IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received September 27, 2020, accepted October 12, 2020, date of publication October 19, 2020, date of current version November 2, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.3032033

Resource Allocation Strategy for D2D-Assisted Edge Computing System With Hybrid Energy Harvesting

JIAFA CHEN, YISHENG ZHAO, (Member, IEEE), ZHIMENG XU, AND HAIFENG ZHENG, (Member, IEEE)

Fujian Key Laboratory for Intelligent Processing and Wireless Transmission of Media Information, College of Physics and Information Engineering, Fuzhou University, Fuzhou 350108, China

Corresponding author: Yisheng Zhao (zhaoys@bupt.edu.cn)

This work was supported in part by the National Natural Science Foundation of China under Grant 61871133, Grant 61971139, and Grant 62071125, in part by the Natural Science Foundation of Fujian Province under Grant 2018J01805, and in part by the Scientific Research Foundation of Fuzhou University under Grant GXRC-18083.

ABSTRACT Due to the limited battery capacity and computing capability of mobile users, the resource allocation strategy in device-to-device (D2D)-assisted edge computing system with hybrid energy harvesting is investigated in this paper. By employing magnetic induction-based wireless reverse charging technology, mobile user can supplement extra energy from nearby users when the energy harvested from ambient radio frequency sources is about to be exhausted. Moreover, mobile user can not only perform local computation, but also offload computing tasks to nearby users for auxiliary computation through D2D communication links or mobile edge computing (MEC) server under base station (BS) for edge computation. Due to the limited computing resources of MEC server, when the computing capability of the MEC server reaches the maximum value, an adjacent user under another nearby BS can be considered as a relay node. The computing tasks of the remaining users under the previous BS can be transferred to the MEC server with sufficient resources under another nearby BS by establishing D2D relay links. The objective of the resource allocation strategy is to maximize the energy efficiency under the constraints of computation delay and energy harvesting. The resource allocation problem is formulated as a mixed-integer nonlinear programming problem, which is not easy to obtain the optimal solution at low computational complexity. A suboptimal solution is obtained by adopting the quantum-behaved particle swarm optimization (QPSO) algorithm. Simulation results show that the performance of the proposed strategy is superior to other benchmark strategies, and QPSO algorithm can achieve higher energy efficiency than the standard particle swarm optimization algorithm.

INDEX TERMS Hybrid energy harvesting, device-to-device communication, mobile edge computing, resource allocation.

I. INTRODUCTION

Radio frequency (RF) energy harvesting is an emerging technology to provide power for smart mobile devices (SMD). The limited battery capacity [2] and computing capability [3], [4] are two shortcomings of SMD. When the battery energy of SMD is exhausted, the service of the SMD will be terminated. For the traditional battery-powered equipment, this can possibly be overcome by using larger batteries or charging the batteries repeatedly. However, using larger batteries at SMDs means not only increasing the cost of hardware, but also inconvenient to carry. Moreover, repeated wired charging is not conducive to the user's experience [5]. SMDs can constantly harvest energy from the received electromagnetic waves to resolve above issues. The SMD offload computing tasks to edge server by exploiting mobile edge computing (MEC) technology [6]–[8], which can significantly reduce the energy consumption of SMD. Meanwhile, the harvested energy can be efficiently utilized by designing resource allocation strategy. Therefore, it is of great significance to study the resource allocation strategy of energy harvesting communication system combined with MEC technology.

The resource allocation problem of energy harvesting communication system combined with MEC has attracted great attention. In [9], wireless power transmitter (WPT) and MEC

The associate editor coordinating the review of this manuscript and approving it for publication was Francisco Rafael Marques Lima.

technology were taken into account in the access point (AP) in order to maximize the computational energy efficiency of low power SMDs. The WPT was used for energy transfer to low power SMDs in the first phase, and in the second phase these SMDs can offload computing tasks to MEC server for edge computation. The WPT-MEC with partial offloading scheme was first time introduced and analyzed to overcome the computational capability and battery limitation problems. Wang *et al.* [10] discussed the total transmission energy consumption at the energy transmitter (ET) over a particular finite horizon in a single-user wireless powered MEC system. By jointly optimizing the transmission energy allocation at the ET and tasks allocation at SMD to ensure the SMD's successful tasks execution. Under the energy and tasks causality constraints, the energy minimization problem was solved through convex optimization techniques. In [11], a reinforcement learning based offloading scheme was studied for an Internet of Things device with energy harvesting. According to the current battery level, the previous radio transmission rate to each edge device, and the predicted amount of the harvested energy, SMDs were able to select the edge device and the offloading rate in order to optimize the offloading policy. A framework for wireless powered cognitive radio (CR)-based MEC-enabled networks was considered in [12]. The objective was to maximize the average calculated number of bits of SMD. The established non-convex optimization problem was solved by using Lagrangian dual decomposition and successive pseudo-convex approximation methods. From the perspective of tasks offloading, there are two main methods of tasks offloading. For the binary offloading [13], [14], computing tasks can be totally remained for local computation or totally offloaded. For the partial offloading [15], [16], computing tasks can be split into several parts, some tasks are remained for local computation and the rest are offloaded.

Although the above studies have demonstrated the effectiveness of the resource allocation strategy of RF energy harvesting combined with MEC to improve the computation performance of communication system, the limited computing resources of MEC server is not always adequate to support all SMDs under the coverage range of base station (BS). To improve this situation, there have been also several works [17]–[19] that investigate the device-to-device (D2D)-assisted MEC system. In [17], the D2D-assisted and non-orthogonal multiple access (NOMA)-based MEC system was investigated to minimize the weighted sum of the energy consumption and delay of all SMDs. A novel D2D-enabled multi-helper MEC system was developed in [18], where the local user could offload its computing tasks to multiple helpers and download the results from them over an orthogonal pre-scheduled time slot. Wen *et al.* [19] attempted to further improve the energy-efficient communication in MEC by proposing a D2D offloading architecture. The aforementioned studies mainly focus on the communication system with energy harvesting from ambient or dedicated RF sources. However, due to the low receiving power based on RF energy harvesting, less energy is harvested in a short period of time. Once the battery energy is exhausted, the system performance will be seriously affected. The wireless energy transfer technology based on magnetic induction (MI) [20]–[22] has much higher receiving power and can obtain more energy in a short time. Motivated by the above observation, we integrate the MI-based wireless energy transfer and the D2D communication into the MEC system. Therefore, we propose a resource allocation strategy for D2D-assisted edge computing system with hybrid energy harvesting to improve the total energy efficiency of communication system. The main contributions of this work are summarized as follows.

- A hybrid energy harvesting method (i.e., RF energy harvesting and MI-based wireless energy transfer) is developed to provide electrical energy for SMDs. By leveraging MI-based wireless energy transfer technology, the SMD can supplement energy from nearby other SMDs (i.e., MI-based wireless reverse charging) when this SMD is about to run out of the energy harvested from ambient RF sources.
- By adopting partial offloading method, computing tasks are split into three parts for computation. One part is remained for local computation. Another part is offloaded to nearby SMDs for auxiliary computation through establishing D2D communication links. The rest part is offloaded to MEC server for edge computation. Moreover, when the computing capability of the MEC server under BS achieves the maximum number of SMDs that can be served, the remainder SMDs can consider an adjacent SMD under another nearby BS as a relay node. This relay node can transfer the rest tasks to the nearby MEC server with sufficient resources under another nearby BS for edge computation by establishing D2D relay links.
- A resource allocation strategy for D2D-assisted edge computing system with hybrid energy harvesting is proposed to maximize the energy efficiency under the constraints of computation delay and energy harvesting. A quantum-behaved particle swarm optimization (QPSO) algorithm is used to obtain a suboptimal solution to the formulated mixed-integer nonlinear programming (MINLP) problem.

The rest of this paper is organized as follows. In Section II, the system model is presented. The problem formulation and solution are shown in Section III. Simulation results and discussions are given in Section IV. Finally, the whole paper is concluded in Section V.

II. SYSTEM MODEL

In this section, the network structure of D2D-assisted edge computing system is presented. Then, the hybrid energy harvesting model of mobile user is given.

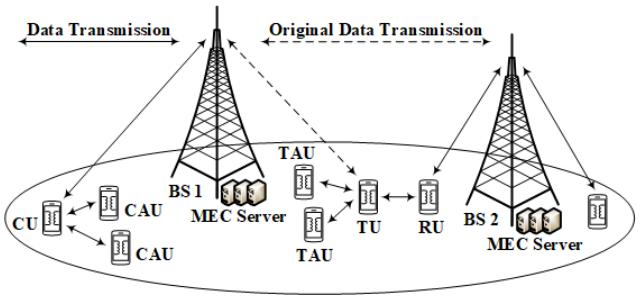

FIGURE 1. Network structure of D2D-assisted edge computing system.

A. NETWORK STRUCTURE

As depicted in Fig. 1, we consider a D2D-assisted edge computing system that consists of two BSs and multiple mobile users. MEC server is deployed near the BS. In order to improve the performance of the whole communication system, except for local computation, mobile users can also offload their computing tasks to the MEC server for edge computation, or offload to nearby users for auxiliary computation through establishing D2D communication links. Mobile users in D2D communication are still controlled by BS [23], [24]. However, offloading computing tasks will consume a certain amount of battery energy. Due to the limited battery capacity of mobile users, if the battery energy of mobile user is about to be exhausted when this user is offloading a computing task, this user can get a timely energy supplement from nearby users through MI-based wireless reverse charging to avoid the interruption of the task offloading. We call these mobile users as common users (CU), denoted as CU i ($i = 1, 2, \cdots, I$), and call these nearby users as common auxiliary users (CAU), denoted as CAU k ($k = 1, 2, \cdots, K = I - 1$). These CAUs actually belong to CUs as well. Auxiliary computation means that when CU i performs computation, these nearby other idle common users can assist CU i for auxiliary computation. Therefore, these idle common users are called CAUs. Assuming that there are $(I + S)$ users under BS 1 that need to offload their computing tasks to MEC server for edge computation. Due to the limited computing resources of MEC server, the computing resources of the MEC server under BS 1 hold only I users to offload their computing tasks for edge computation. In other words, when the computing capability of the MEC server under BS 1 has achieved the maximum number of users that the MEC server can serve (i.e., I users), the rest S users are unable to offload their computing tasks to the MEC server under BS 1 for edge computation. At this point, the number of users under BS 2 that close to BS 1 is small, and the computing resources of MEC server under BS 2 are sufficient. Therefore, these S users under BS 1 can utilize an adjacent user under BS 2 as a relay node, and these S users can transfer their computing tasks to the MEC server under BS 2 for edge computation by establishing D2D relay links. We call this adjacent user as relay user (RU). These S users under BS 1 are called transfer users (TU), denoted as TU s ($s = 1, 2, \cdots, S$). Moreover, TU s can also perform local computation or offload its computing tasks to nearby users for auxiliary computation. We call these nearby users as transfer auxiliary users (TAU), and denoted as TAU q ($q = 1, 2, \cdots, Q = S - 1$).

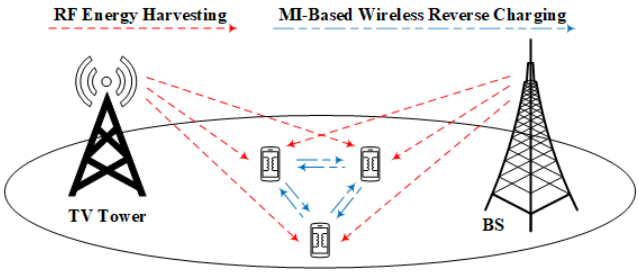

FIGURE 2. Hybrid energy harvesting mode of mobile user.

B. HYBRID ENERGY HARVESTING MODEL OF MOBILE USER

In this subsection, the hybrid energy harvesting model of mobile user is shown. As shown in Fig. 2, mobile user can harvest energy from ambient RF sources, such as television (TV) towers and BSs. When the battery energy of mobile user is about to exhausted, this user can supplement energy from nearby users by leveraging MI-based wireless reverse charging. Take CU i as an example, the energy harvested by CU i from the ambient RF sources and CAUs is denoted as E_i^{CH1} and E_i^{CH2}. We define $P_{i,l}^{\mathrm{CH1}}$ as the power received by CU i from the l-th ambient RF source. According to the Friis equation [25], $P_{i,l}^{\mathrm{CH1}}$ can be obtained by

$$P_{i,l}^{\mathrm{CH1}} = \frac{P_l^{\mathrm{T}} G_l^{\mathrm{T}} G_i^{\mathrm{R}} c^2}{\left(4\pi d_{i,l} f_l^{\mathrm{T}}\right)^2}, \tag{1}$$

where P_l^{T} and G_l^{T} denote the transmitting power and the transmitting antenna gain of the l-th ($l = 1, 2, \cdots, L$) ambient RF source, L is the number of ambient RF sources, G_i^{R} represents the receiving antenna gain of CU i, c indicates the propagation speed of electromagnetic wave in free space, $d_{i,l}$ denotes the distance between the l-th ambient RF source and CU i, and f_l^{T} is transmitting frequency of the l-th ambient RF source. In order to ensure a zero-input/zero-output response for energy harvesting, the constant Ω_i is shown as

$$\Omega_i = \frac{1}{1 + \exp\left(a_i b_i\right)}, \tag{2}$$

where a_i and b_i are constants depending on the practical circuit specifications. According to [26], [27], the practical power harvested by CU i from the l-th ambient RF source is given as

$$\Phi_{i,l}^{\mathrm{CH1}} = \frac{\left[\Psi_{i,l}^{\mathrm{CH1}} - M_i \Omega_i\right]}{1 - \Omega_i}, \tag{3}$$

where M_i indicates the maximal harvested power at CU i when the energy harvesting circuit is driven to saturation.

$\Psi_{i,l}^{\mathrm{CH1}}$ represents the traditional logistic function with respect to received power $P_{i,l}^{\mathrm{CH1}}$, and its expression is

$$\Psi_{i,l}^{\mathrm{CH1}} = \frac{M_i}{1+\exp\left(-a_i\left(P_{i,l}^{\mathrm{CH1}} - b_i\right)\right)}. \tag{4}$$

Therefore, the energy harvesting model of E_i^{CH1} is shown as

$$E_i^{\mathrm{CH1}} = \sum_{l=1}^{L} \Phi_{i,l}^{\mathrm{CH1}} T_{\mathrm{RF}}, \tag{5}$$

where T_{RF} indicates the duration of harvesting energy from ambient RF sources.

The energy harvesting model of E_i^{CH2} is analyzed as follows. Because CU i and CAU k are equipped with magnetic induction coil, there exist mutual inductance between the receiving coil in CU i and the transmitting coil in CAU k, which is calculated as [28]

$$M_{i,k} = \frac{\mu\pi N_k^{\mathrm{t}} N_i^{\mathrm{r}} (a_k^{\mathrm{t}})^2 (a_i^{\mathrm{r}})^2}{2\sqrt{\left((a_k^{\mathrm{t}})^2 + (d_{i,k}^{\mathrm{coil}})^2\right)^3}}, \tag{6}$$

where μ represents the permeability of free space, N_k^{t} and N_i^{r} denote the turns of transmitting and receiving coils, a_k^{t} and a_i^{r} are the radii of transmitting and receiving coils. Moreover, $d_{i,k}^{\mathrm{coil}}$ is the distance between transmitting and receiving coils.

According to the material and the practical implementation of coil, the transmitting coil resistance R_k^{t} and the receiving coil resistance R_i^{r} should be designed to a low value to avoid considerable energy waste, which can be obtained by

$$\begin{cases} R_k^{\mathrm{t}} = 2\pi N_k^{\mathrm{t}} a_k^{\mathrm{t}} R_0 \\ R_i^{\mathrm{r}} = 2\pi N_i^{\mathrm{r}} a_i^{\mathrm{r}} R_0, \end{cases} \tag{7}$$

where R_0 is the resistance per unit length of the coil. Moreover, the self-induction of transmitting and receiving coils can be derived by

$$\begin{cases} L_k^{\mathrm{t}} \approx \frac{1}{2}\mu\pi (N_k^{\mathrm{t}})^2 a_k^{\mathrm{t}} \\ L_i^{\mathrm{r}} \approx \frac{1}{2}\mu\pi (N_i^{\mathrm{r}})^2 a_i^{\mathrm{r}}. \end{cases} \tag{8}$$

In order to maximize the received power, the load impedance is expressed as the complex conjugate of the output impedance of the secondary loop [29]

$$Z_{i,k}^{\mathrm{L}} = R_i^{\mathrm{r}} + \frac{\omega^2 M_{i,k}^2 R_k^{\mathrm{t}}}{R_k^{\mathrm{t}} + \omega^2 (L_k^{\mathrm{t}})^2} + j\left(\frac{\omega^3 M_{i,k}^2 L_k^{\mathrm{t}}}{(R_k^{\mathrm{t}})^2 + 2\omega^2 (L_k^{\mathrm{t}})^2} - \omega L_i^{\mathrm{r}}\right), \tag{9}$$

where ω represents the angular frequency that can be obtained by $\omega = 2\pi f$. Here, f indicates the system operating frequency.

Furthermore, the induced voltage of receiving coil $U_{i,k}^{\mathrm{M}}$ can be derived by

$$U_{i,k}^{\mathrm{M}} = -j\omega M_{i,k} \frac{U_k^{\mathrm{S}}}{R_k^{\mathrm{t}} + j\omega L_k^{\mathrm{t}}}, \tag{10}$$

where U_k^{S} is the power supply voltage of CAU k. According to the principle of equivalent circuit, the received power is equal to the power consumption of $Z_{i,k}^{\mathrm{L}}$, i.e.,

$$P_{i,k}^{\mathrm{CR}} = \mathrm{Re}\left\{\frac{Z_{i,k}^{\mathrm{L}}\left(U_{i,k}^{\mathrm{M}}\right)^2}{\left[\mathrm{Re}\left(Z_{i,k}^{\mathrm{L}}\right)\right]^2}\right\}. \tag{11}$$

Therefore, E_i^{CH2} can be expressed as

$$E_i^{\mathrm{CH2}} = \sum_{k=1}^{K} u_{i,k} P_{i,k}^{\mathrm{CR}} T_{\mathrm{MI}}, \tag{12}$$

where $u_{i,k}$ in energy harvesting case represents that the distance decision that whether CU i can harvest energy from CAU k. When $u_{i,k} = 1$, CU i can harvest energy from CAU k through MI-based wireless reverse charging. On the contrary, $u_{i,k} = 0$ denotes that CU i is too far away from CAU k to harvest energy from CAU k. T_{MI} indicates the duration of charging. As a result, according to the expressions of E_i^{CH1} and E_i^{CH2}, the total energy harvesting model of CU i can be depicted as $E_i^{\mathrm{CH}} = E_i^{\mathrm{CH1}} + E_i^{\mathrm{CH2}}$.

Similarly, TU s can also harvest energy from ambient RF sources and TAUs. The energy harvested by TU s from ambient RF sources is expressed as

$$E_s^{\mathrm{TH1}} = \sum_{l=1}^{L} \Phi_{s,l}^{\mathrm{TH1}} T_{\mathrm{RF}}. \tag{13}$$

Moreover, the energy harvested by TU s from TAUs is denoted as

$$E_s^{\mathrm{TH2}} = \sum_{q=1}^{Q} \varepsilon_{s,q} P_{s,q}^{\mathrm{TR}} T_{\mathrm{MI}}. \tag{14}$$

As a result, the total energy harvesting model of TU s can be shown as $E_s^{\mathrm{TH}} = E_s^{\mathrm{TH1}} + E_s^{\mathrm{TH2}}$.

III. PROBLEM FORMULATION AND SUBOPTIMAL SOLUTION

In this section, the specific resource allocation strategy is analyzed. The optimization model of resource allocation is given first. Then, the suboptimal solution is obtained by using QPSO algorithm.

A. PROBLEM FORMULATION

In this subsection, the problem formulation is presented. As mentioned earlier, mobile users under BS 1 are mainly classified as CUs and TUs. We define C_w as the number of central processing unit (CPU) cycles required for user w ($w = 1, 2, \cdots, I+S$) to perform 1-bit computation data.

By adopting partial offloading method, we assume that computing tasks of user w are split into three parts. One part is remained for local computation, another part is offloaded to nearby users for auxiliary computation through establishing D2D communication links. The rest part is offloaded to MEC server for edge computation. The time and energy consumption models of CU and TU are analyzed in detail as follows.

1) LOCAL COMPUTATION MODEL

We suppose D_i^{CL} and f_i^{CL} as the computation data and the computation capability of CU i for local computation. Therefore, the time required for CU i to perform this computation data D_i^{CL} is

$$T_i^{\mathrm{CL}} = \frac{C_i D_i^{\mathrm{CL}}}{f_i^{\mathrm{CL}}}. \tag{15}$$

At the same time, the energy consumption of CU i is

$$E_i^{\mathrm{CL}} = \kappa \left(f_i^{\mathrm{CL}}\right)^2 C_i D_i^{\mathrm{CL}}, \tag{16}$$

where κ is the effective switched capacitance depending on the chip architecture. In this paper, we set $\kappa = 10^{-26}$ [30]. Similarly, the time and energy consumption of TU s for local computation are given as follows

$$T_s^{\mathrm{TL}} = \frac{C_s D_s^{\mathrm{TL}}}{f_s^{\mathrm{TL}}}, \tag{17}$$

$$E_s^{\mathrm{TL}} = \kappa \left(f_s^{\mathrm{TL}}\right)^2 C_s D_s^{\mathrm{TL}}, \tag{18}$$

where D_s^{TL} and f_s^{TL} indicate the computation data and the computation capability of TU s for local computation.

2) AUXILIARY COMPUTATION MODEL

The time division duplex (TDD) scheme [31] is employed to collect the channel state information. The collection of uplink channel state information is mainly divided into three parts: First, the user transmits uplink pilot signal to the BS. The time required for this period is T^{O}. Then, the BS estimates the channel state information based on the received signal. Finally, the BS feeds back the channel state information to the user. The time required for this period is also T^{O} [32]. For the TDD system, due to channel reciprocity, the downlink has the same channel state information as the uplink. After the computing tasks are finished at the edge server, the computing results will be fed back to the user by downlink. Because the computing process takes some time, the downlink channel state information will be different from the previous uplink channel state information. Therefore, downlink channel estimation is also needed. Similarly, the collection of downlink channel state information is also mainly divided into three parts. The BS first transmits downlink pilot signal to the user. The time required for this period is $3T^{\mathrm{O}}$. Then the user estimates the channel state information based on the received signal, and finally the user feeds back the channel state information to the BS. The time required for feedback process is T^{O} [32]. Therefore, the energy consumption for collecting uplink and downlink channel state information are shown as

$$E^{\mathrm{UC}} = P^{\mathrm{D}} T^{\mathrm{O}} + E^{\mathrm{C}} + P^{\mathrm{B}} T^{\mathrm{O}}, \tag{19}$$

$$E^{\mathrm{DC}} = P^{\mathrm{B}} 3T^{\mathrm{O}} + E^{\mathrm{C}} + P^{\mathrm{D}} T^{\mathrm{O}}, \tag{20}$$

where P^{D} represents the transmitting power of users. E^{C} denotes the energy required to perform the channel estimation process. P^{B} indicates the transmitting power of BS. T^{O} is each orthogonal frequency division multiplexing (OFDM) symbol interval.

Moreover, we adopt orthogonal frequency division multiple access method for data transmission whether auxiliary computation or edge computation. It is assumed that the total channel bandwidth B_1 is divided into $I \times K$ sub-channels when CUs offload their computing tasks to CAUs for auxiliary computation through D2D communication links. We define $\alpha_{i,k}^1$ as the sub-channel ratio coefficient of the total channel bandwidth occupied by the data transmission from CU i to CAU k. According to [33], the achievable uplink data rate from CU i to CAU k is expressed as

$$R_{i,k}^{\mathrm{CU}} = \alpha_{i,k}^1 B_1 \log_2 \left(1 + \frac{P_{i,k}^{\mathrm{CU}} (h_{i,k}^1)^2}{N_0}\right), \tag{21}$$

where $P_{i,k}^{\mathrm{CU}}$ represents the transmitting power of CU i offloading computing tasks to CAU k. $h_{i,k}^1$ denotes the channel gain from CU i to CAU k. N_0 indicates the Gaussian white noise power. We suppose $D_{i,k}^{\mathrm{CA}}$ as the computation data of CU i offloading to CAU k for auxiliary computation. Therefore, the time of data transmission from CU i to CAU k can be obtained by

$$T_{i,k}^{\mathrm{CT}} = \frac{u_{i,k} D_{i,k}^{\mathrm{CA}}}{R_{i,k}^{\mathrm{CU}}}, \tag{22}$$

where $u_{i,k}$ in offloading case denotes that the distance decision that whether CU i can offload computing tasks to CAU k for auxiliary computation. When $u_{i,k} = 1$, CU i can offload computing tasks to CAU k. On the contrary, $u_{i,k} = 0$ indicates that CU i is too far away from CAU k to offload computing tasks to CAU k for auxiliary computation.

The actual uplink time required from CU i to CAU k depends on the greater one between the time of local computation of CAU k and the time of data transmission from CU i to CAU k [34]. In other words, CAU k can help other users for auxiliary computation only when its own local computing tasks has completed. We suppose D_k^{CL} and f_k^{CL} as the computation data and the computation capability of CAU k for local computation. Therefore, the actual uplink time can be given by

$$T_{i,k}^{\mathrm{CU}} = \max \left(T_{i,k}^{\mathrm{CT}}, T_k^{\mathrm{CL}} = \frac{C_k D_k^{\mathrm{CL}}}{f_k^{\mathrm{CL}}}\right). \tag{23}$$

We suppose P^{CIR} as the circuit power consumption. Therefore, the energy consumption of CU i transmitting computation data to CAU k through D2D communication link is

shown as

$$E_{i,k}^{\mathrm{CU}} = \left(u_{i,k}P_{i,k}^{\mathrm{CU}} + P^{\mathrm{CIR}}\right)\frac{u_{i,k}D_{i,k}^{\mathrm{CA}}}{R_{i,k}^{\mathrm{CU}}}. \tag{24}$$

After receiving the computation data of CU i offloading to CAU k for auxiliary computation, CAU k begins to perform this computation data. The time and energy consumption of CAU k performing this computation data are obtained as

$$T_{i,k}^{\mathrm{CP}} = \frac{u_{i,k}C_iD_{i,k}^{\mathrm{CA}}}{f_k^{\mathrm{CL}}}, \tag{25}$$

$$E_{i,k}^{\mathrm{CP}} = \kappa\left(f_k^{\mathrm{CL}}\right)^2 u_{i,k}C_iD_{i,k}^{\mathrm{CA}}. \tag{26}$$

After finishing computing tasks of CU i offloading to CAU k for auxiliary computation, CAU k outputs a computation result. CAU k must transmit this result to CU i. We define $\alpha_{k,i}^2$ as the sub-channel ratio coefficient of the total channel bandwidth occupied by the computation result feedback from CAU k to CU i. Therefore, the achievable downlink data rate from CAU k to CU i is expressed as

$$R_{k,i}^{\mathrm{CD}} = \alpha_{k,i}^2 B_1 \log_2\left(1 + \frac{P_{k,i}^{\mathrm{CD}}(h_{k,i}^2)^2}{N_0}\right), \tag{27}$$

where $P_{k,i}^{\mathrm{CD}}$ represents the transmitting power of CAU k transmitting computation result to CU i. $h_{k,i}^2$ denotes the channel gain from CAU k to CU i. We define ϕ as the conversion coefficient between the output computation result and the input computation data. Therefore, the downlink time required from CAU k to CU i can be given by

$$T_{k,i}^{\mathrm{CD}} = \frac{\phi u_{i,k}D_{i,k}^{\mathrm{CA}}}{R_{k,i}^{\mathrm{CD}}}. \tag{28}$$

Moreover, the energy consumption of CAU k transmitting computation result to CU i is shown as

$$E_{k,i}^{\mathrm{CD}} = \left(u_{i,k}P_{k,i}^{\mathrm{CD}} + P^{\mathrm{CIR}}\right)\frac{\phi u_{i,k}D_{i,k}^{\mathrm{CA}}}{R_{k,i}^{\mathrm{CD}}}. \tag{29}$$

According to the above analysis, the time and energy required of CU i offloading computing tasks to CAU k for auxiliary computation are given by

$$T_{i,k}^{\mathrm{CA}} = T_{i,k}^{\mathrm{CU}} + T_{i,k}^{\mathrm{CP}} + T_{k,i}^{\mathrm{CD}}, \tag{30}$$

$$E_{i,k}^{\mathrm{CA}} = E_{i,k}^{\mathrm{CU}} + E_{i,k}^{\mathrm{CP}} + E_{k,i}^{\mathrm{CD}}. \tag{31}$$

We suppose D_i^{CA} as the total computation data of CU i offloading to CAUs that can be used for auxiliary computation. Therefore, the total computation data can be shown as

$$D_i^{\mathrm{CA}} = \sum_{k=1}^{K} u_{i,k}D_{i,k}^{\mathrm{CA}}. \tag{32}$$

Because of the time and energy consumption of CU i offloading computing tasks to each CAU for auxiliary computation are different, for the time consumption, we only need to take the maximum time of each computation result arrival as the total time consumption of auxiliary computation. Therefore, the total time consumption of CU i for auxiliary computation is given as

$$T_i^{\mathrm{CA}} = \max\left(u_{i,k}T_{i,k}^{\mathrm{CA}}\right). \tag{33}$$

For the energy consumption, the total energy consumption of CU i for auxiliary computation is equal to the sum of the energy consumption of CU i offloading computing tasks to each CAU for auxiliary computation. Therefore, the total energy consumption of CU i for auxiliary computation is expressed as

$$E_i^{\mathrm{CA}} = \sum_{k=1}^{K}\left(E_{i,k}^{\mathrm{CA}}\right). \tag{34}$$

For TU, it can also offload computing tasks to nearby TAUs for auxiliary computation. We suppose B_3 as the total channel bandwidth of TUs offloading their computing tasks to TAUs for auxiliary computation through D2D communication links. B_3 is divided into $S \times Q$ sub-channels. We define $\gamma_{s,q}^1$ as the sub-channel ratio coefficient of the total channel bandwidth occupied by the data transmission from TU s to TAU q. Therefore, the achievable uplink data rate from TU s to TAU q is obtained by

$$R_{s,q}^{\mathrm{TU}} = \gamma_{s,q}^1 B_3 \log_2\left(1 + \frac{P_{s,q}^{\mathrm{TU}}(h_{s,q}^5)^2}{N_0}\right), \tag{35}$$

where $P_{s,q}^{\mathrm{TU}}$ denotes the transmitting power of TU s offloading computing tasks to TAU q. $h_{s,q}^5$ indicates the channel gain from TU s to TAU q. We suppose $D_{s,q}^{\mathrm{TA}}$ as the computation data of TU s offloading to TAU q for auxiliary computation. Therefore, the time of data transmission from TU s to TAU q can be obtained by

$$T_{s,q}^{\mathrm{TT}} = \frac{\varepsilon_{s,q}D_{s,q}^{\mathrm{TA}}}{R_{s,q}^{\mathrm{TU}}}, \tag{36}$$

where the meaning of $\varepsilon_{s,q}$ is similar to that of $u_{i,k}$.

The actual uplink time required from TU s to TAU q depends on the greater one between the time of local computation of TAU q and the time of data transmission from TU s to TAU q. We suppose D_q^{TL} and f_q^{TL} as the computation data and the computation capability of TAU q for local computation. Therefore, the actual uplink time is given as

$$T_{s,q}^{\mathrm{TU}} = \max\left(T_{s,q}^{\mathrm{TT}}, T_q^{\mathrm{TL}} = \frac{C_qD_q^{\mathrm{TL}}}{f_q^{\mathrm{TL}}}\right). \tag{37}$$

Moreover, the energy consumption of TU s transmitting computation data to TAU q through D2D communication link is shown as

$$E_{s,q}^{\mathrm{TU}} = \left(\varepsilon_{s,q}P_{s,q}^{\mathrm{TU}} + P^{\mathrm{CIR}}\right)\frac{\varepsilon_{s,q}D_{s,q}^{\mathrm{TA}}}{R_{s,q}^{\mathrm{TU}}}. \tag{38}$$

After receiving the computation data of TU s offloading to TAU q for auxiliary computation, TAU q starts to perform

this computation data. The time and energy consumption of TAU q performing this computation data are obtained as

$$T_{s,q}^{\mathrm{TP}} = \frac{\varepsilon_{s,q} C_s D_{s,q}^{\mathrm{TA}}}{f_q^{\mathrm{TL}}}, \tag{39}$$

$$E_{s,q}^{\mathrm{TP}} = \kappa \left(f_q^{\mathrm{TL}}\right)^2 \varepsilon_{s,q} C_s D_{s,q}^{\mathrm{TA}}. \tag{40}$$

After TAU q finishes computing tasks of TU s offloading and outputs a computation result, TAU q must transmit this computation result to TU s. We define $\gamma_{q,s}^2$ as the sub-channel ratio coefficient of the total channel bandwidth occupied by the computation result feedback from TAU q to TU s. Therefore, the achievable downlink data rate from TAU q to TU s is expressed as

$$R_{q,s}^{\mathrm{TD}} = \gamma_{q,s}^2 B_3 \log_2 \left(1 + \frac{P_{q,s}^{\mathrm{TD}} (h_{q,s}^6)^2}{N_0}\right), \tag{41}$$

where $P_{q,s}^{\mathrm{TD}}$ is the transmitting power of TAU q transmitting computation result to TU s. $h_{q,s}^6$ represents the channel gain from TAU q to TU s. Therefore, the downlink time required from TAU q to TU s can be given by

$$T_{q,s}^{\mathrm{TD}} = \frac{\phi \varepsilon_{s,q} D_{s,q}^{\mathrm{TA}}}{R_{q,s}^{\mathrm{TD}}}. \tag{42}$$

Furthermore, the energy consumption of TAU q transmitting computation result to TU s is shown as

$$E_{q,s}^{\mathrm{TD}} = \left(\varepsilon_{s,q} P_{q,s}^{\mathrm{TD}} + P^{\mathrm{CIR}}\right) \frac{\phi \varepsilon_{s,q} D_{s,q}^{\mathrm{TA}}}{R_{q,s}^{\mathrm{TD}}}. \tag{43}$$

Finally, the time and energy consumption of TU s offloading computing tasks to TAU q for auxiliary computation can be obtained by

$$T_{s,q}^{\mathrm{TA}} = T_{s,q}^{\mathrm{TU}} + T_{s,q}^{\mathrm{TP}} + T_{q,s}^{\mathrm{TD}}, \tag{44}$$

$$E_{s,q}^{\mathrm{TA}} = E_{s,q}^{\mathrm{TU}} + E_{s,q}^{\mathrm{TP}} + E_{q,s}^{\mathrm{TD}}. \tag{45}$$

We suppose D_s^{TA} as the total computation data of TU s offloading to TAUs that can be used for auxiliary computation. Therefore, the total computation data can be shown as

$$D_s^{\mathrm{TA}} = \sum_{q=1}^{Q} \varepsilon_{s,q} D_{s,q}^{\mathrm{TA}}. \tag{46}$$

Similar to CU i, TU s can also offload its computing tasks to multiple TAUs for auxiliary computation. Therefore, the total time and energy consumption of TU s for auxiliary computation are given as

$$T_s^{\mathrm{TA}} = \max\left(\varepsilon_{s,q} T_{s,q}^{\mathrm{TA}}\right), \tag{47}$$

$$E_s^{\mathrm{TA}} = \sum_{q=1}^{Q} \left(E_{s,q}^{\mathrm{TA}}\right). \tag{48}$$

3) EDGE COMPUTATION MODEL

Assuming that the total channel bandwidth B_2 is divided into I sub-channels when CUs offload their computing tasks to MEC server under BS 1 for edge computation. We define β_i^1 as the sub-channel ratio coefficient of the total channel bandwidth occupied by the data transmission from CU i to BS 1. Therefore, the achievable uplink data rate from CU i to BS 1 is expressed as

$$R_i^{\mathrm{CU}} = \beta_i^1 B_2 \log_2 \left(1 + \frac{P_i^{\mathrm{CU}} (h_i^3)^2}{N_0}\right), \tag{49}$$

where P_i^{CU} represents the transmitting power of CU i offloading computing tasks to MEC server for edge computation. h_i^3 denotes the channel gain from CU i to BS 1. We suppose D_i^{CE} as the computation data of CU i offloading to MEC server for edge computation. Therefore, the uplink transmission time from CU i to BS 1 can be given by

$$T_i^{\mathrm{CU}} = \frac{D_i^{\mathrm{CE}}}{R_i^{\mathrm{CU}}}. \tag{50}$$

Because computing tasks of CU i are split into three parts, the computation data of CU i offloading to MEC server for edge computation can be obtained by $D_i^{\mathrm{CE}} = D_i^{\mathrm{CT}} - D_i^{\mathrm{CL}} - D_i^{\mathrm{CA}}$. Here, D_i^{CT} represents the total computation data of CU i need to be performed. Therefore, the energy consumption of CU i transmitting computation data to MEC server is expressed as

$$E_i^{\mathrm{CU}} = \left(P_i^{\mathrm{CU}} + P^{\mathrm{CIR}}\right) \frac{D_i^{\mathrm{CE}}}{R_i^{\mathrm{CU}}}. \tag{51}$$

After receiving the computation data from CU i, the MEC server starts to perform this computation data. Let f_i^{CE} indicates the computing resources assigned to CU i by the MEC server under BS 1. Specially, the sum of the computing resources assigned to all CUs by the MEC server under BS 1 is exactly equal to the total computing resources F^{E} of the MEC server under BS 1. In other words, the total computing resources of the MEC server under BS 1 can only meet the requirement of I users for edge computation, the MEC server will overload if beyond that. Therefore, the time and energy required for the MEC server under BS 1 to perform this computation data are obtained as

$$T_i^{\mathrm{CP}} = \frac{C_i D_i^{\mathrm{CE}}}{f_i^{\mathrm{CE}}}, \tag{52}$$

$$E_i^{\mathrm{CP}} = \delta D_i^{\mathrm{CE}}, \tag{53}$$

where δ denotes the energy consumption per offloaded bit at the MEC.

After finishing computing tasks of CU i offloading to MEC server for edge computation, the MEC server outputs a computation result. BS 1 must transmit the computation result to CU i. We define β_i^2 as the sub-channels ratio coefficient of the total channel bandwidth occupied by the computation result feedback from BS 1 to CU i. Therefore, the achievable

downlink data rate from BS 1 to CU i is expressed as

$$R_i^{\mathrm{CD}} = \beta_i^2 B_2 \log_2\left(1 + \frac{P^{\mathrm{E}}(h_i^4)^2}{N_0}\right), \tag{54}$$

where P^{E} is the transmitting power of BS, h_i^4 represents the channel gain from BS 1 to CU i. The downlink time required from BS 1 to CU i can be given by

$$T_i^{\mathrm{CD}} = \frac{\phi D_i^{\mathrm{CE}}}{R_i^{\mathrm{CD}}}. \tag{55}$$

Moreover, the energy consumption of BS 1 transmitting the computation result to CU i is shown as

$$E_i^{\mathrm{CD}} = \left(P^{\mathrm{E}} + P^{\mathrm{CIR}}\right)\frac{\phi D_i^{\mathrm{CE}}}{R_i^{\mathrm{CD}}}. \tag{56}$$

According to the above analysis, the total time and energy consumption of CU i for edge computation are given by

$$T_i^{\mathrm{CE}} = T_i^{\mathrm{CU}} + T_i^{\mathrm{CP}} + T_i^{\mathrm{CD}}, \tag{57}$$
$$E_i^{\mathrm{CE}} = E_i^{\mathrm{CU}} + E_i^{\mathrm{CP}} + E_i^{\mathrm{CD}}. \tag{58}$$

Because TU s cannot offload its computing tasks to the MEC server under BS 1 for edge computation, TU s must to utilize RU as a relay node, and transfer its computing tasks to the MEC server under BS 2 for edge computation through establishing D2D relay link. We consider RU to be a relay node of decoded and forwarding type [35]. Therefore, the actual achievable uplink data rate from TU s to BS 2 depends on the smaller one between TU s to RU and RU to BS 2, its expression is

$$R_s^{\mathrm{TU}} = \text{minimum}\left\{R_s^{\mathrm{TU1}}, R_s^{\mathrm{TU2}}\right\}, \tag{59}$$

where R_s^{TU1} and R_s^{TU2} represent the achievable uplink data rate from TU s to RU and from RU to BS 2, respectively. Assuming that the total channel bandwidth B_4 from TU s to RU and from RU to BS 2 are divided into S sub-channels, respectively. We define λ_s^1 as the sub-channel ratio coefficient of the total channel bandwidth occupied by the data transmission from TU s to RU. Therefore, the achievable uplink data rate from TU s to RU is obtained by

$$R_s^{\mathrm{TU1}} = \lambda_s^1 B_4 \log_2\left(1 + \frac{P_s^{\mathrm{TU}}(h_s^7)^2}{N_0}\right), \tag{60}$$

where P_s^{TU} indicates the transmitting power of TU s transferring computing tasks to RU. h_s^7 represents the channel gain from TU s to RU. Meanwhile, we define λ_s^2 as the sub-channel ratio coefficient of the total channel bandwidth occupied by the data transmission from RU to BS 2. Therefore, the achievable uplink data rate from RU to BS 2 is given by

$$R_s^{\mathrm{TU2}} = \lambda_s^2 B_4 \log_2\left(1 + \frac{P^{\mathrm{R}}(h_s^8)^2}{N_0}\right), \tag{61}$$

where P^{R} is the forwarding power of RU. h_s^8 denotes the channel gain from RU to BS 2. We suppose D_s^{TE} as the computation data of TU s offloading to MEC server under BS 2 for edge computation. Similar to CU i, the computation data of TU s offloading to MEC server for edge computation can be obtained by $D_s^{\mathrm{TE}} = D_s^{\mathrm{TT}} - D_s^{\mathrm{TL}} - D_s^{\mathrm{TA}}$. Here, D_s^{TT} indicates the total computation data of TU s need to be performed. Therefore, the time and energy consumption of TU s transmitting computing data to MEC server are expressed as

$$T_s^{\mathrm{TU}} = \frac{D_s^{\mathrm{TE}}}{R_s^{\mathrm{TU}}}, \tag{62}$$
$$E_s^{\mathrm{TU}} = \left(P_s^{\mathrm{TU}} + P^{\mathrm{CIR}}\right)\frac{D_s^{\mathrm{TE}}}{R_s^{\mathrm{TU}}}. \tag{63}$$

After the transmission of the computation data that TU s offloading to the MEC server under BS 2 has completed, the MEC server begins to perform. Let f_s^{TE} indicates the computing resources assigned to TU s by the MEC server under BS 2. Therefore, the time and energy required for the MEC server under BS 2 to perform this computation data are obtained as

$$T_s^{\mathrm{TP}} = \frac{C_s D_s^{\mathrm{TE}}}{f_s^{\mathrm{TE}}}, \tag{64}$$
$$E_s^{\mathrm{TP}} = \delta D_s^{\mathrm{TE}}. \tag{65}$$

Similarly, the MEC server under BS 2 outputs a computation result, and BS 2 must transmit this computation result to TU s through D2D relay link. Therefore, the actual achievable downlink data rate from BS 2 to TU s depends on the smaller one between BS 2 to RU and RU to TU s, its expression is

$$R_s^{\mathrm{TD}} = \text{minimum}\left\{R_s^{\mathrm{TD1}}, R_s^{\mathrm{TD2}}\right\}, \tag{66}$$

where R_s^{TD1} and R_s^{TD2} represent the achievable downlink data rate from BS 2 to RU and from RU to TU s, respectively. We define λ_s^3 and λ_s^4 as the sub-channel ratio coefficient of the total channel bandwidth B_4 occupied by the data transmission from BS 2 to RU and from RU to TU s, respectively. Therefore, the achievable downlink data rate from BS 2 to RU and from RU to TU s can be obtained by

$$R_s^{\mathrm{TD1}} = \lambda_s^3 B_4 \log_2\left(1 + \frac{P^{\mathrm{E}}(h_s^9)^2}{N_0}\right), \tag{67}$$
$$R_s^{\mathrm{TD2}} = \lambda_s^4 B_4 \log_2\left(1 + \frac{P^{\mathrm{R}}(h_s^{10})^2}{N_0}\right), \tag{68}$$

where h_s^9 and h_s^{10} are the channel gain from BS 2 to RU and from RU to TU s. Therefore, the downlink time required from BS 2 to TU s can be given by

$$T_s^{\mathrm{TD}} = \frac{\phi D_s^{\mathrm{TE}}}{R_s^{\mathrm{TD}}}. \tag{69}$$

Moreover, the energy consumption of BS 2 transmitting the computation result to TU s is shown as

$$E_s^{\mathrm{TD}} = \left(P^{\mathrm{E}} + P^{\mathrm{CIR}}\right)\frac{\phi D_s^{\mathrm{TE}}}{R_s^{\mathrm{TD}}}. \tag{70}$$

According to the above analysis, the total time and energy consumption of TU s for edge computation are given by

$$T_s^{\mathrm{TE}} = T_s^{\mathrm{TU}} + T_s^{\mathrm{TP}} + T_s^{\mathrm{TD}}, \tag{71}$$

$$E_s^{\mathrm{TE}} = E_s^{\mathrm{TU}} + E_s^{\mathrm{TP}} + E_s^{\mathrm{TD}}. \tag{72}$$

The optimization objective of the resource allocation strategy is to maximize the energy efficiency [36], [37] while satisfying several constraints. Therefore, the optimization problem can be formulated as

$$\underset{\substack{u_{i,k},P_{i,k}^{\mathrm{CU}},P_{k,i}^{\mathrm{CD}},P_i^{\mathrm{CU}},\\ \varepsilon_{s,q},P_{s,q}^{\mathrm{TU}},P_{q,s}^{\mathrm{TD}},P_s^{\mathrm{TU}}}}{\text{maximize}} \sum_{i=1}^{I}\left(\frac{D_i^{\mathrm{CL}}}{E_i^{\mathrm{CL}}}+\frac{D_i^{\mathrm{CA}}}{E_i^{\mathrm{CA}}}+\frac{D_i^{\mathrm{CE}}}{E_i^{\mathrm{CE}}}\right) + \sum_{s=1}^{S}\left(\frac{D_s^{\mathrm{TL}}}{E_s^{\mathrm{TL}}}+\frac{D_s^{\mathrm{TA}}}{E_s^{\mathrm{TA}}}+\frac{D_s^{\mathrm{TE}}}{E_s^{\mathrm{TE}}}\right), \tag{73a}$$

$$\text{subject to } C1: \max\left[T_i^{\mathrm{CL}}, T_i^{\mathrm{CA}}, T_i^{\mathrm{CE}}\right] \le T^{\mathrm{D}},\ \forall i, \tag{73b}$$

$$C2: \max\left[T_s^{\mathrm{TL}}, T_s^{\mathrm{TA}}, T_s^{\mathrm{TE}}\right] \le T^{\mathrm{D}},\ \forall s, \tag{73c}$$

$$C3: E_i^{\mathrm{CL}} + \sum_{k=1}^{K}\left(E_{i,k}^{\mathrm{CU}}\right) + E_i^{\mathrm{CU}} \le E_i^{\mathrm{CH}},\ \forall i, \tag{73d}$$

$$C4: E_s^{\mathrm{TL}} + \sum_{q=1}^{Q}\left(E_{s,q}^{\mathrm{TU}}\right) + E_s^{\mathrm{TU}} \le E_s^{\mathrm{TH}},\ \forall s, \tag{73e}$$

$$C5: P_{i,k}^{\mathrm{CU}}, P_{k,i}^{\mathrm{CD}}, P_i^{\mathrm{CU}} \le P^{\mathrm{MAX}},\ \forall i,k, \tag{73f}$$

$$C6: P_{s,q}^{\mathrm{TU}}, P_{q,s}^{\mathrm{TD}}, P_s^{\mathrm{TU}} \le P^{\mathrm{MAX}},\ \forall s,q, \tag{73g}$$

$$C7: u_{i,k} \in (0,1),\ \forall i,k, \tag{73h}$$

$$C8: \varepsilon_{s,q} \in (0,1),\ \forall s,q. \tag{73i}$$

In the above, the first and second constraints indicate that the time consumption of CU i and TU s offloading their computing tasks for local computation, auxiliary computation, or edge computation must be not exceed the specific total computation delay T^{D}, respectively. The third and fourth constraints represent that the total energy consumption of CU i and TU s for local computation, auxiliary computation, and edge computation have to less than or equal to the energy harvested by CU i and TU s, respectively. The fifth and sixth constraints specify that the maximum transmitting power P^{MAX} of CU i and TU s, respectively. The seventh constraint denotes the distance decision that whether CU i can harvest energy from CAU k and whether CU i can offload computing tasks to CAU k for auxiliary computation. Similar to the seventh constraint, the eighth is the constraint of the distance decision between TU s and TAU q. It should be noted that the independent variables $P_{i,k}^{\mathrm{CU}}$, $P_{k,i}^{\mathrm{CD}}$, P_i^{CU}, $P_{s,q}^{\mathrm{TU}}$, $P_{q,s}^{\mathrm{TD}}$, and P_s^{TU} are continuous, while $u_{i,k}$ and $\varepsilon_{s,q}$ are discrete. Meanwhile, the objective function is nonlinear. Therefore, the above optimization problem of resource allocation is MINLP problem.

B. SUBOPTIMAL SOLUTION

The main challenge in solving MINLP problem is that it is difficult to obtain the optimal solution at low computational complexity. Therefore, a heuristic algorithm with moderate computational complexity can be used to obtain the suboptimal solution. The QPSO algorithm [38] is a kind of heuristic algorithm based on particle swarm optimization (PSO), which is suitable to solve complex optimization problems. Inspired by the simulation of the foraging process of birds, the PSO algorithm was developed by J. Kennedy and R. Eberhart [39] to solve the optimization problem. Assuming that particle space is a multidimensional space. There are N particles in the space. For the n-th ($n = 1, 2, \cdots, N$) particle, its position vector $\boldsymbol{X}_n$ and velocity vector $\boldsymbol{V}_n$ are initialized. The updating of each particle is generated by comparing its local best position $\boldsymbol{P}_n$ and global best position $\boldsymbol{G}$ with the previous iteration under the solution of fitness function. The update iterative equations of velocity vector and position vector for each particle are shown as

$$\begin{cases} \boldsymbol{V}_n(t+1) = \boldsymbol{V}_n(t) + c_1 r_1(\boldsymbol{P}_n - \boldsymbol{X}_n(t)) \\ \qquad\qquad + c_2 r_2(\boldsymbol{G} - \boldsymbol{X}_n(t)) \\ \boldsymbol{X}_n(t+1) = \boldsymbol{V}_n(t+1) + \boldsymbol{X}_n(t) \end{cases}, \tag{74}$$

where t denotes the iteration number, the iteration is terminated only when the number of iterations reaches the maximal number of iterations T. c_1 and c_2 indicate two acceleration coefficients, r_1 and r_2 are random numbers between 0 and 1. It can be seen in (74) that the update of particle position depends on the step size of velocity. Therefore, the velocity of particle evolution makes particles have the tendency to expand the search space. The particles have the ability to detect new search fields. In order to better control the global detection and local development capability of PSO algorithm, Y. Shi and R. Eberhart introduce the inertial weight m into equations in (74) and become as follows

$$\begin{cases} \boldsymbol{V}_n(t+1) = m\boldsymbol{V}_n(t) + c_1 r_1(\boldsymbol{P}_n - \boldsymbol{X}_n(t)) \\ \qquad\qquad + c_2 r_2(\boldsymbol{G} - \boldsymbol{X}_n(t)) \\ \boldsymbol{X}_n(t+1) = \boldsymbol{V}_n(t+1) + \boldsymbol{X}_n(t) \end{cases}. \tag{75}$$

The modified PSO algorithm is known as the standard particle swarm optimization algorithm (SPSO) [40]. However, the movement of each particle is still described by velocity and position in the SPSO algorithm. With the evolution of iteration time, the trajectory of particle is constant. Meanwhile, the velocity of each particle is limited to a certain extent, so that the search space of particles is a limited and gradually decreasing region, which cannot cover the whole feasible solution space. Therefore, SPSO algorithm cannot guarantee global convergence. To deal with this issue, QPSO algorithm is proposed based on quantum mechanics theory and can obtain a suboptimal solution that is close to globally optimal. The reason is that in quantum space, the aggregation of particles is described by the binding state produced by an attractive potential at the center of the particle's motion. These particles in a quantum bound state can appear at any

point in space with a certain probability density. Particles that satisfying the aggregation property can be searched in the whole feasible solution space, but will not diverge to infinity.

In this paper, QPSO algorithm is adopted to solve the MINLP problem in (73). First, the original constrained optimization problem needs to be transformed to an unconstrained form by using the penalty function method. Therefore, a fitness function that is composed of one objective function and one penalty function is constructed as

$$F(A) = f_{\text{obj}}(A) - \sigma P_{\text{pen}}(A), \tag{76}$$

where A represents all the independent variables $u_{i,k}$, $P_{i,k}^{\text{CU}}$, $P_{k,i}^{\text{CD}}$, P_i^{CU}, $\varepsilon_{s,q}$, $P_{s,q}^{\text{TU}}$, $P_{q,s}^{\text{TD}}$, and P_s^{TU} in (73). $f_{\text{obj}}(A)$ is the objective function, σ indicates the penalty factor, and $P_{\text{pen}}(A)$ is the penalty function that includes twelve items

$$\begin{aligned} P_{\text{pen}}(A) = {} & P_{\text{pen}}^1 + P_{\text{pen}}^2 + P_{\text{pen}}^3 + P_{\text{pen}}^4 + P_{\text{pen}}^5 + P_{\text{pen}}^6 \\ & + P_{\text{pen}}^7 + P_{\text{pen}}^8 + P_{\text{pen}}^9 + P_{\text{pen}}^{10} + P_{\text{pen}}^{11} + P_{\text{pen}}^{12}. \end{aligned} \tag{77}$$

They correspond to the eight constraints of the problem in (73), which are shown as

$$P_{\text{pen}}^1 = \sum_{i=1}^{I} \left[\max\left(0, \max\left[T_i^{\text{CL}}, T_i^{\text{CA}}, T_i^{\text{CE}}\right] - T^{\text{D}}\right)\right]^2, \tag{78a}$$

$$P_{\text{pen}}^2 = \sum_{s=1}^{S} \left[\max\left(0, \max\left[T_s^{\text{TL}}, T_s^{\text{TA}}, T_s^{\text{TE}}\right] - T^{\text{D}}\right)\right]^2, \tag{78b}$$

$$P_{\text{pen}}^3 = \sum_{i=1}^{I} \left[\max\left(0, E_i^{\text{CL}} + E_i^{\text{CO}} - E_i^{\text{CH}}\right)\right]^2, \tag{78c}$$

$$P_{\text{pen}}^4 = \sum_{s=1}^{S} \left[\max\left(0, E_s^{\text{TL}} + E_s^{\text{TO}} - E_s^{\text{TH}}\right)\right]^2, \tag{78d}$$

$$P_{\text{pen}}^5 = \sum_{i=1}^{I} \sum_{k=1}^{K} \left[\max\left(0, P_{i,k}^{\text{CU}} - P^{\text{MAX}}\right)\right]^2, \tag{78e}$$

$$P_{\text{pen}}^6 = \sum_{k=1}^{K} \sum_{i=1}^{I} \left[\max\left(0, P_{k,i}^{\text{CD}} - P^{\text{MAX}}\right)\right]^2, \tag{78f}$$

$$P_{\text{pen}}^7 = \sum_{i=1}^{I} \left[\max\left(0, P_i^{\text{CU}} - P^{\text{MAX}}\right)\right]^2, \tag{78g}$$

$$P_{\text{pen}}^8 = \sum_{s=1}^{S} \sum_{q=1}^{Q} \left[\max\left(0, P_{s,q}^{\text{TU}} - P^{\text{MAX}}\right)\right]^2, \tag{78h}$$

$$P_{\text{pen}}^9 = \sum_{q=1}^{Q} \sum_{s=1}^{S} \left[\max\left(0, P_{q,s}^{\text{TD}} - P^{\text{MAX}}\right)\right]^2, \tag{78i}$$

$$P_{\text{pen}}^{10} = \sum_{s=1}^{S} \left[\max\left(0, P_s^{\text{TU}} - P^{\text{MAX}}\right)\right]^2, \tag{78j}$$

$$P_{\text{pen}}^{11} = \sum_{i=1}^{I} \sum_{k=1}^{K} \left(u_{i,k}^2 - u_{i,k}\right)^2, \tag{78k}$$

$$P_{\text{pen}}^{12} = \sum_{s=1}^{S} \sum_{q=1}^{Q} \left(\varepsilon_{s,q}^2 - \varepsilon_{s,q}\right)^2, \tag{78l}$$

where $\max(\cdot,\cdot)$ means to return the larger one between two numbers. In (78c) and (78d), E_i^{CO} and E_s^{TO} are given as

$$E_i^{\text{CO}} = \sum_{k=1}^{K} \left(E_{i,k}^{\text{CU}}\right) + E_i^{\text{CU}}, \tag{79}$$

$$E_s^{\text{TO}} = \sum_{q=1}^{Q} \left(E_{s,q}^{\text{TU}}\right) + E_s^{\text{TU}}. \tag{80}$$

Secondly, the particle position is defined. Assuming that there are N particles representing the potential solution in the search space. For the n-th particle, its position vector $\boldsymbol{X}_n$ can be expressed as

$$\boldsymbol{X}_n = \left(\boldsymbol{X}_n^1, \boldsymbol{X}_n^2, \cdots, \boldsymbol{X}_n^w, \cdots, \boldsymbol{X}_n^W\right), \ W = I + S, \tag{81}$$

where $\boldsymbol{X}_n^w$ indicates optimization results of distance decision and transmitting power for the w-th user. The specific expression of $\boldsymbol{X}_n^w$ is shown as

$$\begin{aligned} \boldsymbol{X}_n^w = {} & (u_{w,1}, u_{w,2}, \cdots, u_{w,K}, \\ & P_{w,1}^{\text{CU}}, P_{w,2}^{\text{CU}}, \cdots, P_{w,K}^{\text{CU}}, \\ & P_{1,w}^{\text{CD}}, P_{2,w}^{\text{CD}}, \cdots, P_{K,w}^{\text{CD}}, P_w^{\text{CU}}), \ 1 \le w \le I, \end{aligned} \tag{82a}$$

$$\begin{aligned} \boldsymbol{X}_n^w = {} & (\varepsilon_{w,1}, \varepsilon_{w,2}, \cdots, \varepsilon_{w,Q}, \\ & P_{w,1}^{\text{TU}}, P_{w,2}^{\text{TU}}, \cdots, P_{w,Q}^{\text{TU}}, \\ & P_{1,w}^{\text{TD}}, P_{2,w}^{\text{TD}}, \cdots, P_{Q,w}^{\text{TD}}, P_w^{\text{TU}}), \ I+1 \le w \le I+S. \end{aligned} \tag{82b}$$

It can be seen that $\boldsymbol{X}_n^w$ is a multidimensional vector. The optimization results of distance decision and transmitting power for CU are given in (82a). Here, the first K elements denote the distance decision between CU and CAU. The second K elements indicate the transmitting power of CU offloading computing tasks to CAU. The third K elements represent the transmitting power of CAU transmitting computation result to CU. The remaining elements are the transmitting power of CU offloading computing tasks to MEC server. Similarly, the optimization results of distance decision and transmitting power for TU are shown in (82b).

In order to establish an attractive potential to affect the particles in the group, the attractor vector $\boldsymbol{P}$ is defined and its expression is $\boldsymbol{P} = \varphi \boldsymbol{P}_n(t) + (1-\varphi)\boldsymbol{G}(t)$. Here, φ denotes random number between 0 and 1. $\boldsymbol{P}_n(t)$ represents the local best position of the n-th particle in the t-th iteration. $\boldsymbol{G}(t)$ indicates the global best position of all the particles in the t-th iteration. The position of each particle is updated by

$$\begin{cases} \boldsymbol{X}_n(t+1) = \boldsymbol{P} + \beta \left|\boldsymbol{C}(t) - \boldsymbol{X}_n(t)\right| \cdot \ln(1/u), & u > 0.5 \\ \boldsymbol{X}_n(t+1) = \boldsymbol{P} - \beta \left|\boldsymbol{C}(t) - \boldsymbol{X}_n(t)\right| \cdot \ln(1/u), & u \le 0.5, \end{cases} \tag{83}$$

where u represents random number between 0 and 1. β denotes the contraction - expansion coefficient, in the t-th iteration, its value can be calculated by

$$\beta = 0.5\frac{T-t}{T} + 0.5 = 1 - \frac{t}{2T}. \tag{84}$$

Algorithm 1 Resource Allocation Strategy of Maximizing the Energy Efficiency Based on QPSO

1: Initialize N, T, and $\boldsymbol{X}_n(1)$ $(n = 1, 2, \cdots, N)$.
2: Set $\boldsymbol{P}_n(1) = \boldsymbol{X}_n(1)$. Find a best position from $\boldsymbol{P}_n(1)$ $(n = 1, 2, \cdots, N)$ as $\boldsymbol{G}(1)$.
3: Set $t = 1$ and $n = 1$.
4: **while** $t \leq T$ **do**
5: **while** $n \leq N$ **do**
6: Calculate $\boldsymbol{P}$.
7: Calculate β based on (84).
8: Calculate $\boldsymbol{C}(t)$ based on (85).
9: Update the position of the particle based on (83).
10: Compare $F[\boldsymbol{X}_n(t+1)]$ and $F[\boldsymbol{P}_n(t)]$. Then, give the higher value to $\boldsymbol{P}_n(t+1)$.
11: Compare $F[\boldsymbol{P}_n(t+1)]$ and $F[\boldsymbol{G}(t)]$. Then, give the higher value to $\boldsymbol{G}(t+1)$.
12: **end while**
13: **end while**
14: Output the function value of the global best position.

Moreover, in order to control the position of the particle convergence in probability to the attractor, the mean best position $\boldsymbol{C}(t)$ is defined and obtained by

$$\boldsymbol{C}(t) = \frac{1}{N}\sum_{n=1}^{N}\boldsymbol{P}_n(t). \tag{85}$$

For the maximization problem in (73), the greater the objective function value is, the better the corresponding fitness value is. Therefore, according to the fitness function in (76), the local best position of the n-th particle can be obtained by

$$\boldsymbol{P}_n(t+1) = \begin{cases} \boldsymbol{X}_n(t+1), & F[\boldsymbol{X}_n(t+1)] > F[\boldsymbol{P}_n(t)] \\ \boldsymbol{P}_n(t), & F[\boldsymbol{X}_n(t+1)] \leq F[\boldsymbol{P}_n(t)]. \end{cases} \tag{86}$$

The global best position of the group is determined by the following equation

$$\begin{cases} \tau = \arg\max\limits_{1\leq n\leq N}\{F[\boldsymbol{P}_n(t)]\} \\ \boldsymbol{G}(t) = \boldsymbol{P}_\tau(t). \end{cases} \tag{87}$$

In order to clearly illustrate the resource allocation strategy based on QPSO, its detailed steps are shown in **Algorithm 1**.

C. COMPLEXITY ANALYSIS

In this subsection, the computational complexity of **Algorithm 1** is analyzed. The computational complexity is mainly from Line 2 to Line 14. Line 2, which is executed once, has a time complexity of $O(1)$, The specific elapsed time will depend on the function operations in (76) and (87). The time complexity of Line 3 is also $O(1)$. Lines 4-13 consist of two loops: an outer loop and an inner loop. The inner loop given by Lines 6-11 is executed TN times. The time complexity of Lines 4-13 is $O(TN)$. The specific elapsed time will depend on the function operations in (85), (83), and (76). For Line 14, it is also executed once with time complexity $O(1)$. Consequently, the total time complexity of **Algorithm 1** is $O(TN)$.

IV. SIMULATION RESULTS AND DISCUSSIONS

In this section, the performance of the proposed resource allocation strategy is analyzed by simulations. First, the related parameter settings are given. Then, performance comparisons are shown.

A. PARAMETER SETTINGS

In this subsection, the related parameter settings of energy harvesting and resource allocation strategy are given. Assuming that there are two TV towers and two BSs (i.e., TV 1, TV 2, BS 1, and BS 2) as ambient RF sources, their corresponding coordinates at a coordinate system in unit of meter (m) are (1800,1800), (−1800,1800), (160,−160) and (−120,−120). The transmitting power of TV tower, BS 1 and BS 2 are 30 kW, 80 W [41] and 40 W [41]. Their transmitting antenna gains are 15 dBi, 18 dBi [42] and 18 dBi. The frequency of TV 1, TV 2, BS 1 and BS 2 are 558 MHz, 566 MHz, 850 MHz and 1850 MHz. All the users are distributed randomly in a square region, the coordinates of its four vertices are (15,15), (−15,15), (15,−15), (−15,−15), and user's receiving antenna gain is 0 dBi. Moreover, $c = 3 \times 10^8$ m/s, $a_i = 1500$, $b_i = 0.0014$ [43], $M_i = 20$ mW [43], $T_{\mathrm{RF}} = 2000$ s [44]. Besides, the related parameters of MI energy harvesting are set as $\mu = 4\pi \times 10^{-7}$ H/m [21], $N_k^{\mathrm{t}} = N_i^{\mathrm{r}} = 20$, $a_k^{\mathrm{t}} = a_i^{\mathrm{r}} = 0.03$ m, $d_{i,k}^{\mathrm{coil}} \in (0.06, 0.08)$ m, $R_0 = 0.01$ Ω/m [21], $f = 10$ MHz [21], $U_k^{\mathrm{S}} = 3.7$ V, $T_{\mathrm{MI}} = 10$ s. For the resource allocation strategy, the main parameters are set as follows, $P^{\mathrm{D}} = 10$ mW [31], $E^{\mathrm{C}} = 0.06\ \mu$J, $P^{\mathrm{B}} = 1$ W [31], $T^{\mathrm{O}} = 71.4\ \mu$s [45], $C_w \in (500, 1500)$ cycles/bit [34], $D_i^{\mathrm{CL}}, D_s^{\mathrm{TL}}, D_{i,k}^{\mathrm{CA}}, D_{s,q}^{\mathrm{TA}}, D_k^{\mathrm{CL}}, D_q^{\mathrm{TL}} \in (0.1, 0.2)$ Mbits, $D_i^{\mathrm{CT}}, D_s^{\mathrm{TT}} \in (3, 4)$ Mbits [34], $f_s^{\mathrm{TE}} = 2.5$ GHz, $f_i^{\mathrm{CL}}, f_s^{\mathrm{TL}}, f_k^{\mathrm{CL}}, f_q^{\mathrm{TL}} \in (0.5, 2)$ GHz [34], $F^{\mathrm{E}} = 80$ GHz, $P^{\mathrm{CIR}} = 0.3$ W [44], $B_1 = B_2 = B_3 = B_4 = 10$ MHz [33], $P^{\mathrm{R}} = 0.2$ W [34], $P^{\mathrm{E}} = 30$ dBm, $N_0 = 10^{-9}$ W [33], $\delta = 1\ \mu$J/bit, $T^{\mathrm{D}} = 10$ s, $\sigma = 1.5$ [44].

B. AMOUNT OF ENERGY HARVESTED BY DIFFERENT USERS

The amount of energy harvested by different users from different ambient RF sources and MI-based wireless reverse charging is presented in Table 1. In this simulation, the number of users is set as 5. We can observe that the users can harvest more energy from TV tower than the BS. That is because the TV tower has larger transmitting power. Moreover, the users can harvest much more energy from nearby other multiple users through MI-based wireless reverse charging technology than from ambient RF sources in a short time. The reason is that the wireless energy transfer technology based on MI has much higher receiving power and can obtain more energy in a short time. For the energy harvested by the MI-based wireless reverse charging, we can see that the energy obtained by User 2 and User 4 is more than that

TABLE 1. Energy harvested by different users from different ambient RF sources and MI-based wireless reverse charging (unit: J).

Users	User 1	User 2	User 3	User 4	User 5
BS 1	0.4641	0.5414	0.6162	0.5365	0.5638
BS 2	0.1167	0.0944	0.0811	0.0953	0.0895
TV Tower 1	2.0571	2.0442	2.0498	2.0378	2.0629
TV Tower 2	1.9741	1.9867	1.9812	1.9933	1.9688
MI Charging	15.1472	33.3984	17.0903	32.4057	17.2495
Total Energy	19.7592	38.0651	21.8186	37.0686	21.9345

obtained by User 1, User 3, and User 5. That is because the number of nearby other users that can provide charging for User 2 and User 4 is greater than that of User 1, User 3, and User 5.

C. PERFORMANCE COMPARISONS

In this subsection, some simulation results are discussed to analyze the performance of the proposed resource allocation strategy. In the simulation, the performance is analyzed by comparing different strategies. These comparison strategies are described as follows.

1) Local-Edge-Auxiliary-Transfer represents the proposed strategy. The computing tasks can be performed by local computation, edge computation, auxiliary computation, and transferring to nearby MEC server with sufficient resources for edge computation.

2) Local-Edge-Auxiliary indicates the first benchmark strategy. The computing tasks can be performed by local computation, edge computation, and auxiliary computation.

3) Local-Edge denotes the second benchmark strategy. The computing tasks can be performed by local computation and edge computation.

4) Local-Auxiliary is the third benchmark strategy. The computing tasks can be performed by local computation and auxiliary computation.

5) Local-Only refers to the fourth benchmark strategy. The computing tasks only performed by local computation.

Fig. 3 shows the convergence of the QPSO algorithm between the energy efficiency and the number of iterations under different numbers of particles. In the simulation, the number of CUs and TUs are 30 and 10. It can be found that the energy efficiency goes up as the number of iterations increases. When the number of iterations reaches 500, the energy efficiency tends to converge. Moreover, the energy efficiency increases gradually with the growth of the number of particles from 4 to 30, and the rising tendency is not obvious when the number of particles is 20 and 30. The reason is that a result closer to the optimal solution can be obtained through searching from more particles. Therefore, we set the numbers of iterations and particles as 500 and 30 in the following simulations.

Fig. 4 illustrates the time complexity of the QPSO algorithm under different numbers of particles from the perspective of average CPU time. We can find that the average CPU time rises up with the growth of the number of iterations. Furthermore, the average CPU time increases as the number

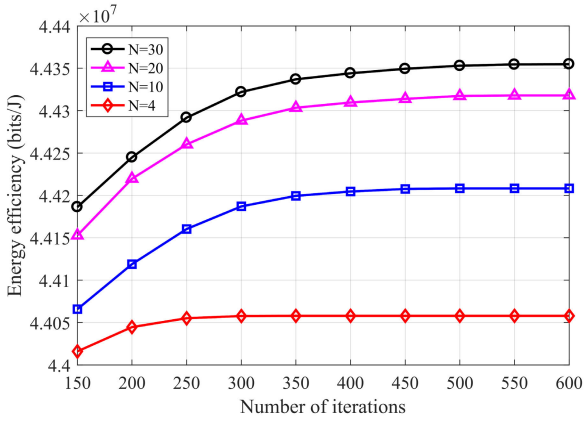

FIGURE 3. Relationship between energy efficiency and number of iterations.

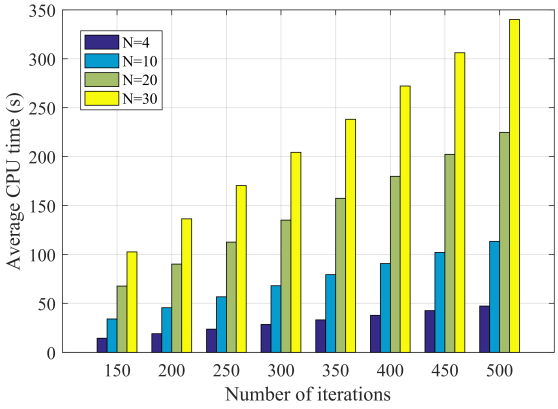

FIGURE 4. Relationship between average CPU time and number of iterations.

of particles increases. That is because when searching in a solution space, the more particles there are, the longer the search time will be.

Fig. 5 depicts the relationship between the energy efficiency and the number of common users under the five different strategies. It can be seen that the energy efficiency increases with the growth of the number of common users. The reason is that each common user consumes some energy to perform its computing tasks. The more common users are, the more total energy consumption and the amount of total computation data are. The growth rate of the total energy consumption is smaller than that of the amount of total computation data. Moreover, compared with the other four strategies, the Local-Edge-Auxiliary-Transfer strategy has the highest energy efficiency. That is because this strategy is the integration of the other four strategies. All the users can perform local computation, auxiliary computation, and edge computation. Meanwhile, transfer users can transfer their computing tasks to nearby MEC server for edge computation through establishing D2D relay links, which greatly improves the total energy efficiency.

Fig. 6 presents the relationship between the energy efficiency and the size of bandwidth under the five different

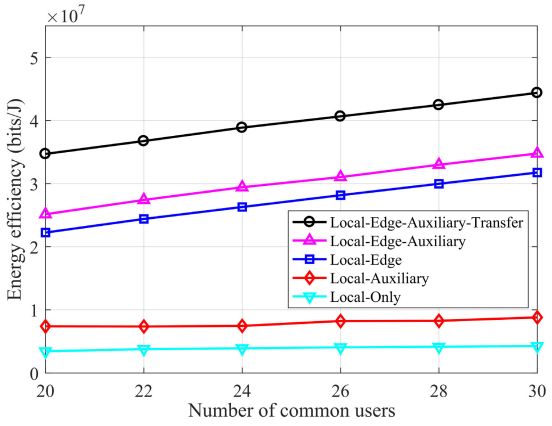

FIGURE 5. Relationship between energy efficiency and number of common users.

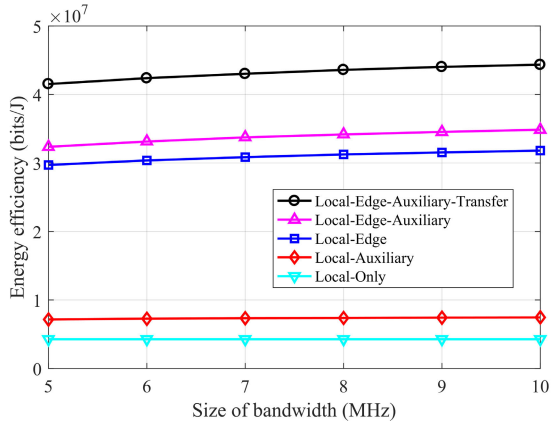

FIGURE 6. Relationship between energy efficiency and size of bandwidth.

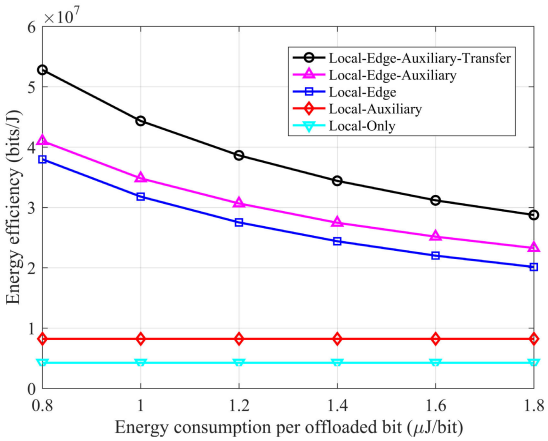

FIGURE 7. Relationship between energy efficiency and energy consumption per offloaded bit.

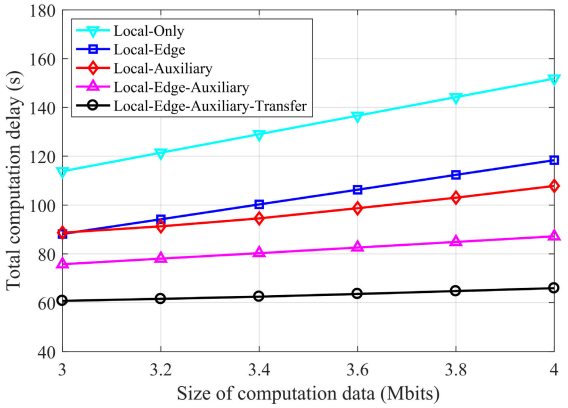

FIGURE 8. Relationship between total computation delay and size of computation data.

strategies. We can observe in the first four strategies that the number of energy efficiency goes up with the growth of the size of bandwidth, while Local-Only strategy keeps invariant. That is because the throughput rises up gradually as the size of bandwidth increases. Therefore, the transmission duration of computation data can be reduced gradually. As a result, the energy consumption of mobile users for edge computation and auxiliary computation can be reduced.

Fig. 7 depicts the relationship between the energy efficiency and the energy consumption per offloaded bit under the five different strategies. It can be observed that the energy efficiency of the first three strategies descend with the growth of the number of energy consumption per offloaded bit. The reason is that mobile users can offload their computing tasks to MEC server for edge computation. The greater the energy required by MEC server to perform per offloaded bit data, the greater the total energy required for edge computation. Meanwhile, the Local-Auxiliary and Local-Only strategies do not offload computing tasks to MEC server for edge computation. Therefore, they are not affected by the variation in energy consumption per offloaded bit.

Fig. 8 presents the relationship between the total computation delay and the size of computation data under the five different strategies. We can see that the total computation delay of Local-Edge-Auxiliary-Transfer is the shortest, while Local-Only is the longest. For the Local-Edge and Local-Auxiliary strategies, when the size of computation data is small, the total computation delay of Local-Edge is shorter than that of Local-Auxiliary, and Local-Edge is longer otherwise. That is because when the size of computation data is small, the computing resources of MEC server are sufficient to perform, and the computing speed of edge computation is faster than that of auxiliary computation. However, the computing resources of MEC server tend to be saturated as the size of computation data increases. Therefore, the computing speed is gradually smaller than that of auxiliary computation.

Fig. 9 shows the relationship between the energy efficiency and the transmitting power of BS under the different algorithm and different conversion coefficient value. We can find that the energy efficiency decreases with the growth of the transmitting power of BS. The reason is that the downlink throughput rises up gradually as the transmitting power of BS grows. However, the growth rate of transmitting power of BS is higher than that of the downlink throughput. As a result, the total energy consumption increases with the growth

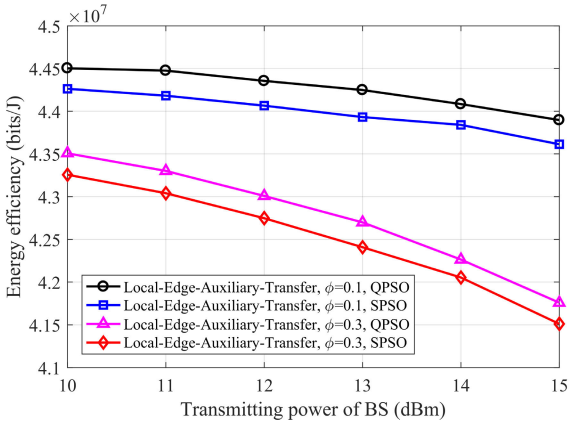

FIGURE 9. Relationship between energy efficiency and transmitting power of BS.

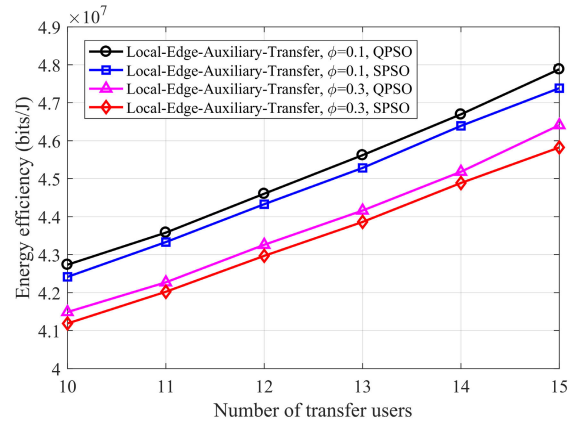

FIGURE 10. Relationship between energy efficiency and number of transfer users.

of the transmitting power of BS. Moreover, the QPSO algorithm outperforms SPSO algorithm. That is because the SPSO algorithm only can obtain local suboptimal solution. The global suboptimal solution can be obtained by using QPSO algorithm.

Fig. 10 illustrates the relationship between the energy efficiency and the number of transfer users under the different algorithm and different conversion coefficient value. It can be observed that the energy efficiency goes up as the number of transfer users increases. The reason is that the growth rate of the total energy consumption of transfer users is smaller than that of the amount of total computation data. Furthermore, we can see that the energy efficiency descends with the growth of conversion coefficient value from 0.1 to 0.3. That is because the greater the conversion coefficient value, the larger the computation result outputs. Consequently, the longer the transmission time required and the more energy is consumed.

V. CONCLUSION

In this paper, aiming at the problem of the limited battery capacity and computing capability of mobile users, a resource allocation strategy in D2D-assisted edge computing system with hybrid energy harvesting has been proposed to maximize the energy efficiency. By leveraging MI-based wireless reverse charging technology, the mobile user can supplement energy from nearby other users when the energy harvested from ambient RF sources is insufficient. Moreover, except for local computation, mobile user can also offload computing tasks to MEC server for edge computation, and to nearby idle users for auxiliary computation through establishing D2D communication links. If the computing resources of MEC server under BS 1 have reached saturation, the remaining users can establish D2D relay links with RU to transfer their computing tasks to nearby MEC server with sufficient resources under BS 2 for edge computation. The formulated resource allocation problem is an MINLP problem. The QPSO algorithm is adopted to obtain the suboptimal solution. Simulation results have shown that the proposed strategy is superior to other benchmark strategies, and QPSO algorithm can obtain higher energy efficiency than SPSO algorithm. We have made some assumptions in this work. First, we assume that there is only one RU to transfer computing tasks of TUs. In the future work, multiple RUs can be considered to transfer computing tasks to further improve the system performance. Second, instead of D2D relay link with limited distance, we will further consider smart device with wider coverage as relay node to broaden the relay distance. Third, only the SPSO algorithm is compared in this paper, we shall also take more heuristic algorithms into consideration for comparison.

ACKNOWLEDGMENT

This article was presented in part at the 2020 IEEE 91st Vehicular Technology Conference (VTC2020-Spring) [1], Antwerp, Belgium, May 2020.

REFERENCES

[1] J. Chen, Y. Zhao, Z. Xu, and H. Zheng, "Resource allocation strategy for mobile edge computing system with hybrid energy harvesting," in *Proc. IEEE 91st Veh. Technol. Conf. (VTC-Spring)*, May 2020, pp. 1–6.
[2] X. Lu, P. Wang, D. Niyato, D. I. Kim, and Z. Han, "Wireless networks with RF energy harvesting: A contemporary survey," *IEEE Commun. Surveys Tuts.*, vol. 17, no. 2, pp. 757–789, 2nd Quart., 2015.
[3] S. Abolfazli, Z. Sanaei, E. Ahmed, A. Gani, and R. Buyya, "Cloud-based augmentation for mobile devices: Motivation, taxonomies, and open challenges," *IEEE Commun. Surveys Tuts.*, vol. 16, no. 1, pp. 337–368, 1st Quart., 2014.
[4] S. Wang, X. Zhang, Y. Zhang, L. Wang, J. Yang, and W. Wang, "A survey on mobile edge networks: Convergence of computing, caching and communications," *IEEE Access*, vol. 5, pp. 6757–6779, Jun. 2017.
[5] Y. Mao, J. Zhang, and K. B. Letaief, "Dynamic computation offloading for mobile-edge computing with energy harvesting devices," *IEEE J. Sel. Areas Commun.*, vol. 34, no. 12, pp. 3590–3605, Dec. 2016.
[6] P. Mach and Z. Becvar, "Mobile edge computing: A survey on architecture and computation offloading," *IEEE Commun. Surveys Tuts.*, vol. 19, no. 3, pp. 1628–1656, 3rd Quart., 2017.
[7] N. Abbas, Y. Zhang, A. Taherkordi, and T. Skeie, "Mobile edge computing: A survey," *IEEE Internet Things J.*, vol. 5, no. 1, pp. 450–465, Feb. 2018.
[8] 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.
[9] A. Mahmood, A. Ahmed, M. Naeem, and Y. Hong, "Partial offloading in energy harvested mobile edge computing: A direct search approach," *IEEE Access*, vol. 8, pp. 36757–36763, Mar. 2020.

[10] F. Wang, J. Xu, and S. Cui, "Optimal energy allocation and task offloading policy for wireless powered mobile edge computing systems," *IEEE Trans. Wireless Commun.*, vol. 19, no. 4, pp. 2443–2459, Apr. 2020.
[11] M. Min, L. Xiao, Y. Chen, P. Cheng, D. Wu, and W. Zhuang, "Learning-based computation offloading for IoT devices with energy harvesting," *IEEE Trans. Veh. Technol.*, vol. 68, no. 2, pp. 1930–1941, Feb. 2019.
[12] B. Liu, W. Li, Y. Ma, J. Wang, and G. Lu, "Wireless powered cognitive-based mobile edge computing with imperfect spectrum sensing," *IEEE Access*, vol. 7, pp. 80431–80442, Jul. 2019.
[13] W. Zhang, Y. Wen, K. Guan, D. Kilper, H. Luo, and D. O. Wu, "Energy-optimal mobile cloud computing under stochastic wireless channel," *IEEE Trans. Wireless Commun.*, vol. 12, no. 9, pp. 4569–4581, Sep. 2013.
[14] C. You, K. Huang, and H. Chae, "Energy efficient mobile cloud computing powered by wireless energy transfer," *IEEE J. Sel. Areas Commun.*, vol. 34, no. 5, pp. 1757–1771, May 2016.
[15] 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.
[16] J. Ren, G. Yu, Y. Cai, and Y. He, "Latency optimization for resource allocation in mobile-edge computation offloading," *IEEE Trans. Wireless Commun.*, vol. 17, no. 8, pp. 5506–5519, Aug. 2018.
[17] X. Diao, J. Zheng, Y. Wu, and Y. Cai, "Joint computing resource, power, and channel allocations for D2D-assisted and NOMA-based mobile edge computing," *IEEE Access*, vol. 7, pp. 9243–9257, Jan. 2019.
[18] H. Xing, L. Liu, J. Xu, and A. Nallanathan, "Joint task assignment and resource allocation for D2D-enabled mobile-edge computing," *IEEE Trans. Commun.*, vol. 67, no. 6, pp. 4193–4207, Jun. 2019.
[19] J. Wen, C. Ren, and A. K. Sangaiah, "Energy-efficient device-to-device edge computing network: An approach offloading both traffic and computation," *IEEE Commun. Mag.*, vol. 56, no. 9, pp. 96–102, Sep. 2018.
[20] Z. Sun, I. F. Akyildiz, S. Kisseleff, and W. Gerstacker, "Increasing the capacity of magnetic induction communications in RF-challenged environments," *IEEE Trans. Commun.*, vol. 61, no. 9, pp. 3943–3952, Sep. 2013.
[21] Z. Sun and I. F. Akyildiz, "Magnetic induction communications for wireless underground sensor networks," *IEEE Trans. Antennas Propag.*, vol. 58, no. 7, pp. 2426–2435, Jul. 2010.
[22] X. Tan, Z. Sun, and I. F. Akyildiz, "Wireless underground sensor networks: MI-based communication systems for underground applications.," *IEEE Antennas Propag. Mag.*, vol. 57, no. 4, pp. 74–87, Aug. 2015.
[23] K. Doppler, M. Rinne, C. Wijting, C. Ribeiro, and K. Hugl, "Device-to-device communication as an underlay to LTE-advanced networks," *IEEE Commun. Mag.*, vol. 47, no. 12, pp. 42–49, Dec. 2009.
[24] R. Wang, J. Yan, D. Wu, H. Wang, and Q. Yang, "Knowledge-centric edge computing based on virtualized D2D communication systems," *IEEE Commun. Mag.*, vol. 56, no. 5, pp. 32–38, May 2018.
[25] H. J. Visser and R. J. M. Vullers, "RF energy harvesting and transport for wireless sensor network applications: Principles and requirements," *Proc. IEEE*, vol. 101, no. 6, pp. 1410–1423, Jun. 2013.
[26] E. Boshkovska, D. W. K. Ng, N. Zlatanov, and R. Schober, "Practical non-linear energy harvesting model and resource allocation for SWIPT systems," *IEEE Commun. Lett.*, vol. 19, no. 12, pp. 2282–2285, Dec. 2015.
[27] B. Clerckx, R. Zhang, R. Schober, D. W. K. Ng, D. I. Kim, and H. V. Poor, "Fundamentals of wireless information and power transfer: From RF energy harvester models to signal and system designs," *IEEE J. Sel. Areas Commun.*, vol. 37, no. 1, pp. 4–33, Jan. 2019.
[28] Y. Li, S. Wang, C. Jin, Y. Zhang, and T. Jiang, "A survey of underwater magnetic induction communications: Fundamental issues, recent advances, and challenges," *IEEE Commun. Surveys Tuts.*, vol. 21, no. 3, pp. 2466–2487, 3rd Quart., 2019.
[29] M. C. Domingo, "Magnetic induction for underwater wireless communication networks," *IEEE Trans. Antennas Propag.*, vol. 60, no. 6, pp. 2929–2939, Jun. 2012.
[30] P. Zhao, H. Tian, C. Qin, and G. Nie, "Energy-saving offloading by jointly allocating radio and computational resources for mobile edge computing," *IEEE Access*, vol. 5, pp. 11255–11268, Jul. 2017.
[31] C.-F. Liu, M. Maso, S. Lakshminarayana, C.-H. Lee, and T. Q. S. Quek, "Simultaneous wireless information and power transfer under different CSI acquisition schemes," *IEEE Trans. Wireless Commun.*, vol. 14, no. 4, pp. 1911–1926, Apr. 2015.
[32] *Physical Channels Modulation*, document TS 36.211 version 10.3.0, 3GPP, Sep. 2011.
[33] 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.
[34] Y. He, J. Ren, G. Yu, and Y. Cai, "D2D communications meet mobile edge computing for enhanced computation capacity in cellular networks," *IEEE Trans. Wireless Commun.*, vol. 18, no. 3, pp. 1750–1763, Mar. 2019.
[35] Z. Hadzi-Velkov, N. Zlatanov, T. Q. Duong, and R. Schober, "Rate maximization of decode-and-forward relaying systems with RF energy harvesting," *IEEE Commun. Lett.*, vol. 19, no. 12, pp. 2290–2293, Dec. 2015.
[36] Z. Zhou, C. Zhang, J. Wang, B. Gu, S. Mumtaz, J. Rodriguez, and X. Zhao, "Energy-efficient resource allocation for energy harvesting-based cognitive machine-to-machine communications," *IEEE Trans. Cognit. Commun. Netw.*, vol. 5, no. 3, pp. 595–607, Sep. 2019.
[37] S. Mumtaz, K. M. Saidul Huq, J. Rodriguez, and V. Frascolla, "Energy-efficient interference management in LTE-D2D communication," *IET Signal Process.*, vol. 10, no. 3, pp. 197–202, May 2016.
[38] J. Sun, W. Xu, and B. Feng, "A global search strategy of quantum-behaved particle swarm optimization," in *Proc. IEEE Conf. Cybern. Intell. Syst.*, Dec. 2004, pp. 111–116.
[39] J. Kennedy and R. Eberhart, "Particle swarm optimization," in *Proc. IEEE Int. Conf. Neural Netw.*, Nov. 1995, pp. 1942–1948.
[40] Y. Shi and R. Eberhart, "A modified particle swarm optimizer," in *Proc. IEEE Int. Conf. Evol. Comput. IEEE World Congr. Comput. Intell.*, May 1998, pp. 69–73.
[41] *Radio Transmission Reception*, document GSM 05.05 Version 5.2.0, ETSI, Jul. 1996.
[42] *Background for Radio Frequency (RF) Requirements*, document GSM 05.50 version 6.1.0, 3GPP, Apr. 2000.
[43] E. Boshkovska, R. Morsi, D. W. K. Ng, and R. Schober, "Power allocation and scheduling for SWIPT systems with non-linear energy harvesting model," in *Proc. IEEE Int. Conf. Commun. (ICC)*, May 2016, pp. 22–27.
[44] Y. Zhao, V. C. M. Leung, C. Zhu, H. Gao, Z. Chen, and H. Ji, "Energy-efficient sub-carrier and power allocation in cloud-based cellular network with ambient RF energy harvesting," *IEEE Access*, vol. 5, pp. 1340–1352, Mar. 2017.
[45] H.-W. Liang, W.-H. Chung, and S.-Y. Kuo, "FDD-RT: A simple CSI acquisition technique via channel reciprocity for FDD massive MIMO downlink," *IEEE Syst. J.*, vol. 12, no. 1, pp. 714–724, Mar. 2018.

JIAFA CHEN received the B.Eng. degree in communication engineering from Putian University, China, in 2017. He is currently pursuing the master's degree with the College of Physics and Information Engineering, Fuzhou University, China. His research interests include energy harvesting communications, mobile edge computing, and radio resource management.

YISHENG ZHAO (Member, IEEE) received the B.Eng. degree in communication engineering from Anhui Normal University, China, in 2006, the M.Eng. degree in communication and information system from the Chongqing University of Posts and Telecommunications, China, in 2010, and the Ph.D. degree in communication and information system from the Beijing University of Posts and Telecommunications, China, in 2013. From April 2016 to April 2017, he was a Visiting Scholar with the Department of Electrical and Computer Engineering, The University of British Columbia, Canada. He is currently an Assistant Professor with the College of Physics and Information Engineering, Fuzhou University, China. His research interests include energy harvesting communications, mobile edge computing, radio resource management, and wireless communications and networks.

ZHIMENG XU received the B.Sc. degree in radio physics from Lanzhou University, China, in 2002, the M.Sc. degree in information and communication engineering from Xidian University, in 2005, and the Ph.D. degree in information and communication engineering from Fuzhou University, in 2013. From August 2011 to January 2012, he was a Visiting Scholar with the Department of Technology, University of Northern Iowa. From March 2016 to March 2017, he was a Postdoctoral Fellow with the Department of Electrical and Computer Engineering, Dalhousie University, Canada. His research interests include ultra-wideband technologies, wireless sensing technologies, and wireless information and power transfer technologies.

HAIFENG ZHENG (Member, IEEE) received the B.Eng. and M.S. degrees in communication engineering from Fuzhou University, Fuzhou, China, and the Ph.D. degree in communication and information system from Shanghai Jiao Tong University, Shanghai, China. From October 2015 to September 2016, he was a Visiting Scholar with the State University of New York at Buffalo. He is currently a Professor with the College of Physics and Information Engineering, Fuzhou University. His research interests include the intelligent Internet of Things, edge computing, traffic modeling, machine learning, and intelligent transportation systems.