Task Scheduling and Trajectory Optimization Based on Fairness and Communication Security for Multi-UAV-MEC System

Yejun He, *Senior Member, IEEE*, Kun Xiang, Xiaowen Cao, *Member, IEEE*, and Mohsen Guizani, *Fellow, IEEE*

***Abstract*—Unmanned aerial vehicles (UAVs) show significant potential in enhancing communication services within the mobile edge computing (MEC) system by taking their advantages on the flexible mobility and reliable line-of-sight links. However, in the scenarios with multiple UAV-MECs (UMs) operating concurrently, potential conflicts in their trajectories need to be mitigated. Thus, the 3-D trajectory needs to be properly designed in a highly reliable manner. Besides, such an infrastructure-free communication paradigm also exposes a potential risk of misuse by malicious parties, which allows them to eavesdrop on private communications, posing a threat to the security and privacy. Therefore, we consider a multi-UAV-assisted MEC communication system, where a UAV maliciously eavesdrops on the data transmission from the user devices (UDs) while a jammer is deployed on the ground to interfere with the eavesdropping channel. In specific, our objective is to minimize the energy consumption and latency while incorporating fairness metrics by optimizing the 3-D trajectories of UMs, transmission power of UDs, and the offloading strategies under the constraints of ensuring communication security and load fairness. Given the complexity of this mixed-integer nonconvex programming problem, we decompose the formulated problem into three subproblems. Specifically, at each time slot, we optimize the transmit power and offloading strategies using theoretical derivation and mathematical analysis, respectively. Additionally, a multiagent deep deterministic policy gradient (MADDPG) algorithm is employed to optimize the trajectories of UMs. Simulation results demonstrate that our proposed joint optimization algorithm successfully minimizes the system energy consumption and delay as compared to benchmarking schemes.**

***Index Terms*—Fairness, mobile edge computing (MEC), multiagent deep deterministic policy gradient (MADDPG), multiunmanned aerial vehicles (UAVs), offloading strategy, security, transmit power.**

I. Introduction

With the development of 5G, more emerging technologies, such as virtual reality and augmented reality, appear and put higher requirements on the network. Traditional cloud computing can no longer meet the increasing computing demands. Mobile edge computing (MEC), a key technology of 5G, provides a wide space for these emerging technologies. As an emerging computing paradigm, MEC works by moving servers closer to the edge of the network and providing powerful computing, storage, and communication services to user devices (UDs) [1]. In order to improve the computational requirements of UDs, a number of researchers have conducted a lot of related studies in recent years. Hou et al. solved the problem of communication reliability and stringent latency requirements of vehicular through resource scheduling [2]. Souza et al. introduced a MEC-network functions virtualization (MEC-NFV) network model designed to meet the demands of ultrareliable and low-latency communication (URLLC) [3].

The extensive implementation of MEC incurs substantial expenses as it resides at the edge of the network, and deploying MEC in areas with insufficient communication infrastructure proves to be a challenging task [4], [5]. Combining the MEC with unmanned aerial vehicles (UAVs) can effectively solve this challenge due to its flexibility and ease of deployment. For traditional MEC networks, the complex terrain makes it difficult to guarantee a good channel quality. While the UAV can be kept at a high altitude, the UAV-assisted MEC system makes it possible for the signal to be transmitted in a Line-of-Sight (LoS) channel. Liu et al. [6] investigated a MISO UAV-assisted MEC network to address the interference caused by multipath effects. The study aims to achieve a minimum energy consumption for the system. El-Emary et al. [7] employed UAVs as relays to facilitate UDs in offloading tasks to MEC servers. They present a task scheduling and allocation algorithm designed to optimize the energy consumption of the system. Qin et al. [8] introduced an enhancement to the channel quality by incorporating the reconfigurable intelligent surface (RIS).

Manuscript received 6 January 2024; revised 20 April 2024; accepted 23 May 2024. Date of publication 11 June 2024; date of current version 25 September 2024. This work was supported in part by the National Key Research and Development Program of China under Grant 2023YFE0107900; in part by the National Natural Science Foundation of China (NSFC) under Grant 62071306; and in part by the Shenzhen Science and Technology Program under Grant JCYJ20200109113601723, Grant JSGG20210802154203011, Grant JSGG20210420091805014, Grant RCBS20231211090520032, and Grant RCBS20231211090520032. *(Corresponding author: Xiaowen Cao.)*

Yejun He, Kun Xiang, and Xiaowen Cao are with the State Key Laboratory of Radio Frequency Heterogeneous Integration, Sino-British Antennas and Propagation Joint Laboratory of Ministry of Science and Technology of China, Guangdong Engineering Research Center of Base Station Antennas and Propagation, Shenzhen Key Laboratory of Antennas and Propagation, College of Electronics and Information Engineering, Shenzhen University, Shenzhen 518060, China (e-mail: heyejun@ieee.org; 1303484745@qq.com; caoxwen@szu.edu.cn).

Mohsen Guizani is with the Machine Learning Department, Mohamed Bin Zayed University of Artificial Intelligence, Abu Dhabi, UAE (e-mail: mguizani@ieee.org).

Digital Object Identifier 10.1109/JIOT.2024.3412825

Within a UAV-assisted MEC system, the predominant share of energy consumption arises from flight-related energy consumption. Therefore, optimization of the UAV trajectory, coupled with resource allocation to the MEC, becomes imperative. Wang et al. [9] determined the shortest path for the UAV by formulating the UAV trajectory optimization subproblem as a traveling salesman problem. Simultaneous optimization of the UAV trajectory, resource allocation, and task scheduling often involves the use of algorithms like successive convex approximation to attain an overall suboptimal solution [10], [11], [12]. To enhance adaptability to terrain changes and the distribution pattern of UDs, the 2-D trajectory of the UAV is extended to 3-D. This extension allows for adjustments in the flight altitude of the UAV to better accommodate variations in the environment [13]. In addition, multi-UAV assisted MEC has been widely emphasized. Each UD can choose one of the UAVs for mission offloading. This is a mixed integer nonlinear programming problem. To solve this problem, the model needs to be simplified to solve the problem step by step [14].

In recent studies, an increasing number of researchers have employed deep reinforcement learning (DRL) in the domain of UAV-assisted MEC. In addressing dynamic conditions, long-term optimization, and ambiguous state information, DRL is superior than traditional convex optimization algorithms. Researchers are committed to applying UAV-assisted MEC systems to more scenarios, which will make the environmental elements more complex and the size of the problems to be solved grows, so the computation time of traditional algorithms cannot meet the demands of UDs. With powerful data processing capabilities, DRL can explore solutions in a broad strategy space and effectively capitulate the characteristics of the problem model. Describing the trajectory planning problem for a UAV as a sequence of decisions, Zhu et al. [15] introduced a sequence-to-sequence pointer network. This model takes the starting point of the UAV and all clusters as input, producing the UAV trajectory as its output. However, this form of node access is only suitable in user-focused scenarios, while sequential UAV trajectories are more adapted to realistic scenarios. Liu et al. [16] optimized the configuration of the VM based on the DQN algorithm. For UAV trajectories, the DDPG algorithm, which is more suitable for large dimensional continuous spaces, is used. Chen et al. [17] addressed a resource allocation problem that takes into account both delay and energy consumption. They utilize DQN to initialize the user association policy and combine it with a greedy policy to swiftly attain the optimal solution. Additionally, the motion trajectories of multiple UAVs are optimized using the DDPG algorithm.

In addition to this, fair competition is also a critical issue. Since UAV is sensitive to energy consumption, when the task allocation is not fair, it will result in a waste of energy and computational resources, making the system less efficient in task processing. Therefore, some researchers have made some studies on the fairness of UAV-assisted MEC. Zhao et al. [18] prevented overloading the UAV computation by limiting the size of the ground device's task volume. Zhou et al. [19] included fairness metrics as part of the soft actor-critic (SAC) reward mechanism when optimizing UAV tracks and allocation of resources based on the SAC algorithm to ensure the fairness of the system in different scenarios. He et al. [20] goal is to optimize the fairness and the energy consumption of the system by optimizing the choice of UDs based on the Nash equilibrium to obtain higher fairness.

Data security is a critical concern in UAV-assisted MEC system. Due to the channel specificity, UAVs are greatly probable to be exposed in a monitored environment, where eavesdroppers can intercept information from UDs through the data link. To protect the data stream, traditional encryption techniques like encrypted communication and authentication can be applied [21], along with emerging physical layer security approaches, such as signal jamming, to increase eavesdropping difficulty [22]. For instance, a lightweight symmetric encryption algorithm was proposed in [21] to ensure data confidentiality, while the Rupasinghe et al. [23] enhanced physical layer security with protected areas. The work [24] designed a protocol for improving security throughput of users with poor channel quality, where confidential messages and artificial noises are sent in a small time slot and remaining time slot, respectively. However, due to the high-speed mobility and dynamic topology of UAVs, traditional techniques may be hard to offer sufficient security and face challenges on location leakage, key management difficulties, and increased communication latency. Toward this end, existing work overcome the above issues via network optimization, such as resource allocation, offloading strategy, and UAV trajectory optimization [25], [26], [27]. In particular, a design in multiple sources of jamming signals emission was proposed in [28] to improve system security by jointly optimizing UAV position, jamming signal emission power, and resource allocation. However, traditional algorithms are hard to obtain the optimal solution and often require several iterations to get the alternative suboptimal results, highlighting the research significance of combining DRL with UAV-MEC.

From the current research status, there are not many studies for multiple UAV-MECs (UMs) in 3-D trajectory scenarios. Under 3-D scenarios, the flight model of drones makes the problem more complicated, and the limitation of the multiintelligent body model to obtain information increases the difficulty of convergence of the algorithm. Therefore, we investigate a multi-UAV assisted MEC system in 3-D scenarios. Specifically, multiple UDs are randomly distributed on the ground, and multiple UMs provide mission offloading computation services for UDs from different starting points. An eavesdropping UAV (EU) is also present in the system, and a jammer is installed on the ground in order to interfere with the quality of the eavesdropping channel. To prevent overloading of individual UMs as well as to reduce the energy and the latency consumed by the system, we optimize the offloading policy of each UD with fairness in mind. In terms of communication security, we adjust the offloading mode according to the channel state of UDs to meet the security requirements of the system. In addition, we study the relationship between the transmit power of UDs and the energy and the latency consumed by the system and obtain the optimal solution by formula derivation. For the trajectory optimization of UMs, we consider the 3-D flight energy model of UMs and find the optimal trajectory of UMs under the

requirement of minimizing the energy and the latency consumed by the system. The main contributions are summarized as follows.

1) We present a 3-D multi-UAV assisted MEC system that considers fairness and communication security. Each UD randomly generates tasks at each time slot, and the UDs can choose to offload to the UM side for processing or local processing. Each drone needs to act on decisions based on changes in the current environment. For fairness between UMs and secure communication between UDs and at the same time reducing the energy consumption and latency of the system, we jointly optimize the launch power and unloading policy of UDs, as well as the 3-D flight tracks of UMs.
2) We decompose our optimization problem into three subproblems given that it is a mixed integer linear programming problem. Specifically, first, we prove the existence of an optimal solution for the launch power by analyzing the mathematical relationship between the energy consumption model and the launch power. Second, an iterative optimization algorithm based on Nash equilibrium is theoretically derived and proposed to obtain the optimal offloading ratio and UDs association policy by combining the safety factor. We describe the UAV flight process as a Markov decision process and optimize drone trajectories on the basis of the multiagent deep deterministic policy gradient (MADDPG) algorithm after obtaining the optimal solutions for the transmit power and offloading strategy of a single time slot.
3) We demonstrate through simulation experiments that our proposed algorithm has good convergence. For energy consumption and delay, our proposed algorithm outperforms other reference algorithms while maintaining high fairness and communication security.

The remainder of this article is organized as follows. Section II describes the system model. Section III gives the formulation of the optimization problem. Section IV describes our power control scheme, UDs offloading strategy, and Drones trajectory optimization algorithm based on MADDPG. Section V gives the simulation results and analysis. Section VI gives the summary of this article.

II. System Model

The model of our proposed 3-D multi-UAV assisted MEC system is shown in Fig. 1. The system consists of I UDs as well as J legal UMs, MEC servers serving the UDs are integrated into each UAV. We denote the set of UDs and UMs by $\mathcal{I}$ and $\mathcal{J}$, respectively. We define the flight time slots of the UMs to be denoted by the set $\mathcal{T} = \{1, 2, \ldots, T\}$. Each UD generates a random size of tasks at each time slot, and the size of the tasks generated by the UDs at time slot t is denoted by the set $\mathcal{L} = \{L_1(t), L_2(t), \ldots, L_I(t)\}$. Considering the user's demand for low-energy consumption and low latency for task processing, UDs need to offload tasks to the UMs for processing. We use time division multiple access (TDMA) protocol to offload tasks to UMs, and UDs have the option of offloading tasks to UMs or keeping them for local processing.

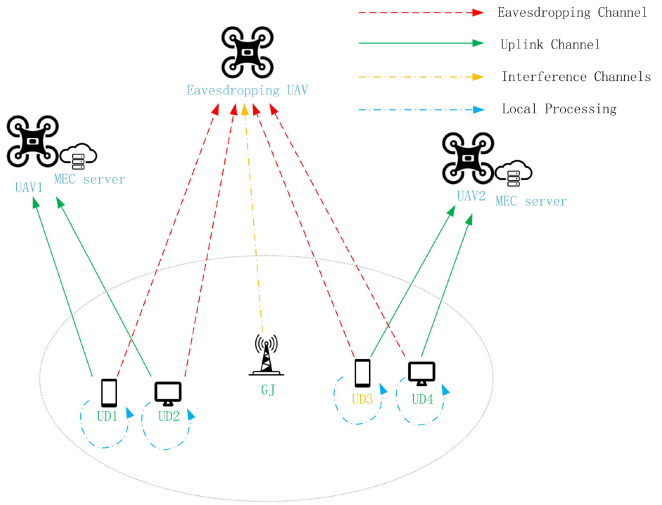


Fig. 1. Three-dimensional multi-UAV-assisted MEC system model.

In addition, the system also contains an EU, which is used to eavesdrop on the data transmitted by the users to the UMs. Assuming that the EU is camouflaged in such a way that we cannot identify whether it is a legitimate UAV or not, and assuming for convenience that the starting point, altitude, and direction of the EU's trajectory are determined, we can enhance the communication security of the system by controlling the offloading ratio of the UDs. In the center of the ground coverage area, we place a ground jammer (GJ), which interferes the EU by transmitting jamming signals to mislead the EU. Since the characteristics of the jamming signal emitted by the GJ are known to the UMs, they can filter out the jamming signal at the receiving end of the signal.

Define the positions of the UDs and UMs at time slot t as $U_i(t) = \{X_i(t), Y_i(t), 0\}, i \in \mathcal{I}$, and $M_j(t) = \{X_j(t), Y_j(t), H_j(t)\}, j \in \mathcal{J}$, respectively. The positions of EU and GJ are $M_{eu}(t) = \{X_{eu}(t), Y_{eu}(t), H_{eu}(t)\}$ and $M_{gj} = \{X_{gj}, Y_{gj}, 0\}$, respectively. The UMs are each from a different starting point $MS_j, j \in \mathcal{J}$ and the displacement constraints for each UM can be expressed as

$$||M_j(t+1) - M_j(t)|| \le q^{\max}, j \in \mathcal{J}, t \in \mathcal{T} \tag{1}$$

where $q^{\max}$ denotes the maximum displacement constraint. The deflection angle constraint for UMs is expressed as

$$\theta_h^{\min} \le \theta_h^j(t) \le \theta_h^{\max} \tag{2}$$

$$\theta_v^{\min} \le \theta_v^j(t) \le \theta_v^{\max} \tag{3}$$

where $\theta_h^j(t)$ and $\theta_v^j(t)$ denote the horizontal and vertical deflection angles of the jth UM, respectively. $\theta_h^{\min}$ and $\theta_h^{\max}$ denote the horizontal deflection angle constraints, and $\theta_v^{\min}$ and $\theta_v^{\max}$ denote the vertical deflection angle constraints.

A. Communication Model

Due to the terrain, we determine the large scale fading of UMs-UD links by using the probability LoS channel model [29]. During the communication process, we only

consider the uplink communication model because the amount of returned data is negligible compared to the processing period. We define the LoS probability between the ith UD and the jth UM as $P_{i,j}^{\mathrm{LoS}}(t)$, $P_{i,j}^{\mathrm{LoS}}(t)$ can be expressed as

$$P_{i,j}^{\mathrm{LoS}}(t) = \frac{1}{1+\lambda_1 \exp\{-\lambda_2(\theta_{i,j}(t)-\lambda_1)\}} \tag{4}$$

where λ_1 and λ_2 are parameters associated with the environment; and $\theta_{i,j}(t)$ denotes the angle of elevation between ith UD and jth UM in time slot t, which can be expressed as

$$\begin{aligned}\theta_{i,j}(t) &= (180/\pi)\arcsin\left(\frac{H_j(t)}{d_{i,j}(t)}\right)\\ &= (180/\pi)\arcsin\left(\frac{H_j(t)}{||M_j(t)-U_i(t)||}\right)\end{aligned} \tag{5}$$

where $d_{i,j}(t)$ denotes at time slot t the Euclidean distance between the ith UD and the jth UM in 3-D space. Therefore, the NLoS probability can be expressed as $P^{\mathrm{NLoS}}(t) = 1 - P^{\mathrm{LoS}}(t)$.

Similarly, the LoS probabilities $P_{i,e}^{\mathrm{LoS}}$ and $P_{g,e}^{\mathrm{LoS}}$ between the ith UD and EU, GJ and EU at time slot t are denoted by

$$P_{i,e}^{\mathrm{LoS}} = \frac{1}{1+\lambda_1 \exp\{-\lambda_2(\theta_{i,e}(t)-\lambda_1)\}} \tag{6}$$

$$P_{g,e}^{\mathrm{LoS}} = \frac{1}{1+\lambda_1 \exp\{-\lambda_2(\theta_{g,e}(t)-\lambda_1)\}} \tag{7}$$

where $\theta_{i,e}(t) = (180/\pi)\arcsin(H_e(t)/d_{i,e}(t))$ and $\theta_{g,e}(t) = (180/\pi)\arcsin(H_e(t)/d_{g,e}(t))$ denote the elevation angles between ith UD and EU, GJ and EU at time slot t, respectively.

In this study, considering the rather high height of UMs and the relatively short flight distance within each time slot, we consider the constancy of the channel gain of UMs during each time slot. The channel gains between ith UD and jth UM, ith UD and EU, and GJ and EU at time slot t are denoted by [29]

$$h_{i,j}(t) = \frac{P_{i,j}^{\mathrm{LoS}}(t)\beta_0 + P_{i,j}^{\mathrm{NLoS}}(t)\kappa\beta_0}{d_{i,j}^{\eta}(t)} \tag{8}$$

$$h_{i,e}(t) = \frac{P_{i,e}^{\mathrm{LoS}}(t)\beta_0 + P_{i,e}^{\mathrm{NLoS}}(t)\kappa\beta_0}{d_{i,e}^{\eta}(t)} \tag{9}$$

$$h_{g,e}(t) = \frac{P_{g,e}^{\mathrm{LoS}}(t)\beta_0 + P_{g,e}^{\mathrm{NLoS}}(t)\kappa\beta_0}{d_{g,e}^{\eta}(t)} \tag{10}$$

where κ is the decay rate of NLoS, β_0 is a channel gain per unit distance, and η is a path loss index.

In this article, we use TDMA protocol for communication. Since the UMs have prior knowledge about the interference signals emitted by the GJ, the GJ only causes interference to the EU. Therefore, for the communication channels between UDs and UMs, and UDs and EU, the signal interference noise ratio (SINR) can be written as

$$r_{i,j}(t) = \frac{p_{i,j}(t)h_{i,j}(t)}{\delta_0^2} \tag{11}$$

$$r_{i,e}(t) = \frac{p_{i,j}(t)h_{i,e}(t)}{p_g h_{g,e}(t) + \delta_0^2} \tag{12}$$

where p_g is the transmit power of GJ. $p_{i,j}(t)$ is the transmit power of the ith UD. δ_0^2 is the power of noise. The transmit power constraint for UDs is defined as

$$0 \le p_{i,j}(t) \le P_{\max} \tag{13}$$

where $P_{\max}$ denotes the maximum transmit power of UDs.

After obtaining the SINR, it can be known from the Shannon formula that the data transmission rate $R_{i,j}(t)$ and the eavesdropping rate $R_{i,e}(t)$ can be expressed as

$$R_{i,j}(t) = B\log_2(1 + r_{i,j}(t)) \tag{14}$$

$$R_{i,e}(t) = B\log_2(1 + r_{i,e}(t)) \tag{15}$$

where B represents the channel bandwidth.

We define two types of connection states between UDs and UMs as connection and disconnection, and the states are updated in real-time for each time slot. The connection state can be expressed as

$$a_{i,j}(t) = \begin{cases} 1, & \text{Connection} \\ 0, & \text{Disconnection.} \end{cases} \tag{16}$$

Assuming that each UD is connected to only one UM per slot and that the ith UD has the option of partially processing the task by unloading it to the UM in the connected state or processing the task locally. The constraint of the offloading policy $\varphi_{i,j}(t)$ is given by

$$0 \le \varphi_{i,j}(t) \le 1. \tag{17}$$

To evaluate the privacy and security tradeoff during the data offloading period, we define a security scaling factor $s_{i,j}(t)$, $s_{i,j}(t)$ is denoted as

$$s_{i,j}(t) = \frac{\sum_{j\in\mathcal{J}} a_{i,j}(t) r_{i,j}(t)}{r_{i,e}(t)}. \tag{18}$$

This security scaling factors indicate that if the security scale factor of the ith UD satisfies $s_{i,j}(t) \ge s_{\min}$, the ith UD can choose the partial offloading policy, otherwise, it can only choose the local processing policy.

B. Computation Model

We give a model of the energy consumption and delay required for the corresponding task processing, depending on the offloading strategy.

1) *Local Computing:* If the policy for ith UD is full local offloading, the latency of local processing is given as

$$T_i^{\mathrm{loc}}(t) = \frac{C_{ud}L_i(t)}{F_{ud}} \tag{19}$$

where F_{ud} is the computing resource of per UD. C_{ud} is the number of CPU cycles required by each UD to process per bit data. The local processing energy consumption of the ith UD is represented as

$$E_i^{\mathrm{loc}}(t) = k_{ud}F_{ud}^3 T_i^{\mathrm{loc}}(t) \tag{20}$$

where k_{ud} is the ith UD's effective capacitance coefficient.

2) *Partial Offloading:* If the policy for ith UD is partial offloading, the energy and the latency consumed by the part of the local processing is written as

$$T_{i,j}^{\text{loc}}(t) = \frac{a_{i,j}(t)C_{ud}(1-\varphi_{i,j}(t))L_i(t)}{F_{ud}} \tag{21}$$

$$E_{i,j}^{\text{loc}}(t) = k_{ud}F_{ud}^3 T_{i,j}^{\text{loc}}(t). \tag{22}$$

Since the consumption formulas for energy and delay for local processing of the partial offloading strategy are the same as for the full local processing strategy, the full local processing strategy can be viewed as $\varphi = 0$, i.e., it is described as a partial offloading strategy in all the subsequent sections. The other part of the tasks of the ith UD will be handed over to UMs. The delay required for this part of the tasks to be uploaded can be expressed as

$$T_{i,j}^{\text{up}}(t) = \frac{a_{i,j}(t)\varphi_{i,j}(t)L_i(t)}{R_{i,j}(t)}. \tag{23}$$

Then the energy consumption required for uploading is

$$E_{i,j}^{\text{up}}(t) = p_{i,j}(t)T_{i,j}^{\text{up}}(t). \tag{24}$$

The consumed latency and energy required to process the portion of the task offloaded to the UMs can be expressed as

$$T_{i,j}^{um}(t) = \frac{a_{i,j}(t)C_{um}\varphi_{i,j}(t)L_i(t)}{F_{um}} \tag{25}$$

$$E_{i,j}^{um}(t) = k_{um}F_{um}^3 T_{i,j}^{um}(t) \tag{26}$$

where F_{um} is the computational resource of UMs and k_{um} is the effective capacitance coefficient of UMs.

During the processes of offloading tasks from UDs to UMs, the EU eavesdrops on the tasks offloaded by the UDs. The size of the data volume that the EU eavesdrops on to the ith UD is written as

$$L_i^e(t) = \begin{cases} \sum_{j\in\mathcal{J}} a_{i,j}(t)R_{i,e}(t)T_{i,j}^{\text{up}}(t), & \text{partial offloading} \\ 0, & \text{local computing.} \end{cases} \tag{27}$$

C. Flight Model

UMs consume energy in addition to processing tasks, and another portion of their energy is used to generate thrust to maintain the UMs in flight. Similar to [30], we build a 3-D quadrotor UAV flight model that only considers acceleration in the same straight line as the velocity of the UMs, and does not consider acceleration in the vertical direction. The thrust of each rotor can be expressed as

$$F(\boldsymbol{v},\boldsymbol{a}) = \frac{1}{n}\left\|(m\|\boldsymbol{a}\| + \frac{1}{2}\rho v^2 S_a)\boldsymbol{v} - m\boldsymbol{g}\right\| \tag{28}$$

where $\boldsymbol{a}$ denotes the acceleration vector, $\boldsymbol{v}$ is the flight velocity vector, $v = |\boldsymbol{v}|$ is the velocity scalar of single UM, m is the mass of a single UM, n is the number of rotors of the UMs, ρ is the air density, S_a denotes the equivalent flat surface area, and $\boldsymbol{g}$ denotes the gravitational acceleration vector. According to [30], the power required for propulsion of an individual UM is expressed in (29), shown at the bottom of the page, where c_r is the blade local cross section drag coefficient, A is the rotor disc area, s_r is the rotor solidity, c_t denotes the thrust coefficient based on the disc area, d_r and c_f are the incremental correction coefficients for the induced power and fuselage drag ratios, respectively, and θ_c denotes the angle between the flight direction and the horizontal direction

The UMs process tasks by using queue scheduling, while the UDs are processing at the same time, so at time slot t the flight time of the jth UM is

$$T_j^{\text{fly}}(t) = \max\left\{\max\left\{T_{i,j}^{\text{loc}}(t)\right\}, T_{i,j}^{\text{up}}(t) + T_{i,j}^{um}(t)\right\}. \tag{30}$$

After obtaining the flight propulsion power and the flight time of individual UM, we can calculate the flight energy consumption of each UM, i.e., at time slot t the jth UM's flight energy consumption is

$$E_j^{\text{fly}}(t) = P^{\text{fly}}(t)T_j^{\text{fly}}(t). \tag{31}$$

III. Problem Formulation

To prevent individual UM connections from being overloaded, in this article, we equalize the number of UDs connected on each UM. According to the Jain's fairness index [31], at time slot t the average load on the jth UM is

$$C_j(t) = \frac{\sum_{i\in\mathcal{I}} a_{i,j}(t)\varphi_{i,j}(t)}{I}. \tag{32}$$

By the general form of the Cauchy-Schwartz inequality, for any real numbers $a_k, b_k, k \in N^+$, there is $(\sum_{k=1}^K a_k b_k)^2 \le \sum_{k=1}^K a_k^2 \sum_{k=1}^K b_k^2$. Hence, we get

$$\sum_{j\in\mathcal{J}} C_j(t)^2 \sum_{j\in\mathcal{J}} \bar{C}_j(t)^2 \ge \left(\sum_{j\in\mathcal{J}} C_j(t)\bar{C}_j(t)\right)^2 \tag{33}$$

where the condition for the equality sign to hold is $[C_1(t)/\bar{C}_1(t)] = [C_2(t)/\bar{C}_2(t)] = \cdots = [C_J(t)/\bar{C}_J(t)]$. Let $\bar{C}_1(t) = \bar{C}_2(t) = \cdots = \bar{C}_J(t)$. We let the inequality transform as

$$\frac{\left(\sum_{j\in\mathcal{J}} C_j(t)\right)^2}{J\left(\sum_{j\in\mathcal{J}} C_j(t)^2\right)} \le 1. \tag{34}$$

Let the left side of (34) be $C(t)$, which denotes the fairness metric between UMs. When $C(t) = 1$ then it indicates that the load is perfectly balanced among UMs.

This article is intended to optimize the system by minimizing the weighted sum of the total delay and total energy

$$P^{\text{fly}}(v,F) = n\left[\frac{c_r}{8}\left(\frac{F}{c_t\rho A} + 3v^2\right)\sqrt{\frac{F\rho s_r^2 A}{c_t}} + (1+c_f)F\left(\sqrt{\frac{F^2}{4\rho^2A^2} + \frac{v^4}{4}} - \frac{v^2}{2}\right)^{0.5} + 0.5d_r v^3 \rho s_r A + \frac{m\|\boldsymbol{g}\|v}{n}\sin\theta_c\right] \tag{29}$$

consumed, under the condition of considering UMs load balancing and communication security. Total energy consumption includes UMs' energy consumption for flight and computation, and the computational and offload energy consumption of the UDs. The total delay includes the computational delay of each task as well as the offloading delay. Specifically, the optimization objective can be expressed as

$$Q(t) = \frac{1}{C(t)} \sum_{j\in\mathcal{J}} \Bigg[\sum_{i\in\mathcal{I}} \Big(E_{i,j}^{\mathrm{loc}}(t) + E_{i,j}^{\mathrm{up}}(t) + E_{i,j}^{um}(t) \Big) + \sum_{i\in\mathcal{I}} \max\Big\{ T_{i,j}^{\mathrm{loc}}(t), T_{i,j}^{\mathrm{up}}(t) + T_{i,j}^{um}(t) \Big\} + \omega E_j^{\mathrm{fly}}(t) \Bigg] \quad (35)$$

where ω denotes the weighted value of flight energy consumption.

By optimizing the trajectory of UMs $M_j(t)$, the offloading strategy of UDs $a_{i,j}(t)$ and the transmit power $p_{i,j}(t)$, it reduces the weighted sum of the total energy and the total delay consumed by the system. We formulate the optimization problem as

$$Q = \min_{\{M_j(t)\},\{A_{i,j}(t)\},\{p_{i,j}(t)\}} \sum_{t=1}^{T} Q(t) \quad (36a)$$

$$\text{s.t.} \quad (1), (2), (3), (13), (16), (17) \quad (36b)$$

$$X_l \le X_j(t) \le X_r \quad \forall j \in \mathcal{J} \quad \forall t \in \mathcal{T} \quad (36c)$$

$$Y_d \le Y_j(t) \le Y_u \quad \forall j \in \mathcal{J} \quad \forall t \in \mathcal{T} \quad (36d)$$

$$H_l \le H_j(t) \le H_h \quad \forall j \in \mathcal{J} \quad \forall t \in \mathcal{T} \quad (36e)$$

$$\sum_{j\in\mathcal{J}} a_{i,j}(t) = 1 \quad \forall i \in \mathcal{I} \quad \forall t \in \mathcal{T} \quad (36f)$$

$$\sum_{i\in\mathcal{I}} a_{i,j}(t) \le N_{\max} \quad \forall j \in \mathcal{J} \quad \forall t \in \mathcal{T} \quad (36g)$$

$$L_i^e(t) \le L_e^{\max} \quad \forall i \in \mathcal{I} \quad \forall t \in \mathcal{T} \quad (36h)$$

where $M_j(t)$ denotes the trajectory optimization variable for UMs. $A_{i,j}(t)$ denotes the optimization variable for the offloading strategy of UDs. $p_{i,j}(t)$ denotes the user transmit power optimization variable. Constraints (36c)–(36e) denote the position constraints of UMs, which can only fly in the specified space. Constraint (36f) denotes that each UD must and can only be associated with one UM at any time slot. Constraint (36g) denotes the number of association constraints of UMs, where the number of UDs associated with each UM in any time slot cannot exceed the tolerance range. Constraint (36h) denotes the communication security constraint, i.e., the amount of information eavesdropped on each UD at any time slot does not exceed the maximum security bound.

IV. Theoretical Analysis and Algorithm Design

We propose an algorithm in this section to optimize the problem in (36) by jointly optimizing the trajectory, the offloading strategy, and the launch power. Since the multivariate coupling, binary constraints, and (36) are nonconvex, (36) is also nonconvex. We divide this nonconvex optimization problem into three subproblems so as to solve it.

A. Power Control

We fix the trajectory of UMs $M_j(t)$ and the offloading strategy of UDs $A_{i,j}(t)$ to discuss the transmit power problem of UDs. Since the variables related to the transmit power $p_{i,j}(t)$ in the t-time slot optimization objective $\bar{Q}(t)$ include only the transmission energy consumption and delay, thus the optimization problem simplifies to

$$\bar{Q}(t) = \min_{p_{i,j}(t)} \sum_{i\in\mathcal{I},\varphi\neq 0} \Big(E_{i,j^*}^{\mathrm{up}}(t) + T_{i,j^*}^{\mathrm{up}}(t) \Big) \quad (37a)$$

$$\text{s.t. } 0 \le p_{i,j^*}(t) \le P_{\max} \quad \forall i \in \mathcal{I} \quad (37b)$$

where j^* denotes the association selection of UDs after fixing the offloading policy. From (37a), $\bar{Q}(t)$ is the sum of the latency and energy consumed by the unloading of per UD. Therefore, the optimization problem can be reduced to the transmit power problem for a single UD, i.e., finding the minimum value of $\bar{Q}_i(p_{i,j^*}(t))$.

Theorem 1: $\bar{Q}_i(p_{i,j^*}(t))$ has only one minimal value in the interval $p_{i,j^*}(t) \in (0, +\infty)$.

Proof: The expansion of $\bar{Q}_i(p_{i,j^*}(t))$ is expressed as

$$\bar{Q}_i(p_{i,j^*}(t)) = \frac{(1 + p_{i,j^*}(t))\varphi_{i,j}(t) L_i(t)}{B \log_2\left(1 + \frac{p_{i,j^*}(t) h_{i,j^*}(t)}{\delta_0^2}\right)} \quad \varphi_{i,j^*}(t) \neq 0. \quad (38)$$

Find the derivative $\bar{Q}_i'(p_{i,j^*}(t))$ with respect to $p_{i,j^*}(t)$ for $\bar{Q}_i(p_{i,j^*}(t))$, which is denoted by

$$\bar{Q}_i'(p_{i,j^*}(t)) = \mu_p \nu_p \quad (39)$$

where μ_p and ν_p are denoted as

$$\mu_p = \frac{\ln 2 \varphi_{i,j^*}(t) L_i(t) h_{i,j^*}(t)}{B\delta_0^2 \left(1 + \frac{h_{i,j^*}(t)}{\delta_0^2} p_{i,j^*}(t)\right) \ln^2\left(1 + \frac{h_{i,j^*}(t)}{\delta_0^2} p_{i,j^*}(t)\right)} \quad (40)$$

$$\nu_p = \left(p_{i,j^*}(t) + \frac{\delta_0^2}{h_{i,j^*}(t)}\right) \ln\left(1 + \frac{h_{i,j^*}(t)}{\delta_0^2} p_{i,j^*}(t)\right) - p_{i,j^*}(t) - 1. \quad (41)$$

From (40), it can be seen that u_p is always higher than 0 regardless of the value of $p_{i,j^*}(t)$. Therefore, we only need to discuss the variation of ν_p. The initial and final values of ν_p in the interval $p_{i,j^*}(t) \in (0, +\infty)$ are

$$\nu_p(0) = -1 < 0 \quad (42)$$

$$\lim_{p_{i,j*}(t)\to+\infty} \nu_p = +\infty. \quad (43)$$

Equations (42) and (43) prove the existence of zero-valued points of ν_p in the interval $(0, +\infty)$. We take the derivative with respect to p for ν_p to get

$$\nu_p' = \ln\left(1 + \frac{h_{i,j^*}(t)}{\delta_0^2} p_{i,j^*}(t)\right). \quad (44)$$

From (44) it follows that the value of ν_p' in the interval $(0, +\infty)$ is constantly greater than 0, so ν_p is monotonically increasing. Let $p_{i,j^*}(t) = p^*$ when $\nu_p = 0$, so that the value of $\bar{Q}_i(p_{i,j^*}(t))$ at that point takes the minimum value. ■

To satisfy the constraints of (37b), comparing p^* with $P_{\max}$, the ith UD's transmit power at t slot is finally given by

$$p^*_{i,j^*}(t) = \begin{cases} p^*, & p^* < P_{\max} \\ P_{\max}, & p^* \geq P_{\max}. \end{cases} \tag{45}$$

B. Offloading Strategy Optimization

To reduce the task processing latency of a single UD, we optimize the unloading ratio to obtain the minimum value, i.e., solving the problem of $\min\{\max\{T^{\text{loc}}_{i,j}(t), T^{\text{up}}_{i,j}(t)+T^{um}_{i,j}(t)\}\}$. From the increasing and decreasing characteristics of the two parameters in the equation, the minimum latency value is achieved when $T^{\text{loc}}_{i,j}(t) = T^{\text{up}}_{i,j}(t) + T^{um}_{i,j}(t)$ [20]. Therefore, the optimal offloading ratio is given by

$$\varphi_{i,j}(t) = \frac{C_{um}F_{i,j}(t)R_{i,j}(t)}{F_{ud}F_{um} + (C_{um}F_{ud} + C_{ud}F_{um})R_{i,j}(t)}. \tag{46}$$

Fixing the trajectory $M_j(t)$ of the UMs and by having optimized the user launch power $p_{i,j}(t)$ and the offloading ratio $\varphi_{i,j}(t)$, we come to discuss the offloading selection problem for UDs. For communication security reasons, in (18) we define a security scaling factor. We assume that the ith UD chooses to associate with the jth UM, and if the $s_{i,j}(t)$ of the communication link between them is lower than the minimum security tolerance limit, we make the offloading ratio $\varphi_{i,j}(t) = 0$ for the ith UD.

In terms of association policies for UDs, since the number of UDs as well as the number of UMs is fixed, the dimensions of the association policies are fixed, we only need to select the optimal policy within a limited combination of policies. What is affected by different strategies is the data transfer rate between UDs and UMs. From (8), (11), and (14), the data transfer rate is inversely proportional to the distance of transmission, so the task processing delay is proportional to the distance of transmission. In [20] it is proved that the energy consumption is proportional to the transmission distance, so the strategy of choosing the UM with the shortest distance can be used as the initial strategy without considering fairness. We take the minimization of $Q(t)$ in (35) as the objective of the optimization of the association strategy and denote it as

$$\begin{aligned} &\min_{a_{i,j}(t)} \quad Q(t) \\ &\text{s.t.} \quad (16),\ (36f),\ (36g). \end{aligned} \tag{47}$$

We define the association policy of UDs and the association policy of removing the ith UD as

$$\bar{A}^*_I = \left\{\bar{A}_1, \bar{A}_2, \ldots, \bar{A}_I\right\} \tag{48}$$

$$\bar{A}^*_{-i} = \left\{\bar{A}_1, \bar{A}_2, \ldots, \bar{A}_{i-1}, \bar{A}_{i+1}, \ldots, \bar{A}_I\right\} \tag{49}$$

where $\bar{A}_i$ is the association choice of the ith UD.

Define $\min Q(\bar{A}^*_{-i}|\bar{A}_i)$ as the basis of the ith UD's choice preference. When in Nash equilibrium, if $\bar{A}^*_{-i}$ is unchanged, a change in association choice $\bar{A}_i$ of any UD will not reduce the value of Q.

Definition 1: If $Q(\bar{A}^*_I) \leq Q(\bar{A}^*_{-i}|\bar{A}_i)$ is satisfied, then the set of associations of the I UDs is a Nash equilibrium in the game process.

Algorithm 1: Matching Based on Nash Equilibrium

```
Initialize: UDs select UMs to match based on nearest distance;
for j = 1 : J do
    if ∑_{i∈I} a_{i,j}(t) > N_max then
        ind = arg min(h_{i,j}(t), N_max − ∑_{i∈I} a_{i,j}(t));
        Corresponding UDs in ind select the next closest UM for association;
    end
end
Calculate the current Q(t);
while No strategy update do
    for i = 1 : I do
        for j = 1 : J do
            if a_{i,j}(t) == 0 and ∑_{i∈I} a_{i,j}(t) < N_max then
                a_{i,j}(t) = 1;
                Calculate the Q'(t) after updating the policy;
                if Q'(t) < Q(t) then
                    Modify the optimal policy to the updated policy;
                    Q(t) = Q'(t);
                else
                    a_{i,j}(t) = 0;
                end
            end
        end
    end
end
```

According to Definition 1, when the associated strategy satisfies the Nash equilibrium condition, then the current associated strategy is the optimal strategy. Therefore, we propose an algorithm for strategy matching based on Nash equilibrium, and the specific steps are shown in Algorithm 1.

C. UMs Trajectory Optimization

Based on the optimization of the launch power as well as the unloading strategy, we come to discuss the optimization of the flight trajectory of UMs. Since the UMs in our designed UAV-assisted MEC system model fly in 3-D space and the parameter space of UMs and UDs is too large, we consider the optimization of the flight trajectories of UMs using a MADRL algorithm. In this article, we discuss a problem of continuous actions optimization in a 3-D space. Therefore, we optimize the trajectories of the UMs using the MADDPG algorithm. We characterize the problem as a Markov decision process, Specifically, the main components are state space, action space, reward, and state transfer probabilities (S, A, R, P).

1) States: The state space primarily encompasses the states of the elements within the environment, so we define S as

$$\begin{aligned} S = &\{M_j(t), M_{eu}(t), M_{gj}, M_i(t), L_i(t)\} \\ &i \in \mathcal{I}, j \in \mathcal{J}, t \in \mathcal{T} \end{aligned} \tag{50}$$

where $M_j(t)$, $M_{eu}(t)$, M_{gj}, and $M_i(t)$ denote the coordinates of the UMs, the coordinate of the EU, the coordinate of the GJ,

and the coordinates of the UDs at time slot t, respectively, and $L_i(t)$ denotes the size of the generated tasks of the UDs at time slot t.

2) Actions: The maneuvering space consists of the flight distances, the horizontal deflection angles, and the vertical deflection angles of the UMs, Define A as

$$A = \left\{ q_j(t), \theta_h^j(t), \theta_v^j(t) \right\}, j \in \mathcal{J}, t \in \mathcal{T} \tag{51}$$

where $q_j(t)$ is the flight distance of the jth UM at t slot. The position update of the jth UM at t slot is given by

$$\begin{aligned} X_j(t+1) &= X_j(t) + q_j(t)\cos(\theta_h^j(t))\cos(\theta_v^j(t)) \\ Y_j(t+1) &= Y_j(t) + q_j(t)\sin(\theta_h^j(t))\cos(\theta_v^j(t)) \\ H_j(t+1) &= H_j(t) + q_j(t)\sin(\theta_v^j(t)). \end{aligned} \tag{52}$$

3) Reward: In this article, our goal is to maximally reduce the weighted sum of the system's energy consumption and delay, considering communication security as well as fairness, while RL is intended to maximize the cumulative reward, so we define R as

$$R = \sum_{j\in\mathcal{J}} R_j = \sum_{j\in\mathcal{J}} (-\sum_{t\in\mathcal{T}} Q_j(t)) \tag{53}$$

where R_j denotes the part of $Q(t)$ associated with the jth UM, i.e., $Q(t) = \sum_{j\in\mathcal{J}} Q_j(t)$.

In order to train the model better, we give the corresponding penalization mechanism for different nonideal cases.

1) *UMs Fly Out of Bounds Penalty:* During the training process, once one of the UMs flies out of the boundary, we give a penalty to the corresponding agent

$$R_j = R_j - R_{\text{out}}. \tag{54}$$

2) *UMs Collision Penalty:* We do not tolerate UMs collisions, so in the event of a collision, negative infinity is added to the total reward and then the training goes straight to the next round

$$R = R - \infty. \tag{55}$$

3) *Failure to Meet Communication Security Requirements:* If the poor trajectory optimization of UMs leads to compromised communication security, we impose a corresponding penalty based on exceeding the communication security requirements

$$R = R - \varepsilon\left(L_i^e(t) - L_e^{\max}\right) \tag{56}$$

where ε denotes the penalty factor for communication security.

4) State Transfer Probabilities: The probability of taking action a to move to the next transitive state s' at state s is defined as $P = \{p(s'|s,a) \quad \forall s, s' \in S, a \in A\}$.

The improvement of MADDPG over the traditional AC algorithm is that MADDPG uses a centralized training and distributed execution framework. During training, each agent's critic network aggregates the states and actions of all agents to compute a Q-value, while each agent's actor network decides actions-based solely on its own local state. The Critic network is extended to allow learning using the strategies of other agents, and a further improvement of this point is that each intelligence approximates a function to the strategies of the other agents. The algorithm is designed along the following lines.

1) *Multiagent AC Design:* We denote the parameters of the strategies of the nth agent by $\theta = [\theta_1, \ldots, \theta_N]$, and $\mu = [\mu_1, \ldots, \mu_N]$ denotes the deterministic strategies of the N agents. The cumulative expected reward for the nth agent $J(\theta_n) = \mathbb{E}_{s,a\sim D}[\sum_{t\in\mathcal{T}} \gamma r_{n,t}]$. γ denotes the rewards discount factor. Then the strategy gradient is

$$\begin{aligned} \nabla_{\theta_n} J(\mu_n) = \mathbb{E}_{s,a\sim D}\Big[& \nabla_{\theta_n} \mu_n(a_n|o_n) \nabla_{a_n} \\ & \times Q_n^{\mu}(s, a_1, a_2, \ldots, a_N)|_{a_n=\mu_n(o_n)} \Big] \end{aligned} \tag{57}$$

where D denotes the experience replay buffer with elements composed of $\{s, a, r, s', \text{done}\}$, o_n denotes the observation of the nth agent. Q_n^{μ} denotes the centralized state-action function of the nth agent. We update the critic network parameters by minimizing the loss function, which is written as

$$\mathcal{L}(\theta_n) = \mathbb{E}_{s,a,r,s'}\Big[(Q_n^{\mu}(s, a_1, a_2, \ldots, a_N) - y)^2 \Big] \tag{58}$$

where y is denotes as

$$y = r_n + \gamma Q_n^{\mu'}(x', a_1', a_2', \ldots, a_N')|_{a_j'=\mu_j'(o_j)} \tag{59}$$

where $Q_n^{\mu'}$ denotes the target network and $\mu' = [\mu_1', \mu_2', \ldots, \mu_N']$ is the parameter θ_j' for which the target policy has a lagged update. θ_j' can be obtained by delayed fitting approximation.

2) *Policies Ensemble:* Since each agent's strategy is updated iteratively causing the environment to be dynamically unstable for a particular agent, in order to be able to better cope with this situation, MADDPG utilizes the idea of strategy ensembles: the nth agent's strategy μ_n consists of a set with K substrategies, and just one substrategy $\mu_n^{(k)}$ is used in each training episodes. for each agent, we maximize the overall reward of its strategy ensemble $J_e(\mu_n) = \mathbb{E}_{k\sim\text{unif}(1,K),s,a\sim D^{(k)}}$. And we construct a memory store $D_i^{(k)}$ for each substrategy k. We optimize the overall effect of the strategy ensemble so that the update gradient for each substrategy is

$$\begin{aligned} \nabla_{\theta_n^{(k)}} J_e(\mu_n) = \frac{1}{K}\mathbb{E}_{s,a\sim D_n^{(k)}}\Big[& \nabla_{\theta_n^{(k)}} \mu_n^{(k)}(a_n|o_n) \\ & \nabla_{a_n} Q_n^{\mu}(s,a)|_{a_n=\mu_n^{(k)}(o_n)} \Big]. \end{aligned} \tag{60}$$

Combining the previous power control as well as the offloading strategy optimization, our proposed joint optimization algorithm is shown in Algorithm 2.

V. Simulation Results

In this section, we give the simulation results of our proposed joint optimization algorithm. The range of the site

Algorithm 2: Proposed Joint Optimization Algorithm

Initialize: actor policy network μ, target policy network μ' with weights θ, θ' for all agents, replay buffer D;
for *episode* $= 1 :$ max_*episode* **do**
  Initialize a random process $\mathcal{N}$ for action exploration;
  Initialize the state s_0 and $step = 0$;
  while *done* $==$ *False* **do**
    $step = step + 1$;
    Select action $a_n = \mu_{\theta_n}(o_n) + \mathcal{N}_{step}$ for each agent based on current policy and exploration;
    Input actions $a = (a_1, \ldots, a_N)$ into the environment;
    According to (41) (46) and Algorithm 1 to obtain the optimal transmit power and offloading strategy;
    Obtain the reward r and the next state s' for all agents;
    Store $(s, a, r, s', done)$ in replay buffer D;
    $s \leftarrow s'$;
    for *agent* $n = 1 : N$ **do**
      Sample a random minbatch of S (s^j, a^j, r^j, s'^j) from D;
      Set y^j by (59);
      Update critic by minimizing the loss $\mathcal{L}(\theta_n)$ in (58);
      Update actor using the sampled policy gradient $\bigtriangledown_{\theta_n} J$ in (57);
    end
    Update target network parameters for each agent: $\theta'_n \leftarrow \tau\theta_n + (1-\tau)\theta'_n$;
    if *step* $==$ max_*step or UMs collide* **then**
      $done_n = True$;
      $R = \sum_{t=1}^{step} r(t)$;
    end
    $done = [done_1, \ldots, done_N]$;
  end
end

for our simulation is 500×500 m^2. The positions of the UDs, UMs, EU, and GJ are within this range at each time slot. The UDs appear randomly within the site. The flight altitude of the UMs ranges from 100 m to 300 m, with a starting altitude of 200 m. The flight altitude of the EU is 100 m, and the trajectory of the EU is fixed as a flight path from (0, 500, 100) to (500, 0, 100). In this article, simulations are performed on Python 3.7, and the parameters of the reinforcement learning algorithm are set as follows: learning rate $lr = 0.001$, reward discount factor $\gamma = 0.95$, and soft update parameter $\tau = 0.001$. In addition, the rest of the simulation parameters are set as specified in Table I.

As shown in Fig. 2, we give the training rewards of each agent and the total training rewards of the system. From the figure, we can see that the training results of UM_1 and UM_2 are similar because they are assigned similar amounts of tasks after equalization, and thus the energy consumption and delay are also similar. In addition, the convergence results

TABLE I
RELEVANT PARAMETERS

Parameters	Value
The number of UDs I	15
The number of UMs J	2
Maximum flight distance q^{max}	40 m
Maximum transmit power P_{max}	0.5 w
transmit power of GJ p_g	0.3 w
Minimum safe distance between UMs d_{min}	5 m
Starting point coordinates of UM_1 MS_1	(0,0,200)
Starting point coordinates of UM_2 MS_2	(500,0,200)
Location of GJ M_{gj}	(250,250,0)
The minimum security scaling factor s_{min}	100
The channel bandwidth B	1 MHz
The noise power δ_0^2	10^{-13} w/Hz
Channel gain per unit distance β_0	10^{-5}
Propagation environment type constant λ_1, λ_2	15, 0.5
The path loss exponent η	2.3
NLoS attenuation κ	0.2
Computing resources per UD F_{ud}	1 GHz
Required CPU cycles per bit computation per UD C_{ud}	1000 cycles/bit
CPU effective capacitance factor per UD k_{ud}	10^{-27}
The amount of data per UD L	$[10^5, 10^6]$ bits
Computing resources for UM F_{um}	5 GHz
Required CPU cycles per bit computation for UM C_{um}	800 cycles/bit
CPU effective capacitance factor for UM k_{um}	10^{-28}
Maximum number of associations per UM N_{max}	10
The number of rotors per UM n	4
The weight per UM m	2 Kg
The equivalent flat surface area S_a	0.01 m^2
The gravitational acceleration g	9.8 m/s^2
The local blade section drag coefficient c_r	0.012
The thrust coefficient based on disk area c_t	0.302
The incremental correction factor c_f	0.121
The fuselage drag ratio d_r	0.834
The air density ρ	1.225 Kg/m^3
The rotor solidity s_r	0.0955
The rotor disk area A	0.0314 m^2
The weight of per UM's flight energy ω	10^{-4}

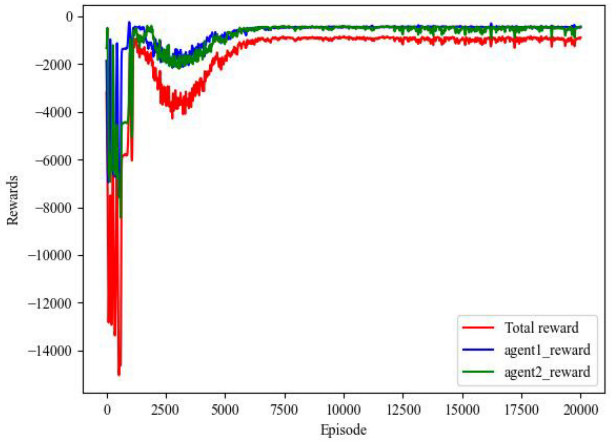


Fig. 2. Training rewards for each agent and all agents.

show that convergence is basically reached after about 7000 episodes of training, so in terms of training effectiveness, our proposed algorithm can train a better model in a short period of time.

The fairness convergence result of the system is shown in Fig. 3, similar to Fig. 2, fairness reaches convergence at about 7000 rounds of training, and the fairness stays around 0.98, which is enough to keep UMs in the load-balanced state at any time slot. Therefore, our

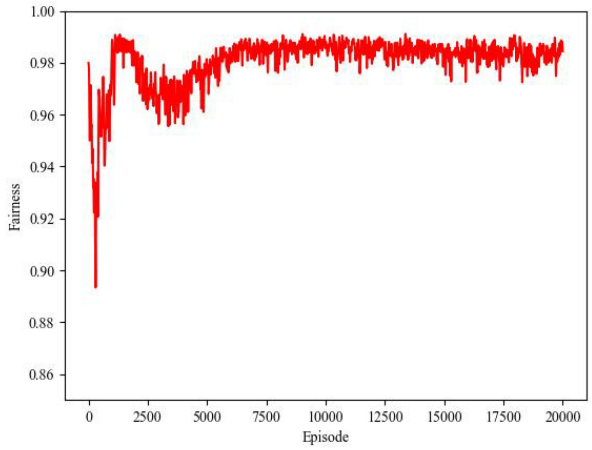


Fig. 3. Fairness convergence results.

proposed algorithm can meet the higher load balancing requirements.

Our proposed algorithm is compared with the algorithm with fixed transmit power. In this case, the transmit power of the comparison algorithms are set to 0.2w and 0.5w, while the other variables are optimized by using our proposed algorithm. As shown in Figs. 4 and 5, we give the average computational energy consumption and average computational delay for each UM per time slot as the number of UDs varies and the number of UMs varies. As shown in Figs. 4(a) and 5(a), as the number of UDs increases, the number of tasks to be processed also increases, and hence the average energy consumption and delay also increase as shown in the figure. As shown in Figs. 4(b) and 5(b), as the number of UMs increases, the computational resources of the MEC server are more sufficient, and the UDs can offload more tasks to the UMs to be processed, so the average computational energy consumption and latency of the system are lower, however, with the increase in the number of UMs, the computational resources of the MEC server are too sufficient, which results in the failure to bring a greater advantage in terms of computational energy consumption and latency. Moreover, over-increasing the number of UMs will result in resource waste and large amount of flight energy loss. Finally, from Figs. 4 and 5, it can be seen that our proposed algorithm outperforms the comparison algorithms in terms of energy consumption and delay due to the optimization of the transmit power of the UDs.

Our proposed algorithm is compared with algorithms for different offloading strategies. The comparison algorithms are as follows.

1) *Nearest:* All UDs select the nearest UM for offloading, if the number of associations of the nearest UM reaches the upper limit then select the next nearest UM. other variables are optimized by using our proposed algorithm.
2) *Random:* All UDs randomly select UMs for offloading, and if the number of associations of the selected UM reaches the upper limit, then randomly select among other UMs. Other variables are optimized by using our proposed algorithm.

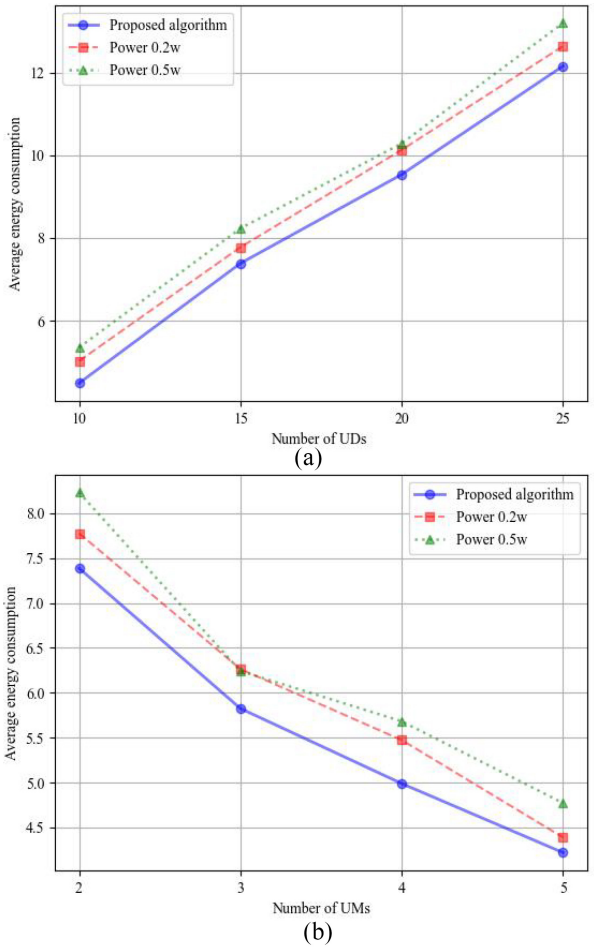


Fig. 4. Comparison of average computational energy consumption of different transmit power optimization algorithms. (a) Effect of the number of UDs on the average computational energy consumption. (b) Effect of the number of UMs on the average computational energy consumption.

As shown in Figs. 6 and 7, we give the average computational energy consumption and latency of different algorithms with varying number of UDs as well as with varying number of UMs. Similarly, as shown in Figs. 6(a) and 7(a), as the number of UDs increases, the number of computational tasks to be processed increases, and the average computational energy consumption and latency also increases. As shown in Figs. 6(b) and 7(b), as the number of UMs increases, the computational resources of the system become more adequate and the average computational energy consumption and latency decrease. From the figures, we can see that our proposed algorithm can obtain the optimal offloading strategy after reaching the Nash equilibrium, so our proposed algorithm has obvious advantages in offloading strategy optimization.

In Fig. 8, we give the effect of different minimum safety scale factors on the safety as well as the reward after training convergence. Since the coefficient varies over a wide range, we take it to be logarithmic. From the figure, it can be seen that the security of the system increases as $s_{\min}$ increases, while the reward after training decreases slower in the initial phase and faster after $\lg(s_{\min}) > 2$. This is due to the fact that when $s_{\min}$ is small, UDs require less

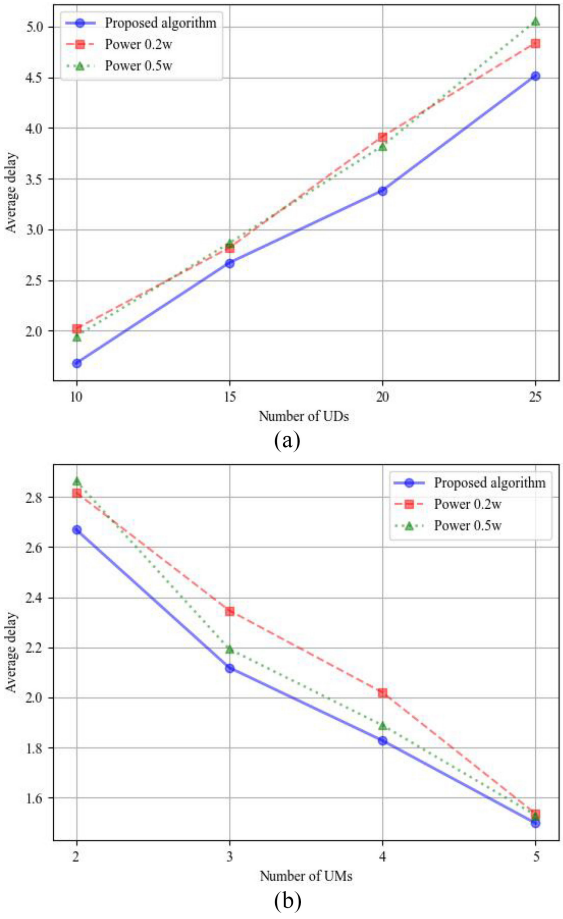


Fig. 5. Comparison of average computational delay of different transmit power optimization algorithms. (a) Effect of the number of UDs on the average computational delay. (b) Effect of the number of UMs on the average computational delay.

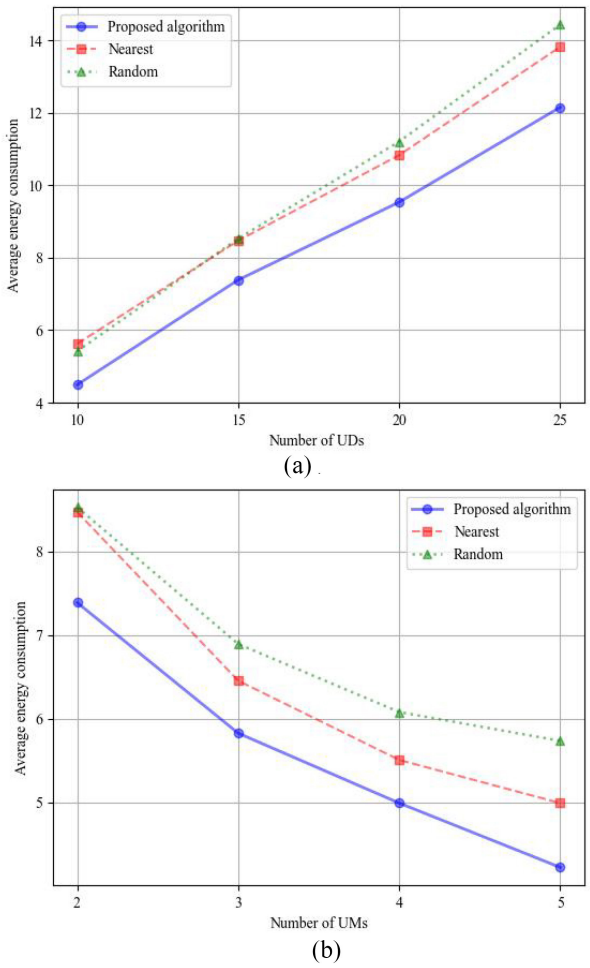


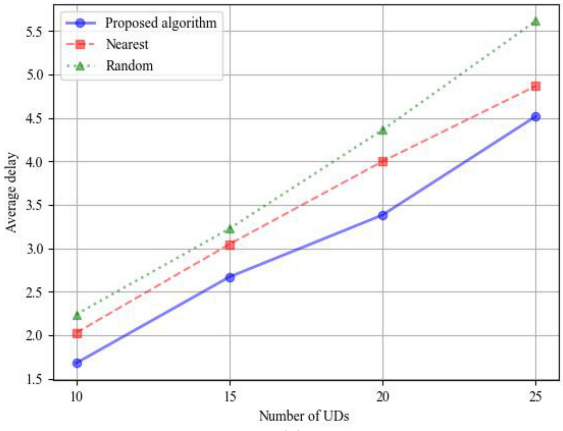
Fig. 6. Comparison of average computational energy consumption of different offloading strategy optimization algorithms. (a) Effect of the number of UDs on the average computational energy consumption. (b) Effect of the number of UMs on the average computational energy consumption.

security in the channel and more UDs can choose to offload tasks to UMs for processing, hence the reward value is higher while security is low. As $s_{\min}$ increases, the security requirements of UDs on the channel increase, and more UDs choose to process tasks locally, thus the reward value is low and security is high. In addition, if we need to satisfy both reward and security requirements, we can see from the figure that we only need to choose $\lg(s_{\min})$ in the range of [2.0, 2.5].

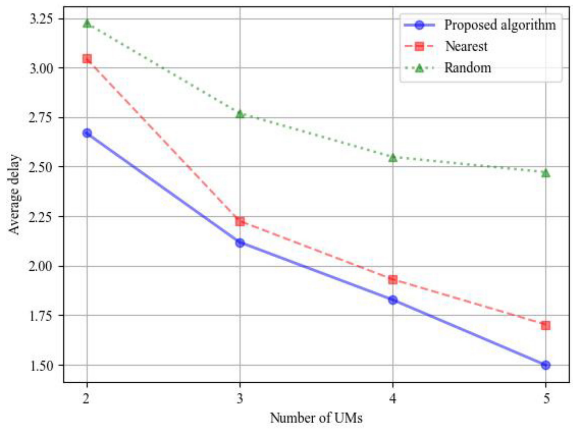
In Fig. 9, we give the trajectories of the UMs, where the green curve and the purple curve indicate the flight trajectories of the UMs, the red straight line indicates the flight trajectory of the EU, the blue dots indicate the distribution of the positions of the UDs, and the yellow dot indicates the position of the GJ. As shown in Fig. 9(b), the altitudes of the UDs are randomly set in the range [0, 10] m, which are represented by varying shades of blue, with darker shades indicating higher altitudes. From Fig. 9(a), it is observed that UMs tend to fly to the areas with dense UDs. When UMs approach the area with fewer UDs, they ascend to increase the elevation angle for enhancing the LoS probability. In contrast, they descend when flying toward denser UD regions or closer to the EU, thereby reducing system energy consumption and delay while improving communication security. In Fig. 9(b), since the altitudes of UDs only affect the channel, it is also observed that our algorithm is well applicable to the practical scenarios with nonzero altitude. This indicates that our proposed algorithm could adjust the flight trajectory in time according to the changes of the environment.

VI. Conclusion

In this article, we investigated a 3-D multi-UAV-assisted MEC communication system. In order to prevent EU maliciously eavesdrops on the data transmission from the ground users, we deployed a jammer on the ground to interfere with the eavesdropping channel. We proposed a joint power control, offloading strategy, and optimization problem for the 3-D trajectories of UMs. Since this is a mixed integer nonconvex programming problem, it was decomposed into three subproblems. Specifically, at each time slot, we obtained the optimal solutions of the power control and offloading strategies through theoretical derivations and mathematical proofs. Then the MADDPG algorithm was used to optimize the trajectories of UMs in the whole time slot. Simulation results show that our proposed algorithm can effectively reduce the system energy consumption and delay while ensuring



(a)



(b)

Fig. 7. Comparison of average computational delay of different offloading strategy optimization algorithms. (a) Effect of the number of UDs on the average computational delay. (b) Effect of the number of UMs on the average computational delay.

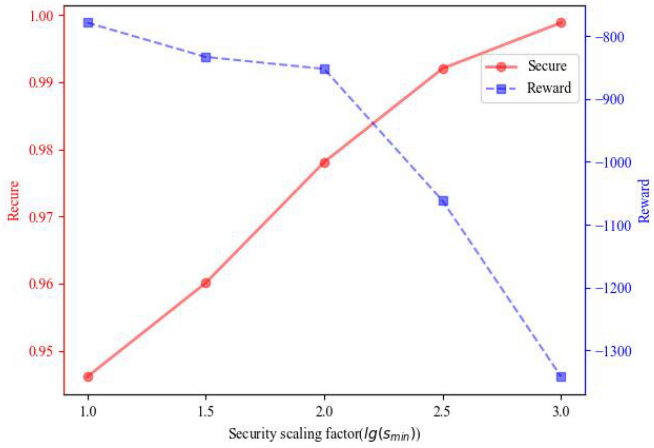
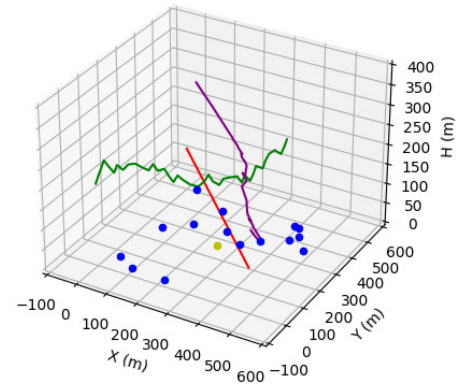
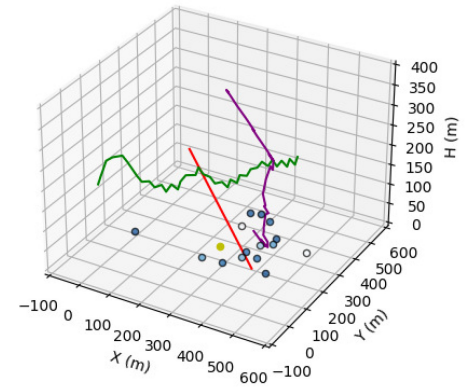


Fig. 8. Impact of different $s_{\min}$ on reward and security.

a high-level of fairness and communication security of the system.



(a)



(b)

Fig. 9. Optimized trajectories for UMs. (a) Altitudes of the UDs are 0. (b) Altitudes of the UDs are randomly set in the range [0, 10] m (the darker the blue color, the higher the altitude of the UD).

REFERENCES

[1] L. Qin, H. Lu, and F. Wu, "When the user-centric network meets mobile edge computing: Challenges and optimization," *IEEE Commun. Mag.*, vol. 61, no. 1, pp. 114–120, Jan. 2023.

[2] Y. Hou, C. Wang, M. Zhu, X. Xu, X. Tao, and X. Wu, "Joint allocation of wireless resource and computing capability in MEC-enabled vehicular network," *China Commun.*, vol. 18, no. 6, pp. 64–76, Jun. 2021.

[3] C. Souza, M. Falcao, A. Balieiro, and K. Dias, "Modelling and analysis of 5G networks based on MEC-NFV for URLLC services," *IEEE Latin America Trans.*, vol. 19, no. 10, pp. 1745–1753, Oct. 2021.

[4] Z. Xie, X. Song, J. Cao, and W. Qiu, "Providing aerial MEC service in areas without infrastructure: A tethered-UAV-based energy-efficient task scheduling framework," *IEEE Internet Things J.*, vol. 9, no. 24, pp. 25223–25236, Dec. 2022.

[5] L. P. Qian, H. Zhang, Q. Wang, Y. Wu, and B. Lin, "Joint multi-domain resource allocation and trajectory optimization in UAV-assisted maritime IoT networks," *IEEE Internet Things J.*, vol. 10, no. 1, pp. 539–552, Jan. 2023.

[6] B. Liu, Y. Wan, F. Zhou, Q. Wu, and R. Q. Hu, "Resource allocation and trajectory design for MISO UAV-assisted MEC networks," *IEEE Trans. Veh. Technol.*, vol. 71, no. 5, pp. 4933–4948, May 2022.

[7] M. El-Emary, A. Ranjha, D. Naboulsi, and R. Stanica, "Energy-efficient task offloading and trajectory design for UAV-based MEC systems," in *Proc. 19th Int. Conf. Wireless Mobile Comput., Netw. Commun. (WiMob)*, Montreal, QC, Canada, 2023, pp. 274–279.

[8] X. Qin, Z. Song, T. Hou, W. Yu, J. Wang, and X. Sun, "Joint optimization of resource allocation, phase shift, and UAV trajectory for energy-efficient RIS-assisted UAV-enabled MEC systems," *IEEE Trans. Green Commun. Netw.*, vol. 7, no. 4, pp. 1778–1792, Dec. 2023.

[9] D. Wang, J. Tian, H. Zhang, and D. Wu, "Task offloading and trajectory scheduling for UAV-enabled MEC networks: An optimal transport theory perspective," *IEEE Wireless Commun. Lett.*, vol. 11, no. 1, pp. 150–154, Jan. 2022.

[10] Z. Yu, Y. Gong, S. Gong, and Y. Guo, "Joint task offloading and resource allocation in UAV-enabled mobile edge computing," *IEEE Internet Things J.*, vol. 7, no. 4, pp. 3147–3159, Apr. 2020.

[11] Y. K. Tun, Y. M. Park, N. H. Tran, W. Saad, S. R. Pandey, and C. S. Hong, "Energy-efficient resource management in UAV-assisted mobile edge computing," *IEEE Commun. Lett.*, vol. 25, no. 1, pp. 249–253, Jan. 2021.
[12] Y. Liao, X. Chen, S. Xia, Q. Ai, and Q. Liu, "Energy minimization for UAV swarm-enabled wireless inland ship MEC network with time windows," *IEEE Trans. Green Commun. Netw.*, vol. 7, no. 2, pp. 594–608, Jun. 2023.
[13] Z. Hu, F. Zeng, Z. Xiao, B. Fu, H. Jiang, and H. Chen, "Computation efficiency maximization and QoE-provisioning in UAV-enabled MEC communication systems," *IEEE Trans. Netw. Sci. Eng.*, vol. 8, no. 2, pp. 1630–1645, Apr.–Jun. 2021.
[14] X. Deng, J. Zhao, Z. Kuang, X. Chen, Q. Guo, and F. Tang, "Computation efficiency maximization in multi-UAV-enabled mobile edge computing systems based on 3D deployment optimization," *IEEE Trans. Emerg. Topics Comput.*, vol. 11, no. 3, pp. 778–790, Jul.–Sep. 2023.
[15] B. Zhu, E. Bedeer, H. H. Nguyen, R. Barton, and J. Henry, "UAV trajectory planning in wireless sensor networks for energy consumption minimization by deep reinforcement learning," *IEEE Trans. Veh. Technol.*, vol. 70, no. 9, pp. 9540–9554, Sep. 2021.
[16] Y. Liu, J. Yan, and X. Zhao, "Deep reinforcement learning based latency minimization for mobile edge computing with virtualization in maritime UAV communication network," *IEEE Trans. Veh. Technol.*, vol. 71, no. 4, pp. 4225–4236, Apr. 2022.
[17] J. Chen et al., "Deep reinforcement learning based resource allocation in multi-UAV-aided MEC networks," *IEEE Trans. Commun.*, vol. 71, no. 1, pp. 296–309, Jan. 2023.
[18] M. Zhao, W. Li, L. Bao, J. Luo, Z. He, and D. Liu, "Fairness-aware task scheduling and resource allocation in UAV-enabled mobile edge computing networks," *IEEE Trans. Green Commun. Netw.*, vol. 5, no. 4, pp. 2174–2187, Dec. 2021.
[19] X. Zhou, L. Huang, T. Ye, and W. Sun, "Computation bits maximization in UAV-assisted MEC networks with fairness constraint," *IEEE Internet Things J.*, vol. 9, no. 21, pp. 20997–21009, Nov. 2022.
[20] Y. He, Y. Gan, H. Cui, and M. Guizani, "Fairness-based 3-D multi-UAV trajectory optimization in multi-UAV-assisted MEC system," *IEEE Internet Things J.*, vol. 10, no. 13, pp. 11383–11395, Jul. 2023.
[21] T. Li et al., "Energy-efficient and secure communication toward UAV networks," *IEEE Internet Things J.*, vol. 9, no. 12, pp. 10061–10076, Jun. 2022.
[22] Y. Shiu, S. Y. Chang, H. Wu, S. C. Huang, and H. Chen, "Physical layer security in wireless networks: A tutorial," *IEEE Wireless Commun*, vol. 18, no. 2, pp. 66–74, Apr. 2011.
[23] N. Rupasinghe, Y. Yapici, I. Guvenc, H. Dai, and A. Bhuyan, "Enhancing physical layer security for NOMA transmission in mmWave drone networks," in *Proc. 52nd Asilomar Conf. Signals, Syst., Comput.*, 2018, pp. 729–733.
[24] Z. Sheng, H. D. Tuan, A. A. Nasir, T. Q. Duong, and H. V. Poor, "Secure UAV-enabled communication using Han–Kobayashi signaling," *IEEE Trans. Wireless Commun.*, vol. 19, no. 5, pp. 2905–2919, May 2020.
[25] W. Lu et al., "Resource and trajectory optimization for secure communications in dual unmanned aerial vehicle mobile edge computing systems," *IEEE Trans. Ind. Informat.*, vol. 18, no. 4, pp. 2704–2713, Apr. 2022.
[26] W. Lu et al., "Secure NOMA-based UAV-MEC network towards a flying eavesdropper," *IEEE Trans. Commun.*, vol. 70, no. 5, pp. 3364–3376, May 2022.
[27] Y. Ding et al., "Online edge learning offloading and resource management for UAV-assisted MEC secure communications," *IEEE J. Sel. Topics Signal Process.*, vol. 17, no. 1, pp. 54–65, Jan. 2023.
[28] Y. Zhou et al., "Secure communications for UAV-enabled mobile edge computing systems," *IEEE Trans. Commun.*, vol. 68, no. 1, pp. 376–388, Jan. 2020.
[29] Z. Yang, S. Bi, and Y.-J. A. Zhang, "Online trajectory and resource optimization for stochastic UAV-enabled MEC systems," *IEEE Trans. Wireless Commun.*, vol. 21, no. 7, pp. 5629–5643, Jul. 2022.
[30] R. Ding, F. Gao, and X. S. Shen, "3D UAV trajectory design and frequency band allocation for energy-efficient and fair communication: A deep reinforcement learning approach," *IEEE Trans. Wireless Commun.*, vol. 19, no. 12, pp. 7796–7809, Dec. 2020.
[31] R. Jain, A. Durresi, and G. Babic, Throughput fairness index: An explanation, ATM Forum Contrib., Mountain View, CA, USA, document 99-0045, Feb. 1999.

Yejun He (Senior Member, IEEE) received the Ph.D. degree in information and communication engineering from Huazhong University of Science and Technology, Wuhan, China, in 2005.

From 2005 to 2006, he was a Research Associate with the Department of Electronic and Information Engineering, The Hong Kong Polytechnic University, Hong Kong. From 2006 to 2007, he was a Research Associate with the Department of Electronic Engineering, Faculty of Engineering, The Chinese University of Hong Kong, Hong Kong. In 2012, he was a Visiting Professor with the Department of Electrical and Computer Engineering, University of Waterloo, Waterloo, ON, Canada. From 2013 to 2015, he was an Advanced Visiting Scholar (Visiting Professor) with the School of Electrical and Computer Engineering, Georgia Institute of Technology, Atlanta, GA, USA. From 2023 to 2024, he is an Advanced Research Scholar (Visiting Professor) with the Department of Electrical and Computer Engineering, National University of Singapore, Singapore. Since 2006, he has been a Faculty of Shenzhen University, Shenzhen, China, where he is currently a Full Professor with the College of Electronics and Information Engineering, the Director of Sino-British Antennas and Propagation Joint Laboratory of Ministry of Science and Technology of the People's Republic of China, the Director of the Guangdong Engineering Research Center of Base Station Antennas and Propagation, and the Director of the Shenzhen Key Laboratory of Antennas and Propagation. He was selected as a Leading Talent with "Guangdong Special Support Program" and Shenzhen "Pengcheng Scholar" Distinguished Professor, China, in 2024 and 2020, respectively. He has authored or coauthored more than 300 refereed journal and conference papers and seven books. He holds about 20 patents. His research interests include wireless communications, antennas, and radio frequency.

Dr. He was also a recipient of the Shenzhen Overseas High-Caliber Personnel Level B (Peacock Plan Award B) and Shenzhen High-Level Professional Talent (Local Leading Talent). He received the Second Prize of Shenzhen Science and Technology Progress Award in 2017, the Three Prize of Guangdong Provincial Science and Technology Progress Award in 2018, the Second Prize of Guangdong Provincial Science and Technology Progress Award in 2023, and the 10th Guangdong Provincial Patent Excellence Award in 2023. He is currently the Chair of IEEE ANTENNAS AND PROPAGATION SOCIETY-Shenzhen Chapter and obtained the 2022 IEEE ANTENNAS AND PROPAGATION SOCIETY Outstanding Chapter Award. He has served as a Technical Program Committee Member or a Session Chair for various conferences, including the IEEE Global Telecommunications Conference, the IEEE International Conference on Communications, the IEEE Wireless Communication Networking Conference, and the IEEE Vehicular Technology Conference. He served as the TPC Chair for IEEE ComComAp 2021 and the General Chair for IEEE ComComAp 2019. He was selected as a Board Member of the IEEE Wireless and Optical Communications Conference (WOCC). He served as the TPC Co-Chair for WOCC 2023/2022/2019/2015, APCAP 2023, UCMMT 2023, ACES-China2023, and NEMO 2020. He acted as the Publicity Chair of several international conferences, such as the IEEE PIMRC 2012. He served as an Executive Chair for 2024 IEEE International Workshop of Radio Frequency and Antenna Technologies. He is serveing as an Executive Chair for 2025 IEEE INTERNATIONAL WORKSHOP OF RADIO FREQUENCY AND ANTENNA TECHNOLOGIES. He is the Principal Investigator for over 40 current or finished research projects, including the National Natural Science Foundation of China, the Science and Technology Program of Guangdong Province, and the Science and Technology Program of Shenzhen City. He has served as a Reviewer for various journals, such as the IEEE TRANSACTIONS ON VEHICULAR TECHNOLOGY, the IEEE TRANSACTIONS ON COMMUNICATIONS, the IEEE TRANSACTIONS ON INDUSTRIAL ELECTRONICS, the IEEE TRANSACTIONS ON ANTENNAS AND PROPAGATION, the IEEE WIRELESS COMMUNICATIONS, the IEEE COMMUNICATIONS LETTERS, the *International Journal of Communication Systems*, and *Wireless Personal Communications*. He is serving as an Associate Editor for IEEE TRANSACTIONS ON ANTENNAS AND PROPAGATION, IEEE TRANSACTIONS ON MOBILE COMPUTING, *IEEE Antennas and Propagation Magazine*, IEEE ANTENNAS AND WIRELESS PROPAGATION LETTERS, *International Journal of Communication Systems*, *China Communications*, and *ZTE Communications*. He is a Fellow of IET, and a Senior Member of the China Institute of Communications and the China Institute of Electronics.

Kun Xiang is currently pursuing the M.S. degree in electronic information with the College of Electronics and Information Engineering, Shenzhen University, Shenzhen, China.

His research interests include wireless communications and mobile edge computing.

Xiaowen Cao (Member, IEEE) received the B.Eng. and Ph.D. degrees from Guangdong University of Technology, Guangzhou, China, in 2017 and 2022, respectively.

She is currently an Assistant Professor with the College of Electronics and Information Engineering, Shenzhen University, Shenzhen, China. Her research interests include edge learning, over-the-air computation, as well as integrated sensing, communication, and computation.

Mohsen Guizani (Fellow, IEEE) received the B.S. (with Distinction), M.S., and Ph.D. degrees in electrical and computer engineering from Syracuse University, Syracuse, NY, USA in 1985, 1987, and 1990, respectively.

He is currently a Professor of Machine Learning with Mohamed Bin Zayed University of Artificial Intelligence, Abu Dhabi, UAE. Previously, he worked in different institutions in USA. He is the author of 11 books, more than 1000 publications, and several U.S. patents. His research interests include applied machine learning and artificial intelligence, smart city, Internet of Things, intelligent autonomous systems, and cybersecurity.

Dr. Guizani was listed as a *Clarivate Analytics Highly Cited Researcher in Computer Science in 2019, 2020, 2021, and 2022*. He has won several research awards, including the "2015 IEEE Communications Society Best Survey Paper Award," the Best ComSoc Journal Paper Award in 2021, as well five Best Paper Awards from ICC and Globecom Conferences. He is also the recipient of the 2017 IEEE Communications Society Wireless Technical Committee Recognition Award, the 2018 AdHoc Technical Committee Recognition Award, and the 2019 IEEE Communications and Information Security Technical Recognition Award. He was the Chair of the IEEE Communications Society Wireless Technical Committee and the Chair of the TAOS Technical Committee. He served as the IEEE Computer Society Distinguished Speaker and is currently the IEEE ComSoc Distinguished Lecturer. He served as the Editor-in-Chief for IEEE NETWORK and is currently serving on the editorial boards for many IEEE transactions and magazines.