

Research Article

Cluster Design and Optimization of SWIPT-Based MEC Networks with UAV Assistance

Xuefei E , Zhonggui Ma , and JunFeng Huang

School of Computer and Communication Engineering, University of Science and Technology Beijing, Beijing 100083, China

Correspondence should be addressed to Zhonggui Ma; zhongguima@ustb.edu.cn

Received 22 May 2021; Revised 10 July 2021; Accepted 2 September 2021; Published 3 November 2021

Academic Editor: Xiaobin Xu

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

In recent years, service isolation and service miniaturization have become very popular. The large services are dismantled into multiple low-cost and simple small services to improve the scalability and disaster tolerance of the entire services. A service network composed of unmanned aerial vehicles (UAVs) and MEC servers is proposed in this paper, which aims at decoupling multiple services of the SWIPT-MEC network. In this network, UAVs take charge of energy transmission and calculation task scheduling and MEC servers are focused on task calculation. To meet the resource requirements of the ground nodes (GNs) in the network, we designed a distributed iterative algorithm to solve the resource allocation decision problem of GNs and used the modified expert bat algorithm to complete the UAV's trajectory planning in a two-dimensional space. The results show that the algorithm can formulate a more fair resource allocation strategy, and its performance is improved by 7% compared with the traditional bat algorithm. In addition, the algorithm in this paper can also adapt to changes in task length and has a certain degree of stability.

1. Introduction

In the past few decades, the Internet of things (IoT) technology has attracted widespread attention from the academic and commercial circles [1, 2]. It is worth mentioning that the communication technology in IoT is one of the current research hotspots [3]. Generally, devices in IoT have low computing ability and energy storage. Most of them cannot perform high-intensity calculations without external auxiliary equipment [4]. Meanwhile, these devices do not have enough energy to undertake the massive data interaction requirements in the intelligent age [5].

Cisco estimates that in 2023, at least 8.7 billion communication devices are connected to the 5G networks [6]. In addition, the data transmission rate in the 5G networks can reach up to 10 Gb/s and the transmission success rate can be maintained at 99.99999% [7]. To support all devices to enjoy real-time and high-throughput services, researchers and engineers proposed a simultaneous wireless information and power transfer- (SWIPT-) based mobile edge computing (MEC) networks.

2. Related Work

MEC was proposed by the European Telecommunications Standards Institute (ETSI) to improve the performance of cloud computing services. It has higher quality of service (QoS), lower cost delay, and faster calculation speed [8, 9]. Compared with mobile cloud computing (MCC), the signal transmission distance between the MEC server and the access device is shorter, which can effectively reduce the transmission energy consumption of the devices [10]. Unlike the uplink, the SWIPT technology allows the networks to return the calculation result and energy to the device simultaneously via the downlink. Furthermore, this technology can make up for the problem of excessive attenuation in traditional long-distance wireless power transfer (WPT) [11]. There are currently two mainstream SWIPT transmission modes: time switching (TS) and power switching (PS). Devices that use the TS mode must have a time-division receiver and those that use PS the mode must equip a frequency-division receiver [12]. Although the energy transmitted by the MEC server is sufficient for devices close to the

base station (BS), it is unsatisfactory for devices that are always at the edge of the MEC service range. Due to insufficient power, the device will be unexpectedly out of service, which is fatal to human-embedded devices or safety detection devices. To make up for this shortcoming, some scholars proposed device-to-device (D2D) networks based on SWIPT [13–16]. This network allows devices with sufficient energy to transmit energy to insufficient ones. However, this method requires each device to be equipped with additional sending and receiving equipment, which causes unnecessary waste of funds. Therefore, it is necessary to design a network that can solve this problem in a low-cost way.

In recent years, more and more scholars have begun to use unmanned aerial vehicles (UAVs) to assist communication [17–19]. Its unique advantages of cheapness and flexibility enable the networks to provide better services to devices at a very low cost. UAVs are divided into two types: fixed-wing UAVs and rotary-wing UAVs. With its vertical takeoff and hovering functions, many scholars have integrated rotary-wing UAVs into their research [20–22]. At the same time, the variability and flexibility of the rotary-wing UAV's trajectory allows the networks to be applied to various scenarios with insufficient BSs (e.g., disaster areas, mountainous areas, and densely populated cities) [23]. Therefore, in this paper, we attempt to introduce a UAV into the SWIPT-based MEC networks of densely populated cities to improve the QoS of the networks and avoid waste of funds.

Compared to fixed BSs, UAVs have a higher probability of establishing better quality communication links with ground nodes (GNs): line-of-sight (LOS) links [24]. Reference [25] assumes that the link between the UAV and GNs is LOS links and the UAV is equipped with a distributed charging device, which can charge multiple GNs at the same time. Both LOS and non-LOS (NLOS) links are considered in [26]. Moreover, the authors assume that a UAV works in urban areas and proposes a hybrid offline-online method to maximize the transmission rate of drones. Reference [27] improves the fading model of the LOS link, so that the model can more accurately express the link in the environment with large obstacles.

The SWIPT-based MEC networks has two main services: computing services and energy services. In terms of computing services, a UAV can be used as a mobile MEC server to meet the computing needs of GNs. In [28], a rotary-wing UAV equipped with an MEC server is used to provide charging and computing services for GNs in a time-division multiple access (TDMA) mode. And the authors design the UAV flight trajectory to minimize the energy consumption of the UAV. Although reference [29] also regards a UAV as a mobile BS, it uses the OFDMA technology instead of the TDMA technology, so that each GN can simultaneously enjoy UAV services. Likewise, reference [30] uses the ant colony system (ACS) algorithm to design the trajectory of the rotary-wing UAV. It is undeniable that the service capability of the MEC server attached to the UAV is far from that installed on the base station. The idea of letting multiple UAVs cooperate to undertake the computing tasks of GNs was proposed by [31]. Due to not requiring BS, this method is indeed very suitable in the disaster area.

But this method will inevitably cause waste of BSs already installed in each cities.

In terms of energy services, a UAV can assist the SWIPT-based MEC networks by acting as a relay node. To overcome the huge fading problem with the target node, reference [32] proposes a method for a UAV to relay energy to bypass large obstacles. Reference [33] is to relay energy through the hovering UAV to reduce the overall energy consumption of the networks. However, this relay method will cause two consecutive attenuations, resulting in that the energy received by the GNs is much less than the energy emitted by the source. In [34–36], the self-collecting function is added to the UAV, so that it can supplement the energy that needs to be forwarded during the relay process, thereby indirectly increasing the energy received by the GNs. But this method is still based on the model of two consecutive attenuations and does not change the essence of the problem.

In addition, although many of those papers mentioned above have studied the flight trajectory of UAVs, most of them assume that GNs are stationary. This is obviously not in line with the reality. Therefore, the SWIPT-based MEC networks in areas where communication is in malfunction in the city need a better solution to solve the shortage of computing services and energy services.

A SWIPT-MEC service network is proposed in this paper that separates computing and energy services to solve the two service problems mentioned above. And GNs in this network are divided into three categories: vehicle-mounted equipment, human equipment, and household equipment, all of them moved randomly. We assume that GNs can sense the location of the UAV and actively initiate a connection. To solve the problem of time slot allocation and UAV flight trajectory, a distributed iterative algorithm and an expert bat algorithm are proposed. The solution can reuse a large number of MECs in the city, thereby saving improvement costs and making service points flexible. The specific contributions of this paper are as follows:

- (i) Compared to the traditional MEC networks, a service cluster network composed of multiple UAVs and one MEC has been proposed in this paper. In order to make the MEC server focus on completing computing tasks, the network allows UAVs to schedule computing tasks and formulate energy distribution strategies. In addition, MEC uses the orthogonal frequency-division multiple access (OFDMA) technology to allocate the spectrum to the UAV, allowing it to communicate with GNs in the time-division multiple access (TDMA) mode through this spectrum
- (ii) Since user information is frequently changed, if the networks apply a centralized algorithm, it will inevitably spend a lot of time to update the data. But the distributed algorithm cannot know global information. Therefore, this paper adopts a semidistributed algorithm and the UAV needs to forward environmental information in the area that it is responsible for. For each GN, the decision information of other GNs can be used as environmental information

- (iii) In order to plan the UAV path more accurately, we use the bat algorithm to dynamically select the best location point within the maximum range of motion. According to the model in this paper, an elite bat colony is introduced to further optimize the algorithm
- (iv) Through the analysis of the results, the distributed iterative algorithm proposed in this paper can maintain stable fairness after the environment is stable. At the same time, the bat algorithm proposed in this paper can improve the performance of the networks by 7% compared with the traditional algorithm. In addition, in the results part, the sensitivity of each parameter of the system is analyzed. The results show that the algorithm in this paper can adapt well to the change of task length, which enables the networks to better adapt to changes in the environment

The rest of the paper are structured as follows. In Section 3, the system model and problem formulation are discussed. In Section 4, the path planning algorithm for the UAV is designed. In Section 5, the distributed iterative algorithm is used to solve the time allocation problem of GNs. Numerical results and analysis are presented in Section 6. Finally, Section 7 makes a summary of the paper.

3. System Model and Problem Formulation

The SWIPT-based MEC networks in this paper within a single base station in areas of communication paralysis are shown in Figure 1. Its service range is $R^{\text{MEC}} = [1 \text{ km} \times 1 \text{ km}]$ [37], and divide R^{MEC} into \mathcal{M} parts; each part is assigned a UAV for cruise work, defined as R_m^{MEC} . This network consists of one MEC server, \mathcal{N} GNs, and \mathcal{M} UAVs. The BS's MEC server with multiple antennas is equipped at height h^{MEC} , while the working height of UAV is at H^{UAV} . The MEC server can simultaneously receive the GN calculation tasks relayed by each UAV and return the calculation results to the GNs through the UAV. UAVs and GNs are only equipped with a full-duplex (FD) antenna and communicate in the MEC networks by TDMA. At the beginning of each slot, GNs will actively send a connection request to the MEC server. After that, the communication address and some basic information in the GN request will be forwarded to the UAV by the MEC server. Finally, GNs will establish a communication link with the corresponding UAV.

The UAV in the networks can not only relay computing tasks but also charge GNs. We assume that the UAV is a mobile energy source so that the energy received by GNs is directly obtained from the energy source, rather than being relayed by the UAV. However, the portable energy of UAV is limited and they cannot work for a long time without charging. Therefore, this paper innovatively installs the UAV charging pile at the h_s^{UAV} of the BS and allows the UAV to have an autonomous navigation function. When the UAV senses that its power is insufficient, it will automatically design a trajectory to return to the charging pile and continue to work after replacing the battery. With the popu-

larization of 5G in the future, base stations will become denser [38] and UAV charging will become more and more convenient. Besides, the interference between different UAVs and between different GNs can be ignored [39]. The energy consumption of transmitters, receivers, and optimized calculations is also not considered [40].

3.1. Mobile Model of GNs and UAVs. We assume that the total working time of the networks is T and divide it into N^T time slots. The GNs on the ground are divided into three categories: fixed in position, moving with humans, and mounted on cars, expressed as GN^H , GN^H , and GN^C , respectively. GN moves continuously in the plane without interruption. To simplify the analysis, this article only samples the position of the GN at the beginning of the time slot. The maximum moving speed of the GN belonging to category GN^H is V_H and GN^C is V_C , and the remaining GNs are always stationary. Similarly, the maximum speed of the UAV is V_{max} . In the three-dimensional space of the SWIPT-based MEC networks, both GNs and UAVs perform a two-dimensional motion in their respective horizontal planes. The specific motion model is shown in Figure 2. At the end of each time slot, GNs and UAVs will randomly select a point in circle O as the starting position of the next time slot. This point is determined by two random numbers. The first is the angle θ with the horizontal axis X . The second is the length d of the movement.

3.2. Binary Computation Offloading Mode. The computational offloading strategy of GNs under the binary offloading mode will be discussed in detail in this section. In this mode, the task is not required to be split and has a wider scope of application. Each GNs in the network will make decisions based on its energy storage and uplink energy consumption. The specific decision plan is as follows:

$$b_n = \begin{cases} 0, & E_n^{LC} \leq E_n^R \text{ and } E_n^{LC} \leq E_n^{UL} \\ 1, & \text{else} \end{cases}, \quad (1)$$

where E_n^R is the GN remaining energy, E_n^{LC} is the energy consumption of local computation, and E_n^{UL} is the upload energy consumption. Let $B = \{b_n \mid b_n \in \{0, 1\} \forall n \in \mathcal{N}\}$ represents the offloading decision of each GN. When $b_n = 1$, GNs use local calculation. Otherwise, GNs will offload calculation to the MEC server. GNs select local calculation only when the local calculation energy consumption is lower than the transmission energy consumption and the local remaining energy.

3.3. Mobile Model of GNs and UAVs. There are both LOS attenuation and NLOS attenuation during the operation of the UAV. However, since the occurrence probability of NLOS is much lower than that of LOS [41, 42], the attenuation problem of NLOS can be ignored. Therefore, the channel attenuation between UAVs, MEC, and GNs can be expressed as [43] follows:

$$h_n^E(r) = k^L r^{\alpha^L}, \quad (2)$$

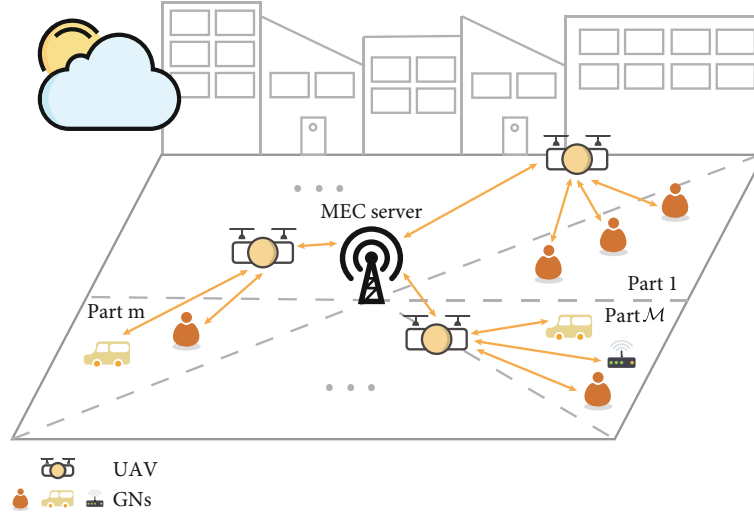


FIGURE 1: Illustration of SWIPT-based MEC networks with UAV assistance.

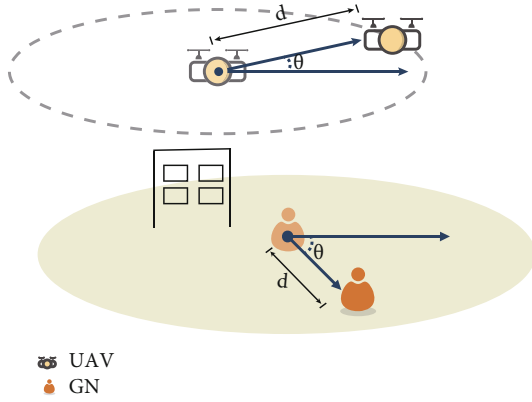


FIGURE 2: Illustration of the mobile model of GNs and UAVs.

where r is the distance between the UAV and the communication target. k^L is the intercepts of the LOS and α^L is the path loss exponents for LOS.

3.3.1. Energy Harvesting. Since the transmission time of the downlink is usually short, it is inefficient to use it to transmit energy. The uplink is more suitable for energy transmission because its transmission time is much longer than the one of the downlinks. Thus, the FD-SWIPT technology [44] is introduced to the networks so that GN can obtain energy from the UAV while offloading computing tasks. The energy collected by GNs can be defined as

$$E_n^{EH} = b_n \alpha_n T_n^{GN} P_m^{UAV} h_n^E(r), \quad (3)$$

where α_n is the energy conversion rate of the receiving equipment used by GNs and P_m^{UAV} represents the power of the transmitter equipped with the UAV. To simplify the model, the energy consumption of the receiving device is ignored in this network [45]. T_n^{GN} is the length of the time slot allocated to GN.

3.3.2. Local Computing. The GNs that select local calculation will no longer upload any calculation tasks to the MEC server, and all calculation tasks are completed locally. In order to prevent the locally calculated GNs from being exhausted, the UAV will also transfer energy to such GNs. Assume that the CPU of GNs needs X revolutions to calculate one bit and the task length is D bits. Then, the local computational energy consumption of each GN can be expressed as

$$E_n^{LC} = (1 - b_n) D_n X \kappa^{LC} F_n^{LC} C^2, \quad (4)$$

$$F_n^{LC} = \frac{D_n X N^T}{T},$$

where F_n^{LC} is the calculation frequency of local calculations and the meaning of κ^{LC} is the correlation coefficient of the CPU of GNs. We assume that the task can be undertaken by GNs.

3.3.3. Computing Offloading. In each time slot, the UAV will dynamically allocate time T_n^{GN} to the GN_n within its service range. Similarly, the UAV also needs to transfer the received tasks to the MEC server at the BS in this time slot. BS will use different antennas to connect to each UAV, so there is no need to worry about conflicts between UAVs. However, conflicts between GNs cannot be ignored, so we use a dynamic TDMA [46] scheme shown in Figure 3 to provide services for GNs. Also, we assume that the length of the returned calculation result is much smaller than the calculation task, so the downlink transmission time can be ignored. Then, we have.

$$C2 : \sum_{n \in \mathcal{N}} T_n^{GN} + T_m^{UAV} = \frac{T}{N^T}, \quad \forall m \in \mathcal{M}, \forall n \in \mathcal{R}_N^{MEC}, \quad (5)$$

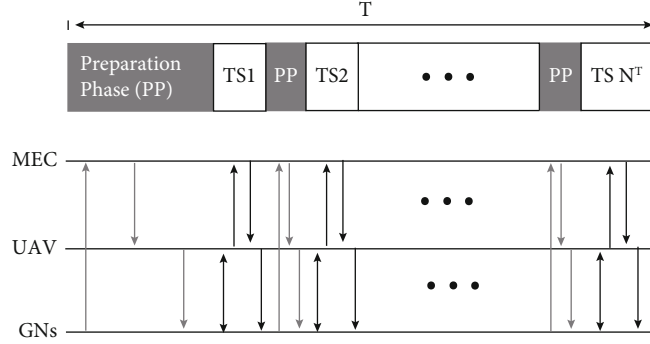


FIGURE 3: Communication interaction model of each device.

where T_m^{UAV} is the time required for the UAV to relay the computing task to the MEC server and can be denoted as

$$T_m^{\text{UAV}} = \frac{\sum_{n \in R_m^{\text{MEC}}} D_n}{\text{Blog}_2 \left(1 + \left(P_m^{\text{UAV}} h_n^E(r) / \sigma^2 \right) \right)}. \quad (6)$$

Here, σ^2 represents the power of the Gaussian white.

Thus, the transmission energy consumption of the UAV can be defined as

$$E_m^{\text{UAV}} = P_m^{\text{UAV}} T_m^{\text{UAV}}. \quad (7)$$

GNs in the computing offloading mode need to offload their computing tasks to the UAV and forward them to the MEC server for computing. In this process, the transmission rate of the GN transmission task is an identity, as shown below

$$\frac{D_n}{T_n^{\text{GN}}} = \text{Blog}_2 \left(1 + \left(P_n^{\text{UL}} h_n E(r) / \sigma^2 \right) \right), \quad (8)$$

where B represents the bandwidth of the channel occupied by the transmission task. After some simple mathematical transformations, the transmission power of GNs in the uplink can be written as

$$P_n^{\text{UL}} = \frac{\sigma^2 \left(2^{D_n / T_n^{\text{GN}} B} - 1 \right)}{h_n E(r)}. \quad (9)$$

Since the transmitter power of GNs in the networks is limited, P_n^{UL} must meet the following constraints

$$C3 : \frac{\sigma^2 \left(2^{D_n / T_n^{\text{GN}} B} - 1 \right)}{h_n E(r)} \leq P_n^{\text{max}}. \quad (10)$$

Here, P_n^{max} represents the maximum transmission power that the transmitter can withstand. C3 can also be written as

$$C3 : \frac{D_n}{\text{Blog}_2 \left(1 + \left(P_n^{\text{max}} h_n^E(r) / \sigma^2 \right) \right)} \leq T_n^{\text{GN}}. \quad (11)$$

According to (9), the energy consumption of GNs when transmitting calculation tasks can be written as

$$E_n^{\text{UL}} = \frac{b_n T_n^{\text{GN}} \sigma^2 \left(2^{D_n / T_n^{\text{GN}} B} - 1 \right)}{h_n E(r)}. \quad (12)$$

3.4. UAV Flight Trajectory Design. The flight trajectory design of the UAV should consider not only the overall situation of the current time instant but also the positions of the previous time. It must be ensured that the service period of each GN is approximate. A fair and high-quality service can ensure that the GNs served by it maintain sufficient power. We believe that the UAV can establish a long-term connection with MEC and GNs until it returns to the charging pile. The UAV can sense its own remaining power and return to the charging pile when the power is low. When the UAV is charging, another fully charged UAV will continue to complete the remaining work.

According to the survey in [47], we conclude that the flight energy consumption of the rotary-wing UAV used in this paper can be modeled as

$$E_m^V(V_t) = \frac{1}{2} M_m V_m^2, \quad (13)$$

where M_m means the weight of the UAV and V_m represents the UAV flight speed with the maximum value of V_m^{max} which can be expressed as follows:

$$C4 : V_t \leq V_m^{\text{max}}. \quad (14)$$

In addition, the position of the UAV cannot exceed R_m^{MEC} and the corresponding constraint is written as

$$C5 : 0 \leq \text{OUT}(t), \quad (15)$$

where $\text{OUT}(\cdot)$ is an out-of-bounds determination function. If the UAV is out of the bound, the result of the function is negative, otherwise, positive. This function can be easily implemented through programming, so there are no more explanations here. And the autonomous back-to-home charging function of the UAV can be realized by the following constraints

$$t = \left\lceil \frac{r^{\text{MEC}}}{V^{\text{max}}} \right\rceil, \quad (16)$$

$$\text{C6} : 0 \leq E_m^R - tE_m^V(V^{\text{max}}),$$

where $\lceil \cdot \rceil$ is the round-up function and r^{MEC} represents the distance between the UAV and the MEC server. t is the pre-judgment time; the UAV can judge whether it can fly back to the charging station after t . And E_m^R is the remaining energy of the UAV. Because tE_m^V increases monotonically when V increases, $tE_m^V(V^{\text{max}})$ is the maximum energy required for the UAV to return to the charging station.

3.5. Utility Function of SWIPT-Based MEC Networks. In this paper, we aim to maximize the remaining energy of GNs and UAV energy consumption in each time slot, meanwhile jointly optimizing the uplink time of GNs and the flight position of the UAV. We assume that the position of the UAV in each time slot is Z_m^{UAV} and the specific model of each time slot can be expressed as

$$\begin{aligned} \max_{Z_m^{\text{UAV}}, T_n^{\text{GN}}, b_n, n \in \mathcal{N}} & \sum (E_n^{\text{EH}} - E_n^{\text{LC}} - E_n^{\text{UL}}) - \sum_{m \in \mathcal{M}} (W_1 E_m^V(V_t) + W_2 E_m^{\text{UAV}}) \\ \text{s.t. C1} & : b_n \in \{0, 1\}, \quad \forall n \in \mathcal{N} \\ & \text{C2} \sim \text{C6}, \end{aligned} \quad (17)$$

where W_1 and W_2 are the weight coefficient used to maintain the same order of magnitude of each variable in the objective function. The position of the UAV is represented as $Z^{\text{UAV}} = \{Z_m^{\text{UAV}} \mid m \in \mathcal{M}\}$.

4. Algorithm Design for Path Planning

The model in this paper is composed of two kinds of agents. The GNs have no way to know the location of the UAV in the next time slot; it can only formulate a strategy based on the environment of the current time slot. However, the UAV needs to weigh the overall situation and find the optimal position of the next time slot. This section aims at designing an algorithm for the UAV, and the algorithm design of GNs will be discussed in the next section. According to the above analysis, the submodel required in this section can be written as

$$\begin{aligned} \max_{Z_m^{\text{UAV}}, T_n^{\text{GN}}, b_n, n \in \mathcal{N}} & \sum (E_n^{\text{EH}} - E_n^{\text{LC}} - E_n^{\text{UL}}) - \sum_{m \in \mathcal{M}} (W_1 E_m^V(V_t) + W_2 E_m^{\text{UAV}}) \\ \text{s.t. C2} & \sim \text{C6}. \end{aligned} \quad (18)$$

However, C2 and C3 in the model are both restrictive con-

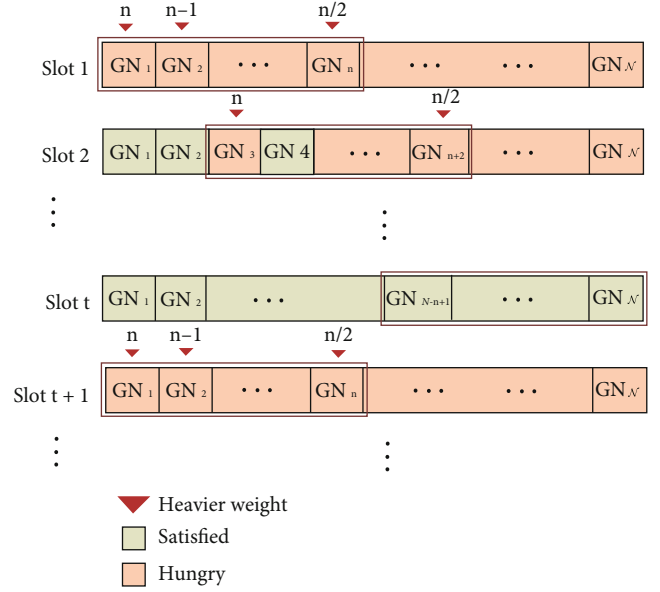


FIGURE 4: Communication interaction model of each device.

ditions for a single time slot. If the algorithm simultaneously considers the transition of the next time slot and the optimization of the next time slot, the complexity of the algorithm will greatly increase. In fact, in C2 and C3, only $h_n^E(r)$ is related to the position of the next UAV slot. After slightly modifying $h_n^E(r)$, the submodel in this section can be written as

$$\begin{aligned} \max_{Z_m^{\text{UAV}}} & W_3 \sum_{n \in \mathcal{N}} r_n - \sum_{m \in \mathcal{M}} (W_1 E_m^V(V_t) + W_2 E_m^{\text{UAV}}) \\ \text{s.t. C4} & \sim \text{C6}. \end{aligned} \quad (19)$$

The specific process will be elaborated in the next section. Furthermore, in the preparation phase of each time slot, MEC will sense the location of the GNs and send the GN list of each area to the corresponding UAV.

Since the position of GNs in this model is constantly changing, reinforcement learning or convex optimization theory cannot achieve good results. Therefore, we consider using the bat algorithm as the basis of the algorithm in this section. It can find the optimal solution while exploring the unknown environment. But this algorithm is not capable of considering multiple time slots jointly. In addition, path planning is also very important, which can help the UAV save flight energy. In order to solve those problems, we introduce the function of path planning and multislot joint consideration in the objective function part. The formula of the objective function can be expressed as (20).

The formulas before $H_n W_7 r$ are all converted from the submodels in this section, where H_n is the magnification weight coefficient. The heavier weight range is n , and the specific selection method is shown in Figure 4. For the left-most hunger GN, its weight H_n value is n . The subsequent GN's value of H_n decreases in turn (the H_n of the next hungry GN is $n-1$) until H_n reaches the minimum value or the edge of heavier weight. It is worth mentioning that since

```

Input  $J, J_c$ : maximum of iterations;
 $NUM^B/NUM^{CB}$ : Number of bats/child bats;
 $A^B/r^B$ : Sonic loudness/pulse of bat;
 $\alpha^A/\alpha^r$ : Loudness/pulse update factor;
1: Use criterion (20) to find the best bat in the initial environment;
2: for  $t < J$  do
3:   for each bat do
4:     Randomly move each bat;
5:     Generate a random number  $P$ ;
6:     if  $P > r^B$  then
7:       Randomly move this bat;
8:       if better than the original then
9:         Adopt current position;
10:      else
11:        Adopt original position;
12:      end if
13:    end if
14:    if better than the better than the optimal bat and  $P < A^B$  then
15:      Update optimal bat position;
16:       $A^B = \alpha^A A^B$ ;
17:       $r^B = (1 - e^{-\alpha^r t}) r^B$ ;
18:    end if
19:  end if
20:  Add  $NUM^{CB}$  child bats as the best bat copy;
21:  for  $t_c < J_c$  do
22:    for each child bats do
23:      Randomly move child bats;
24:      if better than the best then
25:        Update the optimal solution;
26:      end if
27:    end for
28:     $t_c + = 1$ 
29:  end for
30:   $t + = 1$ 
31: end for

```

ALGORITHM 1: Expert bat algorithm.

GNs move randomly, the satisfied GNs may appear after the hungry GN. In order to ensure that the requirements of each GN are met, the high weight range will not change when there are some satisfactory GNs inside.

$$\begin{aligned}
Obj_m^B = & W_3 \sum_{n \in \mathcal{N}} r_n - \sum_{m \in \mathcal{M}} (W_1 E_m^V(V_t) + W_2 E_m^{UAV}) \\
& + W_{4a} (V_t - V^{\min}) + W_{4b} (V^{\max} - V_t) \\
& + W_5 \text{OUT}(t) + W_6 (E_m R - t E_m^V(V^{\max})) + H_n W_7 r.
\end{aligned} \tag{20}$$

Different from the traditional bat algorithm, we try to reduce the overall number of iterations by adding efficient exploration in each iteration. The best bat in each iteration is regarded as an expert bat with self-intelligence. This expert bat can use the bat algorithm in a small area to find a better optimal position to indirectly optimize the movement direction of the nonexpert bat. The specific algorithm is shown in Algorithm 1. Although this algorithm is not fair in each time

slot, it is fair in average time because the path of the UAV is cyclically flying [1].

5. Algorithm Design for Time Slot Management

This section mainly solves the optimization problem of the time occupied by each GNs in each time slot. After separating the relevant content of this section from the main model, the following submodels can be obtained as follows

$$\begin{aligned}
\max_{b_n} \sum_{n \in \mathcal{N}} & \left(\underbrace{\max E_n^{EH} - \max E_n^{UL} - E_n^{LC}}_{P1} \right) \\
\text{s.t. } & C1, C2(P1), C3(P1).
\end{aligned} \tag{21}$$

The separated submodel is split into two parts, one part optimizes the time variable, denoted as $P1$, and the other

```

Input:  $\varepsilon$ : Tolerance error
 $T^G$ : Number of independent calculations
 $W_n^{C2a}/W_n^{C2b}$ : A copy of the auxiliary variable at
each GN
1: for  $T/N^T - T_m^{UAV} - T^{temp} > \varepsilon$  do
2:   • GNs
3:   Update  $W_n^{C2a}, W_n^{C2b}$  according to the received result;
4:   for  $i < T^G$  do
5:     for The change in  $T_n^{GN}$  is more than  $\varepsilon$  do
6:       for each GNs do
7:          $T_n^{GN} - = \Delta T_n^{GN}$ ;
8:       end for
9:     end for;
10:    Update  $W_n^{C2a}, W_n^{C2b}, W^{C3}$  with (28), (29), (30);
11:  end for
12:  Send  $W_n^{C2a}, W_n^{C2b}, T_n^{GN}$  to UAV
13:  • UAV
14:   $W^{C2a}/W^{C2b} =$  Add and average the  $W_n^{C2a}/W_n^{C2b}$  of all GNs in this part;
15:   $T^{temp} = \sum_{n \in \mathcal{N}} T_n^{GN}$ 
16:  Send all GNs'  $T_n^{GN}$  and  $W^{C2a}, W^{C2b}$  to each GN;
17:  end for

```

ALGORITHM 2: Centralized and distributed algorithm of UAV and GN collaboration.

TABLE 1: Simulation parameters.

Parameters	Value
UAV weight (W_m)	9.65 kg
Noise (σ^2)	10^{-13} W
UAV/MEC height (H^{MEC}, H^{UAV})	100/10 m
Bandwidth (B)	10^6 Hz
CPU correlation coefficient (k_0)	10^{-28}
Bit length of the task (D)	1.5 ~ 2 Mbit
Operational capability (X)	1000 cycles/bit
Maximum transmission power of GNs (P^{max})	1 W
Transmission power of UAV (P^{UAV})	10 W
Maximum speed (V^{max})	50 m/s

part optimizes the decision variable. But the $h_n^E(r)$ in C2 and C3 is related to the location of the UAV, which will make the bat algorithm reallocate time every time it evaluates the location of the bat. So, we consider separating the UAV position and time in P1 to reduce the complexity of the algorithm. After taking the logarithm of the objective function, it can be written as

$$\begin{aligned}
\log(E_n^{EH}) &= \log(\alpha_n T_n^{GN} P_m^{UAV}) + \log(h_n^E(r)), \\
-\log(E_n^{UL}) &= -\log\left(T_n^{GN} \sigma^2 \left(2^{D_n/T_n^{GN} B} - 1\right)\right) + \log(h_n^E(r)).
\end{aligned} \tag{22}$$

After removing formulas that have nothing to do with optimization variables, P1 can be written as

$$\begin{aligned}
\max_{T_n^{GN}} W^{P1} \alpha_n T_n^{GN} P_m^{UAV} - T_n^{GN} \sigma^2 \left(2^{D_n/T_n^{GN} B} - 1\right) \\
\text{s.t. C2, C3.}
\end{aligned} \tag{23}$$

The simple process of delogarithmization and merging the maximum function is omitted here. In order to keep the two items in the objective function at the same order of magnitude, we introduce an auxiliary variable W^{P1} . On the other hand, in the C2 constraint, only T^{UAV} is related to the UAV position. According to the maximum speed limit of the UAV, the r of $h_n^E(r)$ in T^{UAV} can be expressed as

$$r' = \sqrt{\left(\sqrt{r^2 - H_2^{UAV}} + V^{max} \frac{T}{NT}\right)^2 + H_2^{UAV}} \geq r. \tag{24}$$

Because $h_n^E(r)$ is inversely proportional to r and T^{UAV} is inversely proportional to $h_n^E(r)$, so, C2 can be shown as

$$\text{C2: } \sum_{n \in \mathcal{N}} T_n^{GN} + T_m^{UAV}(r') = \frac{T}{NT}, \quad \forall m \in \mathcal{M}, \forall n \in R_m^{MEC}. \tag{25}$$

According to $T_m^{UAV}(r') \geq T_m^{UAV}(r)$, it can be known that the spent time will not exceed the time specified by each

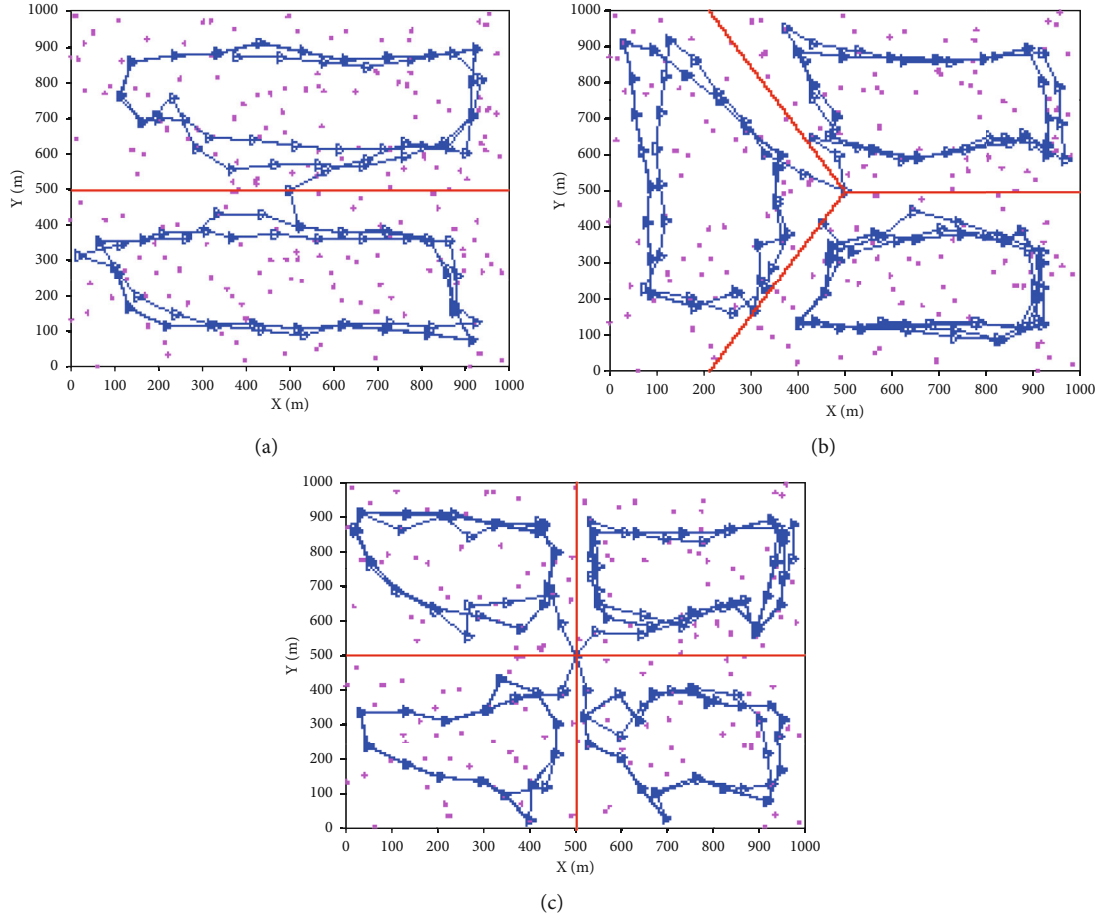


FIGURE 5: The trajectory of UAVs. (a) 2 UAVs; (b) 3 UAVs; (c) 4 UAVs.

endpoint, so the transformation will not affect the next time slot. Similarly, C3 can also be safely transformed into

$$C3 : \frac{D_n}{\text{Blog}_2\left(1 + \left(P_n^{\max} h_n^E(r')/\sigma^2\right)\right)} \leq T_n^{\text{GN}}. \quad (26)$$

Based on the above analysis, it can be simply concluded that the constraints of C3 after the transformation are more stringent than the original one. Besides, to facilitate subsequent processing, we decompose the transformed C2 in the submodel into two constraints.

$$\begin{aligned} C2a : \sum_{n \in \mathcal{N}} T_n^{\text{GN}} + T_m^{\text{UAV}}(r') &\leq \frac{T}{N^T}, \quad \forall m \in \mathcal{M}, \forall n \in R_m^{\text{MEC}}, \\ C2b : \sum_{n \in \mathcal{N}} T_n^{\text{GN}} + T_m^{\text{UAV}}(r') &\geq \frac{T}{N^T}, \quad \forall m \in \mathcal{M}, \forall n \in R_m^{\text{MEC}}. \end{aligned} \quad (27)$$

This section considers using the KKT algorithm as the basic algorithm, and the Lagrangian function of P1 can be

written as (21), where W^{C2a}, W^{C2b}, W^{C3} are the auxiliary variables introduced and the updated formulas are

$$W^{C2a+} = \nabla^{C2a} \left(\sum_{n \in \mathcal{N}} T_n^{\text{GN}} + T_m^{\text{UAV}}(r') - \frac{T}{N^T} \right), \quad (28)$$

$$W^{C2b+} = \nabla^{C2b} \left(\frac{T}{N^T} - \sum_{n \in \mathcal{N}} T_n^{\text{GN}} - T_m^{\text{UAV}}(r') \right), \quad (29)$$

$$W^{C3+} = \nabla^{C3} \left(\frac{D_n}{\text{Blog}_2\left(1 + \left(P_n^{\max} h_n^E(r')/\sigma^2\right)\right)} - T_n^{\text{GN}} \right). \quad (30)$$

Let $(L^{P1})' = 0$ can be obtained

$$\begin{aligned} (L^{P1})' &= T_n^{\text{GN}} B (\alpha_n P_m^{\text{UAV}} W^{P1} + \sigma^2 + W^{C3} \\ &\quad - W^{C2a+} + W^{C2b+}) \sigma^2 2^{D_n/T_n^{\text{GN}} B} \\ &\quad \cdot (-T_n^{\text{GN}} B + \ln 2D_n) - \ln 2D_n \sigma^2 = 0. \end{aligned} \quad (31)$$

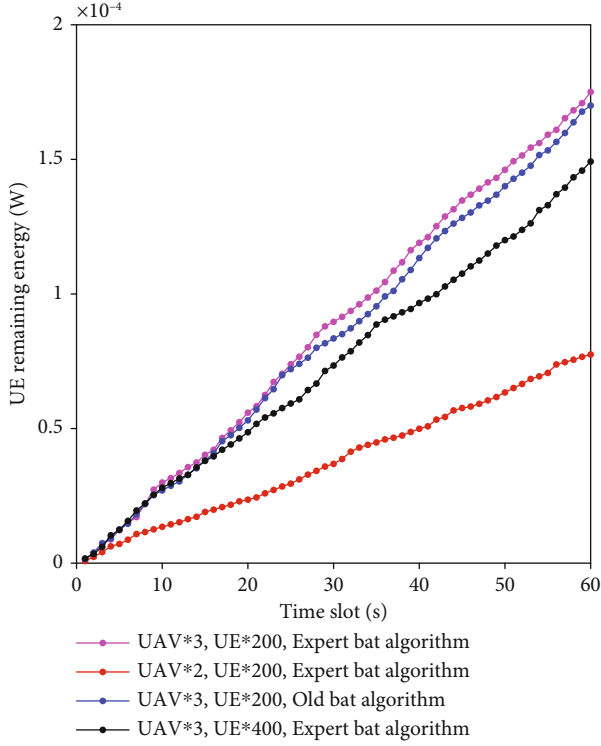


FIGURE 6: The sum of the remaining energy of all GNs in a period.

Theorem 1. Equation $(L^{P1})' = 0$ has a one solution and it is unique.

Proof Of Theorem 1.

Since this equation is a transcendental equation, the answer cannot be calculated directly, so Newton's iteration method is used to find an approximate solution. The iterative step can be expressed as

$$\Delta T_n^{GN} = \frac{(L^{P1})'}{(L^{P1})''}, \quad (32)$$

where

$$(L^{P1})'' = B \left(\alpha_n P_m^{UAV} W^{P1} + \sigma^2 + W^{C3} - W^{C2a} + W^{C2b} \right) - \frac{\sigma^2 2^{D_n/T_n^{GN} B}}{B} \left(\left(\frac{\ln 2 D_n}{T_n^{GN}} \right)^2 - \frac{B \ln 2 D_n}{T_n^{GN}} + B^2 \right). \quad (33)$$

□

□

Furthermore, both W^{C2a} and W^{C2b} need to collect all GN information before they can be calculated, while W^{C3} only needs GN itself. This feature makes this paper more suitable for using centralized and distributed algorithms. In order to minimize the GNs' information stored at UAV, this

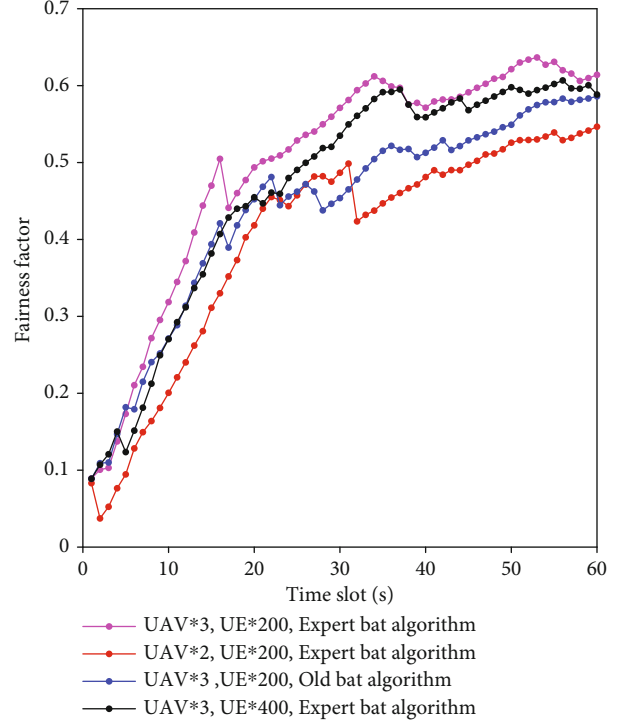


FIGURE 7: Fairness factor of each time slot.

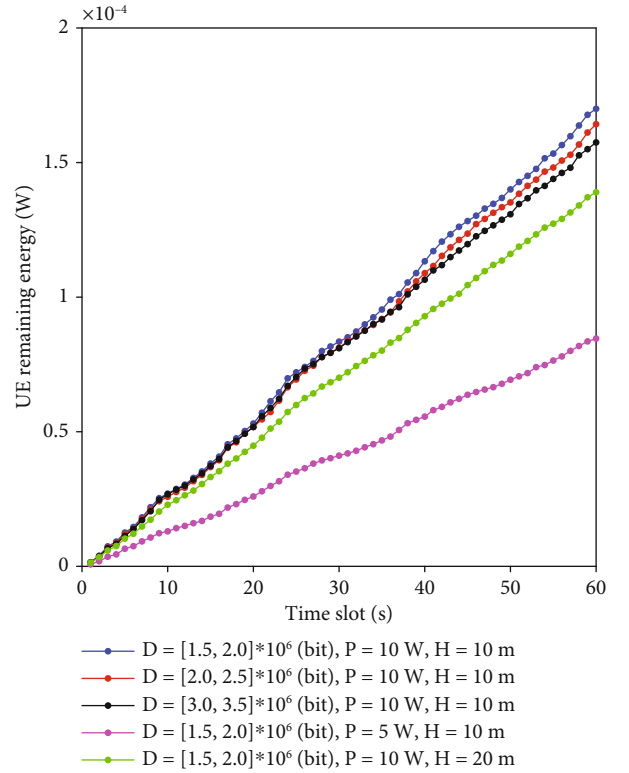


FIGURE 8: Fairness factor of each time slot.

paper allows GN to update W^{C2a} and W^{C2b} according to its own situation, while UAV only plays an auxiliary role. The specific algorithm is as Algorithm 2.

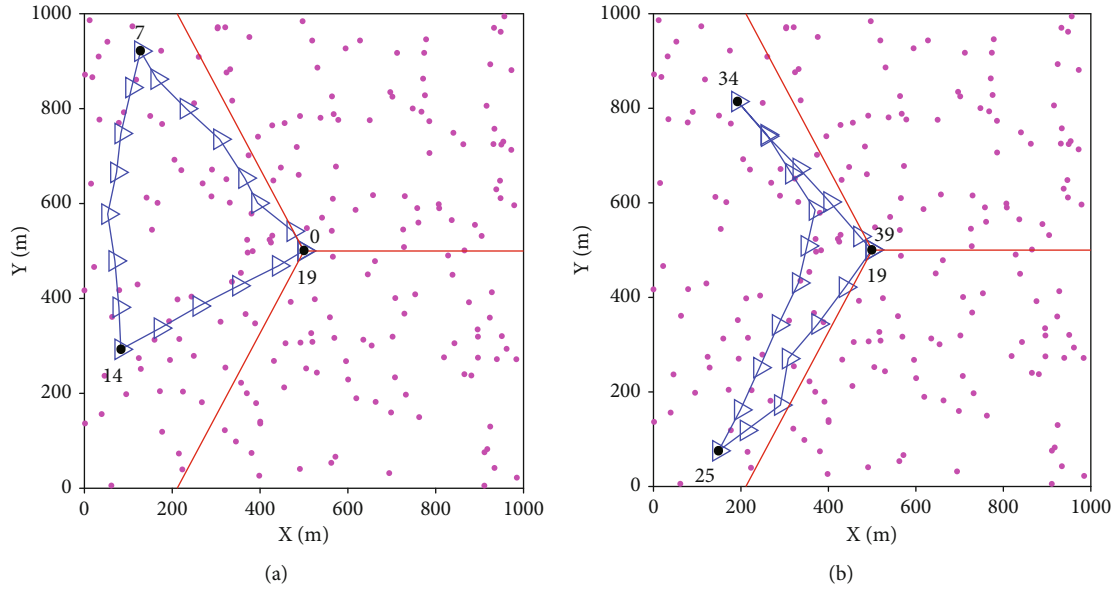


FIGURE 9: Examples of UAV automatic charging: (a) the first charge; (b) the second charge.

6. Simulation and Numerical Results

The numerical evaluation results of the time allocation algorithm and path planning algorithm are shown in this section. Unless specified, the network parameters are set up in Table 1.

When the GNs is stationary, the flight trajectory of the UAVs is shown in Figure 5. Sub-pictures a, b, and c correspond to different UAV numbers. The red dot in the figure represents GNs, the blue line represents the UAV's flight trajectory, and the red line represents the area boundary. It can be seen that the UAV in this paper adopts a dynamic and autonomous way to specify the time used for a service cycle. When the service area of a UAV increases, its service cycle will also increase. And the algorithm of this paper can dynamically change the path with the change of the environment. The environment in which the UAV starts to serve from the base station and from the end of the previous service cycle is different. So, each area in Figure 5 will show two slightly different flight paths. The trace in Figure 5 has protrusions at the corners. This is because UAVs in the networks will consider the needs of every GNs, including GNs in remote corners.

In order to make the performance analysis more realistic, Figure 6 adds the function of random motion to the GNs. In addition, the performance of the system under different GNs and UAV numbers and different bat algorithms is shown in Figure 6. It can be seen that the remaining energy of the GNs in this network increases with time. This is because the network in this paper can guarantee that the received energy of GNs is greater than the consumption in each time slot. When the number of the UAV decreases, the total energy obtained by GNs will also decrease. And when the number of GNs in the networks increases, the total energy obtained by the GNs will decrease by 20%. In addition, Figure 6 also proves that the bat algorithm of this paper

improves the network performance by 7%. Therefore, the number of UAVs in the network can be changed according to the energy requirements of the GNs, thereby avoiding unnecessary waste of resources.

It is not rigorous to only analyze the sum of energy acquired by GNs, because it is very likely that the sum of energy acquired by GNs is very large, but some GNs in the networks have insufficient energy. For this reason, Figure 7 uses the method in [48] to analyze the fairness of energy obtained by GNs under different conditions. It can be seen in Figure 7 that the fairness coefficient in the networks gradually tends to a fixed value through the slow stabilization of the networks. This is because in the previous time slot, the UAV has just entered the networks and has not yet formed a stable circular orbit. In addition, the change in the number of GNs has an impact on fairness of only about 0.01, but when the number of UAVs is reduced by one, the decrease is 0.1. This is because the decrease in the number of UAVs will increase the service area of each UAV, which will make the period for UAVs to provide high-quality services to each GN longer, resulting in uneven energy held by GNs.

The correlation analysis of the model's sensitivity to power, task length, and power is shown in Figure 8. When the mission length increases, the remaining energy of the GNs in the system decreases slightly, only a 7% decrease each 1Mbit. It shows that the algorithm designed in this paper can be well applied to actual scenarios, because the length of the user's task is generally variable and unpredictable. However, it is very sensitive to the power and working height of the UAV. Therefore, it is necessary to lower the working height of the UAV and increase the transmission power of the UAV as much as possible.

Figure 9 shows the working path of the UAV when there are 3 UAVs performing tasks at the same time and the power is insufficient. We deliberately modulate the UAV reserve power to be very low to show that the UAV charging

process and the time slot of each point are marked. According to the constraints of C6, when the UAV senses that it is running out of power, it will enter the return state in advance and continue to work after charging, as shown in Figure 9(b) and the UAV will continue to complete the tasks that were not completed in Figure 9(a) after charging.

7. Conclusion

This paper proposes a SWIPT-based MEC network composed of multiple UAVs and one MEC server. The main task is to provide computing assistance and energy transmission services for GNs. In order to ensure the flexibility of the networks, we proposed a distributed iterative algorithm and designed an expert bat algorithm to ensure the freedom of the UAV's trajectory. Experimental results show that compared with the traditional bat algorithm, this algorithm can guarantee the fairness of network services and can improve the network performance by 7%. In addition, the distributed iterative algorithm in this paper can also be well adapted to fluctuations in task length, ensuring the stability of the networks.

Appendix

A. Proof Of Theorem 1

The equation $(L^{P1})' = 0$ can be regarded as a problem of finding the intersection of a linear function and a nonlinear function, where the linear function $-T_n^{GN} B(\alpha_n P_m^{UAV} W^{P1} + \sigma^2 + W^{C3} - W^{C2a} + W^{C2b})$ is a straight line across the origin and the nonlinear function $\sigma^2 2^{D_n} T_n^{GN} B(-T_n^{GN} B + \ln 2D_n) - \ln 2D_n \sigma^2$ consists of two multiplying monotonically decreasing functions and a constant term, so it is also monotonically decreasing. And when T_n^{GN} approaches infinity, the function value is negative infinity. When T_n^{GN} approaches 0, the function value is infinite. Therefore, the straight line must have only one intersection with the curve.

Data Availability

No data set is required.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This work is supported by the Fundamental Research Funds for the Central Universities (FRF-DF-20-12 and FRF-GF-18-017B).

References

- [1] B. Zhao, P. Zhao, and P. Fan, "ePUF: A lightweight double identity verification in IoT," *Tsinghua Science and Technology*, vol. 25, no. 5, pp. 625–635, 2020.
- [2] J. Li, M. Siddula, X. Cheng, W. Cheng, Z. Tian, and Y. Li, "Approximate data aggregation in sensor equipped IoT networks," *Tsinghua Science and Technology*, vol. 25, no. 1, pp. 44–55, 2020.
- [3] Q. Wu, G. Ding, Y. Xu et al., "Cognitive Internet of things: a new paradigm beyond connection," *IEEE Internet of Things Journal*, vol. 1, no. 2, pp. 129–143, 2014.
- [4] W. Feng, J. Tang, Y. Yu et al., "UAV-enabled SWIPT in IoT networks for emergency communications," *IEEE Wireless Communications*, vol. 27, no. 5, pp. 140–147, 2020.
- [5] F. Wang, J. Xu, X. Wang, and S. Cui, "Joint offloading and computing optimization in wireless powered mobile-edge computing systems," *IEEE Transactions on Wireless Communications*, vol. 17, no. 3, pp. 1784–1797, 2018.
- [6] Cisco, *Cisco Visual Networking Index: Global Mobile Data Traffic Forecast Update, 2018-2023*, Cisco, White Paper, 2020.
- [7] P. Porambage, J. Okwuibe, M. Liyanage, M. Ylianttila, and T. Taleb, "Survey on multi-access edge computing for Internet of things realization," *IEEE Communications Surveys & Tutorials*, vol. 20, no. 4, pp. 2961–2991, 2018.
- [8] M. Patel, B. Naughton, C. Chan, N. Sprecher, S. Abeta, and A. Neal, "Mobile-edge computing introductory technical white paper," *White paper, mobile-edge computing (MEC) industry initiative*, pp. 1089–7801, 2014.
- [9] H. T. Thanh Binh, N. P. Le, N. B. Minh, T. Thu Hai, N. Q. Minh, and D. Bao Son, "A reinforcement learning algorithm for resource provisioning in mobile edge computing network," in *2020 International Joint Conference on Neural Networks (IJCNN)*, pp. 1–7, Glasgow, United Kingdom, 2020.
- [10] Y. Mao, C. You, J. Zhang, K. Huang, and K. B. Letaief, "A survey on mobile edge computing: the communication perspective," *IEEE Communications Surveys & Tutorials*, vol. 19, no. 4, pp. 2322–2358, 2017.
- [11] T. Liu, X. Wang, and L. Zheng, "A cooperative SWIPT scheme for wirelessly powered sensor networks," *IEEE Transactions on Communications*, vol. 65, no. 6, pp. 2740–2752, 2017.
- [12] R. Zhang and C. K. Ho, "MIMO broadcasting for simultaneous wireless information and power transfer," *IEEE Transactions on Wireless Communications*, vol. 12, no. 5, pp. 1989–2001, 2013.
- [13] X. Lu, P. Wang, D. Niyato, D. I. Kim, and Z. Han, "Wireless networks with RF energy harvesting: a contemporary survey," *IEEE Communications Surveys & Tutorials*, vol. 17, no. 2, pp. 757–789, 2015.
- [14] X. Lu, P. Wang, D. Niyato, and E. Hossain, "Dynamic spectrum access in cognitive radio networks with RF energy harvesting," *IEEE Wireless Communications*, vol. 21, no. 3, pp. 102–110, 2014.
- [15] D. T. Hoang, D. Niyato, P. Wang, and D. I. Kim, "Opportunistic channel access and RF energy harvesting in cognitive radio networks," *IEEE Journal on Selected Areas in Communications*, vol. 32, no. 11, pp. 2039–2052, 2014.
- [16] S. Lee, R. Zhang, and K. Huang, "Opportunistic wireless energy harvesting in cognitive radio networks," *IEEE Transactions on Wireless Communications*, vol. 12, no. 9, pp. 4788–4799, 2013.
- [17] M. T. Dabiri, M. Rezaee, V. Yazdaniyan, B. Maham, W. Saad, and C. S. Hong, "3D channel characterization and performance analysis of UAV-assisted millimeter wave links," *IEEE Transactions on Wireless Communications*, vol. 20, no. 1, pp. 110–125, 2021.
- [18] J.-H. Lee, K.-H. Park, Y.-C. Ko, and M.-S. Alouini, "Throughput maximization of mixed FSO/RF UAV-aided mobile

- relaying with a buffer,” *IEEE Transactions on Wireless Communications*, vol. 20, no. 1, pp. 683–694, 2021.
- [19] J. Chen, Q. Wu, Y. Xu et al., “Joint task assignment and spectrum allocation in heterogeneous UAV communication networks: a coalition formation game-theoretic approach,” *IEEE Transactions on Wireless Communications*, vol. 20, no. 1, pp. 440–452, 2021.
- [20] C. Zhan and H. Lai, “Energy minimization in Internet-of-things system based on rotary-wing UAV,” *IEEE Wireless Communications Letters*, vol. 8, no. 5, pp. 1341–1344, 2019.
- [21] H. Ye, X. Kang, J. Joung, and Y. Liang, “Optimization for full-duplex rotary-wing UAV-enabled wireless-powered IoT networks,” *IEEE Transactions on Wireless Communications*, vol. 19, no. 7, pp. 5057–5072, 2020.
- [22] F. Wu, D. Yang, L. Xiao, and L. Cuthbert, “Energy consumption and completion time tradeoff in rotary-wing UAV enabled WPCN,” *IEEE Access*, vol. 7, pp. 79617–79635, 2019.
- [23] C. Zhan and R. Huang, “Energy efficient adaptive video streaming with rotary-wing UAV,” *IEEE Transactions on Vehicular Technology*, vol. 69, no. 7, pp. 8040–8044, 2020.
- [24] Z. Wang, W. Xu, D. Yang, and J. Lin, “Joint trajectory optimization and user scheduling for rotary-wing UAV-enabled wireless powered communication networks,” *IEEE Access*, vol. 7, pp. 181369–181380, 2019.
- [25] J. Xu, Y. Zeng, and R. Zhang, “UAV-enabled wireless power transfer: trajectory design and energy region characterization,” in *2017 IEEE Globecom Workshops (GC Wkshps)*, pp. 1–7, Singapore, 2017.
- [26] C. You and R. Zhang, “Hybrid offline-online design for UAV-enabled data harvesting in probabilistic LoS channels,” *IEEE Transactions on Wireless Communications*, vol. 19, no. 6, pp. 3753–3768, 2020.
- [27] P. S. Bithas, V. Nikolaidis, A. G. Kanatas, and G. K. Karagiannidis, “UAV-to-ground communications: channel modeling and UAV selection,” *IEEE Transactions on Communications*, vol. 68, no. 8, pp. 5135–5144, 2020.
- [28] Y. Du, K. Yang, K. Wang, G. Zhang, Y. Zhao, and D. Chen, “Joint resources and workflow scheduling in UAV-enabled wirelessly-powered MEC for IoT systems,” *IEEE Transactions on Vehicular Technology*, vol. 68, no. 10, pp. 10187–10200, 2019.
- [29] Q. Wu and R. Zhang, “Common throughput maximization in UAV-enabled OFDMA systems with delay consideration,” *IEEE Transactions on Communications*, vol. 66, no. 12, pp. 6614–6627, 2018.
- [30] H. Xiao, Z. Hu, K. Yang, Y. Du, and D. Chen, “An energy-aware joint routing and task allocation algorithm in MEC systems assisted by multiple UAVs,” in *2020 International Wireless Communications and Mobile Computing (IWCMC)*, pp. 1654–1659, Limassol, Cyprus, 2020.
- [31] L. Ye, Y. Zhang, Y. Li, and S. Han, “A dynamic cluster head selecting algorithm for UAV Ad Hoc networks,” in *International Wireless Communications and Mobile Computing Conference, IWCMC*, pp. 225–228, Limassol, Cyprus, 2020.
- [32] B. Ji, Y. Li, B. Zhou, C. Li, K. Song, and H. Wen, “Performance analysis of UAV relay assisted IoT communication network enhanced with energy harvesting,” *IEEE Access*, vol. 7, pp. 38738–38747, 2019.
- [33] A. Agarwal and D. Mishra, “Hovering localization and power allocation for UAV assisted DF relaying ad hoc network,” in *2020 IEEE International Conference on Communications Workshops (ICC Workshops)*, pp. 1–6, Dublin, Ireland, 2020.
- [34] W. Wang, X. Li, M. Zhang et al., “Energy-constrained UAV-assisted secure communications with position optimization and cooperative jamming,” *IEEE Transactions on Communications*, vol. 68, no. 7, pp. 4476–4489, 2020.
- [35] D. N. K. Jayakody, T. D. P. Perera, A. Ghrayeb, and M. O. Hasna, “Self-energized UAV-assisted scheme for cooperative wireless relay networks,” *IEEE Transactions on Vehicular Technology*, vol. 69, no. 1, pp. 578–592, 2020.
- [36] D. N. K. Jayakody, T. D. Ponnimbaduge Perera, M. C. Nathan, and M. Hasna, “Self-energized full-duplex UAV-assisted cooperative communication systems,” in *2019 IEEE International Black Sea Conference on Communications and Networking (Black Sea Com)*, Sochi, Russia, 2019.
- [37] Y. Gu, C. Li, B. Xia, D. Xu, and Z. Chen, “Modeling and performance analysis of stochastic mobile edge computing wireless networks,” in *2019 IEEE 89th Vehicular Technology Conference (VTC2019-Spring)*, pp. 1–5, Kuala Lumpur, Malaysia, 2019.
- [38] P. S. Nithin, N. B. Sai Shibu, S. Sree Lakshmi, and S. Ponnakanti, “Location module for 5G base station to support mobility management of drones,” in *2019 International Conference on Communication and Electronics Systems (ICCES)*, pp. 1336–1340, Coimbatore, India, 2019.
- [39] Q. Zhang, W. Saad, M. Bennis, X. Lu, M. Debbah, and W. Zuo, “Predictive deployment of UAV base stations in wireless networks: machine learning meets contract theory,” *IEEE Transactions on Wireless Communications*, vol. 20, no. 1, pp. 637–652, 2021.
- [40] S. Ahmed, M. Z. Chowdhury, and Y. M. Jang, “Energy-efficient UAV relaying communications to serve ground nodes,” *IEEE Communications Letters*, vol. 24, no. 4, pp. 849–852, 2020.
- [41] C. Liu, M. Ding, C. Ma, Q. Li, Z. Lin, and Y. Liang, “Performance analysis for practical unmanned aerial vehicle networks with LoS/NLoS transmissions,” in *2018 IEEE International Conference on Communications Workshops (ICC Workshops)*, pp. 1–6, Kansas City, MO, 2018.
- [42] Q. Hu, Y. Cai, G. Yu, Z. Qin, M. Zhao, and G. Y. Li, “Joint off-loading and trajectory design for UAV-enabled mobile edge computing systems,” *IEEE Internet of Things Journal*, vol. 6, no. 2, pp. 1879–1892, 2019.
- [43] X. Wang and M. C. Gursoy, “Coverage analysis for energy-harvesting UAV-assisted mmwave cellular networks,” *IEEE Journal on Selected Areas in Communications*, vol. 37, no. 12, pp. 2832–2850, 2019.
- [44] K. Huang and E. Larsson, “Simultaneous information and power transfer for broadband wireless systems,” *IEEE Transactions on Signal Processing*, vol. 61, no. 23, pp. 5972–5986, 2013.
- [45] Y. Liu and X. Wang, “Information and energy cooperation in OFDM relaying: protocols and optimization,” *IEEE Transactions on Vehicular Technology*, vol. 65, no. 7, pp. 5088–5098, 2016.
- [46] F. Zhou, Y. Wu, R. Q. Hu, and Y. Qian, “Computation rate maximization in UAV-enabled wireless-powered mobile-edge computing systems,” *IEEE Journal on Selected Areas in Communications*, vol. 36, no. 9, pp. 1927–1941, 2018.

- [47] J. Zhang, L. Zhou, Q. Tang et al., "Stochastic computation offloading and trajectory scheduling for UAV-assisted mobile edge computing," *IEEE Internet of Things Journal*, vol. 6, no. 2, pp. 3688–3699, 2019.
- [48] H. V. Abeywickrama, Y. He, E. Dutkiewicz, B. A. Jayawickrama, and M. Mueck, "A reinforcement learning approach for fair user coverage using UAV mounted base stations under energy constraints," *IEEE Open Journal of Vehicular Technology*, vol. 1, pp. 67–81, 2020.