vol. 3, pp. 2306–2316, 2015.

[16] J. Liu, Y. Mao, J. Zhang, and K. B. Letaief, “Delay-optimal computation task scheduling for mobile-edge computing systems,” in *2016 IEEE International Symposium on Information Theory (ISIT)*, pp. 1451–1455, Barcelona, Spain, July 2016.