Joint Task Offloading and Computation in Cooperative Multicarrier Relaying-Based Mobile-Edge Computing Systems

Dieli Hu, Gaofei Huang, *Member, IEEE*, Dong Tang, *Member, IEEE*, Sai Zhao, and Hui Zheng

***Abstract*—This article studies a mobile-edge computing (MEC) system, where an access point (AP) and a relay node serve a user terminal over multicarrier subchannels. In the MEC system, the relay can assist not only task offloading but also task computation. Aiming at minimizing total energy consumption at the user terminal and the relay, the resource allocations, such as subcarrier allocation, power allocation, task partition, and offloading time and computation time allocation, are to be optimized, subject to a given task computation delay constraint. To achieve this goal, a novel cooperative MEC protocol is designed, where multicarrier subchannels are utilized for parallel task offloading by integrating the rateless coding technique. Then, under the newly designed protocol, the resource allocation optimization problem is formulated as a mixed-integer programming (MIP) problem that is challenging to solve. To tackle this MIP problem, continuous relaxation and algebraic transformation techniques are applied to transform it into a convex problem in order to reveal the lower bound of energy consumption performance. After that, by equivalently rewriting the integer subcarrier allocation constraint in the original optimization problem as the intersection of a convex set and a d.c. (difference of two convex sets) set, the problem is solved by the successive convex approximation to achieve a practical and efficient resource allocation scheme. Simulation results show that the proposed jointly cooperative task offloading and computation scheme can significantly reduce the energy consumption as compared to the baseline schemes, where the relay only assists the task offloading or task computation.**

***Index Terms*—Cooperative communication, mobile-edge computing (MEC), subcarrier allocation, task offloading.**

I. Introduction

MOBILE-EDGE computing (MEC) is a promising technique that enables mobile terminals to wirelessly offload their computation tasks to the servers at the edge of cellular networks. For various new applications with ultralow-latency demand in 5G networks, e.g., augmented reality (AR), autonomous driving, and mobile online gaming, MEC plays a critical role due to the small size and limited computation resources of the wireless devices. Therefore, how to deploy MEC for these applications has become a critical issue in 5G networks [1], [2].

In MEC systems, task offloading can be categorized into two types, namely, binary offloading and partial offloading [1]. In binary offloading, a computation tasks can only be executed as a whole either locally at the mobile terminal or remotely at the MEC server [3]. In partial offloading, a computation task can be partitioned into several parts, each of which is to be executed at the mobile terminal or at the MEC server. As compared with the binary offloading, the partial offloading has drawn more attention since the communication and computation resources in partial offloading can be scheduled more flexibly. However, because of the flexible resource scheduling, it is also more challenging to achieve superior performance in the MEC systems with partial offloading and, thus, it has become a hot topic in wireless communications in recent years [4]–[16].

A. Related Works

The multicarrier transmission technique, such as orthogonal frequency-division multiplexing (OFDM), can address the frequency-selective fading issue to realize wireless broadband communication with a high data transmission rate. Therefore, by employing the multicarrier transmission technique in the MEC system, task offloading can be more efficient, so the system performance is promising to be improved. In [4], an orthogonal frequency-division multiple access (OFDMA)-based MEC system was studied, where the energy consumption in task offloading and computation was minimized by optimizing subcarrier allocation, power allocation, and task partition subject to the task computation delay constraints at multiple user terminals. In [5], the physical-layer security issue is further considered in an OFDMA-based MEC system in the presence of a malicious eavesdropper, and the weighted-sum energy consumption for multiple user terminals was minimized subjected to secrecy offloading rate constraints and the computation delay constraints. In [6] and [7], communication and computation resource allocations were jointly optimized to minimize the maximal computation delay of each user terminal. In [8], an OFDMA-based system with multiple MEC servers was studied. In [9], by integrating the in-network

Manuscript received October 25, 2020; accepted January 10, 2021. Date of publication January 13, 2021; date of current version July 7, 2021.This work was supported in part by the National Natural Science Foundation of China under Grant 61872098 and Grant 61902084; in part by the Guangdong Natural Science Foundation under Grant 2017A030313363; in part by the Featured Innovation Project of Guangdong Education Department under Grant 2018KTSCX174; and in part by the Basic Innovation Program for Post-Graduate Students by Guangzhou University under Grant 2019GDJC-M24. *(Corresponding author: Gaofei Huang.)*

The authors are with the Faculty of Electronics and Communication Engineering, Guangzhou University, Guangzhou 510006, China (e-mail: 2111807054@e.gzhu.edu.cn; huanggaofei@gzhu.edu.cn; tangdong@gzhu.edu.cn; zhaosai@gzhu.edu.cn; hzheng@gzhu.edu.cn).

Digital Object Identifier 10.1109/JIOT.2021.3051234

caching technique in an OFDMA-based MEC system, computation offloading and content caching strategies were designed to maximize the total revenue of the system.

In addition to the multicarrier transmission technique, cooperative relaying is also a key technique in wireless communications since it can increase the coverage area by communicating over a larger distance or around obstacles. Therefore, how to employ cooperative relaying to design high-performance MEC systems is another attractive topic in wireless communications in recent years [10], [11]. In [10], a relay-based MEC system was studied, where a wireless-powered mobile terminal near an access point (AP) connected with the MEC server acted as a relay node to help a wireless-powered mobile terminal far from the AP to offload tasks to the AP. However, it was assumed in [10] that the relay node could not assist the mobile terminal in task computing. In [11], the MEC system in which a relay node could assist both task offloading and task computing was considered, and task offloading from a mobile terminal to the relay and AP was carried out over a single-carrier channel in a time-division manner.

B. Motivations

In this article, the MEC system deployed over a cooperative multicarrier relay network is studied, which consists of a user terminal, a relay node, and an AP connected with a MEC server. Although multicarrier transmission-based MEC systems and cooperative relaying-based MEC systems have been studied separately in previous works, how to deploy MEC in a cooperative multicarrier relay network has not been known yet. In the literature, it has been demonstrated that cooperative multicarrier relaying can significantly enhance system capacity while increasing the coverage area [17]–[19]. Therefore, by deploying MEC over a cooperative multicarrier relay network, task offloading can be carried out with a high transmission rate over a large distance such that the system performance is promising to be improved. To this end, the cooperative multicarrier relaying-based MEC system is investigated in this article, where an efficient joint task offloading and computing scheme is to be developed to improve the system performance. Moreover, as in [11], the relay node in the investigated MEC system is assumed to have computing capability since it can achieve more performance gain.

However, when designing a high-performance cooperative multicarrier relaying-based MEC system, there are two challenges to be tackled. First, a new task offloading and computation protocol is required, and the reasons are illustrated as follows. In the cooperative multicarrier relaying-based MEC system, where the relay can cooperate in task computing, some subcarriers can be used for task offloading from the user terminal to the relay for cooperative computing at the relay, and the other subcarriers can be used for task offloading from the user terminal to AP. That is, task offloading to the relay and AP can be accomplished simultaneously over different subcarriers. However, in the cooperative single-carrier relaying-based MEC system studied in [11], task offloading to the relay and AP was carried out in a time-division manner. Thus, the employed protocol does not apply to the cooperative multicarrier relaying-based MEC system. Moreover, the communication protocol employed in traditional cooperative multicarrier relay systems [20]–[22] cannot apply to the cooperative multicarrier relaying-based MEC system either. The reason is that the relay in the traditional cooperative multicarrier relay system is required to forward all its received information to a destination node. However, the relay in the cooperative multicarrier relaying-based MEC system uses some received information for computing and the other for forwarding to AP. Second, to achieve superior performance under the newly designed protocol, subcarriers are required to be properly allocated for task offloading from the user terminal to the relay and from the user terminal to AP, respectively, and power allocation over these subcarriers is required to be optimized. The involved problem is a mixed-integer programming (MIP) and nonconvex problem, which is NP-hard and hard to solve. Due to the two reasons mentioned above, it is nontrivial to design a cooperative multicarrier relaying-based MEC system with superior performance.

In traditional cooperative relay networks, rateless code (RC) [23], [24] has been advocated to accomplish dynamic decode-and-forward (DDF) relaying to enhance throughput performance in the last decade [25]–[33]. In rateless coded systems, a transmitter broadcasts an infinite coded stream that is produced by encoding packets/symbols with an RC, and the receiver periodically attempts to decode the received message; once the received mutual information marginally exceeds the entropy of the message, the receiver is able to successfully decode and recover the original information. When employing RC in a cooperative relay system with DDF relaying, the relay is enabled to autonomously switch from listening to the source node to transmitting to the destination node to enhance spectral efficiency. Motivated by the advantages of RC in cooperative relay networks, the rateless coding technique is integrated into the newly designed cooperative multicarrier relaying-based MEC protocol to improve the task offloading efficiency in our work.

C. Contributions

In this article, by designing a cooperative multicarrier relaying-based MEC system, our contributions are summarized as follows.

1) We propose a novel protocol that enables the user terminal to simultaneously offload the task to the relay and AP for cooperative computation over parallel multicarrier subchannels. Moreover, in the proposed protocol, the rateless coding technique is integrated to enhance the performance of the studied MEC system.
2) Under the proposed protocol, we optimize CPU frequencies at the user terminal and the relay, subcarrier allocation, power allocation, task partition, as well as the allocation of task offloading and task computation time to minimize the energy consumption at the user terminal and the relay. Because subcarrier allocation optimization involves integer programming, the formulated optimization problem is an MIP and nonconvex

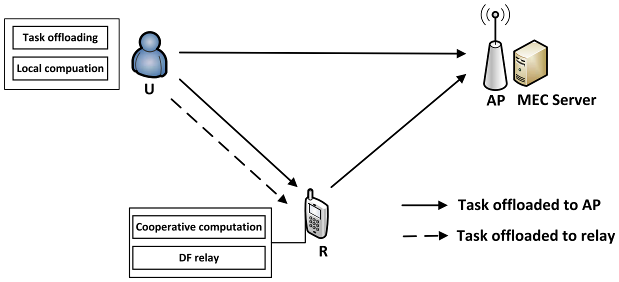


Fig. 1. System model.

problem that is hard to solve directly. By employing continuous relaxation and algebraic transformation techniques, we propose to solve this problem to achieve a lower bound of the energy consumption performance. Furthermore, by equivalently rewriting the integer subcarrier allocation constraint as the intersection of a convex set and a d.c. (difference of two convex sets) set, we achieve a practical optimization scheme by a successive convex approximation (SCA) method.

3) We provide numerical results to validate our proposed joint task offloading and computing scheme. The provided numerical results show that as compared with the baseline schemes where the relay only assists in task offloading or task computing, our proposed scheme can achieve much lower energy consumption in the studied MEC system.

D. Organization

The remainder of this article is organized as follows. The system model and the proposed cooperative multicarrier relaying base MEC protocol are introduced in Section II. In Section III, the joint task offloading and computing algorithm is presented. Performance evaluations are given in Section IV, and conclusions are drawn in Section V.

II. System Model and Cooperative Multicarrier Relaying-Based MEC Protocol

The cooperative multicarrier relaying-based MEC system considered in this article is illustrated in Fig. 1, which consists of a user terminal (U), a half-duplex decode-and-forward (DF) relay (R), and an AP connected with a MEC server. The three nodes send and receive their signals over multicarrier subchannels. The number of subcarriers is N, and the set of subcarriers is denoted as $\mathbb{N} \triangleq \{1, \ldots, N\}$. There is a direct link between the user terminal and AP, but the communications between the user terminal and AP need to be assisted by the relay due to a long distance or obstacles. It is assumed that the relay is a wireless node that is capable of computation, e.g., a mobile smartphone. Thus, the relay can assist not only task offloading from the user terminal to AP but also cooperative task computing for the user terminal.

As the user terminal can offload its tasks to both the relay and AP, it is assumed that partial offloading is adopted in the MEC system. Denote the size (in bits) of the computation task data (e.g., the program codes and input parameters) at the user terminal as $L > 0$. As the computation task is partitioned into three parts, denote l_U, l_R, and l_A as the data size for local computing at the user terminal, the offloaded data size for cooperative computing at the relay, and the offloaded data size for cooperative computing at AP, respectively. Then, it follows that:

$$l_U + l_R + l_A = L. \tag{1}$$

Furthermore, it is assumed that the L-bit computation task is required to be successfully executed within duration T, which is denoted as a time frame in this article.

Note that the equality in (1) indicates full granularity in task data partition. In other words, the task data could be partitioned into the subsets of any size. Such an assumption has been adopted in many works in the literature, e.g., [4]–[16], to simplify the analysis. By taking this assumption, the fundamental tradeoffs in the offloading process can be understood, and the achieved system performance can provide an upper bound for realistic offloading strategies.

Within the duration of one time frame, it is assumed that the wireless channels do not vary over each subcarrier. Let h_n^{UR}, h_n^{RA}, and h_n^{UA} denote the nth subcarrier channel coefficients of the UR (from the user terminal to the relay) link, the RA (from the relay to AP) link, and the UA (from the user terminal to AP) link, respectively. The corresponding normalized channel gains are denoted as $\gamma_n^{\mathrm{UR}} = (|h_n^{\mathrm{UR}}|^2/\sigma_R^2)$, $\gamma_n^{\mathrm{RA}} = (|h_n^{\mathrm{RA}}|^2/\sigma_A^2)$, and $\gamma_n^{\mathrm{UA}} = (|h_n^{\mathrm{UA}}|^2/\sigma_A^2)$, respectively, where σ_R^2 and σ_A^2 are the variances of the additive white Gaussian noise (AWGN) at the receivers of the relay and AP, respectively. Note that the perfect channel state information (CSI) is assumed to be available in this article, which can facilitate the design of the cooperative multicarrier relaying-based MEC system. The perfect CSI assumption has been adopted in many works on multicarrier relay systems in the literature, e.g., [11], [20]–[22], [49], [50], which can provide an insight into the system design. To achieve accurate the CSI in practical implementation, the user terminal transmits training signals during the first phase while the relay and AP measure the UR and UA channels, respectively. The relay then transmits training signals in the second phase to let the AP measure the RA channel. The measured channel gains can be fed back to the user terminal on dedicated reverse control channels.

A. Proposed Cooperative Multicarrier Relaying-Based MEC Protocol

For the cooperative multicarrier relaying-based MEC system illustrated in Fig. 1, a new protocol is proposed to enable the user terminal to offload its computation task to the relay and AP over parallel multicarrier subchannels in this article. The time frame structure of the proposed cooperative multicarrier relaying-based MEC protocol is illustrated in Fig. 2. In the proposed protocol, DDF relaying is employed and a time frame consists of three phases, whose time lengths are τ_1, τ_2, and τ_3, respectively. In the first phase, the user terminal sends its offloaded computation task data to AP and the relay over two subcarrier sets, i.e., $\mathbb{N}_A^{(1,o)}$ and $\mathbb{N}_{UR}^{(1,c)}$, respectively, where $\mathbb{N} = \mathbb{N}_A^{(1,o)} \bigcup \mathbb{N}_{UR}^{(1,c)}$ and $\mathbb{N}_A^{(1,o)} \bigcap \mathbb{N}_{UR}^{(1,c)} = \emptyset$ with $\emptyset$ denotes

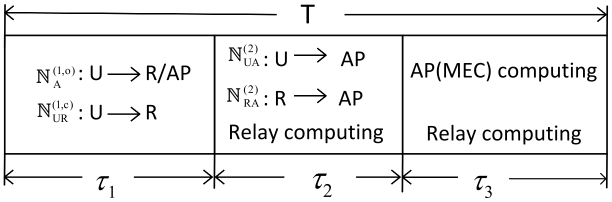

T
$\mathbb{N}_{A}^{(1,o)}$: U ⟶ R/AP
$\mathbb{N}_{UR}^{(1,c)}$: U ⟶ R
$\mathbb{N}_{UA}^{(2)}$: U ⟶ AP
$\mathbb{N}_{RA}^{(2)}$: R ⟶ AP
Relay computing
AP(MEC) computing
Relay computing
τ_1
τ_2
τ_3

Fig. 2. Time frame structure of the proposed cooperative multicarrier relaying-based MEC protocol.

the empty set. Over the subcarriers in $\mathbb{N}_A^{(1,o)}$, the user terminal broadcasts the signals of the computation task data offloaded to AP; over the subcarriers in $\mathbb{N}_{UR}^{(1,c)}$, the user terminal sends the computation task data offloaded to the relay for cooperative computing. The computation task data offloaded to AP modulated over the subcarriers in $\mathbb{N}_A^{(1,o)}$ are encoded with an RC, and both the relay and AP try to decode these data when they receive the signals broadcast by the user terminal. In the case that the relay successfully decodes before AP, the second phase starts and the relay re-encodes the left-over data that have not been decoded at AP and transmits them to AP over the subcarriers in the subcarrier set $\mathbb{N}_{RA}^{(2)}$. Meanwhile, the user terminal continuously sends the computation task data offloaded to AP over the other subcarriers in the subcarrier set $\mathbb{N}_{UA}^{(2)}$, and the relay starts executing the cooperative computing based on the computation task data received over the subcarriers in $\mathbb{N}_{UR}^{(1,c)}$. Here, $\mathbb{N} = \mathbb{N}_{UA}^{(2)} \bigcup \mathbb{N}_{RA}^{(2)}$ and $\mathbb{N}_{UA}^{(2)} \bigcap \mathbb{N}_{RA}^{(2)} = \emptyset$. In the third phase, the MEC server starts executing the computation task based on the computation task data offloaded to AP in the first and second phases, while the relay continues computing the cooperative computation task.

To facilitate the expression of subcarrier allocation, let $\omega_{A,n}^{(1,o)}, \omega_{R,n}^{(1,c)}, \omega_{RA,n}^{(2)}, \omega_{UA,n}^{(2)} \in \{0, 1\}$ be subcarrier allocation indicators. In other words, if subcarrier n is allocated to one of the four subcarrier sets [i.e., $\mathbb{N}_A^{(1,o)}$, $\mathbb{N}_{UR}^{(1,c)}$, $\mathbb{N}_{RA}^{(2)}$, and $\mathbb{N}_{UA}^{(2)}$], the corresponding subcarrier allocation indicator is 1; otherwise, the corresponding subcarrier allocation indicator is 0. In the first phase, subcarrier n can be allocated to either $\mathbb{N}_A^{(1,o)}$ or $\mathbb{N}_{UR}^{(1,c)}$. Thus, it follows that:

$$\omega_{A,n}^{(1,o)} + \omega_{R,n}^{(1,c)} \leq 1 \ \forall n \in \mathbb{N}. \tag{2}$$

Similarly, in the second phase, subcarrier n can be allocated to either $\mathbb{N}_{RA}^{(2)}$ or $\mathbb{N}_{UA}^{(2)}$. Thus, it follows that:

$$\omega_{RA,n}^{(2)} + \omega_{UA,n}^{(2)} \leq 1 \ \forall n \in \mathbb{N}. \tag{3}$$

As the size of task computing results is normally of much smaller than that of computation task data, the time for downloading the task computing results to the user terminal is negligible compared to the offloading time [10], [11]. Thus, the downloading time is ignored in this article. As a result, to accomplish the computation task within duration T, τ_i ($i \in \{1, 2, 3\}$) should satisfy the inequality as follows:

$$\tau_1 + \tau_2 + \tau_3 \leq T. \tag{4}$$

B. Task Offloading and Energy Consumption Model

1) Task Offloading for Cooperative Computing at the Relay and the Involved Energy Consumption: According to the time structure of the proposed cooperative relaying-based MEC protocol, the computation task data offloaded to the relay are sent over the subcarriers in $\mathbb{N}_{UR}^{(1,c)}$ in the first phase. Let $p_{U,n}^{(1)}$ denote the transmit power over subcarrier n at the user terminal in the first phase. Then, the energy consumption for task offloading from the user terminal to the relay is calculated as

$$E_R = \sum_{n=1}^{N} \tau_1 \omega_{R,n}^{(1,c)} p_{U,n}^{(1)}. \tag{5}$$

Moreover, the achievable data transmission rate from the user terminal to the relay over subcarrier n in the first phase can be expressed as

$$r_{UR,n}\left(p_{U,n}^{(1)}\right) = B\log_2\left(1 + p_{U,n}^{(1)}\gamma_n^{UR}\right). \tag{6}$$

Thus, the computation task data offloaded to the relay should satisfy the inequality as follows:

$$l_R \leq \sum_{n=1}^{N} \tau_1 \omega_{R,n}^{(1,c)} r_{UR,n}\left(p_{U,n}^{(1)}\right). \tag{7}$$

2) Task Offloading for Cooperative Computing at AP and the Involved Energy Consumption: The computation task data offloaded to AP consist of two parts. The first part is received from the direct link and the relay link, which is transmitted with rateless coding technique via DDF cooperative relaying. The computation task data received from the direct link are transmitted by the user terminal over the subcarriers in $\mathbb{N}_A^{(1,o)}$ in the first phase. Because the achievable data transmission rate from the user terminal to AP over subcarrier n in the first phase is calculated as $r_{UA,n}^{(1,o)}(p_{U,n}^{(1)}) = B\log_2(1 + p_{U,n}^{(1)}\gamma_n^{UA})$, the maximum size of computation task data decoded over the subcarriers in $\mathbb{N}_A^{(1,o)}$ at AP can be expressed as

$$l_{UA}^{(1)} = \sum_{n=1}^{N} \tau_1 \omega_{A,n}^{(1,o)} r_{UA,n}^{(1,o)}\left(p_{U,n}^{(1)}\right). \tag{8}$$

Meanwhile, the relay can also receive the involved signals over the subcarriers in $\mathbb{N}_A^{(1,o)}$, and the maximum data size decoded at the relay can be obtained as

$$l_{UR}^{(1)} = \sum_{n=1}^{N} \tau_1 \omega_{A,n}^{(1,o)} r_{UR,n}\left(p_{U,n}^{(1)}\right) \tag{9}$$

where $r_{UR,n}(p_{U,n}^{(1)})$ is provided in (6). If the amount of decoded data at the relay is less than that at AP, the relay need not forward any data to AP. Otherwise, the relay re-encodes the decoded data that have not been decoded at AP and forwards them over the subcarriers in $\mathbb{N}_{RA}^{(2)}$. Let $p_{R,n}$ denote the transmit power over subcarrier n at the relay. Then, the achievable data transmission rate from the relay to AP over subcarrier n is calculated as $r_{RA,n}(p_{R,n}) = B\log_2(1 + p_{R,n}\gamma_n^{RA})$. Thus, the maximum size of computation task data decoded over the subcarriers in $\mathbb{N}_{RA}^{(2)}$ at AP can be expressed as

$$l_{RA} = \sum_{n=1}^{N} \tau_2 \omega_{RA,n}^{(2)} r_{RA,n}\left(p_{R,n}\right). \tag{10}$$

As a result, as for the first part of computation task data offloaded to AP, the size can be obtained as [33]

$$\begin{aligned} l_A^I &= \min\left\{\max\left\{l_{UR}^{(1)}, l_{UA}^{(1)}\right\}, l_{UA}^{(1)} + l_{RA}\right\} \\ &= \min\Bigg\{\max\Bigg\{\sum_{n=1}^{N} \tau_1 \omega_{A,n}^{(1,o)} r_{UR,n}\left(p_{U,n}^{(1)}\right) \\ &\qquad \sum_{n=1}^{N} \tau_1 \omega_{A,n}^{(1,o)} r_{UA,n}^{(1)}\left(p_{U,n}^{(1)}\right)\Bigg\} \\ &\qquad \sum_{n=1}^{N} \tau_1 \omega_{A,n}^{(1,o)} r_{UA,n}^{(1)}(p_{U,n}^{(1)}) + \sum_{n=1}^{N} \tau_2 \omega_{RA,n}^{(2)} r_{RA,n}\left(p_{R,n}\right)\Bigg\}. \end{aligned} \tag{11}$$

Besides, the energy consumption for offloading these task data can be calculated as

$$E_A^I = \sum_{n=1}^{N} \tau_1 \omega_{A,n}^{(1,o)} p_{U,n}^{(1)} + \sum_{n=1}^{N} \tau_2 \omega_{RA,n}^{(2)} p_{R,n}. \tag{12}$$

The second part of computation task data offloaded to AP is transmitted by the user terminal over the subcarriers in $\mathbb{N}_{UA}^{(2)}$ via the direct link. Let $p_{U,n}^{(2)}$ denote the transmit power at the user terminal over subcarrier n in the second phase. Then, the achievable data transmission rate from the user terminal to AP over subcarrier n in the second phase is calculated as $r_{UA,n}^{(2)}(p_{U,n}^{(2)}) = B\log_2(1 + p_{U,n}^{(2)}\gamma_n^{UA})$. Thus, as for the second part of computation task data offloaded to AP, the size can be calculated as

$$l_A^{II} = \sum_{n=1}^{N} \tau_2 \omega_{UA,n}^{(2)} r_{UA,n}^{(2)}\left(p_{U,n}^{(2)}\right). \tag{13}$$

Therefore, the size of computation task data offloaded to AP within one time frame should satisfy the inequality as follows:

$$\begin{aligned} l_A &\le l_A^I + l_A^{II} \\ &= \min\Bigg\{\max\Bigg\{\sum_{n=1}^{N} \tau_1 \omega_{A,n}^{(1,o)} r_{UR,n}\left(p_{U,n}^{(1)}\right) \\ &\qquad \sum_{n=1}^{N} \tau_1 \omega_{A,n}^{(1,o)} r_{UA,n}^{(1)}\left(p_{U,n}^{(1)}\right)\Bigg\} \\ &\qquad \sum_{n=1}^{N} \tau_1 \omega_{A,n}^{(1,o)} r_{UA,n}^{(1)}\left(p_{U,n}^{(1)}\right) + \sum_{n=1}^{N} \tau_2 \omega_{RA,n}^{(2)} r_{RA,n}\left(p_{R,n}\right)\Bigg\} \\ &\quad + \sum_{n=1}^{N} \tau_2 \omega_{UA,n}^{(2)} r_{UA,n}^{(2)}\left(p_{U,n}^{(2)}\right). \end{aligned} \tag{14}$$

Also, the energy consumption for offloading the second part of computation task data to AP can be obtained as

$$E_A^{II} = \sum_{n=1}^{N} \tau_2 \omega_{UA,n}^{(2)} p_{U,n}^{(2)}. \tag{15}$$

Then, the energy consumption for offloading the computation task data to AP can be calculated as

$$\begin{aligned} E_A &= E_A^I + E_A^{II} \\ &= \sum_{n=1}^{N} \tau_1 \omega_{A,n}^{(1,o)} p_{U,n}^{(1)} + \sum_{n=1}^{N} \tau_2 \omega_{RA,n}^{(2)} p_{R,n} \\ &\quad + \sum_{n=1}^{N} \tau_2 \omega_{UA,n}^{(2)} p_{U,n}^{(2)}. \end{aligned} \tag{16}$$

As a result, the total energy consumption for offloading computation task data in the MEC system can be calculated as

$$\begin{aligned} E_o &= E_A + E_R \\ &= \sum_{n=1}^{N} \tau_1 \omega_{A,n}^{(1,o)} p_{U,n}^{(1)} + \sum_{n=1}^{N} \tau_2 \omega_{RA,n}^{(2)} p_{R,n} \\ &\quad + \sum_{n=1}^{N} \tau_2 \omega_{UA,n}^{(2)} p_{U,n}^{(2)} + \sum_{n=1}^{N} \tau_1 \omega_{R,n}^{(1,c)} p_{U,n}^{(1)}. \end{aligned} \tag{17}$$

C. Task Computing and Energy Consumption Model

As partial offloading is adopted in this article, task computing can be carried out jointly at the user terminal, relay, and AP. In the following, the computing models at the three nodes are described, respectively.

For the user terminal and the relay, it is assumed that the numbers of CPU cycles for executing one-bit computation task are C_U and C_R, respectively. Recall that the sizes of computation task data at the user and the relay are l_U and l_R, respectively. Therefore, the numbers of CPU cycles for task computation at the user and the relay can be calculated as $C_U l_U$ and $C_R l_R$, respectively. Denote the CPU frequencies for the nth cycle at the user terminal and the relay as $f_{U,i}$ and $f_{R,i}$, respectively, and let the maximum values of $f_{U,i}$ and $f_{R,i}$ be $f_{U,\max}$ and $f_{R,\max}$, respectively, i.e.,

$$0 \le f_{U,i} \le f_{U,\max}\ \forall i \in \{1, \ldots, C_U l_U\} \tag{18}$$

$$0 \le f_{R,i} \le f_{R,\max}\ \forall i \in \{1, \ldots, C_R l_R\}. \tag{19}$$

Because the l_U-bit computation task is required to be successfully executed at the user terminal within duration T, $f_{U,i}$ should satisfy

$$\sum_{i=1}^{C_U l_U} \frac{1}{f_{U,i}} \le T. \tag{20}$$

Meanwhile, because the l_R-bit computation task is required to be successfully executed at the relay within duration $T - \tau_1$, $f_{R,i}$ should satisfy

$$\sum_{i=1}^{C_R l_R} \frac{1}{f_{R,i}} \le T - \tau_1. \tag{21}$$

Moreover, the maximum data size of the computation task that can be accomplished at the user terminal within duration T is obtained as $(T f_{U,\max}/C_U)$. Thus, l_U should satisfy

$$C_U l_U \le T f_{U,\max}. \tag{22}$$

The maximum data size of the computation task that can be accomplished at the relay within duration $T - \tau_1$ is obtained as $[((T - \tau_1) f_{R,\max})/C_R]$. Thus, l_R should satisfy

$$C_R l_R \le (T - \tau_1) f_{R,\max}. \tag{23}$$

Besides, let K_U and K_R denote the effective capacitance coefficient that depends on the chip architecture [36] at the user and the relay, respectively. Then, the energy consumption for executing the l_U-bit and l_R-bit computation tasks at the user and the relay can be calculated as [1]

$$E_{U,c} = \sum_{i=1}^{C_U l_U} K_U f_{U,i}^2 \tag{24}$$

and

$$E_{R,c} = \sum_{i=1}^{C_R l_R} K_R f_{R,i}^2 \tag{25}$$

respectively.

In this article, it is assumed that AP and the MEC server are powered by the grid, and the user terminal and relay are powered by batteries. Our goal is to minimize the energy consumption at the user terminal and the relay. Thus, the computing energy consumption at the MEC server is ignored. As a result, the total energy consumed in the considered MEC system can be obtained as

$$\begin{aligned} E_c &= E_{U,c} + E_{R,c} \\ &= \sum_{i=1}^{C_U l_U} K_U f_{U,i}^2 + \sum_{i=1}^{C_R l_R} K_R f_{R,i}^2. \end{aligned} \tag{26}$$

Moreover, to reduce the computation delay, the MEC server should execute the computation task with its maximum CPU's frequency $f_{A,\max}$ since the task computing energy consumption at the MEC server is ignored. Thus, the time for computing l_A-bit task at the MEC server is obtained as

$$\tau_3 = \frac{C_A l_A}{f_{A,\max}}. \tag{27}$$

Then, the delay constraint for computing the L-bit task described in (4) can be modified as

$$\tau_1 + \tau_2 + \frac{C_A l_A}{f_{A,\max}} \leq T. \tag{28}$$

The mainly used notations and their description are summarized in Table I.

TABLE I
NOTATION DEFINITIONS

Symbol	Description
$\mathbb{N}$	set of subcarriers
N	the number of subcarriers
n	the subcarrier index $n \in \mathbb{N}$
U, R, A	user terminal / DF relay / AP
L	the size (in bits) of the computation task data at the user terminal
l_{U}, l_{R}, l_{A}	the size (in bits) for computing at the user terminal / relay / AP
γ_n^{UR}, γ_n^{RA}, γ_n^{UA}	the corresponding normalized channel gains of the UR link / RA link / UA link
T	the task delay constraint
τ_1, τ_2, τ_3	the length of the first phase / second phase / third phase
$\mathbb{N}_{\mathrm{A}}^{(1,\mathrm{o})}$	the subcarrier set used for user to broadcast the signals of the task data to AP in the first phase
$\mathbb{N}_{\mathrm{UR}}^{(1,\mathrm{c})}$	the subcarrier set used for user to offload the task data to the relay for cooperative computing in the first phase
$\mathbb{N}_{\mathrm{RA}}^{(2)}$	the subcarrier set used for relay to offload the task data to AP in the second phase
$\mathbb{N}_{\mathrm{UA}}^{(2)}$	the subcarrier set used for user to offload the task data to AP in the second phase
$\omega_{\mathrm{A},n}^{(1,\mathrm{o})}$, $\omega_{\mathrm{R},n}^{(1,\mathrm{c})}$, $\omega_{\mathrm{RA},n}^{(2)}$, $\omega_{\mathrm{UA},n}^{(2)}$	the subcarrier allocation indicators representing subcarrier n is allocated to $\mathbb{N}_{\mathrm{A}}^{(1,\mathrm{o})}$ / $\mathbb{N}_{\mathrm{UR}}^{(1,\mathrm{c})}$ / $\mathbb{N}_{\mathrm{RA}}^{(2)}$ / $\mathbb{N}_{\mathrm{UA}}^{(2)}$
$p_{\mathrm{U},n}^{(1)}$	the transmit power over subcarrier n at the user in the first phase
$p_{\mathrm{R},n}$, $p_{\mathrm{U},n}^{(2)}$	the transmit power over subcarrier n at the user / relay in the second phase
$f_{\mathrm{U},i}$, $f_{\mathrm{R},i}$	the CPU frequencies for the n-th cycle at the user / relay
$f_{\mathrm{U},max}$, $f_{\mathrm{R},max}$	the maximum values of $f_{\mathrm{U},i}$ / $f_{\mathrm{R},i}$
C_{U}, C_{R}, C_{A}	the numbers of CPU cycles for executing one-bit computation task at the user / relay / AP
K_{U}, K_{R}	the effective capacitance coefficient at the user / relay / AP

Remark: In this article, in order to facilitate analysis and provide an insight into the design strategy for the cooperative multicarrier relaying-based MEC system, it is assumed that the relay does not have its own computation task. Nevertheless, it is worth pointing out that the cooperative multicarrier relaying-based MEC protocol proposed in this article and the jointly offloading and computation algorithm provided in the next section can be easily extended to the scenario where the relay has its own computation task. The details on the extensions are omitted in this article for brevity.

III. Jointly Cooperative Offloading and Computation Algorithm

A. Problem Statement

In this article, our goal is to minimize the total energy consumption of the studied MEC system, subject to the user terminal's computation task execution delay constraint in (28), task offloading constraints in (7), (14), (22), and (23), the CPU frequency constraints at the user and the relay in (18)–(21), subcarrier allocation constraints in (2) and (3), and the maximum transmit power constraints at the user terminal and the relay. To achieve this goal, CPU frequencies at the user terminal and the relay, subcarrier allocation, power allocation, time lengths of the three phases in one time frame and task partition are jointly optimized. In the MEC system, the total energy consumption consists of two parts, i.e., task offloading energy consumption and computing energy consumption at the user and the relay, which can be expressed as

$$\begin{aligned} E_{\text{total}}(\boldsymbol{\omega}, \boldsymbol{p}, \boldsymbol{\tau}, \boldsymbol{f}) &= E_o + E_c \\ &= \sum_{n=1}^{N} \tau_1 \omega_{A,n}^{(1,o)} p_{U,n}^{(1)} + \sum_{n=1}^{N} \tau_2 \omega_{RA,n}^{(2)} p_{R,n} \\ &\quad + \sum_{n=1}^{N} \tau_2 \omega_{UA,n}^{(2)} p_{U,n}^{(2)} \\ &\quad + \sum_{n=1}^{N} \tau_1 \omega_{R,n}^{(1,c)} p_{U,n}^{(1)} + \sum_{i=1}^{C_U l_U} K_U f_{U,i}^2 \\ &\quad + \sum_{i=1}^{C_R l_R} K_R f_{R,i}^2 \end{aligned} \tag{29}$$

where $\boldsymbol{\omega} = [\omega_{A,n}^{(1,o)}, \omega_{R,n}^{(1,c)}, \omega_{RA,n}^{(2)}, \omega_{UA,n}^{(2)}]$, $\boldsymbol{p} = [p_{U,n}^{(1)}, p_{R,n}, p_{U,n}^{(2)}]$, $\boldsymbol{\tau} = [\tau_1, \tau_2]$, and $\boldsymbol{f} = [f_{U,i}, f_{R,i}]$.

Therefore, the problem to optimize the resource allocation for the studied MEC system can be formulated as

$$\min_{\boldsymbol{\omega},\boldsymbol{p},\boldsymbol{\tau},\boldsymbol{f},\boldsymbol{l}} \; E_{\text{total}}(\boldsymbol{\omega}, \boldsymbol{p}, \boldsymbol{\tau}, \boldsymbol{f}) \tag{30a}$$

$$\text{s.t.} \;\; (1)-(3), (7), (14), (18)-(23), (28) \text{ and} \tag{30b}$$

$$0 \le \tau_i \le T \quad \forall i = \{1, 2\} \tag{30c}$$

$$l_U \ge 0,\; l_R \ge 0,\; l_A \ge 0 \tag{30d}$$

$$p_{U,n}^{(1)} \ge 0,\; p_{U,n}^{(2)} \ge 0,\; p_{R,n} \ge 0 \quad \forall n \in \mathbb{N} \tag{30e}$$

$$\omega_{A,n}^{(1,o)},\; \omega_{R,n}^{(1,c)},\; \omega_{RA,n}^{(2)},\; \omega_{UA,n}^{(2)} \in \{0, 1\} \; \forall n \in \mathbb{N} \tag{30f}$$

$$\sum_{n=1}^{N} \Big[\omega_{A,n}^{(1,o)} + \omega_{R,n}^{(1,c)}\Big] p_{U,n}^{(1)} \le P_{U,\max} \tag{30g}$$

$$\sum_{n=1}^{N} \omega_{UA,n}^{(2)} p_{U,n}^{(2)} \le P_{U,\max} \tag{30h}$$

$$\sum_{n=1}^{N} \omega_{RA,n}^{(2)} p_{R,n} \le P_{R,\max} \tag{30i}$$

where $\boldsymbol{l} = [l_U, l_R, l_A]$, $P_{U,\max}$ and $P_{R,\max}$ are the maximum transmit power at the user terminal and the relay, respectively, (30g) and (30h) are the transmit power constraints at the user terminal in the first and second phases, respectively, and (30i) is the transmit power constraint at the relay.

B. Simplifying the Problem and Problem Transformation by Continuous Relaxation

Problem (30) is an MIP and nonconvex problem, thus it is hard to solve. Moreover, the task offloading constraint in (14) makes this problem more complicated. To simplify the constraint (14), problem (39) is proposed to be solved with two steps. First, by assuming that the relay can decode more data than AP, constraint (14) is simplified as

$$\begin{aligned} l_A \le \min\Bigg\{ &\sum_{n=1}^{N} \tau_1 \omega_{A,n}^{(1,o)} r_{UR,n}\Big(p_{U,n}^{(1)}\Big) \\ &\sum_{n=1}^{N} \tau_1 \omega_{A,n}^{(1,o)} r_{UA,n}^{(1)}\Big(p_{U,n}^{(1)}\Big) + \sum_{n=1}^{N} \tau_2 \omega_{RA,n}^{(2)} r_{RA,n}\big(p_{R,n}\big) \Bigg\} \\ &+ \sum_{n=1}^{N} \tau_2 \omega_{UA,n}^{(2)} r_{UA,n}^{(2)}\Big(p_{U,n}^{(2)}\Big). \end{aligned} \tag{31}$$

Then, replacing constraint (14) in (30b) with (31), problem (30) becomes

$$\min_{\boldsymbol{\omega},\boldsymbol{p},\boldsymbol{\tau},\boldsymbol{f},\boldsymbol{l}} \; E_{\text{total}}(\boldsymbol{\omega}, \boldsymbol{p}, \boldsymbol{\tau}, \boldsymbol{f}) \tag{32a}$$

$$\begin{aligned} \text{s.t.} \;\; &(1)-(3), (7), (18)-(23), (28) \\ &(30\text{c})-(30\text{i}), \text{ and } (31). \end{aligned} \tag{32b}$$

If the solution to problem (32) satisfies $\sum_{n=1}^{N} \tau_1 \omega_{A,n}^{(1,o)} r_{UR,n}(p_{U,n}^{(1)}) > \sum_{n=1}^{N} \tau_1 \omega_{A,n}^{(1,o)} r_{UA,n}^{(1)}(p_{U,n}^{(1)})$, the relay can decode more data than AP over the subcarriers in $\mathbb{N}_A^{(1,o)}$. Thus, the obtained solution is also applied to problem (30). Otherwise, the user terminal offloads task data to AP directly over the subcarriers in $\mathbb{N}_A^{(1,o)}$ in the first phase and over all subcarriers in $\mathbb{N}$ in the second phase, respectively. In this case, the second step is required, where problem (30) is solved by letting $p_{R,n} = 0$, $\omega_{RA,n}^{(2)} = 0$, and $\omega_{UA,n}^{(2)} = 1 \; \forall n \in \mathbb{N}$, and replacing constraint (14) in (30b) by the inequality as follows:

$$\begin{aligned} l_A \le &\sum_{n=1}^{N} \tau_1 \omega_{A,n}^{(1,o)} r_{UA,n}^{(1)}\Big(p_{U,n}^{(1)}\Big) \\ &+ \sum_{n=1}^{N} \tau_2 \omega_{UA,n}^{(2)} r_{UA,n}^{(2)}\Big(p_{U,n}^{(2)}\Big). \end{aligned} \tag{33}$$

Because the obtained problem can be solved by the approach that is used to solve problem (32), it is only focused on how to solve problem (32) in the follows.

To solve problem (32), the following lemma is provided to achieve the optimal solution of $\boldsymbol{f}$ to simplify the problem.

Lemma 1: The optimal solutions of the user's CPU frequency and the relay's CPU frequency (i.e., $\boldsymbol{f}$) to problem (32) should satisfy

$$f_{U,i}^{*} = C_U l_U / T \quad \forall i \in \{1, \ldots, C_U l_U\} \tag{34}$$

$$f_{R,i}^{*} = C_R l_R / (T - \tau_1) \quad \forall i \in \{1, \ldots, C_R l_R\}. \tag{35}$$

Proof: See Appendix A. ■

Based on Lemma 1, the energy consumption for executing the l_U-bit computation task at the user described in (24) can be modified as

$$E_{U,c} = \frac{K_U C_U^3 l_U^3}{T^2} \tag{36}$$

and the energy consumption for executing the l_R-bit computation task at the relay described in (25) can be modified as

$$E_{R,c} = \frac{K_R C_R^3 l_R^3}{(T - \tau_1)^2}. \tag{37}$$

Thus, $E_{\text{total}}(\boldsymbol{\omega}, \boldsymbol{p}, \boldsymbol{\tau}, \boldsymbol{f})$ described in (29) can be rewritten as

$$\begin{aligned} E'_{\text{total}}(\boldsymbol{\omega}, \boldsymbol{p}, \boldsymbol{\tau}, \boldsymbol{l}) = &\sum_{n=1}^{N} \tau_1 \omega_{A,n}^{(1,o)} p_{U,n}^{(1)} + \sum_{n=1}^{N} \tau_2 \omega_{RA,n}^{(2)} p_{R,n} \\ &+ \sum_{n=1}^{N} \tau_2 \omega_{UA,n}^{(2)} p_{U,n}^{(2)} \\ &+ \sum_{n=1}^{N} \tau_1 \omega_{R,n}^{(1,c)} p_{U,n}^{(1)} + \frac{K_U C_U^3 l_U^3}{T^2} \\ &+ \frac{K_R C_R^3 l_R^3}{(T - \tau_1)^2}. \end{aligned} \tag{38}$$

Furthermore, constraints (18)–(21) in (30b) can be removed from problem (32). Thus, problem (32) can be simplified as

$$\min_{\boldsymbol{\omega},\boldsymbol{p},\boldsymbol{\tau},\boldsymbol{l}} \; E'_{\text{total}}(\boldsymbol{\omega}, \boldsymbol{p}, \boldsymbol{\tau}, \boldsymbol{l}) \tag{39a}$$

$$\text{s.t.} \;\; (1)-(3), (7),\; (22), (23), (28)\, (30\text{c})-(30\text{i}) \text{ and } (31). \tag{39b}$$

In the simplified problem (39), to deal with the integer constraint (30f) in (39b), the continuous relaxation method is employed. That is, the subcarrier allocation factors in $\boldsymbol{\omega}$ are allowed to take a value on the set of non-negative real numbers, which indicates that the computation task data can be offloaded to the relay and AP over the subcarriers by a time-sharing approach [37]. Furthermore, define $s_{A,n}^{(1,o)} = \tau_1\omega_{A,n}^{(1,o)}$, $s_{R,n}^{(1,c)} = \tau_1\omega_{R,n}^{(1,c)}$, $s_{RA,n}^{(2)} = \tau_2\omega_{RA,n}^{(2)}$, and $s_{UA,n}^{(2)} = \tau_2\omega_{UA,n}^{(2)}$. Then, for the constraints in (39b), (2) and (3) can be rewritten as

$$s_{A,n}^{(1,o)} + s_{R,n}^{(1,c)} \le \tau_1 \tag{40}$$

and

$$s_{RA,n}^{(2)} + s_{UA,n}^{(2)} \le \tau_2 \tag{41}$$

respectively, (7) and (31) can be rewritten as

$$l_R \le \sum_{n=1}^{N} s_{R,n}^{(1,c)} r_{UR,n}\left(p_{U,n}^{(1)}\right) \tag{42}$$

and

$$\begin{aligned} l_A \le \min\Bigg\{ & \sum_{n=1}^{N} s_{A,n}^{(1,o)} r_{UR,n}\left(p_{U,n}^{(1)}\right) \\ & \sum_{n=1}^{N} s_{A,n}^{(1,o)} r_{UA,n}^{(1)}\left(p_{U,n}^{(1)}\right) + \sum_{n=1}^{N} s_{RA,n}^{(2)} r_{RA,n}\left(p_{R,n}\right)\Bigg\} \\ & + \sum_{n=1}^{N} s_{UA,n}^{(2)} r_{UA,n}^{(2)}\left(p_{U,n}^{(2)}\right) \end{aligned} \tag{43}$$

respectively, (30f) can be rewritten as

$$s_{A,n}^{(1,o)}, s_{R,n}^{(1,c)} \in [0, \tau_1], s_{RA,n}^{(2)}, s_{UA,n}^{(2)} \in [0, \tau_2] \quad \forall n \in \mathbb{N} \tag{44}$$

and (30g)–(30i) can be rewritten as

$$\sum_{n=1}^{N}\left[s_{A,n}^{(1,o)} + s_{R,n}^{(1,c)}\right] p_{U,n}^{(1)} \le \tau_1 P_{U,\max} \tag{45}$$

$$\sum_{n=1}^{N} s_{UA,n}^{(2)} p_{U,n}^{(2)} \le \tau_2 P_{U,\max} \tag{46}$$

and

$$\sum_{n=1}^{N} s_{RA,n}^{(2)} p_{R,n}^{(2)} \le \tau_2 P_{R,\max} \tag{47}$$

respectively.

Moreover, the objective function in (39a), i.e., $E'_{\text{total}}(\boldsymbol{\omega}, \boldsymbol{p}, \boldsymbol{\tau}, \boldsymbol{l})$ that is described in (38), can be rewritten as

$$\begin{aligned} E''_{\text{total}}(\boldsymbol{s}, \boldsymbol{p}, \boldsymbol{\tau}, \boldsymbol{l}) = & \sum_{n=1}^{N} s_{A,n}^{(1,o)} p_{U,n}^{(1)} \\ & + \sum_{n=1}^{N} s_{RA,n}^{(2)} p_{R,n} + \sum_{n=1}^{N} s_{UA,n}^{(2)} p_{U,n}^{(2)} \\ & + \sum_{n=1}^{N} s_{R,n}^{(1,c)} p_{U,n}^{(1)} + \frac{K_U C_U^3 l_U^3}{T^2} \\ & + \frac{K_R C_R^3 l_R^3}{(T-\tau_1)^2} \end{aligned} \tag{48}$$

where $\boldsymbol{s} = [s_{A,n}^{(1,o)}, s_{R,n}^{(1,c)}, s_{RA,n}^{(2)}, s_{UA,n}^{(2)}]$.

Therefore, problem (39) can be rewritten as the following nonlinear programming problem:

$$\min_{\boldsymbol{s},\boldsymbol{p},\boldsymbol{\tau},\boldsymbol{l}} \; E''_{\text{total}}(\boldsymbol{s}, \boldsymbol{p}, \boldsymbol{\tau}, \boldsymbol{l}) \tag{49a}$$

$$\text{s.t.} \;\; (22), (23), \;(28), (30c) - (30e), \text{ and } (40)–(47). \tag{49b}$$

However, the objective function in (49a) and constraints (42) and (43) in (49b) are nonconvex, so problem (49) is still difficult to solve directly. To solve this nonconvex problem, let $e_{U,n}^{(1,o)} = s_{A,n}^{(1,o)} p_{U,n}^{(1)}$, $e_{R,n} = s_{RA,n}^{(2)} p_{R,n}$, $e_{U,n}^{(2)} = s_{UA,n}^{(2)} p_{U,n}^{(2)}$, and $e_{U,n}^{(1,c)} = s_{R,n}^{(1,c)} p_{U,n}^{(1)}$. Then, $E''_{\text{total}}(\boldsymbol{s}, \boldsymbol{p}, \boldsymbol{\tau}, \boldsymbol{l})$ in (48) can be rewritten as

$$\begin{aligned} E'''_{\text{total}}(\boldsymbol{e}, \boldsymbol{\tau}, \boldsymbol{l}) = & \sum_{n=1}^{N} e_{U,n}^{(1,o)} + \sum_{n=1}^{N} e_{R,n} + \sum_{n=1}^{N} e_{U,n}^{(2)} \\ & + \sum_{n=1}^{N} e_{U,n}^{(1,c)} + \frac{K_U C_U^3 l_U^3}{T^2} + \frac{K_R C_R^3 l_R^3}{(T-\tau_1)^2} \end{aligned} \tag{50}$$

where $\boldsymbol{e} = [e_{U,n}^{(1,o)}, e_{U,n}^{(1,c)}, e_{R,n}, e_{U,n}^{(2)}]$.

Meanwhile, (42) and (43) can be rewritten as

$$l_R \le \sum_{n=1}^{N} s_{R,n}^{(1,c)} r_{UR,n}\left(\frac{e_{U,n}^{(1,c)}}{s_{R,n}^{(1,c)}}\right) \tag{51}$$

and

$$\begin{aligned} l_A \le \min\Bigg\{ & \sum_{n=1}^{N} s_{A,n}^{(1,o)} r_{UR,n}\left(\frac{e_{U,n}^{(1,o)}}{s_{A,n}^{(1,o)}}\right) \\ & \sum_{n=1}^{N} s_{A,n}^{(1,o)} r_{UA,n}^{(1)}\left(\frac{e_{U,n}^{(1,o)}}{s_{A,n}^{(1,o)}}\right) + \sum_{n=1}^{N} s_{RA,n}^{(2)} r_{RA,n}\left(\frac{e_{R,n}}{s_{RA,n}^{(2)}}\right)\Bigg\} \\ & + \sum_{n=1}^{N} s_{UA,n}^{(2)} r_{UA,n}^{(2)}\left(\frac{e_{U,n}^{(2)}}{s_{UA,n}^{(2)}}\right) \end{aligned} \tag{52}$$

respectively, and (45)–(47) can be rewritten as

$$\sum_{n=1}^{N}\left[e_{U,n}^{(1,o)} + e_{U,n}^{(1,c)}\right] \le \tau_1 P_{U,\max} \tag{53}$$

$$\sum_{n=1}^{N} e_{U,n}^{(2)} \le \tau_2 P_{U,\max} \tag{54}$$

and

$$\sum_{n=1}^{N} e_{R,n} \le \tau_2 P_{R,\max} \tag{55}$$

respectively. Then, problem (49) can be equivalently rewritten as

$$\min_{\boldsymbol{s},\boldsymbol{e},\boldsymbol{\tau},\boldsymbol{l}} \; E'''_{\text{total}}(\boldsymbol{e}, \boldsymbol{\tau}, \boldsymbol{l}) \tag{56a}$$

$$\text{s.t.} \;\; (22), (23), \;(28), (30c), \;(30d), (40), (41), (44) \tag{56b}$$

$$(51)–(55), \text{ and} \tag{56c}$$

$$e_{U,n}^{(1,o)} \ge 0, e_{U,n}^{(1,c)} \ge 0, e_{R,n} \ge 0, e_{U,n}^{(2)} \ge 0 \;\forall n \in \mathbb{N}. \tag{56d}$$

It can be proved that problem (56) is convex and can be effectively solved by using the interior-point method [38].

Note that the derived subcarrier allocation vector ω by solving problem (56) does not necessarily satisfy the integer constraint given by (30f). Therefore, the achieved solution to problem (56) may not be feasible to the original problem (32), which causes that the solution achieved by solving problem (56) is not practical. Nevertheless, by using the solution to problem (56), a lower bound of energy consumption performance can be obtained for the studied MEC system.

C. Successive Convex Approximation-Based Iterative Optimization Algorithm

In this section, a practically feasible solution to the original problem (32) is derived, which is locally optimal and can achieve the performance similar to the lower bound achieved by the optimal solution obtained by solving problem (56). Recall that problem (39) is the simplified problem that is equivalent to problem (32). Therefore, the practically feasible solution is derived based on problem (39).

It can be observed that when the integer constraint on $\boldsymbol{\omega}$ described in (30f) holds, the corresponding constraint on $\boldsymbol{s}$ described in (44) can be expressed as

$$s_{A,n}^{(1,o)}, s_{R,n}^{(1,c)} \in \{0, \tau_1\}, s_{RA,n}^{(2)}, s_{UA,n}^{(2)} \in \{0, \tau_2\} \quad \forall n \in \mathbb{N}. \tag{57}$$

Recall that (44) is one of the constraints in (56b). Thus, problem (56) can be equivalently rewritten as

$$\begin{aligned} \min_{\boldsymbol{s},\boldsymbol{e},\boldsymbol{\tau},\boldsymbol{l}} \quad & E'''_{\text{total}}(\boldsymbol{e}, \boldsymbol{\tau}, \boldsymbol{l}) && (58a)\\ \text{s.t.} \quad & (22), (23), (28), (30c), (30d), (40), (41) && (58b)\\ & (51)-(55), (56d), \text{ and } (57). && (58c) \end{aligned}$$

Due to the discrete constraint (57) in (58c), problem (58) cannot be solved directly. Thus, to deal with this discrete constraint, a theorem is provided as follows.

Theorem 1: Define

$$\begin{aligned} X := \Big\{ \boldsymbol{s} : \; & s_{A,n}^{(1,o)}, \; s_{R,n}^{(1,c)} \in [0, \tau_1] \\ & s_{RA,n}^{(2)}, s_{UA,n}^{(2)} \in [0, \tau_2] \quad \forall n \in \mathbb{N} \Big\} \end{aligned} \tag{59}$$

and

$$Y := \left\{ \boldsymbol{s} : \sum_{n=1}^{N} (\theta_n \tau_1 + \vartheta_n \tau_2) - \sum_{n=1}^{N} \chi_n \le 0 \right\} \tag{60}$$

with $\theta_n = s_{A,n}^{(1,o)} + s_{R,n}^{(1,c)}$, $\vartheta_n = s_{RA,n}^{(2)} + s_{UA,n}^{(2)}$ and $\chi_n = (s_{A,n}^{(1,o)})^2 + (s_{R,n}^{(1,c)})^2 + (s_{RA,n}^{(2)})^2 + (s_{UA,n}^{(2)})^2$, and define a set based on (57) as

$$\begin{aligned} W := \Big\{ \boldsymbol{s} : \; & s_{A,n}^{(1,o)}, \; s_{R,n}^{(1,c)} \in \{0, \tau_1\} \\ & s_{RA,n}^{(2)}, s_{UA,n}^{(2)} \in \{0, \tau_2\} \; \forall n \in \mathbb{N} \Big\}. \end{aligned} \tag{61}$$

Then, it follows that $W = X \bigcap Y$.

Proof: See Appendix B. ■

According to Theorem 1, problem (58) can be equivalently transformed into the nonlinear programming problem as follows:

$$\begin{aligned} \min_{\boldsymbol{s},\boldsymbol{e},\boldsymbol{\tau},\boldsymbol{l}} \quad & E'''_{\text{total}}(\boldsymbol{e}, \boldsymbol{\tau}, \boldsymbol{l}) && (62a)\\ \text{s.t.} \quad & (22), (23), (28), (30c), (30d), (40), (41) && (62b)\\ & (51)-(55), (56d), \text{ and } && (62c)\\ & s_{A,n}^{(1,o)}, s_{R,n}^{(1,c)} \in [0, \tau_1], s_{RA,n}^{(2)}, s_{UA,n}^{(2)} \in [0, \tau_2] \; \forall n \in \mathbb{N} && (62d)\\ & \sum_{n=1}^{N} (\theta_n \tau_1 + \vartheta_n \tau_2) - \sum_{n=1}^{N} \chi_n \le 0. && (62e) \end{aligned}$$

In problem (62), the constraint (62e) is nonconvex. Thus, problem (62) is still a nonconvex optimization problem that cannot be solved directly. To tackle this problem, the constraint (62e) is rewritten as

$$H(\boldsymbol{s}, \boldsymbol{\tau}) - J(\boldsymbol{s}, \boldsymbol{\tau}) - M(\boldsymbol{s}) \le 0 \tag{63}$$

where $H(\boldsymbol{s}, \boldsymbol{\tau}) = (1/2) \sum_{n=1}^{N} [(s_{A,n}^{(1,o)} + \tau_1)^2 + (s_{R,n}^{(1,c)} + \tau_1)^2 + (s_{RA,n}^{(2)} + \tau_2)^2 + (s_{UA,n}^{(2)} + \tau_2)^2]$, $J(\boldsymbol{s}, \boldsymbol{\tau}) = (1/2) \sum_{n=1}^{N} \chi_n + N(\tau_1^2 + \tau_2^2)$ and $M(\boldsymbol{s}) = \sum_{n=1}^{N} \chi_n$. Because $H(\boldsymbol{s}, \boldsymbol{\tau})$, $J(\boldsymbol{s}, \boldsymbol{\tau})$, and $M(\boldsymbol{s})$ are convex functions, (63) is a d.c. constraint [35]. Moreover, it can be observed that the objective function and all the constraints except (62e) in problem (62) are convex. Thus, problem (62) can be expressed as a d.c. programming problem [39].

It has been known that SCA methods can effectively achieve a close-to-optimal local optimum of d.c. programming problems with polynomial-time computational complexity [40], [41]. Therefore, problem (62) is proposed to solve by SCA methods in the follows. To solve problem (62) via SCA, the left-hand side of (63) is introduced as a penalty term into the objective function in (62a) and, thus, problem (62) can be rewritten as

$$\begin{aligned} \min_{\boldsymbol{s},\boldsymbol{e},\boldsymbol{\tau},\boldsymbol{l}} \quad & E'''_{\text{total}}(\boldsymbol{e}, \boldsymbol{\tau}, \boldsymbol{l}) + \mu[H(\boldsymbol{s}, \boldsymbol{\tau}) - J(\boldsymbol{s}, \boldsymbol{\tau}) - M(\boldsymbol{s})] && (64a)\\ \text{s.t.} \quad & (62b)-(62d) && (64b) \end{aligned}$$

where $\mu > 0$ is the penalty factor. It has been shown in [40] and [41] that there is $\mu_0 > 0$ so that problem (64) is equivalent to problem (62) when $\mu > \mu_0$.

Problem (64) can be solved by SCA. To achieve this, the first-order Taylor approximations of $J(\boldsymbol{s}, \boldsymbol{\tau})$ and $M(\boldsymbol{s})$ around given $\boldsymbol{s}^{(k)}$ and $\boldsymbol{\tau}^{(k)}$ at the kth iteration are expressed as

$$\begin{aligned} \widehat{J}\Big(\boldsymbol{s}^{(k)}, \boldsymbol{\tau}^{(k)}, \boldsymbol{s}, \boldsymbol{\tau}\Big) = \; & J\Big(\boldsymbol{s}^{(k)}, \boldsymbol{\tau}^{(k)}\Big) + \nabla_{\boldsymbol{s}} J\Big(\boldsymbol{s}^{(k)}, \boldsymbol{\tau}^{(k)}\Big)^{\mathrm{T}} \Big(\boldsymbol{s} - \boldsymbol{s}^{(k)}\Big) \\ & + \nabla_{\boldsymbol{\tau}} J\Big(\boldsymbol{s}^{(k)}, \boldsymbol{\tau}^{(k)}\Big)^{\mathrm{T}} \Big(\boldsymbol{\tau} - \boldsymbol{\tau}^{(k)}\Big) \end{aligned} \tag{65}$$

$$\widehat{M}\Big(\boldsymbol{s}^{(k)}, \boldsymbol{s}\Big) = M\Big(\boldsymbol{s}^{(k)}\Big) + \nabla_{\boldsymbol{s}} M\Big(\boldsymbol{s}^{(k)}\Big)^{\mathrm{T}} \Big(\boldsymbol{s} - \boldsymbol{s}^{(k)}\Big) \tag{66}$$

where $\nabla_{\boldsymbol{s}}$ represents the gradient with respect to $\boldsymbol{s}$, $\nabla_{\boldsymbol{\tau}}$ represents the gradient with respect to $\boldsymbol{\tau}$, and A^{T} represents the transpose operation of A. Then, the convex optimization problem for the kth iteration of the sequential convex programming for problem (64) can be obtained as

$$\begin{aligned} \min_{\boldsymbol{s},\boldsymbol{e},\boldsymbol{\tau},\boldsymbol{l}} \quad & E'''_{\text{total}}(\boldsymbol{e}, \boldsymbol{\tau}, \boldsymbol{l}) + \mu\big[H(\boldsymbol{s}, \boldsymbol{\tau}) - \widehat{J}(\boldsymbol{s}, \boldsymbol{\tau}) - \widehat{M}(\boldsymbol{s})\big] \\ \text{s.t.} \quad & (64b). \end{aligned} \tag{67}$$

Because problem (67) is convex, it can be effectively solved by using the interior-point method to obtain $\boldsymbol{s}^{(k+1)} = \boldsymbol{s}^*$ and

Algorithm 1 Algorithm for Solving Problem (64) Based on SCA

```
Initialize: Set $k = 0$, $s^{(0)}$, $\tau^{(0)}$ and tolerance $\varepsilon > 0$;
Repeat:
    Solve problem (67) with $s^{(k)}$ and $\tau^{(k)}$ to obtain $(s^*, e^*, \tau^*, l^*)$;
    Let $s^{(k+1)} = s^*$ and $\tau^{(k+1)} = \tau^*$;
    Set $k = k + 1$;
Until $\|\Gamma^{(k+1)} - \Gamma^{(k)}\| < \varepsilon$.
```

$\boldsymbol{\tau}^{(k+1)} = \boldsymbol{\tau}^*$. In summary, the SCA-based iterative algorithm to solve problem (64) is illustrated in Algorithm 1, where $\Gamma \triangleq [\boldsymbol{s}, \boldsymbol{\tau}]$ and the initial $s^{(0)}$ and $\boldsymbol{\tau}^{(0)}$ are obtained by solving the convex optimization problem (67) with $\mu = 0$. Note that according to the theory of d.c. programming [39], Algorithm 1 converges as the iteration k increases.

Although Algorithm 1 can achieve a local optimal solution $(\boldsymbol{s}^*, \boldsymbol{e}^*, \boldsymbol{\tau}^*, \boldsymbol{l}^*)$ to problem (64), the subcarrier allocation indicator vector $\boldsymbol{\omega}^*$ derived from $(\boldsymbol{s}^*, \boldsymbol{e}^*, \boldsymbol{\tau}^*, \boldsymbol{l}^*)$ may be still not integral but only near integral. In order to achieve the feasible solution to the original problem (32), the derived $\boldsymbol{\omega}^*$ needs to be rounded to integral values, which is denoted as $\bar{\boldsymbol{\omega}} = \lfloor \boldsymbol{\omega}^* + (1/2) \rfloor$. Then, substituting $\boldsymbol{\omega} = \bar{\boldsymbol{\omega}}$ into problem (56), the objective function in (56a) is rewritten as

$$\tilde{E}'''_{\text{total}}(\boldsymbol{e}, \boldsymbol{\tau}, \boldsymbol{l}) = \sum_{n=1}^{N} \bar{\omega}_{A,n}^{(1,o)} e_{U,n}^{(1,\text{o})} + \sum_{n=1}^{N} \bar{\omega}_{RA,n}^{(2)} e_{R,n} + \sum_{n=1}^{N} \bar{\omega}_{UA,n}^{(2)} e_{U,n}^{(2)} + \sum_{n=1}^{N} \bar{\omega}_{R,n}^{(1,c)} e_{U,n}^{(1,\text{c})} + \frac{K_U C_U^3 l_U^3}{T^2} + \frac{K_R C_R^3 l_R^3}{(T - \tau_1)^2}. \tag{68}$$

Meanwhile, in constraint (56c), (51), and (52) are rewritten as

$$l_R \le \sum_{n=1}^{N} \tau_1 \bar{\omega}_{R,n}^{(1,c)} r_{UR,n} \left(\frac{e_{U,n}^{(1,c)}}{\tau_1} \right) \tag{69}$$

and

$$l_A \le \min \left\{ \sum_{n=1}^{N} \tau_1 \bar{\omega}_{A,n}^{(1,o)} r_{UR,n} \left(\frac{e_{U,n}^{(1,\text{o})}}{\tau_1} \right), \sum_{n=1}^{N} \tau_1 \bar{\omega}_{A,n}^{(1,o)} \times r_{UA,n}^{(1)} \left(\frac{e_{U,n}^{(1,\text{o})}}{\tau_1} \right) + \sum_{n=1}^{N} \tau_2 \bar{\omega}_{RA,n}^{(2)} r_{RA,n} \left(\frac{e_{R,n}}{\tau_2} \right) \right\} + \sum_{n=1}^{N} \tau_2 \bar{\omega}_{UA,n}^{(2)} r_{UA,n}^{(2)} \left(\frac{e_{U,n}^{(2)}}{\tau_2} \right) \tag{70}$$

respectively, and (53)–(55) are rewritten as

$$\sum_{n=1}^{N} \left[\bar{\omega}_{A,n}^{(1,o)} e_{U,n}^{(1,\text{o})} + \bar{\omega}_{R,n}^{(1,c)} e_{U,n}^{(1,c)} \right] \le \tau_1 P_{U,\max} \tag{71}$$

$$\sum_{n=1}^{N} \bar{\omega}_{UA,n}^{(2)} e_{U,n}^{(2)} \le \tau_2 P_{U,\max} \tag{72}$$

and

$$\sum_{n=1}^{N} \bar{\omega}_{RA,n}^{(2)} e_{R,n} \le \tau_2 P_{R,\max} \tag{73}$$

respectively. As a result, problem (56) can be rewritten as

$$\min_{\boldsymbol{e}, \boldsymbol{\tau}, \boldsymbol{l}} \; \tilde{E}'''_{\text{total}}(\boldsymbol{e}, \boldsymbol{\tau}, \boldsymbol{l}) \quad \text{s.t} \; (22), (23), \; (28), (30\text{c}), (30\text{d}), (69){-}(73). \tag{74}$$

As problem (74) is convex, the optimal solution $(\bar{\boldsymbol{e}}, \bar{\boldsymbol{\tau}}, \bar{\boldsymbol{l}})$ can be obtained, where $(\bar{\boldsymbol{\tau}}, \bar{\boldsymbol{l}})$ is the final solution to the original problem (32), and the final solution $\bar{\mathbf{p}}$ to problem (32) can be calculated by

$$\bar{p}_{U,n}^{(1)} = \begin{cases} \frac{\bar{e}_{U,n}^{(1,\text{o})}}{\bar{\tau}_1}, & \text{if } \bar{\omega}_{A,n}^{(1,o)} = 1 \\ \frac{\bar{e}_{U,n}^{(1,c)}}{\bar{\tau}_1}, & \text{if } \bar{\omega}_{A,n}^{(1,o)} = 0 \end{cases} \tag{75}$$

$\bar{p}_{R,n} = (\bar{e}_{R,n}/\bar{\tau}_2)$ and $\bar{p}_{U,n}^{(2)} = (\bar{e}_{U,n}^{(2)}/\bar{\tau}_2)$.

Complexity Analysis: The computational complexity of Algorithm 1 is mainly determined by solving problem (67). In the convex optimization theory, when solving a nonlinear convex optimization problem with m optimization variables by the interior-point method, its computational complexity is $\mathcal{O}(m^{3.5} \log(1/\xi))$, where ξ is a given solution accuracy. The number of optimization variables in problem (64) is $8N + 5$, so the computational complexity of Algorithm 1 is $\mathcal{O}(X(8N + 5)^{3.5} \log(1/\xi))$, where X is the number of iterations of Algorithm 1.

IV. Simulation Results

This section verifies the performance of the proposed joint task offloading and computation scheme for the cooperative multicarrier relaying-based MEC system by computer simulations. In the simulations, the number of subcarriers is set as $N = 16$. The channel coefficient of each subcarrier follows a complex Gaussian distribution with zero-mean value and unit variance. The distance of the UA link is set as $d_{UA} = 30$ m, the distance of the UR link is denoted as d_{UR}, and the distance of the RA link is denoted as d_{RA}. Given the reference distance of 1 m, the average signal power attenuation is estimated as -30 dB. Accordingly, the path loss of any two nodes is modeled as $10^{-3} d_{pq}^{-\alpha_{pq}}$, where d_{pq} is the distance between nodes p and q, and α_{pq} is the path-loss exponent of the link between nodes p and q. It is assumed that $\alpha_{UR} = 2$ and $\alpha_{UA} = \alpha_{RA} = 3$. Such a setting corresponds to the scenario, where the UR channel is free space and there are obstacles in the UA and RA channels. The total bandwidth of the system is $B_N = 1$ MHz. The variances of AWGN are set as $\sigma_R^2 = \sigma_A^2 = 10^{-10}$ W.

Furthermore, the computation parameters of the three nodes in the considered MEC system are set as $C_U = C_R = C_A = 10^3$ cycle/bit, $f_{U,\max} = 0.5$ GHz, $f_{A,\max} = 5$ GHz, $K_U = 10^{-27}$, and $K_R = 3 \times 10^{-28}$ [34]. The maximum transmit power constraints at the user terminal and the relay are set as $P_{U,\max} = 0.2$ W and $P_{R,\max} = 1$ W, respectively. The penalty factor μ is set as 4×10^{-4}.

In the following simulations, our proposed joint task offloading and computing scheme is compared with three partial offloading baseline schemes, namely, only direct-link to

AP scheme, relay cooperative computing scheme, and relay cooperative communication scheme, and three binary offloading baseline schemes, namely, only user computation scheme, only relay computing scheme, and only AP computing scheme. These baseline schemes are described in detail as follows.

1) *Only Direct-Link to AP Scheme:* In this scheme, the user terminal and AP jointly execute the computation tasks. The user terminal only offloads the tasks to the AP via the direct link, and the relay does not involve in task offloading and task computing. This corresponds to solving problem (30) by setting $\omega_{R,n}^{(1,c)} = \omega_{RA,n}^{(2)} = \omega_{UA,n}^{(2)} = 0$ and $l_R = 0$, and modifying constraint (14) as $l_A \le \sum_{n=1}^{N} \tau_1 \omega_{A,n}^{(1,o)} r_{UA,n}^{(1)}(p_{U,n}^{(1)})$.
2) *Relay Cooperative Computing Scheme:* In this scheme, the user terminal, relay, and AP jointly execute the computation tasks, but the user terminal only offloads the tasks to the AP via the direct link, and the relay does not assist forwarding the computation task data from the user terminal to AP. This corresponds to solving problem (30) by setting $\omega_{RA,n}^{(2)} = 0$ and modifying constraint (14) as $l_A \le \sum_{n=1}^{N} \tau_1 \omega_{A,n}^{(1,o)} r_{UA,n}^{(1)}(p_{U,n}^{(1)}) + \sum_{n=1}^{N} \tau_2 \omega_{UA,n}^{(2)} r_{UA,n}^{(2)}(p_{U,n}^{(2)})$.
3) *Relay Cooperative Communication Scheme:* In this scheme, the relay does not assist task computing for the user terminal and only assists in forwarding the task data from the user terminal to AP. Therefore, in this scheme, the user terminal and AP jointly execute the computation task. This corresponds to solving problem (30) by setting $\omega_{R,n}^{(1,c)} = 0$ and $l_R = 0$.
4) *Only User Computing Scheme:* In this scheme, the user terminal executes the task computation by itself. The energy consumption can be calculated by (36) with $l_U = L$.
5) *Only Relay Computing Scheme:* In this scheme, the user terminal offloads all computation task data to the relay for remote execution. This corresponds to solving problem (30) by setting $\omega_{A,n}^{(1,o)} = \omega_{UA,n}^{(2)} = \omega_{RA,n}^{(2)} = 0$ and $l_R = L$.
6) *Only AP Computing Scheme:* In this scheme, the user terminal offloads all computation task data to AP with the assistance of the relay for remote execution. This corresponds to solving problem (30) by setting $\omega_{R,n}^{(1,c)} = 0$ and $l_A = L$.

Moreover, to evaluate the performance of our proposed scheme, the energy consumption lower bound achieved by solving problem (56) is also illustrated in the following involved simulation results.

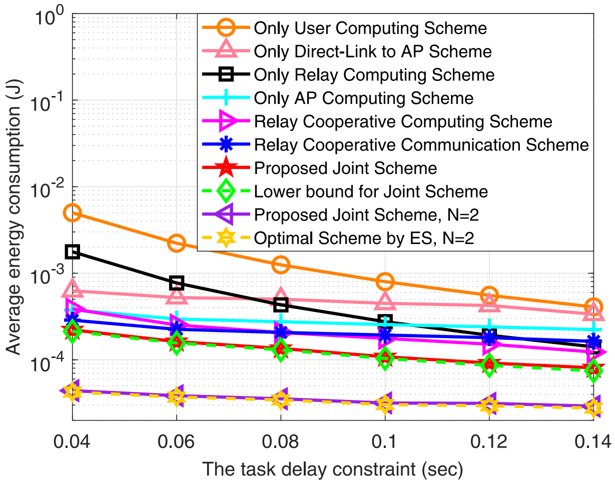


Fig. 3. Average energy consumption versus the task delay constraint; performance comparison of our proposed joint task offloading and computing scheme and the six baseline schemes with $L = 20$ kb, $d_{\text{UR}} = 5$ m, $d_{\text{RA}} = 25$ m, and $f_{\text{R,max}} = 1$ GHz.

A. Performance Comparison With Varying Task Execution Delay Constraint T

When the task delay constraint T varies, the energy consumption performance of our proposed joint task offloading and computing scheme is compared with those of the six baseline schemes in Fig. 3, where $L = 20$ b, $d_{UR} = 5$ m, $d_{RA} = 25$ m, and $f_{R,\max} = 1$ GHz. Moreover, to verify the optimality of our proposed scheme, the gap between the energy consumption achieved by our proposed scheme and that achieved by the global optimal solution to the original problem (30) is also illustrated, where the global optimal solution is achieved by the exhaustive search (ES) method and N is set as 2 to facilitate ES. From Fig. 3, it can be seen that MEC can significantly save energy consumption since the amount of energy consumed by the only user computing scheme is much larger than that consumed by the other schemes. Also, it can be found in Fig. 3 that the energy consumption decreases in all the illustrated schemes when the task delay constraint T increases. This is because when the task delay constraint T increases, the CPU frequency for executing the computation task and the transmit power for task offloading can be decreased, which results in the energy consumption for computation and task offloading can be reduced. More importantly, it can be observed in Fig. 3 that as compared with the six baseline schemes, the proposed joint scheme can remarkably reduce the energy consumption. For instance, as compared with the relay cooperative computing scheme and the relay cooperative communication scheme, the proposed joint scheme can save up to 50% of energy. Moreover, it can be found in Fig. 3 that the energy consumption achieved by the proposed joint scheme is much close to its lower bound. Furthermore, for $N = 2$, the amount of energy consumption achieved our proposed joint scheme is approximately equal to that achieved by the ES scheme. This verifies that Algorithm 1 can achieve superior energy consumption performance for the studied MEC system, although the solution achieved by Algorithm 1 is locally optimal.

Furthermore, Fig. 3 also reveals that the partial offloading schemes generally outperform the binary offloading schemes due to the flexibility of resource allocation in partial offloading. Nevertheless, Fig. 3 also shows that the only relay computing scheme can achieve better energy consumption performance than the relay cooperative communication scheme when T becomes large (i.e., $T = 0.14$ s). This result may be due to that the computation energy consumed at the relay can be reduced as T increases [see (21), (25)].

Moreover, the benefit from the employment of the relay can be found in Fig. 3 by comparing the only direct-link to AP scheme with the only AP computing scheme and comparing the relay cooperative communication scheme with our proposed joint scheme. First, it is observed from Fig. 3 that the only AP computing scheme outperforms the only direct-link to AP scheme. The reason is that by using the high-quality relay channel, the energy consumption for task offloading in the only AP computing scheme is less than that in the only AP computing scheme. Second, it is observed from Fig. 3 that our proposed joint scheme outperforms the relay cooperative communication scheme. This is because as the relay assists task computing, the amount of task data offloaded to AP can be reduced. As a result, the energy consumption for data offloading is reduced, which can improve the energy consumption performance of the MEC system.

Furthermore, Fig. 3 also reveals that the partial offloading schemes generally outperform the binary offloading schemes due to the flexibility of resource allocation in partial offloading. However, there are two exceptional cases. The first is that the only AP computing scheme outperforms the only direct-link to AP scheme, which verifies that the energy consumption performance of the MEC system can benefit from the relay's assistance in data offloading. The second is that the only relay computing scheme can achieve better energy consumption performance than the relay cooperative communication scheme when T becomes large (i.e., $T = 0.14$ s). This result may be due to that the computation energy consumed at the relay can be reduced as T increases [see (21), (25)].

B. Performance Comparison With Varying Task Data Size L

Fig. 4 illustrates the energy consumption performance of our proposed joint scheme and the six baseline schemes when the size of computation task data (i.e., L) varies. Here, it is assumed that $T = 0.04$ s, $d_{UR} = 5$ m, $d_{RA} = 25$ m, and $f_{R,\max} = 1$ GHz. As can be seen in Fig. 4, our proposed joint scheme can outperform the six baseline schemes in all regions of computation task data size. More specifically, as compared with the relay cooperative communication scheme, which performs the best among the six baseline schemes, our proposed joint scheme saves about 40% of energy when L is small (e.g., $10 \leq L \leq 14$) and saves about 30% of energy when L is large (e.g., $16 \leq L \leq 20$). This also demonstrates the benefit of enabling the relay to assist cooperative computation in the MEC system. Again, Fig. 4 verifies that the proposed joint scheme can achieve approximate optimal resource allocation since the energy consumption achieved by the proposed joint scheme is very close to its lower bound.

Another interesting finding in Fig. 4 is that as compared with the only user computing and only relay computing schemes, the amount of energy consumed in the other five schemes is much smaller when the computation task data size L is large. The reason is that the energy consumption can be reduced by utilizing the high computation capacity of the MEC server located at AP, especially when the computation task data size L is large and the task computation delay T is small.

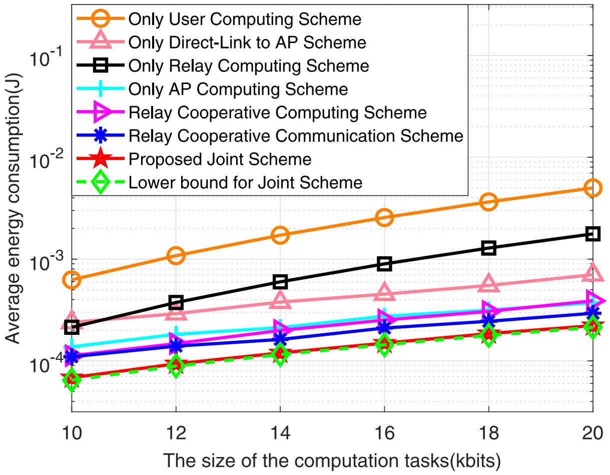


Fig. 4. Average energy consumption versus the size of computation task data; performance comparison of our proposed joint task offloading and computing scheme and the six baseline schemes with $T = 0.04$ s, $d_{\mathrm{UR}} = 5$ m, $d_{\mathrm{RA}} = 25$ m, and $f_{R,max} = 1$ GHz.

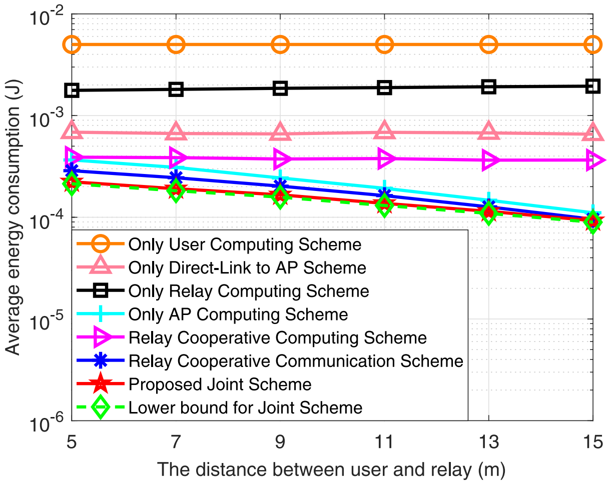


Fig. 5. Average energy consumption versus the distance between user and relay; performance comparison of our proposed joint task offloading and computing scheme and the six baseline schemes with $T = 0.04$ s, $L = 20$ kb, and $f_{R,\max} = 1$ GHz.

C. Performance Comparison With Varying d_{UR}

Fig. 5 plots the energy consumption performance of our proposed joint scheme and the six baseline schemes when the distance between relay and user terminal varies from 5 to 15 m, where $L = 20$ kb, $T = 0.04$ s, $d_{RA} = d_{UA} - d_{UR}$, and $f_{R,\max} = 1$ GHz. As shown in Fig. 5, our proposed joint scheme achieves the lowest energy consumption among the illustrated schemes wherever the relay locates. Also, Fig. 5 further confirms that the energy consumption achieved by our proposed joint scheme approaches its lower bound.

Another finding revealed by Fig. 5 is that when $d_{UR} = 15$ m (i.e., $d_{UR} = d_{RA}$), the only AP computing scheme and the relay cooperative communication scheme can achieve a similar energy consumption performance as that achieved by our

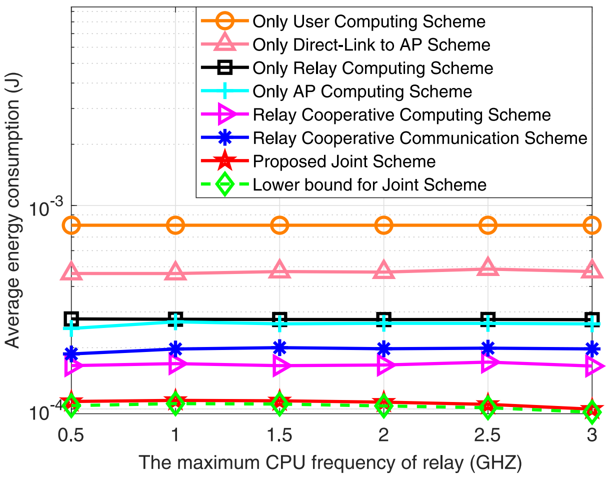


Fig. 6. Average energy consumption versus the maximum CPU frequency of relay; performance comparison of our proposed joint task offloading and computing scheme and the six baseline schemes with $T = 0.1$ s, $L = 20$ kb, $d_{\mathrm{UR}} = 5$ m, and $d_{\mathrm{RA}} = 25$ m.

proposed joint scheme. This result can be explained as follows. According to the cooperative communication theory, the end-to-end achievable data transmission rate from the user terminal to AP achieves its maximum when the relay locates at the midpoint of d_{UA}. As a result, the amount of energy consumed in task offloading from the user terminal to AP is minimized. Because the MEC server at AP has highest computation capacity, most of the computation task data are offloaded to AP and only a small portion of the computation tasks are executed at the relay and the user. Therefore, the proposed joint scheme performs as almost the same as the only AP computing scheme and the relay cooperative communication scheme when $d_{UR} = d_{RA}$. Nevertheless, it is noted that the proposed joint scheme outperforms these two schemes when the relay d_{UR} is small (e.g., $d_{UR} \leq 11$ m).

D. Performance Comparison With Varying $f_{R,\max}$

Fig. 6 plots the energy consumption performance of our proposed joint scheme and the six baseline schemes when the maximum CPU frequency of the relay (i.e., $f_{R,\max}$) varies from 0.5 to 3 GHz, where $T = 0.1$s, $L = 20$ kb, $d_{UR} = 5$ m, and $d_{RA} = 25$ m. From Fig. 6, it is observed that the energy consumption remains constant as $f_{R,\max}$ varies for all the illustrated schemes, which indicates that increasing $f_{R,\max}$ cannot save the energy consumption in the MEC system. The reason can be illustrated as follows. According to (35) in Lemma 1 and the proof of Lemma 1, the optimal relay's computing frequency is determined by l_R and τ_1. In other words, the optimal relay's computing frequency is independent of $f_{R,\max}$. Therefore, when the optimal solution to problem (30) is achieved, the relay's computing frequency constraint in (19) is not necessary to be active. As a result, it can be concluded that the energy consumption cannot be decreased by increasing $f_{R,\max}$. Note that all the schemes illustrated in Fig. 6 can be achieved by solving problem (30), thus the conclusion can apply to all the schemes in Fig. 6.

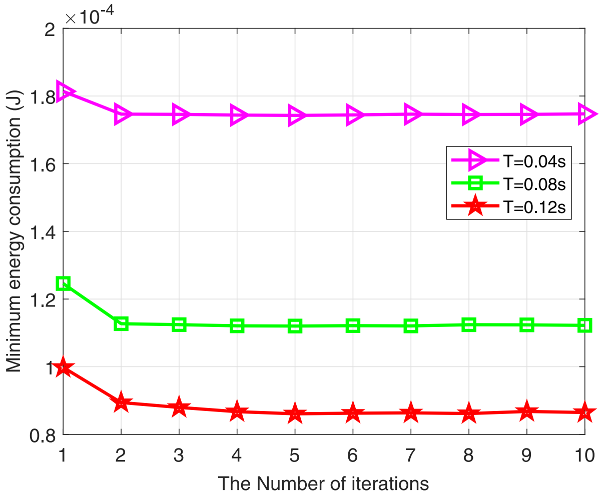


Fig. 7. Convergence of the proposed algorithm (Algorithm 1) for different T's; where $L = 20$ kb, $d_{\mathrm{UR}} = 5$ m, $d_{\mathrm{RA}} = 25$ m, and $f_{\mathrm{R,max}} = 1$ GHz.

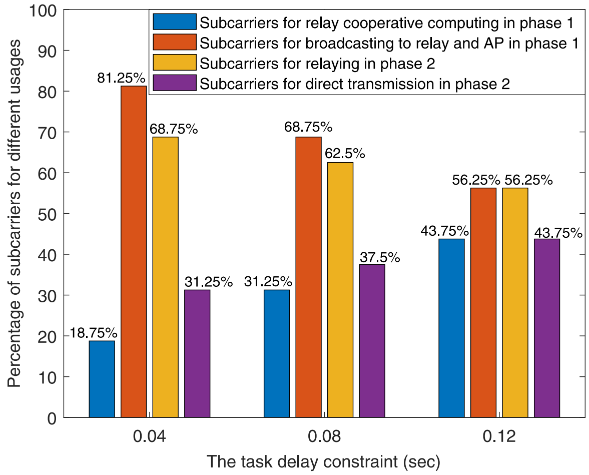


Fig. 8. Percentage of subcarriers for different usages for $T = 0.04$ s, $T = 0.08$ s, and $T = 0.12$ s, where $L = 20$ kb, $d_{\mathrm{UR}} = 5$ m, $d_{\mathrm{RA}} = 25$ m, and $f_{\mathrm{R,max}} = 1$ GHz.

E. Convergence Performance of Algorithm 1

Fig. 7 plots the convergence of the proposed algorithm (Algorithm 1) for $T = 0.04$ s, $T = 0.08$ s, and $T = 0.12$ s. As can be seen from Fig. 7, Algorithm 1 always converges to the optimal solution in less than 5 iterations. Moreover, it can be observed from Fig. 7 that the achieved energy consumption is equal to about 90% of its minimum value in less than 3 iterations. Recall that in Figs. 3–6, it has been verified that Algorithm 1 can achieve the energy consumption that is very close to its lower bound. Thus, it can be concluded that Algorithm 1 is very efficient although it achieves a locally optimal solution.

F. Analysis of Subcarrier Allocation for Different Offloading Usages

Fig. 8 plots the percentage of the subcarriers for different offloading usages (i.e., the subcarriers in the sets $\mathbb{N}_A^{(1,o)}$, $\mathbb{N}_{UR}^{(1,c)}$, $\mathbb{N}_{RA}^{(2)}$, and $\mathbb{N}_{UA}^{(2)}$) for $T = 0.04$ s, $T = 0.08$ s, and $T = 0.12$ s, respectively, where $L = 20$ kb, $d_{UR} = 5$ m, $d_{RA} = 25$ m, and

$f_{R,\max}$. As can be seen from Fig. 8, as T increases, the number of subcarriers used for relay cooperative computing in phase 1 (i.e., the subcarriers in $\mathbb{N}_{UR}^{(1,c)}$) and the number of subcarriers used for direct transmission in phase 2 (i.e., the subcarriers in $\mathbb{N}_{UA}^{(2)}$) increase, while the number of subcarriers used for broadcasting to relay and AP in phase 1 (i.e., the subcarriers in $\mathbb{N}_{A}^{(1,o)}$) and the number of subcarriers used for relaying in phase 2 (i.e., the subcarriers in $\mathbb{N}_{RA}^{(2)}$) decrease. This indicates that in this case, the user terminal is apt to offload tasks to the relay for cooperative computing when T increases, which can also be verified by the results in Fig. 3.

V. Conclusion

In this work, we studied a joint task offloading and computation design for a cooperative multicarrier relaying-based MEC system consisting of a user terminal, a relay with computation capability, and an AP equipped with an MEC server. In order to reduce the task offloading and computing energy consumed at the user terminal and the relay, we proposed a new cooperative MEC protocol that integrates the rateless coding technique to fully utilize the multicarrier subchannels for task offloading, and then formulated a problem to jointly optimize CPU frequencies at the user terminal and the relay, subcarrier allocation, power allocation, task partition, and offloading time and computation time allocation for the MEC system. By employing convex optimization and SCA techniques to solve the formulated problem, we proposed an efficient algorithm to achieve superior energy consumption performance in the MEC system.

It is worth pointing out that many emerging techniques on MEC have not been involved in this article, such as how to employ an intelligent reflecting surface [42]–[44] to enhance the performance MEC systems, how to deploy MEC in securing unmanned aerial vehicle communication systems [45]–[48], etc. The combination of our cooperative multicarrier relaying-based MEC system with these emerging techniques is an interesting topic, which will be studied in our future work.

Appendix A
Proof of Lemma 1

In order to prove Lemma 1, (34) is first to be proved. Define an auxiliary variable $f_U \triangleq (\sum_{i=1}^{C_U l_U} f_{U,i}/C_U l_U)$. When $x > 0$, $g_1(x) = (1/x)$ and $g_2(x) = x^2$ are both convex functions. Thus, according to Jensen's inequality, we have

$$C_U l_U / f_U \leq \sum_{i=1}^{C_U l_U} \frac{1}{f_{U,i}} \tag{76}$$

$$C_U l_U f_U^2 \leq \sum_{i=1}^{C_U l_U} f_{U,i}^2. \tag{77}$$

For (76) and (77), both the equalities hold if and only if $f_{U,i} = f_U \ \forall i \in \{1, \ldots, C_R l_R\}$. To minimize the objective function value of problem (30), the equality in (77) should be satisfied. Thus, it follows that $f_{U,i} = f_U \ \forall i \in \{1, \ldots, C_R l_R\}$. Then, the objective function of problem (30) can be rewritten as

$$\begin{aligned} E_{\text{total}}(\boldsymbol{\omega}, \boldsymbol{p}, \boldsymbol{\tau}, \boldsymbol{f}) = &\sum_{n=1}^{N} \tau_1 \omega_{A,n}^{(1,o)} p_{U,n}^{(1)} + \sum_{n=1}^{N} \tau_2 \omega_{RA,n}^{(2)} p_{R,n} \\ &+ \sum_{n=1}^{N} \tau_2 \omega_{UA,n}^{(2)} p_{U,n}^{(2)} + \sum_{n=1}^{N} \tau_1 \omega_{R,n}^{(1,c)} p_{U,n}^{(1)} \\ &+ C_U l_U K_U f_U^2 + \sum_{i=1}^{C_R l_R} K_R f_{R,i}^2. \end{aligned} \tag{78}$$

Furthermore, according to the constraint in (20), it is obtained that $f_U \geq C_U l_U / T$. To minimize the value of $E_{\text{total}}(\boldsymbol{\omega}, \boldsymbol{p}, \boldsymbol{\tau}, \boldsymbol{f})$ in (A03), there should be $f_U = C_U l_U / T$. Thus, it follows that (34) holds.

It can be verified that (35) can be proved via a similar approach for proving (34). Thus, the proof of Lemma 1 is completed.

Appendix B
Proof of Theorem 1

To prove Theorem 1, define a set

$$\begin{aligned} Z := \Bigg\{ \mathbf{s} : &\sum_{n=1}^{N} \Big[\Big(s_{A,n}^{(1,o)} + s_{R,n}^{(1,c)} \Big) \tau_1 + \Big(s_{RA,n}^{(2)} + s_{UA,n}^{(2)} \Big) \tau_2 \Big] \\ &- \sum_{n=1}^{N} \Big[\Big(s_{A,n}^{(1,o)} \Big)^2 + \Big(s_{R,n}^{(1,c)} \Big)^2 + \Big(s_{RA,n}^{(2)} \Big)^2 \\ &+ \Big(s_{UA,n}^{(2)} \Big)^2 \Big] > 0 \Bigg\}. \end{aligned} \tag{79}$$

Then, $X \cap Y = X \backslash Z$, where $X \backslash Z$ denotes the difference between X and Z. Therefore, if and only if $W = X \backslash Z$, $W = X \cap Y$. To prove $W = X \backslash Z$, it is observed that $W \subset X \backslash Z$. On the other hand, it can be also observed that if and only if $\mathbf{s} \in W$, the following equations hold:

$$\begin{cases} s_{A,n}^{(1,o)} \tau_1 - \Big(s_{A,n}^{(1,o)} \Big)^2 = 0 \\ s_{R,n}^{(1,c)} \tau_1 - \Big(s_{R,n}^{(1,c)} \Big)^2 = 0 \\ s_{RA,n}^{(2)} \tau_2 - \Big(s_{RA,n}^{(2)} \Big)^2 = 0 \\ s_{UA,n}^{(2)} \tau_2 - \Big(s_{UA,n}^{(2)} \Big)^2 = 0. \end{cases} \tag{80}$$

Meanwhile, if $\mathbf{s} \in X$, it follows that:

$$\begin{cases} s_{A,n}^{(1,o)} \tau_1 - \Big(s_{A,n}^{(1,o)} \Big)^2 \geq 0 \\ s_{R,n}^{(1,c)} \tau_1 - \Big(s_{R,n}^{(1,c)} \Big)^2 \geq 0 \\ s_{RA,n}^{(2)} \tau_2 - \Big(s_{RA,n}^{(2)} \Big)^2 \geq 0 \\ s_{UA,n}^{(2)} \tau_2 - \Big(s_{UA,n}^{(2)} \Big)^2 \geq 0 \end{cases} \tag{81}$$

which results in that

$$\begin{aligned} &\Big[\Big(s_{A,n}^{(1,o)} + s_{R,n}^{(1,c)} \Big) \tau_1 + \Big(s_{RA,n}^{(2)} + s_{UA,n}^{(2)} \Big) \tau_2 \Big] \\ &- \Big[\Big(s_{A,n}^{(1,o)} \Big)^2 + \Big(s_{R,n}^{(1,c)} \Big)^2 + \Big(s_{RA,n}^{(2)} \Big)^2 + \Big(s_{UA,n}^{(2)} \Big)^2 \Big] \geq 0. \end{aligned} \tag{82}$$

Moreover, if $\mathbf{s} \notin Z$, it is obtained that

$$\sum_{n=1}^{N}\left[\left(s_{A,n}^{(1,o)}+s_{R,n}^{(1,c)}\right)\tau_1+\left(s_{RA,n}^{(2)}+s_{UA,n}^{(2)}\right)\tau_2\right] -\sum_{n=1}^{N}\left[\left(s_{A,n}^{(1,o)}\right)^2+\left(s_{R,n}^{(1,c)}\right)^2+\left(s_{RA,n}^{(2)}\right)^2+\left(s_{UA,n}^{(2)}\right)^2\right]\leq 0. \tag{83}$$

Therefore, if $\mathbf{s} \in X\backslash Z$, it follows that:

$$\left[s_{A,n}^{(1,o)}\tau_1-\left(s_{A,n}^{(1,o)}\right)^2\right]+\left[s_{R,n}^{(1,c)}\tau_1-\left(s_{R,n}^{(1,c)}\right)^2\right] +\left[s_{RA,n}^{(2)}\tau_2-\left(s_{RA,n}^{(2)}\right)^2\right]+\left[s_{UA,n}^{(2)}\tau_2-\left(s_{UA,n}^{(2)}\right)^2\right]=0 \tag{84}$$

and thus (80) can be obtained since $\mathbf{s} \in X$. Then, it follows that $W \supset X\backslash Z$. As a result, it is obtained that $W = X\backslash Z$. The proof follows.

References

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

[2] C. Wang, Y. He, F. R. Yu, Q. Chen, and L. Tang, "Integration of networking, caching, and computing in wireless systems: A survey, some research issues, and challenges," *IEEE Commun. Surveys Tuts.*, vol. 20, no. 1, pp. 7–38, 1st Quart., 2017.

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

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

[5] J. Xu and J. Yao, "Exploiting physical-layer security for multiuser multicarrier computation offloading," *IEEE Wireless Commun. Lett.*, vol. 8, no. 1, pp. 9–12, Feb. 2019.

[6] M. Li, S. Yang, Z. Zhang, J. Ren, and G. Yu, "Joint subcarrier and power allocation for OFDMA based mobile edge computing system," in *Proc. IEEE 28th Annu. Int. Symp. Pers. Indoor Mobile Radio Commun.*, Montreal, QC, Canada, 2017, pp. 1–6.

[7] S. Wang, C. Pan, and C. Yin, "Joint heterogeneous tasks offloading and resource allocation in mobile edge computing systems," in *Proc. 10th Int. Conf. Wireless Commun. Signal Process.*, Hangzhou, China, Oct. 2018, pp. 1–6.

[8] K. Cheng, Y. Teng, W. Sun, A. Liu, and X. Wang, "Energy-efficient joint offloading and wireless resource allocation strategy in multi-MEC server systems," in *Proc. IEEE Int. Conf. Commun. (ICC)*, Kansas City, MO, USA, May 2018, pp. 1–6.

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

[10] X. Hu, K.-K. Wong, and K. Yang, "Wireless powered cooperation-assisted mobile edge computing," *IEEE Trans. Wireless Commun.*, vol. 17, no. 4, pp. 2375–2388, Apr. 2018.

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

[12] 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.

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

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

[15] M. Liu, F. R. Yu, Y. Teng, V. C. M. Leung, and M. Song, "Computation offloading and content caching in wireless blockchain networks with mobile edge computing," *IEEE Trans. Veh. Technol.*, vol. 67, no. 11, pp. 11008–11021, Nov. 2018.

[16] J. Feng, Q. Pei, F. R. Yu, X. Chu, and B. Shang, "Computation offloading and resource allocation for wireless powered mobile edge computing with latency constraint," *IEEE Wireless Commun. Lett.*, vol. 8, no. 5, pp. 1320–1323, Oct. 2019.

[17] L. Le and E. Hossain, "Multihop cellular networks: Potential gains, research challenges, and a resource allocation framework," *IEEE Commun. Mag.*, vol. 45, no. 9, pp. 66–73, Sep. 2007.

[18] R. Pabst *et al.*, "Relay-based deployment concepts for wireless and mobile broadband radio," *IEEE Commun. Mag.*, vol. 42, no. 9, pp. 80–89, Sep. 2004.

[19] M. Salem *et al.*, "An overview of radio resource management in relay-enhanced OFDMA-based networks," *IEEE Commun. Surveys Tuts.*, vol. 12, no. 3, pp. 422–438, 3rd Quart., 2010.

[20] C.-N. Hsu, H.-J. Su, and P.-H. Lin, "Joint subcarrier pairing and power allocation for OFDM transmission with decode-and-forward relaying," *IEEE Trans. Signal Process.*, vol. 59, no. 1, pp. 399–414, Jan. 2011.

[21] G. Huang, Q. Zhang, and J. Qin, "Joint time switching and power allocation for multicarrier decode-and-forward relay networks with SWIPT," *IEEE Signal Process. Lett.*, vol. 22, no. 12, pp. 2284–2288, Dec. 2015.

[22] G. Huang and D. Tang, "Wireless information and power transfer in two-way OFDM amplify-and-forward relay networks," *IEEE Commun. Lett.*, vol. 20, no. 8, pp. 1563–1566, Aug. 2016.

[23] M. Luby, "LT codes," in *Proc. 43rd Annu. IEEE Symp. Found. Comput. Sci.*, Vancouver, BC, Canada, 2002, pp. 271–280.

[24] A. Shokrollahi, "Raptor codes," *IEEE Trans. Inf. Theory*, vol. 52, no. 6, pp. 2551–2567, Jun. 2006.

[25] J. Castura and Y. Mao, "Rateless coding for wireless relay channels," in *Proc. IEEE Int. Symp. Inf. Theory*, Adelaide, SA, Australia, Sep. 2005, pp. 810–814.

[26] J. Buhler and S. Stanczak, "Energy-efficient relaying using rateless codes," in *Proc. IEEE Int. Conf. Acoust. Speech Signal Process. (ICASSP)*, Vancouver, BC, Canada, 2013, pp. 4943–4947.

[27] R. Urgaonkar and M. J. Neely, "Optimal routing with mutual information accumulation in wireless networks," *IEEE J. Sel. Areas Commun.*, vol. 30, no. 9, pp. 1730–1737, Oct. 2012.

[28] K. Pang, Z. Lin, B. F. Uchoa-Filho, and B. Vucetic, "Distributed network coding for wireless sensor networks based on rateless LT codes," *IEEE Wireless Commun. Lett.*, vol. 1, no. 6, pp. 561–564, Dec. 2012.

[29] J. Yue, Z. Lin, B. Vucetic, G. Mao, and T. Aulin, "Performance analysis of distributed Raptor codes in wireless sensor networks," *IEEE Trans. Commun.*, vol. 61, no. 10, pp. 4357–4368, Oct. 2013.

[30] X. Di, K. Xiong, P. Fan, and H.-C. Yang, "Simultaneous wireless information and power transfer in cooperative relay networks with rateless codes," *IEEE Trans. Veh. Technol.*, vol. 66, no. 4, pp. 2981–2996, Apr. 2017.

[31] A. Ravanshid, L. Lampe, and J. B. Huber, "Dynamic decode-and-forward relaying using Raptor codes," *IEEE Trans. Wireless Commun.*, vol. 10, no. 5, pp. 1569–1581, May 2011.

[32] A. Molisch, N. Mehta, J. Yedidia, and J. Zhang, "Performance of fountain codes in collaborative relay networks," *IEEE Trans. Wireless Commun.*, vol. 6, no. 11, pp. 4108–4119, Nov. 2007.

[33] S. Zhang and V. K. N. Lau, "Resource allocation for OFDMA system with orthogonal relay using rateless code," *IEEE Trans. Wireless Commun.*, vol. 7, no. 11, pp. 4534–4540, Nov. 2008.

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

[35] H. Tuy, *Convex Analysis and Global Optimization*. Dordrecht, The Netherlands: Kluwer, 1998.

[36] T. D. Burd and R. W. Brodersen, "Processor design for portable systems," *Kluwer J. VLSI Signal Process. Syst. Signal Image Video Technol.*, vol. 13, no. 2, pp. 203–221, Aug. 1996.

[37] C. Y. Wong, R. S. Cheng, K. B. Letaief, and R. D. Murch, "Multiuser OFDM with adaptive subcarrier, bit, and power allocation," *IEEE J. Sel. Areas Commun.*, vol. 17, no. 10, pp. 1747–1758, Oct. 1999.

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

[39] A. J. Smola, S. V. N. Vishwanathan, and T. Hofmann, "Kernel methods for missing variables," in *Proc. 10th Int. Workshop Artif. Intell. Stat.*, Mar. 2005, pp. 325–332.

[40] U. Rashid, H. D. Tuan, H. H. Kha, and H. H. Nguyen, "Joint optimization of source precoding and relay beamforming in wireless MIMO relay networks," *IEEE Trans. Commun.*, vol. 12, no. 2, pp. 488–499, Feb. 2014.

[41] E. Che, H. D. Tuan, and H. H. Nguyen, "Joint optimization of cooperative beamforming and relay assignment in multi-user wireless relay networks," *IEEE Trans. Wireless Commun.*, vol. 13, no. 10, pp. 5481–5495, Oct. 2014.

[42] Q. Wu and R. Zhang, "Towards smart and reconfigurable environment: Intelligent reflecting surface aided wireless network," *IEEE Commun. Mag.*, vol. 58, no. 1, pp. 106–112, Jan. 2020.

[43] M. Cui, G. Zhang, and R. Zhang, "Secure wireless communication via intelligent reflecting surface," *IEEE Wireless Commun. Lett.*, vol. 8, no. 5, pp. 1410–1414, Oct. 2019.

[44] T. Bai, C. Pan, Y. Deng, M. Elkashlan, A. Nallanathan, and L. Hanzo, "Latency minimization for intelligent reflecting surface aided mobile edge computing," *IEEE J. Sel. Areas Commun.*, vol. 38, no. 11, pp. 2666–2682, Nov. 2020.

[45] Y. Xu, T. Zhang, D. Yang, Y. Liu, and M. Tao, "Joint resource and trajectory optimization for security in UAV-assisted MEC systems," *IEEE Trans. Commun.*, vol. 69, no. 1, pp. 573–588, Jan. 2021, doi: 10.1109/TCOMM.2020.3025910.

[46] Y. Zhou et al. "Secure communications for UAV-enabled mobile edge computing systems," *IEEE Trans. Commun.*, vol. 68, no. 1, pp. 376–388, Jan. 2020.

[47] T. Bai, J. Wang, Y. Ren, and L. Hanzo, "Energy-efficient computation offloading for secure UAV-edge-computing systems," *IEEE Trans. Veh. Technol.*, vol. 68, no. 6, pp. 6074–6087, Jun. 2019.

[48] G. Zhang, Q. Wu, M. Cui, and R. Zhang, "Securing UAV communications via joint trajectory and power control," *IEEE Trans. Wireless Commun.*, vol. 18, no. 2, pp. 1376–1389, Feb. 2019.

[49] Y. Iraqi and A. Al-Dweik, "Efficient information transmission using smart OFDM for IoT applications," *IEEE Internet Things J.*, vol. 7, no. 9, pp. 8397–8409, Sep. 2020.

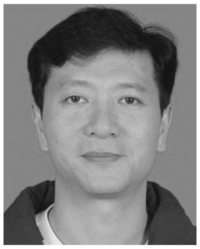
[50] X. Chen, Y. Cai, Q. Shi, M. Zhao, B. Champagne, and L. Hanzo, "Efficient resource allocation for relay-assisted computation offloading in mobile-edge computing," *IEEE Internet Things J.*, vol. 7, no. 3, pp. 2452–2468, Mar. 2020.

Dieli Hu received the B.E. degree in electrical engineering and its automation from China Three Gorges University, Yichang, China, in 2017. She is currently pursuing the M.E. degree in the Faculty of Electronics and Communication, Guangzhou University, Guangzhou, China.

Her current research interests include mobile edge computing, wireless cooperative communication, and vehicular networks.

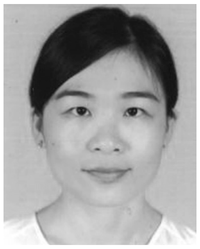
Gaofei Huang (Member, IEEE) received the M.S. and Ph.D. degrees from Sun Yat-sen University, Guangzhou, China, in 2004 and 2012, respectively.

He is currently an Associate Professor with Guangzhou University, Guangzhou. His current research interests include energy-harvesting wireless communications, ambient backscatter communications, mobile-edge computing, UAV-aided communications, intelligent reflecting surface-aided communications, the resource allocation in next-generation wireless communication systems, and cross-layer design of wireless communication systems.



Dong Tang (Member, IEEE) received the Ph.D. degree from Sun Yat-sen University, Guangzhou, China, in 2007.

He is currently a Professor with Guangzhou University, Guangzhou. His interests are in the area of energy harvesting in wireless communication networks, wireless MIMO, and OFDM communication networks.



Sai Zhao received the bachelor's and master's degrees in communication engineering from Central South University, Changsha, China, in 2006 and 2003, respectively, and the Ph.D. degree in communication and information system from Sun Yat-sen University, Guangzhou, China, in 2015.

She is currently a Lecturer with Guangzhou University, Guangzhou. Her current research interests include cooperative wireless communication, physical-layer security, and nonorthogonal multiple access.

Hui Zheng received the M.S. and Ph.D. degrees from the South China University of Technology, Guangzhou, China, in 1996 and 2000, respectively.

He is currently an Associate Professor with Guangzhou University, Guangzhou. His current research interests include WSN protocol stack and FPGA-SoC architecture design.