IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

SPECIAL SECTION ON MOBILE MULTIMEDIA: METHODOLOGY AND APPLICATIONS

Received December 30, 2019, accepted January 18, 2020, date of publication February 11, 2020, date of current version March 2, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.2973359

Energy-Efficient Multimedia Task Assignment and Computing Offloading for Mobile Edge Computing Networks

YANG SUN, (Member, IEEE), TINGTING WEI, HUIXIN LI, YANHUA ZHANG, AND WENJUN WU, (Member, IEEE)
Faculty of Information Technology, Beijing University of Technology, Beijing 100124, China

Corresponding author: Wenjun Wu (wenjunwu@bjut.edu.cn)

This work was supported in part by the Beijing Post-Doctoral Funding Project under Grant Q6042001201903, in part by the Chaoyang District Post-Doctoral Funding Project under Grant Q1042001201901, in part by the National Natural Science Foundation of China under Grant U1633115 and Grant 61701010, and in part by the Science and Technology Foundation of Beijing Municipal Commission of Education under Grant KM201810005027.

ABSTRACT With the rapid development of 5G technology in recent years, multimedia communication services, such as live online and short video, have received wide attention and become the important means of people's daily social intercourse. However, the rapid growth of multimedia communication demands pose greater challenges to both the wireless network communication capacity and the network processing capacity. Mobile Edge Computing (MEC) is widely regarded as a promising technology to cope with the above challenges. To satisfy the growing demands and improve the quality of experience for users, it is in urge need to seek the effective and efficient task assignment and computing offloading strategy for MEC networks. In this paper, we focused on the multimedia services which need to be processed, uploaded and shared in the network and research the long-term task assignment and resource coordination problem. We formulate the optimization problem as a stochastic optimization problem with the aim of the minimizing the time-average energy consumption of the system. By using the Lyapunov optimization technique, we decompose the original problem into several subproblems which can be solved with current system information and low computational complexities. On this basis, we propose an online energy-efficient task assignment and computing offloading strategy to adaptively decide the task assignment, coordinate and optimize the wireless and computation resource allocation by taking the dynamic wireless condition and service delay constraints into consideration. Extensive simulation results show that our proposed algorithm can achieve considerate energy consumption and delay performances under different conditions.

INDEX TERMS Multimedia transmission, task assignment, computing offloading, Lyapunov optimization, delay constraints.

I. INTRODUCTION

With the rapid development of 5G technology, multimedia communication and social communication are becoming more and more popular [1], [2], a variety of intelligent applications are emerging, e.g. multimedia services, virtual reality and augmented reality [3]. Among them, as one of the most popular applications, multimedia communication services, such as live online and short video, put forward higher requirements for data transmission and data processing. On one hand, users' real-time multimedia traffic is increasing rapidly. According to Cisco's latest data traffic forecast [4], the monthly global mobile data traffic will achieve 77 exabytes by 2022, among which nearly 61 exabytes are generated by mobile video services. On the other hand, the rapidly increasing multimedia services have also made higher put forward new challenges and requirements to the computing ability of the devices and the network to accomplish the multimedia processing operations, such as the video transcoding and compression operations [5].

In order to meet the above challenges, the emerging mobile edge computing (MEC) has been widely concerned

The associate editor coordinating the review of this manuscript and approving it for publication was Dapeng Wu.

by academia and industry [6]–[8]. MEC implements the MEC servers at the edge of wireless network, which can push the computing, communication, storage and other capabilities closer to the user devices [9]–[11]. The users can offload the computing-intensive tasks from the user devices with limited computing capacities to edge servers with sufficient computing and storage resources. By doing this, the MEC has the potential to further reduce the delay, promote the efficiency of network and improve the end user's service experience. However, the above goals have also posed greater challenges to the task assignment and computing offloading optimization among the user devices and MEC servers in the wireless network.

To solve the above challenges, the existing literatures mainly studied the problem of task assignment and computing offloading problem based on two optimization targets including the service delay reduction and energy consumption minimzation [13]–[17]. The authors in [13] used iterative and gradient descent methods to analyze the tradeoff between local offloading and edge offloading, thereby reduce energy consumption. Using the optimized video segmentation strategy and the optimized resource allocation strategy, the authors in [14] studied the resource allocation problem of the multi-user mobile edge computation offloading system to minimize the delay of the system. The authors in [15] considered the performance and energy tradeoffs of multi-core computation offloading problem, and proposed a heuristic algorithm to minimize energy consumption under the completion time constraints of multiple applications. Using particle swarm optimization (PSO) algorithm, the authors in [16] proposed an energy-aware multi-controller layout scheme and a delay-aware resource management model to minimize the system energy consumption with latency constraints. The authors in [17] used variable relaxation and majorization minimization (MM) methods to minimize energy consumption. However, none of these literatures took network and service requirment dynamics into account.

To deal with the long-term computing offloading problem, many literatures applied with the Lyapunov method to assist in optimizing computing offloading decisions [18]–[23]. The authors in [18] studied the problem of partial computation offloading for data distribution applications. The local computing rate, transmission power, cloud computing rate allocation and offloading ratio are jointly optimized to minimize the weighted sum of mobile energy consumption. The authors in [19] improved the Lyapunov optimization method, and deduced the closed expression of the transmission rate, thus avoiding the error caused by prediction and saving energy for mobile devices. On the basis of considering offloading profit and offloading cost, the authors in [20] proposed a multi-stage stochastic programming method to determine the optimal offloading. The authors in [21] considered a parallel virtual queuing model, an adaptive queuing right (AQW) resource allocation strategy and a novel task buffering and offloading strategy, and took both the laxity time and execution time of tasks into account, to achieve the tradeoff between throughput and task completion rate optimization. By developing a fine-grained queue model to solve the challenges such as on-line decision-making efficiency, dynamic system and power-delay trade-off, the authors in [22] conducted systematic study on predictive offloading in fog systems. A novel hybrid edge computing framework for Chimera was proposed in [23], which integrated the emerging edge cloud wireless access network, thus bringing long-term benefits. However, few of them focused on the long-term task assignment and computing offloading optimization for the computing-intensive and data-intensive [28], [29] multimedia services which need to be processed, uploaded and shared in the network.

There exist two ways to process and upload the computing-intensive and data-intensive multimedia service in the MEC networks, which are determined by multimedia data processing location. Here we name the local processing mode as the way to firstly complete the data processing locally and upload the processed data to the application server in the cloud via the MEC-BS. On the other hand, we name the MEC processing mode as the way to transmit the raw multimedia data through the wireless channel to the MEC-BS and utilize the MEC-server with sufficient computation capacity to execute the data processing. Therefore, under dynamic system and task arrival condition, how to effective and efficient assign the tasks and coordinate the wireless and computation resources between the two processing modes turn to be an interesting problem. In this paper, we propose an online energy-efficient task assignment and computing offloading strategy to solve the above problem. The contributions of the paper can be summarized as follows.

1) We formulate the optimization problem as a stochastic optimization problem with the aim of the minimizing the time-average energy consumption of the system. To tackle the complex problem, we introduce several queue models and decompose the original problem into several subproblems which can be solved with current system information and low computational complexities by using the Lyapunov optimization technique.
2) We develop an online algorithm called energy-efficient task assignment and computing offloading strategy to adaptively decide the the task assignment, coordinate and optimize the wireless and computation resource allocation. Specifically, we propose an iterative wireless resource allocation algorithm to solve the joint wireless transmit power and sub-channel resource allocation problem by using the Lagrangian dual composition method.
3) We finally provide both theoretical analyses and simulation results to verify that our proposed algorithm can flexibly strike a balance between energy consumption and delay performances by tuning the control parameter while keeping the system stable.

The whole paper is organized as follows. We describe the system model in section II. The problem formulation is given in section III. In section IV, the solution based on Lyapunov

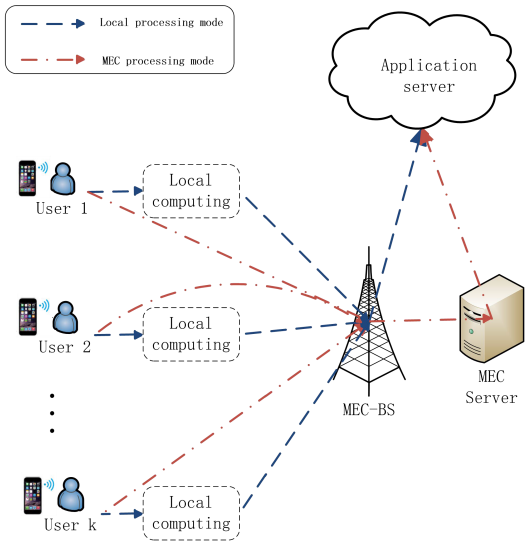


FIGURE 1. System model.

optimization technique is proposed, and the simulation results of the proposed algorithm performance are given in section V. Finally, some conclusions are drawn in section VI.

II. SYSTEM MODEL

As shown in Fig.1, here we consider a multi-user mobile edge computing network consisting of one MEC-BS and K users. The MEC-BS is equipped with a MEC server to provide the computing offloading service to the users under its coverage. The set of users within the MEC-BS's coverage is denoted as $\mathcal{K} = \{1, 2, \cdots, k, \cdots, K\}$. Each user has some multimedia tasks which need be processed (like transcoding [24], compression [25]) and uploaded to the application server in the cloud for saving and sharing. Here we consider a time-slotted system, in which the multimedia tasks' processing and transmission operations are performed in discrete time. The set of time slot is denoted as $\mathcal{T} = \{1, 2, \cdots, t, \cdots, T\}$, and the length of each time slot is ς. Here we use $\lambda_k(t)$ to denote the arriving data size from user k at tth time slot, and we assume that the arriving requested data size $\lambda_k(t)(k \in \mathcal{K})$ is an independent and identically distributed (i.i.d.) Poisson process and $\mathbb{E}[\lambda_k(t)] = \bar{\lambda}_k$. Then the multimedia task requirement for each user $k(k \in \mathcal{K})$ can be characterized by a tuple of three main parameters, which can be described by $\{\lambda_k(t), \rho\lambda_k(t), \varepsilon\lambda_k(t)\}$. Here, $\lambda_k(t)$ is the raw multimedia data size that requested from user k, $\rho\lambda_k(t)$ is the number of CPU cycles that needed to complete the processing operation, and $\varepsilon\lambda_k(t)(0 < \varepsilon \leq 1)$ is the processed multimedia data size.

Due to the limited resource constraints of the system, we propose two kinds of processing models: the local processing model and the MEC processing model.

A. LOCAL PROCESSING MODEL

In local processing model, the arriving tasks are firstly processed locally, and then the processed data are uploaded to the application server through the wireless channels.

TABLE 1. Summary of key notations.

Notation	Meaning
K	Number of users in MEC-BS
T	Number of time slots
J	The available CPU cores number in the MEC server
ς	The length of each time slot
ρ	The CPU cycles number per bit to complete the processing operation
ε	The ratio between raw data and processed data
ξ	The bandwidth of each sub-channel
τ_l	The energy parameter that depends on the CPU hardware condition of user device
τ_m	The energy parameter that depends on the CPU hardware condition of MEC server
P_{max}	The maximum uplink transmit power
F^l_{max}	The maximum CPU cycles number of the user device
F^m_{max}	The maximum CPU cycles number of each CPU core in the MEC server
$\lambda_k(t)$	The raw multimedia data size that requested from user k at tth time slot
$g_{k,n}(t)$	Channel gain between user k and MEC-BS at tth time slot
$p^l_{k,n}(t)$	Transmit power assigned to upload the locally processed data at tth time slot
$p^m_{k,n}(t)$	Transmit power assigned to transmit the MEC offloading data at tth time slot
$w^l_{k,n}(t)$	Wireless bandwidth assigned to upload the locally processed data at tth time slot
$w^m_{k,n}(t)$	Wireless bandwidth assigned to transmit the MEC offloading data at tth time slot
σ^2	The noise power
$f^l_k(t)$	Number of local CPU cycles assigned to user k at tth time slot
$f^m_{k,j}(t)$	Number of CPU cycles of jth CPU core in the MEC server assigned to user k at tth time slot
$c^l_k(t)$	The data size processed locally at tth time slot
$c^m_k(t)$	The data size processed by the MEC server at tth time slot
$r^l_k(t)$	The transmit data size of the locally processed data at tth time slot
$r^m_k(t)$	The transmit data size of the MEC offloading data at tth time slot
$a^l_k(t)$	Arriving locally unprocessed data size of user k at tth time slot
$a^m_k(t)$	Arriving MEC offloading data size of user k at tth time slot
$R_k(t)$	Data queue of the locally unprocessed data of user k at tth time slot
$D_k(t)$	Data queue of the locally processed but un-transmitted data of user k at tth time slot
$Z_k(t)$	Data queue of the MEC unprocessed and un-transmitted data of user k at tth time slot
$H_k(t)$	Data queue of the MEC received but unprocessed data of user k at tth time slot
V	Lyapunov control parameter

1) LOCAL COMPUTING MODEL

Here we let $f^l_k(t)$ be the scheduled local CPU cycles number of user k at tth time slot, which can't exceed the maximum CPU cycles number of the user device F^l_{max}. Then, the data size $c^l_k(t)$ which already have been processed at tth time slot can be denoted as:

$$c^l_k(t) = \varsigma \frac{f^l_k(t)}{\rho}. \tag{1}$$

According to [30], under given allocated local CPU cycles number $f^l_k(t)$, the energy consumption $E^{l,ex}_k(t)$ for local processing at tth time slot can be calculated as follows:

$$E^{l,ex}_k(t) = \varsigma \tau_l (f^l_k(t))^3, \tag{2}$$

where τ_l is the parameter that depends on the CPU hardware condition of the user device.

2) TRANSMISSION MODEL OF LOCALLY PROCESSED DATA

When the raw data are processed locally, the processed data would be transmit to the application server via the wireless channels. Here we denote $p^l_{k,n}(t)$, $w^l_{k,n}(t)$ as the allocated uplink transmission power and bandwidth of the user k to transmit the local processed data. According to the Shannon theory, the achievable transmission rate of user k on sub-channel n is given by:

$$\gamma^l_{k,n}(t) = \xi \log_2(1 + \frac{p^l_{k,n}(t) g_{k,n}(t)}{\sigma^2}), \tag{3}$$

where ξ is the wireless bandwidth per sub-channel.

And the total achievable transmission rate $r^l_k(t)$ to upload the local processed data can be expressed as follows:

$$r^l_k(t) = \varsigma \sum_{n=1}^{N} w^l_{k,n}(t) \gamma^l_{k,n}(t). \tag{4}$$

where $g_{k,n}(t)$ is the channel gain on sub-channel n between the user k and MEC-BS, σ^2 is the noise power.

Thus, the energy consumption to transmit the locally processed data at tth time slot is

$$E^{l,tr}_k(t) = \varsigma \sum_{n=1}^{N} w^l_{k,n}(t) p^l_{k,n}(t). \tag{5}$$

B. MEC PROCESSING MODEL

When the raw data are applied the MEC processing model, the user should firstly upload the raw data to the MEC-BS, and then utilize the computation resource of the MEC server to execute the data processing operation.

1) TRANSMISSION MODEL OF MEC OFFLOADING DATA

There are two kinds of data that needed to be transmitted via wireless channels: the locally processed data and the MEC offloading data. Taking the difference between the two kinds of data transmission requirements into consideration, here we apply different power and bandwidth allocation strategies. Here we denote $p^m_{k,n}(t)$, $w^m_{k,n}(t)$ as the allocated uplink transmission power and bandwidth of the user k to transmit the raw data that would be processed at the MEC server. Similarly, the achievable transmission rate of user k on sub-channel n is given by:

$$\gamma^m_{k,n}(t) = \xi \log_2(1 + \frac{p^m_{k,n}(t) g_{k,n}(t)}{\sigma^2}). \tag{6}$$

The total transmission rate $r^m_k(t)$ to transmit the raw data to the MEC-BS can be expressed as:

$$r^m_k(t) = \varsigma \sum_{n=1}^{N} w^m_{k,n}(t) \gamma^m_{k,n}(t). \tag{7}$$

The energy transmission to upload the MEC offloading data transmission at tth time slot is

$$E^{m,tr}_k(t) = \varsigma \sum_{n=1}^{N} w^m_{k,n}(t) p^m_{k,n}(t). \tag{8}$$

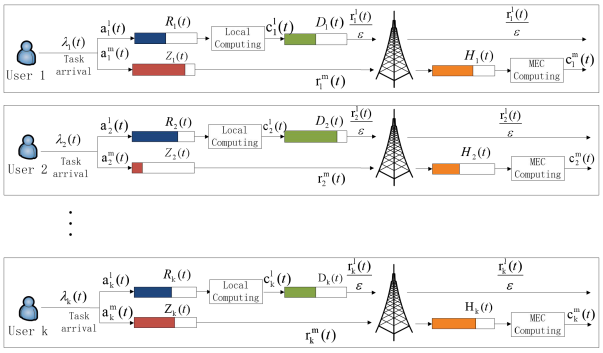


FIGURE 2. Queue model.

2) MEC COMPUTING MODEL

After receiving the raw data from the users, the MEC server with larger computation capacity starts to execute the processing operation. The MEC server is equipped with a multiple core CPU and each core can be occupied by only one task at every time slot. Here we assume the available number of CPU cores is J, and the maximum CPU cycles number of the each core is set to be F^m_{max}. Let $f^m_{k,j}(t)$ denote the allocated CPU cycles number of jth CPU core in MEC server for user k at tth time slot, the amount of data that have completed to be processing is given by

$$c^m_k(t) = \varsigma \frac{\sum_{j=1}^{J} f^m_{k,j}(t)}{\rho}. \tag{9}$$

The corresponding energy consumption for data processing operation of total users can be denoted as

$$E^{m,ex}(t) = \varsigma \tau_m \sum_{k=1}^{K} \sum_{j=1}^{J} f^m_{k,j}(t)^3, \tag{10}$$

where τ_m is the parameter that depends on the CPU hardware condition of MEC server.

According to the system model described above, the energy consumption $E^{total}(t)$ of all users in the system in tth time slot can be defined as:

$$E^{total}(t) = \sum_{k=1}^{K} [E^{l,tr}_k(t) + E^{l,ex}_k(t) + E^{m,tr}_k(t)] + E^{m,ex}(t). \tag{11}$$

C. QUEUE MODEL

Here we describe the all actual and virtual queues according to the raw data size for uniformity. As shown in Fig.2, there exist four types of queues in the system: $R_k(t)$ is referred as local processing data queue of user k in which the data are waiting to be processed locally. $c^l_k(t)$ is referred as the data size processed locally at tth time slot, $a^l_k(t)$ is referred as the arriving task data of user k at tth time slot that would be assigned to be processed locally in the near future. $D_k(t)$ is referred as the transmit queue consists of the local processed data awaiting to be uploaded to the application

server via the MEC-BS. $r_k^l(t)$ is the transmit data size of the locally processed data at tth time slot. $Z_k(t)$ is referred as the data queue in which the data are waiting to be uploaded to the MEC-BS to complete the processing operation. $a_k^m(t)$ is referred as the arriving data of user k that would be offloaded and uploaded to the MEC-BS at tth time slot. $r_k^m(t)$ is the transmit data size of the MEC offloading data at tth time slot. $H_k(t)$ is referred as the data queue awaiting to be processed by the MEC server, $c_k^m(t)$ is the data size processed by the MEC server at tth time slot. Here we can update the queue dynamics of $R_k(t+1)$, $D_k(t+1)$, $Z_k(t+1)$, $H_k(t+1)$ as follows:

$$R_k(t+1) = [R_k(t) - c_k^l(t)]^+ + a_k^l(t), \tag{12}$$

$$D_k(t+1) = [D_k(t) - \frac{r_k^l(t)}{\varepsilon}]^+ + c_k^l(t), \tag{13}$$

$$Z_k(t+1) = [Z_k(t) - r_k^m(t)]^+ + a_k^m(t), \tag{14}$$

$$H_k(t+1) = [H_k(t) - c_k^m(t)]^+ + r_k^m(t). \tag{15}$$

where $[x]^+ = \max(x, 0)$

III. PROBLEM FORMULATION

In this section, we formulate the multi-user multimedia task assignment and computing offloading problem as a stochastic optimization problem, the objective of which is to minimize the time-averaged energy consumption subject to queue stability. The problem formulation is given by

$$\min_{\mathbf{a},\mathbf{f^l},\mathbf{p},\mathbf{w},\mathbf{f^m}} \lim_{T\to\infty} \sum_{t=0}^{T-1} \mathbb{E}\{E^{total}(t)\}$$

$$s.t.\ \text{C1:} \sum_{k=1}^{K} w_{k,n}^l(t) + w_{k,n}^m(t) \le 1, \forall n$$

$$\text{C2: } w_{k,n}^l(t) \in \{0,1\}, w_{k,n}^m(t) \in \{0,1\}, \forall k, n$$

$$\text{C3: } \sum_{n=1}^{N} w_{k,n}^l(t) p_{k,n}^l(t) + w_{k,n}^m(t) p_{k,n}^m(t) \le P_{max}, \forall k$$

$$\text{C4: } 0 \le f_{k,j}^m(t) \le F_{max}^m, \forall k, j$$

$$\text{C5: } 0 \le f_k^l(t) \le F_{max}^l, \forall k$$

$$\text{C6: All queues } R_k(t), D_k(t), Z_k(t) \text{ and } H_k(t) \text{ are stable .} \tag{16}$$

where $\mathbf{a} = \{\{a_k^l(t)\}, \{a_k^m(t)\}\}$ is the a concatenated vector of local processing data size $\{a_k^l(t)\}$ and MEC offloading data size $\{a_k^m(t)\}$. $\mathbf{f^l} = \{f_k^l(t)\}$ and $\mathbf{f^m} = \{f_{k,l}^m(t)\}$ are the local and MEC computation resource vectors, respectively. $\mathbf{p} = \{\{p_{k,n}^l(t)\}, \{p_{k,n}^m(t)\}\}$,$\mathbf{w} = \{\{w_{k,n}^l(t)\}, \{w_{k,n}^m(t)\}\}$ are the concatenated vectors that combine transmit power $\{p_{k,n}^l(t)\}$ and $\{p_{k,n}^m(t)\}$, wireless bandwidth $\{w_{k,n}^l(t)\}$ and $\{w_{k,n}^m(t)\}$, respectively. F_{max}^l and F_{max}^m are the maximal local computation capacity and each CPU core's computation capacity in the MEC server, respectively. Specifically, C1-C2 are the bandwidth allocation constraints and C3 are the uplink transmit power constraints. C4 and C5 imply that the overall allocated computation resources cannot exceed the maximal computation capacities of MEC server and user devices. C6 guarantees the network stability.

Clearly, we can find out that the problem (16) is a stochastic program which can be solved by dynamic programming if we have a statistical knowledge of the system (i.e. the wireless channel state information) in advance. However, it is costly to get the dynamic and future system information in practical scenarios. To address the above challenge, in this paper, we introduce Lyapunov optimization technique to develop an online task assignment and computing offloading algorithm to solve the stochastic program which only utilizes the current system information and has a lower computational complexity [26].

IV. ONLINE ENERGY-EFFICIENT TASK ASSIGNMENT AND COMPUTING OFFLOADING ALGORITHM

According to the update equation (12)-(15), here we define $\Theta(t) = [R_k(t), D_k(t), Z_k(t), H_k(t)]$ as a concatenated vector of all queues and virtual queues. The Lyapunov function can be defined as:

$$L(\Theta(t)) \triangleq \frac{1}{2}\sum_{k=1}^{K} R_k(t)^2 + \frac{1}{2}\sum_{k=1}^{K} D_k(t)^2 + \frac{1}{2}\sum_{k=1}^{K} Z_k(t)^2 + \frac{1}{2}\sum_{k=1}^{K} H_k(t)^2. \tag{17}$$

Here we define the one-slot conditional Lyapunov drift $\Delta(\Theta(t))$ as follows

$$\Delta(\Theta(t)) \triangleq \mathbb{E}\{L(\Theta(t+1)) - L(\Theta(t))|\Theta(t)\}. \tag{18}$$

Theorem 1: Under any control algorithm, the drift-plus-penalty expression has the following upper bound for all t, all possible values of $\Theta(t)$, and all parameters $V \ge 0$ [26]:

$$\begin{aligned}
&\Delta(\Theta(t)) + V\mathbb{E}\{E^{total}(t)|\Theta(t)\} \\
&\le B + \sum_{k=1}^{K} \mathbb{E}\{R_k(t)a_k^l(t) + Z_k(t)a_k^m(t)|\Theta(t)\} \\
&\quad + \sum_{k=1}^{K} \mathbb{E}\{V\varsigma\tau_l(f_k^l(t))^3 + \frac{D_k(t) - R_k(t)}{\rho}\varsigma f_k^l(t)|\Theta(t)\} \\
&\quad + \sum_{k=1}^{K} \mathbb{E}\{\sum_{k=1}^{K}[V\varsigma \sum_{n=1}^{N}(w_{k,n}^l(t)p_{k,n}^l(t) + w_{k,n}^m(t)p_{k,n}^m(t)) \\
&\quad - \frac{D_k(t)}{\varepsilon} r_k^l(t) - (Z_k(t) - H_k(t))r_k^m(t)]|\Theta(t)\} \\
&\quad + \mathbb{E}\{V\varsigma\tau_m \sum_{k=1}^{K}(\sum_{j=1}^{J}(f_{k,j}^m(t))^3 - \frac{H_k(t)}{\rho}\varsigma \sum_{j=1}^{J} f_{k,j}^m(t))|\Theta(t)\}.
\end{aligned}$$

where B is a positive constant that satisfies the following for all t:

$$\begin{aligned} B \geq & \frac{1}{2}\sum_{k=1}^{K}\mathbb{E}\{a_k^l(t)^2 + a_k^l(t)^2|\Theta(t)\} \\ & + \frac{1}{2}\sum_{k=1}^{K}\mathbb{E}\{c_k^l(t)^2 + (\frac{r_k^l(t)}{\varepsilon})^2|\Theta(t)\} \\ & + \frac{1}{2}\sum_{k=1}^{K}\mathbb{E}\{a_k^m(t)^2 + r_k^m(t)^2|\Theta(t)\} \\ & + \frac{1}{2}\sum_{k=1}^{K}\mathbb{E}\{r_k^m(t)^2 + c_k^m(t)^2|\Theta(t)\}. \end{aligned} \tag{19}$$

Proof: See Appendix A.

According to (19), the original problem can be transformed into a series of instantaneous optimization subproblems which can be solved with the current system information. Here we propose an online algorithm to minimize the right-hand-side of (19) by observing $R_k(t)$,$D_k(t)$,$Z_k(t)$,$H_k(t)$ at every time slot. At each time slot, the instantaneous problem can be further decomposed into the following four subproblems, and the online energy-efficient task assignment and computing offloading algorithm is summarized in Algorithm 1.

Algorithm 1 Online Energy-Efficient Task Assignment and Computing Offloading Algorithm

1: Initialization: $t = 0$, $\mathbf{R}(0) = 0$, $\mathbf{D}(0) = 0$, $\mathbf{Z}(0) = 0$, $\mathbf{H}(0) = 0$, T.
2: **repeat**
3: Each user generates its arrival task size λ_k;
4: Solve the task assignment decision subproblem according to (20) and (21)
5: Solve the local computation resource allocation subproblem according to (23);
6: Solve the wireless resource allocation subproblem according to Algorithm 2;
7: Solve the MEC computation resource allocation subproblem according to (41);
8: Update all queues $\mathbf{R}(t)$, $\mathbf{D}(t)$, $\mathbf{Z}(t)$, $\mathbf{H}(t)$ according to (12),(13),(14),(15).
9: **until** $t > T$.

A. TASK ASSIGNMENT DECISION

The task assignment decision subproblem determines the amount of data to be processed in the local device and MEC server. The task assignment decision subproblem is formulated as follows:

$$\begin{aligned} \min \ & \sum_{k=1}^{K}[R_k(t)a_k^l(t) + Z_k(t)a_k^m(t)], \\ s.t. \ & a_k^l(t) + a_k^m(t) = \lambda_k(t), \\ & 0 \leq a_k^l(t) \leq A_{max}^l, \\ & 0 \leq \sum_{k=1}^{K} a_k^m(t) \leq A_{max}^m. \end{aligned}$$

Since the subproblem (20) is a linear programming problem, we can easily obtain the optimal solutions of $a_k^l(t)$ and $a_k^m(t)$. The solutions of (20) are obtained based on the following criteria.

For the user set $\mathcal{K}_l = \{k|R_k(t) < Z_k(t), \forall k \in \mathcal{K}\}$, the optimal task assignment decisions are given by:

$$\begin{cases} a_k^l(t) = \min\{\lambda_k(t), A_{max}^l\} \\ a_k^m(t) = \max\{0, \lambda_k(t) - A_{max}^l\}. \end{cases} \tag{20}$$

For the user set $\mathcal{K}_m = \{k|R_k(t) \geq Z_k(t), \forall k \in \mathcal{K}\}$, we can obtain the optimal task assignment decisions one by one in ascending order of $Z_k(t)$. The optimal task assignment decisions are given by:

$$\begin{cases} a_k^l(t) = \max\{0, \lambda_k(t) - (A_{max}^m - \sum\limits_{k' \in \mathcal{K}_k} a_{k'}^m(t))\} \\ a_k^m(t) = \min\{\lambda_k(t), A_{max}^m - \sum\limits_{k' \in \mathcal{K}_k} a_{k'}^m(t)\}. \end{cases} \tag{21}$$

where $\mathcal{K}_k = \{k|Z_k(t) \geq Z_{k'}'(t) \& a_{k'}^m > 0, \forall k' \in \mathcal{K}_m\}$.

B. LOCAL COMPUTATION RESOURCE ALLOCATION

At each time slot, the user device can get its optimal computation resource allocation strategy by solving the following subproblem:

$$\begin{aligned} \min \ & \sum_{k=1}^{K} V\tau_l(f_k^l(t))^3 + \frac{D_k(t) - R_k(t)}{\rho} f_k^l(t), \\ s.t. \ & C5. \end{aligned} \tag{22}$$

As the subproblem (22) is a standard convex problem, we can easily obtain the optimal number of CPU cycles $f_k^l(t)$ allocated to the user k as follows:

$$f_k^l(t) = \begin{cases} \min\{\sqrt{\dfrac{R_k(t) - D_k(t)}{3V\tau_l\rho}}, F_{max}^l\} & R_k(t) > D_k(t) \\ 0 & \text{otherwise.} \end{cases}$$

C. WIRELESS RESOURCE ALLOCATION

In order to transmit the locally processed data and the raw data that offloading to the MEC server, we decompose the wireless resource allocation subproblem which involves the wireless resource allocation decisions $p_{k,n}^l(t)$, $p_{k,n}^m(t)$, $w_{k,n}^l(t)$, $w_{k,n}^m(t)$ from the right-hand-side of (19). The wireless resource allocation subproblem can be formulated as follows:

$$\begin{aligned} \min_{\mathbf{p},\mathbf{w}} \ & \sum_{k=1}^{K}\sum_{n=1}^{N} V(p_{k,n}^l(t) + p_{k,n}^m(t)) - \frac{D_k(t)}{\varepsilon} w_{k,n}^l(t)\gamma_{k,n}^l(t) \\ & - (Z_k(t) - H_k(t))w_{k,n}^m(t)\gamma_{k,n}^m(t) \\ s.t. \ & C1, C2, C3 \end{aligned} \tag{23}$$

Due to the integer constraints for the sub-channel allocation in C2, the subproblem (23) is a non-convex mixed integer programming problem which can be solved by the Brute-force method with a high computational complexity.

To make the subproblem tractable, here we relax $w^l_{k,n}$, $w^m_{k,n}$ to be the continuous real variables in the range [0, 1] and regard $w^l_{k,n}$, $w^m_{k,n}$ as the time-sharing factors for sub-channel n. Let $\tilde{p}^l_{k,n} = w^l_{k,n}p^l_{k,n}$ and $\tilde{p}^m_{k,n} = w^m_{k,n}p^m_{k,n}$ denote the actual transmit power allocated to user n to upload the locally processed data and the MEC offloading data, respectively. Similarly, denote $\tilde{\gamma}^l_{k,n}(t) = B\log_2(1 + \tilde{p}^l_{k,n}(t)g_{k,n}(t)/\sigma^2)$, $\tilde{\gamma}^m_{k,n}(t) = B\log_2(1 + \tilde{p}^m_{k,n}(t)g_{k,n}(t)/\sigma^2)$ as the achievable rate of user k on sub-channel n to transmit the two kinds of data, respectively. Now, the subproblem (23) can be converted into:

$$\begin{aligned}
\min_{\tilde{\mathbf{p}},\mathbf{w}} \ & \sum_{k=1}^{K}\sum_{n=1}^{N} V(w^l_{k,n}(t)\tilde{p}^l_{k,n}(t) + w^m_{k,n}(t)\tilde{p}^m_{k,n}(t)) \\
& - \frac{D_k(t)}{\varepsilon} w^l_{k,n}(t)\tilde{\gamma}^l_{k,n}(t) - (Z_k(t) - H_k(t))w^m_{k,n}(t)\tilde{\gamma}^m_{k,n}(t), \\
s.t. \ \ & \text{C9: } \sum_{k=1}^{K} w^l_{k,n}(t) + w^m_{k,n}(t) \le 1, \forall n \\
& \text{C10: } w^l_{k,n}(t) \in [0,1], w^m_{k,n}(t) \in [0,1], \forall k, n \\
& \text{C11: } \sum_{n=1}^{N} \tilde{p}^l_{k,n}(t) + \tilde{p}^m_{k,n}(t) \le P_{max}, \forall k
\end{aligned} \tag{24}$$

According to [27], we can solve the subproblem (24) by using the Lagrangian dual composition method. The associated Lagrange function can be seen to be:

$$\begin{aligned}
& L(\tilde{\mathbf{p}}, \mathbf{w}, \boldsymbol{\nu}, \boldsymbol{\delta}) \\
& = \sum_{k=1}^{K}\sum_{n=1}^{N}[V(\tilde{p}^l_{k,n}(t) + \tilde{p}^m_{k,n}(t)) - \frac{D_k(t)}{\varepsilon} w^l_{k,n}(t)\tilde{\gamma}^l_{k,n}(t) \\
& \quad -(Z_k(t) - H_k(t))w^m_{k,n}(t)\tilde{\gamma}^m_{k,n}(t)] \\
& \quad + \sum_{n=1}^{N} \delta_n[(\sum_{k=1}^{K} w^l_{k,n}(t) + w^m_{k,n}(t)) - 1] \\
& \quad + \sum_{k=1}^{K} \nu_k[(\sum_{n=1}^{N} \tilde{p}^l_{k,n}(t) + \tilde{p}^m_{k,n}(t)) - P_{max}],
\end{aligned} \tag{25}$$

where $\boldsymbol{\nu} = \{\nu_k\}$ and $\boldsymbol{\delta} = \{\delta_n\}$ are the Lagrange multipliers for the transmit power and bandwidth constraints.

Thus, the Lagrangian dual function can be defined as:

$$g(\boldsymbol{\nu}, \boldsymbol{\delta}) = \min_{\tilde{\mathbf{p}},\mathbf{w}} L(\tilde{\mathbf{p}}, \mathbf{w}, \boldsymbol{\nu}, \boldsymbol{\delta}) \tag{26}$$

The dual Lagrangian problem can be given by:

$$\max_{\boldsymbol{\nu},\boldsymbol{\delta}} \ g(\boldsymbol{\nu}, \boldsymbol{\delta}). \tag{27}$$

$$s.t. \ \boldsymbol{\nu}, \boldsymbol{\delta} > 0 \tag{28}$$

The Lagrangian function (25) can be further rewritten as:

$$\begin{aligned}
& L(\tilde{\mathbf{p}}, \mathbf{w}, \boldsymbol{\nu}, \boldsymbol{\delta}) \\
& = \sum_{n=1}^{N} L_n(\tilde{\mathbf{p}}, \mathbf{w}, \boldsymbol{\nu}, \boldsymbol{\delta}) - \sum_{n=1}^{N}\delta_n - \sum_{k=1}^{K}\nu_k P_{max},
\end{aligned} \tag{29}$$

where

$$\begin{aligned}
& L_n(\tilde{\mathbf{p}}, \mathbf{w}, \boldsymbol{\nu}, \boldsymbol{\delta}) \\
& = \sum_{k=1}^{K}(V + \nu_k)(\tilde{p}^l_{k,n}(t) + \tilde{p}^m_{k,n}(t)) \\
& \quad - \frac{D_k(t)}{\varepsilon} w^l_{k,n}(t)\tilde{\gamma}^l_{k,n}(t) + \delta_n(w^l_{k,n}(t) + w^m_{k,n}(t)) \\
& \quad - (Z_k(t) - H_k(t))w^m_{k,n}(t)\tilde{\gamma}^m_{k,n}(t).
\end{aligned} \tag{30}$$

Then we first take the partial derivatives of $L_n(\cdots)$ with respect to $\tilde{p}^l_{k,n}(t)$ and $\tilde{p}^m_{k,n}(t)$, respectively.

$$\frac{\partial L_n(\cdots)}{\partial \tilde{p}^l_{k,n}} = V + \nu_k - \frac{1}{\ln 2}\frac{D_k(t)w^l_{k,n}(t)g_{k,n}(t)\xi}{\varepsilon(\sigma^2 + \tilde{p}^l_k(t)g_{k,n}(t))}, \tag{31}$$

$$\begin{aligned}
\frac{\partial L_n(\cdots)}{\partial \tilde{p}^m_{k,n}} &= V + \nu_k \\
& \quad - \frac{1}{\ln 2}\frac{(Z_k(t) - H_k(t))w^m_{k,n}(t)g_{k,n}(t)\xi}{\sigma^2 + p^m_{k,n}(t)g_{k,n}(t)}.
\end{aligned} \tag{32}$$

According to the Karush-Kuhn-Tucker (KKT) conditions, the optimal solutions $\tilde{p}^{l*}_{k,n}(t)$, $\tilde{p}^{m*}_{k,n}(t)$ of the subproblem (24) can be easily obtained as follows:

$$\tilde{p}^{l*}_{k,n}(t) = (\frac{1}{\ln 2}\frac{D_k(t)w^l_{k,n}(t)\xi}{\varepsilon(V + \nu_k)} - \frac{\sigma^2}{g_{k,n}(t)})^+, \forall k, n \tag{33}$$

$$\tilde{p}^{m*}_{k,n}(t) = (\frac{1}{\ln 2}\frac{(Z_k(t) - H_k(t))w^m_{k,n}(t)\xi}{V + \nu_k} - \frac{\sigma^2}{g_{k,n}(t)})^+, \forall k, n \tag{34}$$

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

Substituting (33), (34) into (24), the partial derivatives of $L_n(\cdots)$ with respect to $\tilde{w}^l_{k,n}$ and $\tilde{w}^m_{k,n}$ can be expressed as:

$$\begin{aligned}
\eta^l_{k,n} &= \frac{\partial L_n(\cdots)}{\partial w^l_{k,n}(t)} \\
&= (V + \nu_k(t))\tilde{p}^{m*}_{k,n}(t) + \delta_n - \frac{D_k(t)}{\varepsilon}\tilde{\gamma}^{l*}_{k,n}(t),
\end{aligned} \tag{35}$$

$$\begin{aligned}
\eta^m_{k,n} &= \frac{\partial L_n(\cdots)}{\partial w^m_{k,n}(t)} \\
&= (V + \nu_k(t))\tilde{p}^{m*}_{k,n}(t) + \delta_n - (Z_k(t) - H_k(t))\tilde{\gamma}^{m*}_{k,n}(t),
\end{aligned} \tag{36}$$

where $\tilde{\gamma}^{l*}_{k,n}(t) = \xi\log_2(1 + \tilde{p}^{l*}_{k,n}(t)g_{k,n}(t)/\sigma^2)$, $\tilde{\gamma}^{m*}_{k,n}(t) = \xi\log_2(1 + \tilde{p}^{m*}_{k,n}(t)g_{k,n}(t)/\sigma^2)$.

Due to the objective of the sub-channel allocation is to minimize the for all users in the system. Therefore, for any sub-channel n, it will be assigned to the user k who has the smallest and negative $\eta^\rho_{k,n}(\rho \in \{l, m\})$, that is,

$$w^{\rho^*}_{k^*,n} = 1|_{(\rho^*,k^*)=\min\limits_{\rho,k}\eta^\rho_{k,n}}, \forall \rho, k, n \tag{37}$$

We resort to the subgradient method to iteratively find the optimal solution to problem (26). Then, the Lagrange multipliers are updated by

$$\nu^{(i+1)}_k = [\nu^{(i)}_k + \beta^{(i)}_1(\sum_{n=1}^{N}(\tilde{p}^l_{k,n}(t) + \tilde{p}^m_{k,n}(t)) - P_{max})]^+, \tag{38}$$

where $\beta_1^{(i)}$ are the positive step sizes of iteration $i \in \{1, 2, \cdots, I_{max}\}$. I_{max} is the maximum number of iterations. The whole procedure to solve (23) is summarized in Algorithm 2.

Algorithm 2 Iterative Wireless Resource Allocation Algorithm

1: Initialization: $i = 0$, set $\tilde{\mathbf{p}}$,$\mathbf{w}$, $\boldsymbol{\nu}$, $\boldsymbol{\delta}$, I_{max}.
2: **repeat**
3: **for** $k = 1$ to K **do**
4: Each user updates $\tilde{p}_{k,n}^l$, $\tilde{p}_{k,n}^m$ according to (33) and (34);
5: Calculate $\eta_{k,n}^l$, $\eta_{k,n}^m$ according to (35) and (36);
6: **end for**
7: MEC-BS update $w_{k,n}^l$, $w_{k,n}^m$ according to (37).
8: Update $\boldsymbol{\nu}$ according to (38).
9: $i = i + 1$;
10: **until** Convergence or $i = I_{max}$.

D. MEC COMPUTATION RESOURCE ALLOCATION

We formulate the MEC computation resource allocation subproblem as follows:

$$\begin{aligned} \min \ & \sum_{k=1}^{K}\sum_{j=1}^{J}[V\tau_m(f_{k,j}^m(t))^3 - \frac{H_k(t)}{\rho}f_{k,j}^m(t)] \\ s.t. \ & 0 \le f_{k,j}^m(t) \le F_{max}, \forall k, j \end{aligned} \tag{39}$$

It is easily to verify that the above subproblem is a convex problem which can be further decomposed into J independent subproblems. For each CPU core, we can easily obtain the optimal number of CPU cycles $f_{k,j}^m(t)$ when the CPU core j is occupied by the user k can be derived as follows:

$$\tilde{f}_{k,j}^m(t) = \min\{\sqrt{\frac{H_k(t)}{3V\tau_m\rho}}, F_{max}^m\}. \tag{40}$$

As each core can be occupied by only one task at every time slot, the optimal number of CPU cycles $f_{k,j}^m(t)(\forall k, j)$ can be determined by follows:

$$f_{k,j}^m(t) = \begin{cases} \tilde{f}_{k,j}^m(t) & O(\tilde{f}_{k,j}^m(t)) = \min\limits_{k' \in \mathcal{K}} O(\tilde{f}_{k',j}^m(t)) \\ 0 & \text{otherwise}, \end{cases} \tag{41}$$

where $O(\tilde{f}_{k,j}^m(t)) = V\tau_m(\tilde{f}_{k,j}^m(t))^3 - \frac{H_k(t)}{\rho}\tilde{f}_{k,j}^m(t)$.

E. PERFORMANCE ANALYSIS

Here, we mathematically analyze the performance bounds of the proposed algorithm based on the Lyapunov optimization technique.

1) QUEUEING BOUNDS

Let $\omega(t)$ denote as a random event which can describe the dynamic system condition, $\alpha(t) \in A_{\omega(t)}$ denote the control decision of the system, where $A_{\omega(t)}$ is an abstract set of all possible actions. Here we introduce Slater condition to ensure the stability of the system queues.

Definition 1 (Slater Condition): There are values $\epsilon > 0$ and $\Psi(\epsilon)$ and an ω-only policy $\alpha^*(t)$ that satisfies:

$$\begin{aligned} \mathbb{E}\{\hat{E}^{total}(\alpha^*(t), \omega(t))\} &= \Psi(\epsilon) \\ \mathbb{E}\{\hat{a}_k^l(\alpha^*(t), \omega(t)) - c_k^l(\alpha^*(t), \omega(t))\} &\le \epsilon \\ \mathbb{E}\{\hat{c}_k^l(\alpha^*(t), \omega(t)) - \frac{\hat{r}_k^l(\alpha^*(t), \omega(t))}{\varepsilon}\} &\le \epsilon \\ \mathbb{E}\{\hat{a}_k^m(\alpha^*(t), \omega(t) - \hat{r}_k^m(\alpha^*(t), \omega(t))\} &\le \epsilon \\ \mathbb{E}\{\hat{r}_k^m(\alpha^*(t), \omega(t) - \hat{c}_k^m(\alpha^*(t), \omega(t))\} &\le \epsilon \end{aligned}$$

In addition, it is assumed that the expectation of $\mathbb{E}\{\hat{E}^{total}(\alpha^*(t), \omega(t))\}$ is bounded by some finite constants E^{min}, E^{max}:

$$E^{min} \le \mathbb{E}\{\hat{E}^{total}(\alpha^*(t), \omega(t))\} \le E^{max} \tag{42}$$

When the system satisfies the above boundedness assumptions, then the time-average of real queue lengths can be upper bounded by the following theorem.

Theorem 2: Suppose there are constants $\epsilon > 0$ and $\Psi(\epsilon)$ for which the Slater condition holds, then the average queue bounds satisfy

$$\begin{aligned} \lim_{T\to\infty} \sup \sum_{t=0}^{T-1}\sum_{k=1}^{K} & \mathbb{E}\{R_k(t) + D_k(t) + Z_k(t) + H_k(t)\} \\ & \le \frac{B + V[\Psi(\epsilon) - E^{opt}]}{\epsilon} \end{aligned} \tag{43}$$

where E^{opt} is the infimum time-average energy consumption achievable by any policy that meets the required constraints and B is defined in (19).

Proof: A similar proof can be found in [26].

2) PERFORMANCE OF THE PROPOSED ALGORITHM

The performance of the proposed algorithm based on Lyapunov optimization is given by Theorem 3.

Theorem 3: Suppose $\omega(t)$ is i.i.d. over the time slots, the problem (20)-(39) are feasible. Then time average expected energy consumption satisfies:

$$\lim_{T\to\infty} \sup \sum_{t=0}^{T-1} \mathbb{E}\{E^{total}\} \le E^{opt} + \frac{B}{V} \tag{44}$$

Proof: See Appendix B.

V. SIMULATION RESULTS

Here we conduct extensive simulations to evaluate the performances of the proposed online energy-efficient multimedia task assignment and computing offloading algorithm.

A. SIMULATION SETUP

In our simulation, there exists a MEC-BS with 4 users randomly distributed within its coverage. The coverage radius of the MEC-BS is set to 400m. For the MEC-BS, The total sub-channel number is 50, each sub-channel bandwidth ξ

is 180kHz. The available number of CPU cores in the MEC server is 6, and the maximum computation capacity for each CPU core is set to be 2G cycles/s. The unit time slot is 1s. For the user device, the maximum uplink transmit power is set to be 500mW, the noise power density is -174dBm/Hz. The maximum computation capacity of the user device is 2G cycles/s. During each time slot, the tasks arrival of each user is Poisson distributed with average arrival rate $\bar{\lambda}$. The service processing parameter ρ is set to be 737.5 cycles/bits, and the ratio between processed data and raw data is $\varepsilon = 0.5$. The simulation results are averaged over 1000 consecutive time slots. We list some detailed simulation parameters in TABLE.2.

TABLE 2. Simulation parameters.

Parameter	Value	Parameter	Value
K	4	N	50
T	1000	J	6
ς	1s	ξ	180kHz
P_{max}	500mW	σ^2	-174dBm/Hz
F^l_{max}	2G cycles/s	F^m_{max}	2G cycles/s
ε	0.5	ρ	737.5 cycles/bit
τ_l	10^{-27}	τ_m	10^{-27}
A^l_{max}	2Mbps	A^m_{max}	16Mbps

Here we compared the performance of the proposed algorithm with two reference schemes that apply the single processing modes. For convenience, we user *only local processing mode* to denote the scheme in which the multimedia services are only processed locally. Meanwhile, we use *only MEC processing mode* to describe the scheme that all the multimedia services are uploaded to the MEC server to process. According to the Little's law, the time-averaged delay d_k can be approximated by:

$$d_k = \frac{\lim\limits_{T\to\infty} \sum\limits_{t=0}^{T-1} \mathbb{E}\{R_k(t) + D_k(t) + Z_k(t) + H_k(t)\}}{\bar{\lambda}_k}, \quad (45)$$

which indicates that the delay is proportional to the total queue length.

B. SIMULATION RESULTS AND ANALYSIS

First, we present the system stability in Fig.3 with the control parameter $V = [10^{11}, 5 * 10^{11}, 10^{12}, 10^{13}]$ and task average arrival rate $\bar{\lambda} = 2$Mbps. We observe that the time-averaged energy consumption of our proposed algorithm is tending to stabilize and converge towards a steady value as time went on, which verifies the Theorem 3.

In Fig.4, we evaluate the energy consumption performance of proposed algorithm compared with two reference schemes under different Lyapunov control parameters V. We observe that the energy consumption decreases as the Lyapunov control parameter V increases. Meanwhile, compared to the two reference schemes, our proposed algorithm can effectively reduce the energy consumption of the system. That's because our proposed algorithm can dynamically assign the multimedia tasks to the more energy efficient processing mode based on the current system condition.

Fig.5 shows the total queue backlogs performance of the proposed algorithm compared with two reference schemes. It is shown that the average queue length increases

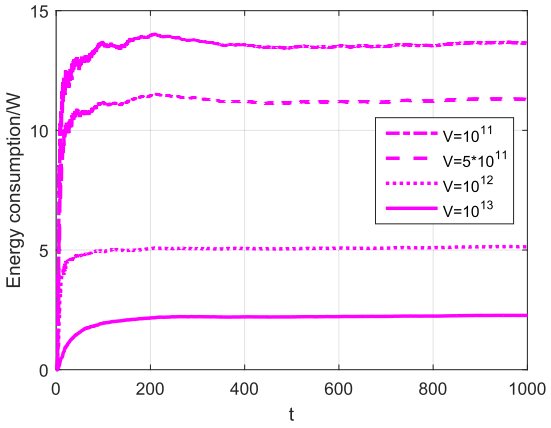


FIGURE 3. Energy consumption vs. time, $\bar{\lambda} = 2$*Mbps*.

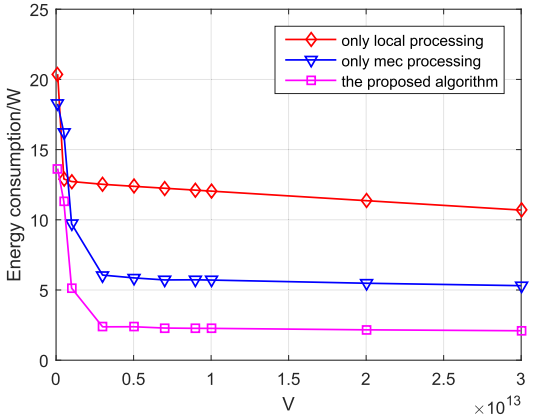


FIGURE 4. Energy consumption vs. Lyapunov control parameter V, $\bar{\lambda} = 2$*Mbps*.

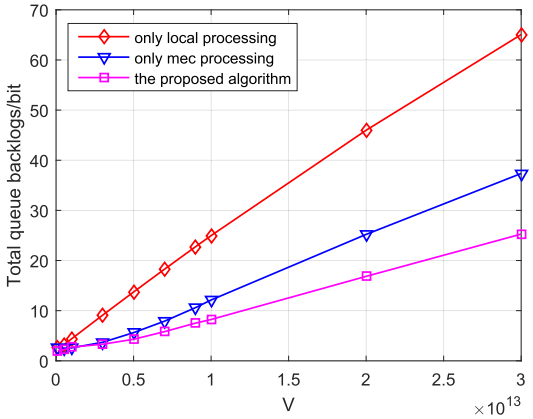


FIGURE 5. Total queue backlogs vs. Lyapunov control parameter V, $\bar{\lambda} = 2$*Mbps*.

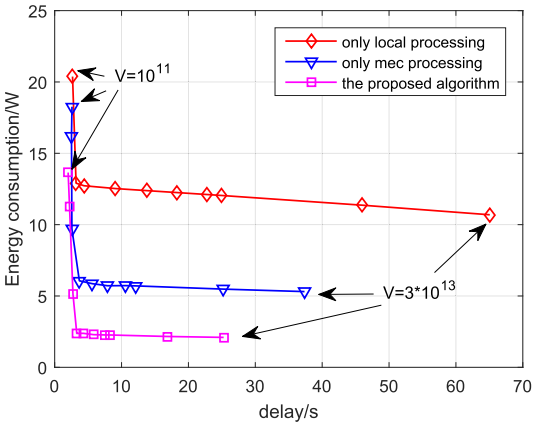


FIGURE 6. Energy consumption vs. delay, $\bar{\lambda} = 2$*Mbps*.

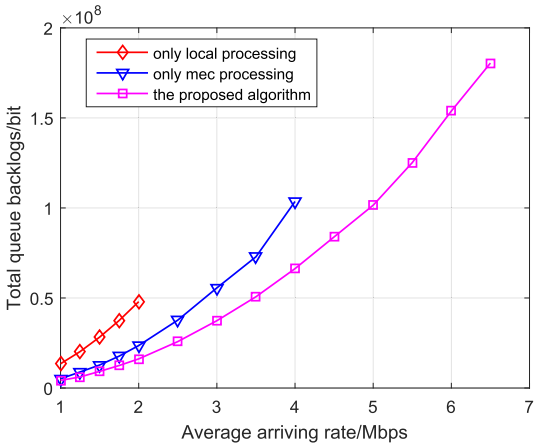


FIGURE 8. Total queue backlogs vs. average arriving rate, $V = 10^{13}$.

while the Lyapunov control parameter V increases, which verifies Theorem 2. Specifically, the total queue length of the proposed algorithm is significantly less than the two reference schemes, which means the proposed algorithm has a better delay performance.

Fig.6 shows the relationship between the energy consumption and the average delay performances. From the figure, we can observe the trade-off between the energy consumption and the delay performances. Combined with Fig.4 and Fig.5, it is easily to figure out that the energy consumption decreases and the delay increases when V is increasing, which indicates that we should select an appropriate value of V based on the requirements of the two objectives.

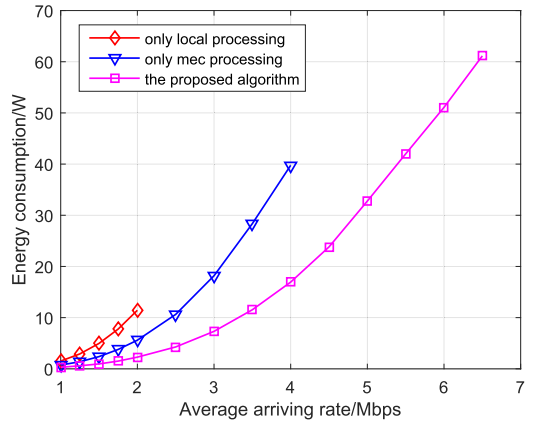


FIGURE 7. Energy consumption vs. average arriving rate, $V = 10^{13}$.

In Fig.7 and Fig.8, we evaluate the energy consumption and the total queue backlogs performances versus the average arriving rate. From the two figures, we can see that the energy consumption and the total queue backlogs both increase as the average arriving rate increases. This is because the increasing arriving rate bring more requirements to the wireless and computation resource, which further increase the energy consumption and service delay. Compared with the two reference scheme, the proposed algorithm can accommodate more service requirements while guarantee considerate energy and delay performances.

VI. CONCLUSION

In this paper, we focus on the long-term task assignment and computing offloading problem for the computing-intensive and data-intensive multimedia services in the MEC networks by taking the dynamics of system and task arrival into consideration. We formulate the optimization problem as a stochastic optimization problem with the aim of the minimizing the time-averaged energy consumption of the system and propose an online algorithm to adaptively decide the the task assignment, coordinate and optimize the wireless and computation resource allocation. Specifically, we propose an iterative wireless resource allocation algorithm to solve the joint wireless transmit power and sub-channel resource allocation problem by using the Lagrangian dual composition method. Simulation results show that our proposed algorithm can achieve a considerate better energy consumption and service delay performances.

APPENDIXES

APPENDIX A

PROOF OF THEOREM 1

According to [26], for any $Q \geq 0$, $b \geq 0$, $a \geq 0$, we have:

$$(\max[Q-b, 0]+a)^2 \leq Q^2+a^2+b^2+2Q(a-b). \quad (46)$$

For queue $R_k(t)$, we have:

$$R_k(t+1)^2 \leq (R_k(t)-c_k^l(t))^2+a_k^l(t)^2+2(R_k(t)-c_k^l(t))a_k^l(t).$$

Therefore:

$$\frac{R_k(t+1)^2 - R_k(t)^2}{2} \leq \frac{a_k^l(t)^2 + c_k^l(t)^2}{2} - c_k^l(t)a_k^l(t) + R_k(t)[a_k^l(t) - c_k^l(t)].$$

Similarly, we have

$$\frac{D_k(t+1)^2 - D_k(t)^2}{2} \le \frac{c_k^l(t)^2 + \frac{r_k^l(t)^2}{\varepsilon}}{2} - c_k^l(t)\frac{r_k^l(t)}{\varepsilon} + D_k(t)[c_k^l(t) - \frac{r_k^l(t)}{\varepsilon}], \quad (47)$$

$$\frac{Z_k(t+1)^2 - Z_k(t)^2}{2} \le \frac{a_k^m(t)^2 + r_k^m(t)^2}{2} - r_k^m(t)a_k^m(t) + Z_k(t)[a_k^m(t) - r_k^m(t)], \quad (48)$$

$$\frac{H_k(t+1)^2 - H_k(t)^2}{2} \le \frac{r_k^m(t)^2 + c_k^m(t)^2}{2} - r_k^m(t)c_k^m(t) + H_k(t)[r_k^m(t) - c_k^m(t)]. \quad (49)$$

Therefore, by adding $V\mathbb{E}\{E^{total}(t)|\Theta(t)\}$ to both sides, we have:

$$\begin{aligned}
&\Delta(\Theta(t)) + V\mathbb{E}\{E^{total}(t)|\Theta(t)\} \\
&\le B + V\mathbb{E}\{E^{total}(t)|\Theta(t)\} \\
&\quad + \sum_{k=1}^{K} R_k(t)\mathbb{E}\{a_k^l(t) - c_k^l(t)|\Theta(t)\} \\
&\quad + \sum_{k=1}^{K} D_k(t)\mathbb{E}\{c_k^l(t) - \frac{r_k^l(t)}{\varepsilon}|\Theta(t)\} \\
&\quad + \sum_{k=1}^{K} Z_k(t)\mathbb{E}\{a_k^m(t) - r_k^m(t)|\Theta(t)\} \\
&\quad + \sum_{k=1}^{K} H_k(t)\mathbb{E}\{r_k^m(t) - c_k^m(t)|\Theta(t)\}.
\end{aligned}$$

By introducing (1),(4),(7),(9) into the above function, we obtain

$$\begin{aligned}
&\Delta(\Theta(t)) + V\mathbb{E}\{E^{total}(t)|\Theta(t)\} \\
&\le B + \sum_{k=1}^{K} \mathbb{E}\{R_k(t)a_k^l(t) + Z_k(t)a_k^m(t)|\Theta(t)\} \\
&\quad + \sum_{k=1}^{K} \mathbb{E}\{V\varsigma\tau_l(f_k^l(t))^3 + \frac{D_k(t) - R_k(t)}{\rho}\varsigma f_k^l(t)|\Theta(t)\} \\
&\quad + \sum_{k=1}^{K} \mathbb{E}\{\sum_{k=1}^{K}[V\varsigma\sum_{n=1}^{N}(w_{k,n}^l(t)p_{k,n}^l(t) + w_{k,n}^m(t)p_{k,n}^m(t)) \\
&\quad - \frac{D_k(t)}{\varepsilon}r_k^l(t) - (Z_k(t) - H_k(t))r_k^m(t)]|\Theta(t)\} \\
&\quad + \mathbb{E}\{V\varsigma\tau_m\sum_{k=1}^{K}(\sum_{j=1}^{J}(f_{k,j}^m(t))^3 - \frac{H_k(t)}{\rho}\varsigma\sum_{j=1}^{J}f_{k,j}^m(t))|\Theta(t)\}.
\end{aligned}$$

Hence, the proof of Theorem 1 is completed.

APPENDIX B
PROOF OF THEOREM 3

Here we denote $a_k^{l*}(t)$, $a_k^{m*}(t)$, $r_k^{l*}(t)$, $r_k^{m*}(t)$, $c_k^{l*}(t)$, $c_k^{m*}(t)$ as the resulting task assignment, transmitted and processed values and denote $E^{total*}(t)$ as the corresponding energy consumption value under any alternative control decision $\alpha(t) \in A_{\omega(t)}$, respectively. We can easily get the following function for every time slot t:

$$\begin{aligned}
&\Delta(\Theta(t)) + V\mathbb{E}\{E^{total}(t)|\Theta(t)\} \\
&\le B + V\sum_{k=1}^{K}\mathbb{E}\{E^{total*}(t)|\Theta(t)\} \\
&\quad + \sum_{k=1}^{K} R_k(t)\mathbb{E}\{a_k^{l*}(t) - c_k^{l*}(t)|\Theta(t)\} \\
&\quad + \sum_{k=1}^{K} D_k(t)\mathbb{E}\{c_k^{l*}(t) - \frac{r_k^{l*}(t)}{\varepsilon}|\Theta(t)\} \\
&\quad + \sum_{k=1}^{K} Z_k(t)\mathbb{E}\{a_k^{m*}(t) - r_k^{m*}(t)|\Theta(t)\} \\
&\quad + \sum_{k=1}^{K} H_k(t)\mathbb{E}\{r_k^{m*}(t) - c_k^{m*}(t)|\Theta(t)\}.
\end{aligned}$$

According to Definition 1, we take $\epsilon \to \infty$ and we get:

$$\lim_{T\to\infty}\sup\sum_{t=0}^{T-1}\mathbb{E}\{E^{total}\} \le E^{opt} + \frac{B}{V}. \quad (50)$$

Hence, the proof of Theorem 3 is completed.

REFERENCES

[1] D. Wu, Q. Liu, H. Wang, D. Wu, and R. Wang, "Socially aware energy-efficient mobile edge collaboration for video distribution," *IEEE Trans. Multimedia*, vol. 19, no. 10, pp. 2197–2209, Oct. 2017.
[2] D. Wu, J. Yan, H. Wang, D. Wu, and R. Wang, "Social attribute aware incentive mechanism for device-to-device video distribution," *IEEE Trans. Multimedia*, vol. 19, no. 8, pp. 1908–1920, Aug. 2017.
[3] S. Wang, X. Zhang, Y. Zhang, L. Wang, J. Yang, and W. Wang, "A survey on mobile edge networks: Convergence of computing, caching and communications," *IEEE Access*, vol. 5, pp. 6757–6779, 2017.
[4] Cisco. *Cisco Visual Networking Index: Global Mobile Data Traffic Forecast Update, 2017-2022*. Accessed: Feb. 18, 2019. [Online]. Available: https://www.cisco.com/c/en/us/solutions/collateral/serviceprovider/visual-networking-index-vni/white-paper-c11-738429.html
[5] J. Ren, G. Yu, Y. Cai, and Y. He, "Latency optimization for resource allocation in mobile-edge computation offloading," *IEEE Trans. Wireless Commun.*, vol. 17, no. 8, pp. 5506–5519, Aug. 2018.
[6] P. Mach and Z. Becvar, "Mobile edge computing: A survey on architecture and computation offloading," *IEEE Commun. Surveys Tuts.*, vol. 19, no. 3, pp. 1628–1656, 3rd Quart., 2017.
[7] R. Wang, J. Yan, D. Wu, H. Wang, and Q. Yang, "Knowledge-centric edge computing based on virtualized D2D communication systems," *IEEE Commun. Mag.*, vol. 56, no. 5, pp. 32–38, May 2018.
[8] R. Wang, H. Liu, H. Wang, Q. Yang, and D. Wu, "Distributed security architecture based on blockchain for connected health: Architecture, challenges, and approaches," *IEEE Wireless Commun.*, vol. 26, no. 6, pp. 30–36, Dec. 2019.
[9] T. Taleb, K. Samdanis, B. Mada, H. Flinck, S. Dutta, and D. Sabella, "On multi-access edge computing: A survey of the emerging 5G network edge cloud architecture and orchestration," *IEEE Commun. Surveys Tuts.*, vol. 19, no. 3, pp. 1657–1681, 3rd Quart., 2017.

[10] X. Wang, Y. Han, C. Wang, Q. Zhao, X. Chen, and M. Chen, "In-edge AI: Intelligentizing mobile edge computing, caching and communication by federated learning," *IEEE Netw.*, vol. 33, no. 5, pp. 156–165, Sep. 2019.

[11] P. Zhang, X. Kang, Y. Liu, and H. Yang, "Cooperative willingness aware collaborative caching mechanism towards cellular D2D communication," *IEEE Access*, vol. 6, pp. 67046–67056, 2018.

[12] H. Flores, P. Hui, S. Tarkoma, Y. Li, S. Srirama, and R. Buyya, "Mobile code offloading: From concept to practice and beyond," *IEEE Commun. Mag.*, vol. 53, no. 3, pp. 80–88, Mar. 2015.

[13] H. Sun, F. Zhou, and R. Q. Hu, "Joint offloading and computation energy efficiency maximization in a mobile edge computing system," *IEEE Trans. Veh. Technol.*, vol. 68, no. 3, pp. 3052–3056, Mar. 2019.

[14] J. Ren, G. Yu, Y. Cai, Y. He, and F. Qu, "Partial offloading for latency minimization in mobile-edge computing," in *Proc. IEEE Global Commun. Conf. (GLOBECOM)*, Singapore, Dec. 2017, pp. 1–6.

[15] Y. Geng, Y. Yang, and G. Cao, "Energy-efficient computation offloading for multicore-based mobile devices," in *Proc. IEEE Conf. Comput. Commun. (INFOCOM)*, Honolulu, HI, USA, Apr. 2018, pp. 46–54.

[16] F. Li, X. Xu, H. Yao, J. Wang, C. Jiang, and S. Guo, "Multi-controller resource management for software-defined wireless networks," *IEEE Commun. Lett.*, vol. 23, no. 3, pp. 506–509, Mar. 2019.

[17] A. Khalili, S. Zarandi, and M. Rasti, "Joint resource allocation and offloading decision in mobile edge computing," *IEEE Commun. Lett.*, vol. 23, no. 4, pp. 684–687, Apr. 2019.

[18] M. Sheng, Y. Wang, X. Wang, and J. Li, "Energy-efficient multiuser partial computation offloading with collaboration of terminals, radio access network, and edge server," *IEEE Trans. Commun.*, to be published.

[19] S. Pan and Y. Chen, "Energy-optimal scheduling of mobile cloud computing based on a modified Lyapunov optimization method," *IEEE Trans. Green Commun. Netw.*, vol. 3, no. 1, pp. 227–235, Mar. 2019.

[20] Y. Li, S. Xia, M. Zheng, B. Cao, and Q. Liu, "Lyapunov optimization based trade-off policy for mobile cloud offloading in heterogeneous wireless networks," *IEEE Trans. Cloud Comput.*, to be published.

[21] L. Li, Q. Guan, L. Jin, and M. Guo, "Resource allocation and task offloading for heterogeneous real-time tasks with uncertain duration time in a fog queueing system," *IEEE Access*, vol. 7, pp. 9912–9925, 2019.

[22] X. Gao, X. Huang, S. Bian, Z. Shao, and Y. Yang, "PORA: Predictive offloading and resource allocation in dynamic fog computing systems," *IEEE Internet Things J.*, vol. 7, no. 1, pp. 72–87, Jan. 2020.

[23] L. Pu, X. Chen, G. Mao, Q. Xie, and J. Xu, "Chimera: An energy-efficient and deadline-aware hybrid edge computing framework for vehicular crowdsensing applications," *IEEE Internet Things J.*, vol. 6, no. 1, pp. 84–99, Feb. 2019.

[24] H. Yuan, C. Guo, J. Liu, X. Wang, and S. Kwong, "Motion-homogeneous-based fast transcoding method from H.264/AVC to HEVC," *IEEE Trans. Multimedia*, vol. 19, no. 7, pp. 1416–1430, Jul. 2017.

[25] H. Wang, T. Wang, L. Liu, H. Sun, and N. Zheng, "Efficient compression-based line buffer design for image/video processing circuits," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 27, no. 10, pp. 2423–2433, Oct. 2019.

[26] M. J. Neely, *Stochastic Network Optimization with Application to Communication and Queueing Systems* (Synthesis Lectures on Communication Networks), vol. 3. San Rafael, CA, USA: Morgan & Claypool, 2010, p. 1–211.

[27] H. Zhang, C. Jiang, N. C. Beaulieu, X. Chu, X. Wen, and M. Tao, "Resource allocation in spectrum-sharing OFDMA femtocells with heterogeneous services," *IEEE Trans. Commun.*, vol. 62, no. 7, pp. 2366–2377, Jul. 2014.

[28] Z. Li, J. Chen, and Z. Zhang, "Socially aware caching in D2D enabled fog radio access networks," *IEEE Access*, vol. 7, pp. 84293–84303, 2019.

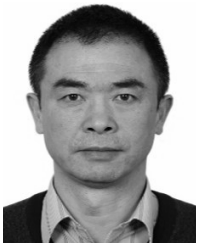
[29] Z. Li, Y. Jiang, Y. Gao, L. Sang, and D. Yang, "On buffer-constrained throughput of a wireless-powered communication system," *IEEE J. Sel. Areas Commun.*, vol. 37, no. 2, pp. 283–297, Feb. 2019.

[30] Y. Mao, J. Zhang, S. H. Song, and K. B. Letaief, "Stochastic joint radio and computational resource management for multi-user mobile-edge computing systems," *IEEE Trans. Wireless Commun.*, vol. 16, no. 9, pp. 5994–6009, Sep. 2017.

YANG SUN (Member, IEEE) received the Ph.D. degree from the School of Information and Communication Engineering, Beijing University of Posts and Telecommunications, in 2018, China. She is currently a Lecturer with the Beijing University of Technology. Her current research interests focus on ultra-dense heterogeneous networks, interference management, massive MIMO, and green telecommunications.

TINGTING WEI is currently pursuing the M.S. degree with the Faculty of Information Technology, Beijing University of Technology, China. Her current research interests include computation offloading, fog radio access networks, mobile edge computing, reinforcement learning, and resource allocation.

HUIXIN LI is currently pursuing the M.S. degree with the Faculty of Information Technology, Beijing University of Technology, China. Her current research interests in resource allocation, mobile edge computing, and mobile caching optimization.



YANHUA ZHANG received the B.E. degree from the Xi'an University of Technology, Xi'an, China, in 1982, and the M.S. degree from Lanzhou University, Lanzhou, China, in 1988. From 1982 to 1990, he was with the Jiuquan Satellite Launch Center (JSLC), Jiuquan, China. In 1990, he was a Visiting Professor with Concordia University, Montreal, Canada. In 1997, he joined the Beijing University of Technology, Beijing, China, where he is currently a Professor. His research interest includes QoS-aware networking and radio resource management in wireless networks.

WENJUN WU (Member, IEEE) received the B.S. and Ph.D. degrees from the Beijing University of Posts and Telecommunications, China, in 2007 and 2012, respectively. From 2012 to 2015, she was a Postdoctoral Researcher with Beihang University, China. She is currently an Associate Professor with the Beijing University of Technology, China. Her research interests are in the field of mobile edge computing, blockchain, and deep reinforcement learning.