Energy-Efficient Multi-UAV-Enabled Multiaccess Edge Computing Incorporating NOMA

Xiaochen Zhang, Jiao Zhang, Jun Xiong, Li Zhou, and Jibo Wei

***Abstract*—Multiaccess edge computing (MEC) is regarded as a promising solution to overcome the limit on the computation capacity of mobile devices. This article investigates an energy-efficient unmanned aerial vehicle (UAV)-enabled MEC framework incorporating nonorthogonal multiple access (NOMA), where multiple UAVs are deployed as edge servers to provide computation assistance to terrestrial users and NOMA is adopted to reduce the energy consumption of task offloading. A utility is formed to mathematically evaluate the weighted energy cost of the system. Due to the coupling of parameters, the minimization of utility is a highly nonconvex problem and therefore, the problem is decomposed into two more tractable subproblems, i.e., the optimal allocation of radio and computation resources given UAV trajectories, and the trajectory planning based on given resource allocation schemes. These two problems are converted to convex ones via successive convex approximation (SCA) and quadratic approximation, respectively. Then, an efficient iterative algorithm is proposed where these two subproblems are alternately solved to gradually approach the optimal resource management of the proposed system. Sufficient numerical results show that our proposed strategy has a remarkable advantage over existing systems in terms of energy efficiency.**

***Index Terms*—Energy efficiency, multiaccess edge computing (MEC), nonorthogonal multiple access (NOMA), unmanned aerial vehicles (UAVs).**

I. Introduction

THE UPCOMING 5G communications aim to provide networks with the extreme data rate, deep coverage, and ultrahigh reliability [1]. The versatility of networks urges the emergence of numerous new services, e.g., high-definition (HD) live video streaming, virtual reality (VR), and augmented reality (AR). Not only do these applications depend on ubiquitous and competent communications networks but they also raise critical requirements to the computation capacity at user terminals. Moreover, the idea of Internet of Things (IoT) intends to integrate extensive devices into networks to construct smart cities and offer seamless services to citizens. However, a ceiling is set on the computation capacity of most devices for longer battery life and better portability.

Manuscript received May 17, 2019; revised August 15, 2019, October 18, 2019, and February 24, 2020; accepted March 8, 2020. Date of publication March 11, 2020; date of current version June 12, 2020. This work was supported in part by the National Natural Science Foundation of China under Grant 61601482 and Grant 61601480. *(Corresponding author: Jun Xiong.)*

The authors are with the College of Electronic Science and Engineering, National University of Defense Technology, Changsha 410073, China (e-mail: zhangxiaochen14@nudt.edu.cn; zhangjiao16@nudt.edu.cn; xj8765@nudt.edu.cn; zhouli2035@nudt.edu.cn; wjbhw@nudt.edu.cn).

Digital Object Identifier 10.1109/JIOT.2020.2980035

Multiaccess edge computing (MEC) is considered as a promising paradigm to circumvent these problems by offloading the computation tasks of user devices to servers deployed in base stations, i.e., the edge of networks [2]. These servers have a much higher computation capacity than smart mobile devices (SMDs). Therefore, the execution capacity of tasks can be improved dramatically with the aid of MEC servers, which makes previously impossible applications now possible at SMDs. Especially, instead of a centralized structure used in cloud computing, MEC adopts a distributed structure that provides lower latency as the edge servers can directly exchange data with user terminals. It should be noticed that the application of MEC is definitely not restricted to mobile radio networks, and the concept of edge computing can be realized in a variety of architectures such as fiber-wireless (FiWi) access networks [3]. The scalability of network structures may further improve the performance of edge computing in terms of coverage, latency, and energy efficiency.

Despite the limited battery life and carrying capacity, unmanned aerial vehicles (UAVs) have already been widely used as cheap and flexible relay platforms in emergent conditions where cellular networks are disabled or insufficient. Moreover, the performance of UAVs is being rapidly upgraded, which also prompts copious research intending to integrate UAVs into more advanced network architectures [4], [5] in the era of 5G and beyond. Likewise, to further increase the flexibility of MEC networks, UAVs can be used to load servers to provide services for scattered ground SMDs [6]–[11]. The movement of UAVs results in a dynamic topology of MEC networks, which can significantly decrease the energy cost of task offloading at SMDs and further truncate the system energy consumption.

In addition, because SMDs offload data to MEC servers and the computation results are then returned via wireless communications, a heavy burden is imposed on the scarce radio spectrum by MEC networks. In order to increase the rate of task offloading, transmit power has to be increased, which exacerbates the interference to other communications and cuts down the battery life of SMDs. In this case, nonorthogonal multiple access (NOMA) [12] can be used to manage the task offloading more efficiently. By employing superposition coding (SC) and successive interference cancellation (SIC) [13], NOMA enables the full exploitation of the limited spectrum during the offloading/downloading process in MEC.

In this article, we consider a flexible MEC network architecture incorporating both UAVs and NOMA. Before we

display our proposed scheme, a review of related work is presented.

A. Related Work

Taleb *et al.* [2] first introduced the potential architecture as well as some test results of MEC. In MEC networks, the system performance greatly depends on the allocation of radio and computation resources. Sardellitti *et al.* considered the energy minimization of MEC in a multiple-input–multiple-output (MIMO) multicell system [14]. Furthermore, Zhang *et al.* [15] investigated the tradeoff between energy consumption and latency of MEC networks. Typically, a nonconvex problem is formulated in these studies and iterative algorithms adopting successive convex approximation (SCA) are proposed to find the optimal resource allocation policy. However, Sardellitti *et al.* [14] and Zhang *et al.* [15] treated the sizes of computation tasks as deterministic variables. Mao *et al.* [16] introduced a stochastic model to describe the arrival of computation tasks and an energy minimization algorithm is given based on Lyapunov optimization techniques. In addition, Hu *et al.* [17] modeled a MEC network supported by the wireless power transfer (WPT) technology to further minimize energy consumption at mobile devices. Apart from traditional methods based on convex optimization, machine learning (ML) can also be an alternative. Cao *et al.* [18] summarized the prospective ML-based approaches that could be used in MEC. A full comparison between ML methods and traditional methods is provided in this article. Particularly, the work in [19] acclimatizes MEC networks to nonstationary environments with a stochastic online learning method.

As summarized above, MEC in typical cellular networks has been well investigated. To further reduce the energy consumption and latency, many studies become attentive to the incorporation of other state-of-the-art technology into MEC networks. For instance, the ideas of UAV-enabled MEC and NOMA-assisted MEC have newly emerged.

1) UAV-Enabled MEC: UAVs have gained much popularity recently because of its carrying capacity, low cost, and high maneuverability. They have been considered as an important part of future communications networks to expand coverage and increase throughput in numerous scenarios [20], [21]. The function of UAVs as relays is especially irreplaceable when terrestrial infrastructures are disabled or insufficient to meet the demands.

Embedding UAVs into MEC networks has become popular lately because it further raises the flexibility of edge server deployment. In the meantime, the complexity of system optimization increases significantly as the scheduling of UAV trajectories has to be addressed. Du *et al.* [6] and Zhou *et al.* [7] presented an energy cost minimization design of the UAV-enabled MEC networks. Furthermore, Hu *et al.* [9] minimized the sum/average delay of all users in a UAV-MEC system by proposing two algorithms with different complexities. Additionally, Qian *et al.* [22] maximized the throughput of UAV-MEC networks subject to the energy and Quality-of-Service (QoS) constraints. Zhang *et al.* [10] applied the stochastic model to the single UAV scenario to characterize a more realistic arrival process of computation tasks, where Lyapunov optimization techniques and alternating direction method of multipliers (ADMMs) are used to minimize the energy consumption of the system. For further reduction in energy consumption of wireless devices, WPT techniques are applied to UAV-MEC networks in [8]. Unlike the aforementioned studies in which UAVs could hover, Hu *et al.* [23] deployed the UAV as a stationary platform. The most advantageous position of UAV deployment is found by a 2-D search and the corresponding optimal resource allocation policy is given to minimize the energy consumption of users. Apart from the convex optimization methods, Messous *et al.* [11] adopted a noncooperative theoretical gaming method to achieve a tradeoff between energy consumption, time delay, and computation cost. Furthermore, Asheralieva and Niyato [24] considered a MEC network operated by multiple service providers, where reinforcement learning (RL)-based algorithms are proposed to manage interactions between servers on either UAVs or base stations and maximize their long-term payoff.

2) NOMA-Based MEC: Existing orthogonal multiple access (OMA) strategies allocate orthogonal resources like time slots or subcarriers to different users, whereas NOMA enables multiple users to share the same time slot or spectrum for data transmission. Abundant research has been carried out, focusing on the optimization of NOMA system performance [25], [26]. Lei *et al.* [27] and Ding *et al.* [28] maximized the throughput of a NOMA system and also considered several practical issues, such as user clustering and subchannel allocation. On the other hand, Fang *et al.* [29] and Yang *et al.* [30] considered the NOMA system from the perspective of energy efficiency.

When NOMA is applied to MEC, it can also result in a marked improvement in system performance. Although NOMA has intrigued many researchers as a promising solution to multiuser access, the research into NOMA-MEC networks is still in the initial stage. Kiani and Ansari [31] first improved edge computing with NOMA by completing user clustering and subchannel allocation with a heuristic algorithm. Furthermore, Pan *et al.* [32] proposed a method to minimize the total system energy consumption in NOMA-based MEC and iterative algorithms are given in [33] to minimize the delay of task completion. Most recently, Ding *et al.* [34] provided a comprehensive study on the influence of NOMA on energy consumption and latency of MEC in both uplink and downlink.

It can be found from related work that energy efficiency is the primary concern of plentiful studies on MEC. A prospective method to further improve the energy efficiency of MEC systems is to integrate multiple techniques like UAVs and NOMA to them. There has been research looking into the combination of NOMA and UAV in wireless networks. For instance, Hou *et al.* [35] applied MIMO-NOMA to a UAV network in which a fixed UAV could provide services for users within the coverage. The outage probability and ergodic rate of the system are analyzed. Mei and Zhang [36] proposed a cooperative NOMA to reduce the interference during communications between UAVs and cellular base stations. However,

until the completion of this article, there is no prior work providing an overall evaluation of the joint influence of NOMA and UAV on MEC networks. Would the cooperation of the two cutting-edge techniques further improve the performance of MEC networks? This intricate system requires further investigation to fully exploit its potential, which motivates the work in this article.

B. Our Contributions

Specifically, the major concern of this article is to establish a multi-UAV-enabled network architecture adopting NOMA to provide edge computing for terrestrial SMDs. UAVs should adjust their positions dynamically to provide more flexible services to the area than stationary platforms. The main contributions of this article are summarized as follows.

1) We first establish a MEC network where multiple UAVs are equipped with servers to provide edge computing services for SMDs. NOMA is applied to task offloading in the UAV-MEC network. The deployment of UAVs expands the network coverage and improves the system flexibility, while the introduction of NOMA further brings a remarkable reduction in system energy consumption.
2) The minimization of weighted energy consumption in such a system is of considerable complexity since various factors, including radio and computation resources and UAV trajectories are involved. After formulation, this optimization problem is proved to be highly nonconvex due to the coupling of numerous parameters. We uniquely transform the original problem into two more tractable subproblems, i.e., the optimal allocation of radio and computation resources given UAV trajectories and the trajectory scheduling for UAVs according to the current resource allocation strategy.
3) Regarding the foregoing problem, an efficient iterative optimization algorithm is proposed to find the optimal allocation of radio and computation resources and UAV trajectory planning. By adopting SCA and quadratic approximation, respectively, the two subproblems are converted into solvable convex problems. Then, our proposed algorithm finds the solution to the original problem by iteratively solving the two reformulated subproblems. It is shown that our proposed system has a distinct edge over previous UAV-MEC systems in terms of energy efficiency.

The remainder of this article is organized as follows. Section II illustrates the system model of the proposed MEC network and gives a mathematical formulation of the weighted energy cost minimization problem. Section III presents the solution to this problem by proposing a joint optimization algorithm of radio and computation resource allocation as well as trajectory scheduling of UAVs. The simulation results are discussed in Section IV to demonstrate the benefits of our proposed scheme. Finally, Section V draws a conclusion of this article and envisages the future trend of this article.

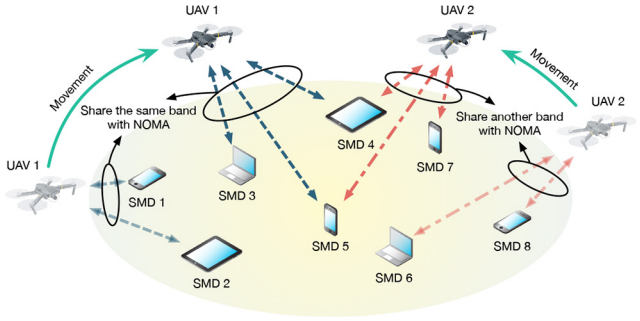

Fig. 1. Diagram of the proposed system.

II. System Model

Consider a MEC network where $U(U \geq 1)$ terrestrial users are uniformly distributed in a given area. Each user has computation tasks to be completed at their SMDs within an interval of T. Due to the limited operational capability of SMDs, it is much demanding for them to run computation-intensive applications on their own. Thus, we employ M UAVs and each of them is equipped with a MEC server. Both SMDs and UAVs have communications capability and therefore, data exchange between SMDs and MEC servers is available. Because MEC servers have a much more powerful computational capacity, part of tasks are offloaded by users to MEC servers. NOMA enables multiple users to share the same band when offloading their tasks to the same server and frequency-division multiple access (FDMA) is used for interference management of transmission to different servers. A schematic of the proposed model is illustrated in Fig. 1.

We adopt sets $\mathcal{U} = \{1, 2, \ldots, U\}$ and $\mathcal{M} = \{1, 2, \ldots, M\}$ to index users and UAVs, respectively. A 3-D Cartesian coordinate system is established to describe locations of UAVs and users. The location of user $i \in \mathcal{U}$ is described by a vector $\mathbf{r}_i^{\text{user}} = (x_i^{\text{user}}, y_i^{\text{user}}, 0)$. Although the entire computation process is continuous in time, the interval can be discretized into $N(N > 1)$[1] time slots with equal length τ. Define a set $\mathcal{N} = \{1, \ldots, N\}$ to index time slots. In each time slot, coordinates of UAVs are updated and the position of UAV m in slot n is given as $\mathbf{r}_m^{\text{uav}}[n] = (x_m^{\text{uav}}[n], y_m^{\text{uav}}[n], h_m^{\text{uav}})$. Similar to [6], [7], and [9], we assume that only horizontal coordinates of UAVs are variable and their altitudes remain constant. Our model primarily consists of three parts: 1) channel model; 2) task execution model; and 3) energy consumption model.

A. Channel Model

Most UAV trajectory optimization problems adopt the free space pathloss (FSPL) model to establish a deterministic relation between the geographic coordinates and channels [6]–[11], [37]. A direct path is often assumed between UAVs and SMDs to justify the use of the FSPL model. This deterministic relation is vital to trajectory scheduling and yet

[1] $N = 1$ is a special case where no task offloading or edge computing is carried out. To properly enable edge computing, we specify $N > 1$. This is due to the causal relation between offloaded tasks and executed tasks on UAVs.

TABLE I
SUMMARY OF KEY NOTATIONS

Notions	Description
$U/M/N$	Number of users/UAVs/time slots
$\mathcal{U}/\mathcal{M}/\mathcal{T}$	Set to index different users/UAVs/time slots
T	Time interval allowed to complete computation tasks
τ	Duration of each time slot
$\mathbf{r}_i^{\text{user}}$	Coordinates of user i
$\mathbf{r}_m^{\text{uav}}[n]$	Coordinates of UAV m in the n-th time slot
$\bar{L}_{i,m}[n]$	Average path loss between user i and UAV m in time slot n
$g_{i,m}[n]$	Average channel gain between user i and UAV m in time slot n
D_i	Amount of tasks at user i
$f_i^{\text{user}}[n]$	CPU frequency of user i in time slot n
C_i^{user}	Number of CPU cycles required to finish one bit of computation task at user i
$d_i^{\text{user}}[n]$	Number of bits finished in time slot n by user i
$f_m^{\text{uav}}[n]$	CPU frequency for tasks of user i by server m in time slot n
C_m^{uav}	Number of CPU cycles required to finish one bit of computation task at UAV m.
$d_{i,m}^{\text{uav}}[n]$	Number of bits finished by server m in time slot n for user i
κ_i^{user}	Effective switched capacitance of CPU at user i
$E_{\text{comp}}^{\text{user}}$	Total energy cost of computation by local SMDs
κ_m^{uav}	Effective switched capacitance of CPU of the server on UAV m
$E_{\text{comp}}^{\text{uav}}$	Total energy cost of computation by servers on UAVs
B_{tot}	Total available bandwidth
B	Bandwidth allocated to each server
N_0	PSD of AWGN
$R_{i,m}[n]$	Number of bits offloaded from user i to UAV m in time slot n
$p_{i,m}[n]$	Transmit power of user i to UAV m in time slot n
$r_k^m[n]$	Index of user with the k-th smallest channel gain to UAV m in time slot n
E_{off}	Total energy consumption of task offloading
W_m	Mass of UAV m
$\mathbf{v}_m[n]$	Average velocity of UAV m in time slot n
E_{mov}	Total energy consumption of UAV flight
$\omega_{\text{user}}/\omega_{\text{uav}}$	Weight factor of users/UAVs

the FSPL model is insufficient to provide an accurate characterization of real channels. In fact, there are a variety of objects in the surroundings acting as scatterers or obstacles. As UAVs hovering around, links between UAVs and SMDs may become intermittent and this phenomenon certainly depends on characteristics of varying environments. Therefore, an alternative is required to address the inaccuracy of the oversimplified FSPL model.

The work in [38] provides a comprehensive characterization of the air-to-ground (ATG) channel for UAVs by taking into account the features of surroundings. We embrace this model in this article to characterize the ATG channels between UAVs and terrestrial nodes. ATG channels are generally categorized into two scenarios: 1) Line of Sight (LoS) and 2) Non-LoS (NLoS). Model parameters are extracted from measurement data in different surroundings. According to [38], the expectation of channel gain over both cases is used to determine the channel gain based on the configuration of UAVs and SMDs.

Despite the fact that at any instant a channel is either LoS or NLoS, this model obtains the channel gain by averaging both cases according to their probability of occurrence, thus only providing an approximate description of the channel which is neither LoS nor NLoS. It seems preferable to distinguish between LoS and NLoS during the optimization, while such an idea is virtually infeasible since channels are influenced by the layout of the system as well as the environment in an unpredictable manner. Channel conditions may change considerably as a consequence of slight displacement of UAVs, and there is no way to identify whether a link is LoS or NLoS prior to the deployment of a UAV in the particular position. Consequently, a compromise plan, as referred to above, is needed to maintain the deterministic dependence of trajectory scheduling on channel conditions and meanwhile describe the changeable channels more accurately than the FSPL model does. The model in [38] predicts the expectation of channel path loss from the perspective of statistics and this has been adopted by several studies including [39]–[43] for the same purpose.

The probability of LoS/NLoS depends on the elevation angles of UAVs. The straightline distance and elevation angle between user i and UAV m in slot n can be calculated by

$$\begin{cases} \theta_{i,m}[n] = (180/\pi) \cdot \arcsin\left(h_m^{\text{uav}}/d_{i,m}[n]\right) \\ d_{i,m}[n] = \sqrt{\left(x_i^{\text{user}} - x_m^{\text{uav}}[n]\right)^2 + \left(y_i^{\text{user}} - y_m^{\text{uav}}[n]\right)^2 + \left(h_m^{\text{uav}}\right)^2}. \end{cases} \tag{1}$$

Then, the probability of the LoS/NLoS path between user i and UAV m in the nth time slot is

$$\begin{cases} \mathbb{P}_{\text{LoS}}^{i,m}[n] = \frac{1}{1+a\exp\left(-b(\theta_{i,m}[n]-a)\right)} \\ \mathbb{P}_{\text{NLoS}}^{i,m}[n] = 1 - \mathbb{P}_{\text{LoS}}^{i,m}[n] \end{cases} \tag{2}$$

where a and b are parameters contingent on different environments [38]. Furthermore, a surface fitting method is given in [38] to find parameters a and b for typical environments, including suburban, urban, dense urban, and high-rise urban. Thus, the path loss in each case is given as [38]

$$L_\xi^{i,m}[n] = \eta_\xi \left(4\pi f_c d_{i,m}[n]/c\right)^\alpha, \quad \xi \in \{\text{LoS}, \text{NLoS}\} \tag{3}$$

where f_c is the carrier frequency, c is the speed of light, α stands for the path-loss exponent, and η_ξ is the excessive path loss depending on the existence of the LoS path. We use the average path loss of both scenarios $\bar{L}_{i,m}[n]$ to determine channel gain $g_{i,m}[n]$ in this article, i.e.,

$$\bar{L}_{i,m}[n] = \mathbb{P}_{\text{LoS}}^{i,m}[n] L_{\text{LoS}}^{i,m}[n] + \mathbb{P}_{\text{NLoS}}^{i,m}[n] L_{\text{NLoS}}^{i,m}[n]. \tag{4}$$

Then, the channel gain $g_{i,m}[n]$ is the reciprocal of average path loss $\bar{L}_{i,m}[n]$.

B. Task Execution Model

We assume that the number of bits to be executed by user i is D_i. The partial offloading strategy is adopted, which means that each user can conduct local computing and meanwhile

offload part of their tasks to MEC servers on UAVs [10]. After MEC servers have received the offloaded data, they would start to allocate computation resources to proceed these tasks from the next time slot. The number of bits offloaded to UAV m by user i in the nth time slot is $R_{i,m}[n]$.

Furthermore, it is assumed that each SMD only holds a single-core CPU. The CPU frequency of user i in the nth time slot is $f_i^{\text{user}}[n]$. The maximum CPU frequency of user i is $f_{i,\max}^{\text{user}}$ and therefore, $f_i^{\text{user}}[n]$ should not exceed $f_{i,\max}^{\text{user}}$. Process density C_i^{user} denotes the number of CPU cycles required to finish one bit of the computation task for user i. Then, the number of bits finished by user i in time slot n is $d_i^{\text{user}}[n] = f_i^{\text{user}}[n] \cdot \tau / C_i^{\text{user}}$.

The MEC server on UAV m has a maximum CPU frequency of $f_{m,\max}^{\text{uav}}$. The CPU frequency allocated to the tasks of user i by UAV m in the nth time slot is $f_{i,m}^{\text{uav}}[n]$. Accordingly, we have the relation $\sum_{i=1}^{U} f_{i,m}^{\text{uav}}[n] \le f_{m,\max}^{\text{uav}}\ \forall n \in \mathcal{N}, m \in \mathcal{M}$. Process density C_m^{uav} denotes the number of CPU cycles required to finish one bit of the computation task. Then, the number of bits finished by server m for user i in time slot n is $d_{i,m}^{\text{uav}}[n] = f_{i,m}^{\text{uav}}[n] \cdot \tau / C_m^{\text{uav}}$.

Note that there is also a causal constraint that the number of bits finished at each MEC server for certain users cannot exceed the total amount of tasks already offloaded from the particular user. As the process is discretized and a time slot is the smallest unit of action, the MEC server can only compute the bits offloaded in preceding time slots. Accordingly, edge computing should not be activated in the first time slot when no task has been offloaded. On the other hand, task offloading to UAVs in the Nth time slot is unreasonable since offloaded tasks are no longer executed after the termination of the period.

C. Energy Consumption Model

The energy consumption of the system consists of three parts, i.e., task execution, task offloading, and UAV flight.

1) Energy Cost of Task Execution: As both SMDs and MEC servers conduct computation, the system expenditure of energy takes both into consideration. The instantaneous power cost by CPU is proportional to the cube of CPU frequency. Accordingly, the local energy consumed by user i in time slot n is $e_i^{\text{user}}[n] = \kappa_i^{\text{user}} f_i^{\text{user}}[n]^3 \tau$, where κ_i^{user} is a constant called the effective switched capacitance of CPU at user i [44]. Likewise, the energy cost of computation for user i by UAV m in time slot n is $e_{i,m}^{\text{uav}}[n] = \kappa_m^{\text{uav}} f_{i,m}^{\text{uav}}[n]^3 \tau$. Therefore, the overall energy consumption of computation at SMDs and MEC servers is given as

$$\begin{cases} E_{\text{comp}}^{\text{user}} = \sum_{i=1}^{U} \sum_{n=1}^{N} \kappa_i^{\text{user}} f_i^{\text{user}}[n]^3 \tau \\ E_{\text{comp}}^{\text{uav}} = \sum_{m=1}^{M} \sum_{i=1}^{U} \sum_{n=1}^{N} \kappa_m^{\text{uav}} f_{i,m}^{\text{uav}}[n]^3 \tau. \end{cases} \tag{5}$$

2) Task Offloading Scheme: Power-domain NOMA is applied to the UAV-enabled MEC network to further minimize the energy consumption during task offloading. Due to the diversity of SMDs in the network, we assume that omnidirectional antennas are used on all SMDs. In order to avoid interference at UAVs when users offload data to multiple UAVs, the entire spectrum B_{tot} is evenly divided into M bands. Different servers exclusively occupy a band through FDMA while SMDs share the same band when offloading data to a specific server.

Due to the motion of UAVs, channel conditions vary with time. To make the continuous process feasible for resource allocation, we assume that positions of UAVs remain constant during one time slot. As a result, SMDs are ranked by each MEC server in the ascending order of channel gain in each time slot. Therefore, each MEC server has a list updated to time. The list of SMDs for UAV m in time slot n is denoted as $\mathcal{R}_m[n] = \{r_1^m[n], r_2^m[n], \ldots, r_U^m[n]\}$, where $r_k^m[n] \in \mathcal{U}\ \forall k \in \mathcal{U}$. $r_k^m[n]$ denotes the index of the user with the kth smallest channel gain within links to UAV m during time slot n.

In each time slot, UAVs will conduct SIC to demodulate signals from multiple users. The sequence of SIC follows the descending order of each user's current channel gain. Note that only signals transmitted to the same UAV can cause interference. Each UAV will not be interfered by signals targeting at other UAVs because they work on different sub-bands. When the server on UAV m decodes the signal from user $r_k^m[n]$, the received signals from user $r_1^m[n]$ to user $r_{k-1}^m[n]$ are treated as interference. After the signal from user $r_k^m[n]$ is decoded, it is removed from the superimposed signal. Then, the server starts to decode the signal from user $r_{k-1}^m[n]$ and this iterative process goes on until the signal from user $r_1^m[n]$ is decoded. When decoding the signal from user $r_1^m[n]$, the server no longer suffers from interference. Suppose the transmit power used by user $r_k^m[n]$ for offloading tasks to UAV m is $p_{r_k,m}[n]$, and then the maximum amount of data that can be offloaded to UAV m by user $r_k^m[n]$ in the nth time slot is

$$R_{r_k,m}[n] = B \cdot \log\left(1 + \frac{p_{r_k,m}[n] g_{r_k,m}[n]}{\sum_{l=1}^{k-1} p_{r_l,m}[n] g_{r_l,m}[n] + BN_0}\right) \cdot \tau \tag{6}$$

where B is the bandwidth allocated to each MEC server and N_0 is the power spectral density (PSD) of additive white Gaussian noise at receivers. The total energy cost for task offloading is

$$E_{\text{off}} = \sum_{m=1}^{M} \sum_{i=1}^{U} \sum_{n=1}^{N} p_{i,m}[n] \tau. \tag{7}$$

Note that the download of results from edge servers is omitted from our proposed optimization scheme on account of several concerns. In general, the size of computation results is often trivial compared with that of original tasks, thereby taking few communication resources like bandwidth and power. This article concentrates on the energy efficiency improvement of MEC networks and accordingly disregarding the download of results will not noticeably affect the system performance. Additionally, overlooking the download process reduces the model complexity significantly, which has been adopted by numerous studies, including [6]–[8], [10], [16], [31], and [45]. Last but not least, although the evaluation of latency is contingent on the delay of computation results, this is not our primary concern. As long as all tasks can be completed within the time limit, we will like to cut down the energy cost of the system to its minimum, as considered in [7], [10], and [16]. This is compatible with a number of applications such as environment monitoring where analysis results need only

to be returned periodically from edge servers to terrestrial infrastructures.

3) UAV Flight: The initial and final positions of UAVs are specified, the maximum speed of UAV m is $v_{\max}^m$ and no further constraint is set on trajectories. To attain flight energy consumption, we assume that UAVs do uniform rectilinear flight between two positions, which is widely adopted in many related studies, such as [7], [10], and [42]. Thus, no acceleration or turning radius is considered in our model. The average velocity of UAV m in the nth time slot is $\mathbf{v}_m[n]$. Denote the mass of UAV m by W_m, and then the energy consumption of UAV flight following [7], [46], and [47] is:

$$\begin{aligned} E_{\text{mov}} &= \sum_{m=1}^{M}\sum_{n=1}^{N}\frac{W_m}{2}\|\mathbf{v}_m[n]\|^2\tau \\ &= \frac{W_m\tau}{2}\sum_{m=1}^{M}\sum_{n=1}^{N}\left(\left\|\mathbf{r}_m^{\text{uav}}[n]-\mathbf{r}_m^{\text{uav}}[n-1]\right\|/\tau\right)^2. \end{aligned} \tag{8}$$

D. Problem Formulation

A system utility is constructed to measure the overall weighted cost of energy consumption, which is given as

$$\begin{aligned} \rho &= \omega_{\text{user}}\left(E_{\text{comp}}^{\text{user}}+E_{\text{off}}\right)+\omega_{\text{uav}}\left(E_{\text{comp}}^{\text{uav}}+\gamma E_{\text{mov}}\right) \\ &= \omega_{\text{user}}\left(\underbrace{\sum_{i=1}^{U}\sum_{n=1}^{N}\kappa_i^{\text{user}}f_i^{\text{user}}[n]^3\tau}_{\text{local computing}}+\underbrace{\sum_{m=1}^{M}\sum_{i=1}^{U}\sum_{n=1}^{N}p_{i,m}[n]\tau}_{\text{offloading}}\right) \\ &\quad+\omega_{\text{uav}}\left(\underbrace{\sum_{m=1}^{M}\sum_{i=1}^{U}\sum_{n=1}^{N}\kappa_m^{\text{uav}}f_{i,m}^{\text{uav}}[n]^3\tau}_{\text{edge computing}}+\gamma\underbrace{\sum_{m=1}^{M}\sum_{n=1}^{N}\frac{W_m}{2}\|\mathbf{v}_m[n]\|^2\tau}_{\text{flight}}\right) \end{aligned} \tag{9}$$

where the nonnegative parameters ω_{user} and ω_{uav} denote the weight factors of users and UAVs, respectively. We restrict $\omega_{\text{user}}+\omega_{\text{uav}}=1$. γ is the coefficient for energy consumption of UAV flight. In practical systems, UAV flight consumes much more energy than other components, while UAVs are generally more tolerable to energy cost than SMDs, because UAVs, as a part of infrastructures, often carry much larger batteries and can be recharged more frequently. In contrast, task offloading and execution at SMDs exhaust the battery duration at SMDs considerably, despite their relatively small proportion in the utility. Therefore, weight coefficients are needed to adjust the priority of different components in optimization.

Next, we consider the minimum weighted energy consumption by jointly optimizing radio and computation resources as well as UAV trajectories. The optimization problem mentioned above is formulated as follows:

$$\begin{aligned} \mathbf{P_0}: \min_{\mathbf{r}_{\text{uav}},\mathbf{p},\mathbf{f}_{\text{user}},\mathbf{f}_{\text{uav}}} \quad & \rho \\ \text{s.t.} \quad & C1\text{: } 0\le f_i^{\text{user}}[n]\le f_{i,\max}^{\text{user}}, \qquad i\in\mathcal{U}, n\in\mathcal{N} \\ & C2\text{: } \sum_{i=1}^{U}f_{i,m}^{\text{uav}}[n]\le f_{m,\max}^{\text{uav}}, \qquad n\in\mathcal{N},\ \ m\in\mathcal{M} \\ & C3\text{: } 0\le\sum_{m=1}^{M}p_{i,m}[n]\le p_{\max}^{i}\ \ i\in\mathcal{U},\ \ n\in\mathcal{N} \\ & C4\text{: } \sum_{m=1}^{M}\sum_{n=1}^{N}\left(d_i^{\text{user}}[n]+d_{i,m}^{\text{uav}}[n]\right)\ge D_i,\ \ i\in\mathcal{U} \\ & C5\text{: } \sum_{t=1}^{n-1}R_{i,m}[t]\ge\sum_{t=1}^{n}d_{i,m}^{\text{uav}}[t],\ \ i\in\mathcal{U}, \\ & \quad n\in\mathcal{N},\ \ m\in\mathcal{M} \\ & C6\text{: } \left\|\mathbf{r}_m^{\text{uav}}[n]-\mathbf{r}_m^{\text{uav}}[n-1]\right\|\le v_{\max}^m\cdot\tau \\ & \quad n\in\mathcal{N},\ \ m\in\mathcal{M} \\ & C7\text{: } \mathbf{r}_m^{\text{uav}}[0]=\mathbf{r}_I^m,\ \ \mathbf{r}_m^{\text{uav}}[N]=\mathbf{r}_F^m,\ \ m\in\mathcal{M} \end{aligned} \tag{10}$$

where $\mathbf{r}_{\text{uav}}, \mathbf{p}, \mathbf{f}_{\text{user}}$, and $\mathbf{f}_{\text{uav}}$ denote the UAV trajectories, transmit power, CPU frequencies at SMDs, and edge servers, respectively. $C1$ and $C2$ restrict the maximum CPU frequencies. $C3$ sets an upperbound to transmit power. $C4$ ensures that the system finishes all computation tasks within the given interval. $C5$ indicates the causal relation between task offloading and edge computing. $C6$ and $C7$ limit the speed and initial/final coordinates of UAVs.

The objective function of $\mathbf{P_0}$ is the sum of polynomials and quadratic functions of the decision variables. As the second-order derivative of the objective function is positive, it is a convex function [48]. However, the amount of offloaded tasks $R_{i,m}[n]$ in $C5$ involves the coupling of UAV trajectories and transmit power in both numerator and denominator, which is not a convex function of decision variables. Therefore, $C5$ makes $\mathbf{P_0}$ a nonconvex problem. Next, we decompose the original problem $\mathbf{P_0}$ into two more tractable subproblems, and subsequently, an efficiently iterative optimization algorithm is proposed where the two subproblems are solved alternately to find a convergent optimal solution to the original problem.

III. Joint Optimization Algorithm for Management of Radio and Computation Resources and Trajectory Planning

Due to the nonconvexcity of $\mathbf{P_0}$, it is impractical to come up with a straight solution. Nevertheless, the optimal solution to the original problem $\mathbf{P_0}$ can be found by iteratively solving two subproblems, i.e., the optimal management of radio and computation resources given the current UAV trajectories and trajectory scheduling for UAVs when the assignment of radio and computation resources is fixed. First, the solutions to these two subproblems are presented, respectively. Later, an algorithm to solve $\mathbf{P_0}$ is obtained.

A. Optimal Management of Radio and Computation Resources

Given the UAVs' trajectories, subproblem $\mathbf{SP_1}$ is formulated to seek the optimal allocation strategy of radio and computation resources, which is given by

$$\begin{aligned} \mathbf{SP_1}: \min_{\mathbf{p},\mathbf{f}_{\text{user}},\mathbf{f}_{\text{uav}}} \quad & \omega_{\text{user}}\left(E_{\text{comp}}^{\text{user}}+E_{\text{off}}\right)+\omega_{\text{uav}}\left(E_{\text{comp}}^{\text{uav}}+\gamma E_{\text{mov}}\right) \\ \text{s.t.} \quad & C1-C5. \end{aligned} \tag{11}$$

Even if the UAV trajectories are determined, $C5$ still results in the intractability of $\mathbf{SP_1}$. To solve $\mathbf{SP_1}$, the problem has to be reformulated. We first transform $\mathbf{SP_1}$ into an equivalent form of which the objective function is convex and the structures of constraints are transformed. Afterward, we show that the problem can be solved as it is approximated by a convex optimization problem.

The objective function of $\mathbf{SP_1}$ is the weighted sum of four parts, i.e., the energy cost for local computing, task offloading, edge computing, and UAV flight. Since the trajectories of UAVs are predetermined, the energy cost of UAV flight becomes a constant which can be neglected in $\mathbf{SP_1}$. $E_{\text{comp}}^{\text{user}}$ and $E_{\text{comp}}^{\text{uav}}$ are polynomial functions of CPU frequencies with positive coefficients. Therefore, these two terms are convex. Next, we show that the energy expenditure on offloading can be transformed into the sum of multiple exponential functions with positive coefficients, which turns out to be a convex function.

According to (6), it can be found that

$$e^{R_{r_k,m}[n]\ln 2/B\tau} = 1 + \frac{p_{r_k,m}[n]g_{r_k,m}[n]}{\sum_{l=1}^{k-1} p_{r_l,m}[n]g_{r_l,m}[n] + BN_0}. \quad (12)$$

Set

$$\begin{cases} J = \ln 2/B\tau \\ M_{r_k,m}[n] = \sum_{l=1}^{k} p_{r_l,m}[n]g_{r_l,m}[n] + BN_0 \end{cases} \quad (13)$$

and then the amount of transmitted data given in (6) can be written in a recursive style as

$$M_{r_k,m}[n] = e^{JR_{r_k,m}[n]} M_{r_{k-1},m}[n], \quad k \in \mathcal{U}\backslash\{1\}. \quad (14)$$

For $k = 1$, $M_{r_1,m}[n] = p_{r_1,m}[n]g_{r_1,m}[n] + BN_0$ according to definition in (13). From (6), we can obtain

$$R_{r_1,m}[n] = B \cdot \log\left(1 + \frac{p_{r_1,m}[n]g_{r_1,m}[n]}{BN_0}\right) \cdot \tau. \quad (15)$$

Thus, $M_{r_1,m}[n] = BN_0 \exp(JR_{r_1,m}[n])$, which is consistent with the iterative fashion in (14). A complete expression for all user indices is, therefore, given as

$$M_{r_k,m}[n] = BN_0 e^{J\sum_{l=1}^{k} R_{r_l,m}[n]} \quad \forall k \in \mathcal{U}. \quad (16)$$

The transmit power of user $r_k^m[n]$ to UAV m in time slot n can be equivalently written as

$$p_{r_k,m}[n] = \frac{M_{r_k,m}[n] - M_{r_{k-1},m}[n]}{g_{r_k,m}[n]}, \quad k \in \mathcal{U}\backslash\{1\}. \quad (17)$$

This expression will also be valid for $k = 1$ when we set $\sum_{l=1}^{0} R_{r_l,m}[n] = 0$. Furthermore, if we let

$$x_{k,m}[n] = \sum_{l=1}^{k} R_{r_l,m}[n] \quad (18)$$

it can be shown that

$$M_{r_k,m}[n] = BN_0 e^{Jx_{k,m}[n]}. \quad (19)$$

Compared with $M_{r_k,m}[n]$, $x_{k,m}[n]$ has a more tangible meaning, i.e., the sum amount of data transmitted to UAV m from users with indices from $r_1^m[n]$ to $r_k^m[n]$ in time slot n. Then, the transmit power is transformed into

$$p_{r_k,m}[n] = \frac{BN_0}{g_{r_k,m}[n]}\left(e^{Jx_{k,m}[n]} - e^{Jx_{k-1,m}[n]}\right) \quad \forall k \in \mathcal{U}. \quad (20)$$

Unfortunately, the difference of two exponential functions in (20) is generally not a convex function. However, as the sum of transmit power at each SMD holds some recursive features, next we try to exploit them to demonstrate that the sum of power is indeed a convex function. The sum of transmit power to UAV m from all SMDs is

$$\begin{aligned} \sum_{k=1}^{U} p_{r_k,m}[n] &= \sum_{k=1}^{U} \frac{BN_0}{g_{r_k,m}[n]}\left(e^{Jx_{k,m}[n]} - e^{Jx_{k-1,m}[n]}\right) \\ &= BN_0 \sum_{k=0}^{U}\left(-\frac{1}{g_{r_{k+1},m}[n]} + \frac{1}{g_{r_k,m}[n]}\right)e^{Jx_k}. \end{aligned} \quad (21)$$

As $g_{r_1,m}[n] \le g_{r_2,m}[n] \le \cdots \le g_{r_U,m}[n]$, the coefficient of each exponential function is nonnegative. Set $\beta_{k,m}[n] = 1/g_{r_k,m}[n]$, and we have $\beta_{U+1,m}[n] = 0$ and $\beta_{0,m}[n] = 0$ as special cases. Therefore, the sum transmit power for offloading tasks to UAV m in time slot n becomes

$$\sum_{k=1}^{U} p_{r_k,m}[n] = BN_0 \sum_{k=0}^{U}\left[\left(-\beta_{k+1,m}[n] + \beta_{k,m}[n]\right)e^{Jx_{k,m}[n]}\right] \quad (22)$$

which is a convex function of arguments in $\mathbf{SP_1}$ [48].

The sum of three convex functions and a constant is still a convex function. Consequently, the objective function of $\mathbf{SP_1}$ is convex.

The deterministic relation between transmit power $p_{i,m}[n]$ and maximal amount of transmitted data $R_{r_k,m}[n]$ is given by (6). Because $R_{r_k,m}[n]$ can be expressed in the form of $x_{k,m}[n]$, the allocation of transmit power $p_{i,m}[n]$ is equivalent to that of $x_{k,m}[n]$. Accordingly, variables in constraints, including $p_{i,m}[n]$ and $R_{i,m}[n]$ need to be expressed by $x_{k,m}[n]$ as well. Define a function $\Psi(i,n,m)$ to indicate the rank of user i for UAV m in time slot n, and then the amount of data transmitted by user i to UAV m in time slot n becomes

$$R_{i,m}[n] = x_{\Psi(i,n,m),m}[n] - x_{\Psi(i,n,m)-1,m}[n]. \quad (23)$$

Therefore, $\mathbf{SP_1}$ is transformed into its equivalent form as $\mathbf{SP_1'}$

$$\begin{aligned} \mathbf{SP_1'}: \min_{\mathbf{x},\mathbf{f}_{\text{user}},\mathbf{f}_{\text{uav}}} \quad & \omega_{\text{user}} \sum_{i=1}^{U}\sum_{n=1}^{N} \kappa_i^{\text{user}} f_i^{\text{user}}[n]^3\tau + \omega_{\text{user}} BN_0 \\ & \times \sum_{i=1}^{U}\sum_{n=1}^{N}\sum_{m=1}^{M}\left(\beta_{k,m}[n] - \beta_{k+1,m}[n]\right)e^{Jx_{k,m}[n]} \\ & + \omega_{\text{uav}} \sum_{i=1}^{U}\sum_{n=1}^{N}\sum_{m=1}^{M} \kappa_i^{\text{uav}} f_{i,m}^{\text{uav}}[n]^3\tau \\ \text{s.t.} \quad & C1, C2, C4 \\ & C3'\text{: } 0 \le \sum_{m=1}^{M} \frac{e^{Jx_{\Psi(i,n,m),m}[n]} - e^{Jx_{\Psi(i,n,m)-1,m}[n]}}{g_{\Psi(i,n,m),m}[n]} \\ & \le \frac{p_{\max}^i}{BN_0}, \quad i \in \mathcal{U}, \quad n \in \mathcal{N} \end{aligned}$$

$$C5': \sum_{t=1}^{n-1}\left(x_{\Psi(i,t,m),m}[t] - x_{\Psi(i,t,m)-1,m}[t]\right) \geq \sum_{t=1}^{n} f_{i,m}^{\text{uav}}[t]\cdot\tau/C_i^{\text{user}} \quad i\in\mathcal{U},\quad n\in\mathcal{N},\quad m\in\mathcal{M} \tag{24}$$

where $C3'$ and $C5'$ are mathematically equivalent to $C3$ and $C5$ in $\mathbf{SP_1}$. Note that in $\mathbf{SP_1}$, trajectories of UAVs are fixed and accordingly coordinates are invariable. As a result, the last term in ρ accounting for energy consumption of UAV flight is irrelevant in $\mathbf{SP_1}$. We denote the objective function in $\mathbf{SP_1'}$ as ζ. $C1$, $C2$, $C4$, and $C5'$ are all affine functions of variables. Yet $C3'$ is the addition and subtraction of multiple exponential functions whose convexity is not guaranteed.

Similar to the method in [32], SCA is adopted to approximate the original constraint function $C3'$. Let $y(x) = \exp(Jx)$, and its local approximation can be obtained by substituting it with its first-order Taylor expansion, i.e.,

$$\tilde{y}(x|x_0) = y(x_0) + \frac{\partial y(x)}{\partial x}(x - x_0) = e^{Jx_0} + Je^{Jx_0}(x - x_0). \tag{25}$$

Apparently, (25) is a linear function of x. By replacing $C3'$ with its first-order Taylor expansion $C3''$, $\mathbf{SP_1'}$ is formulated into $\mathbf{SP_1''}$

$$\mathbf{SP_1''}: \min_{\mathbf{x},\mathbf{f}_{\text{user}},\mathbf{f}_{\text{uav}}} \ \zeta \quad \text{s.t.} \quad C1, C2, C3,'' C4, C5'. \tag{26}$$

In fact, the solution to $\mathbf{SP_1}$ can be found by iteratively solving $\mathbf{SP_1''}$. In each iteration, the center of series expansion is updated by the optimal $x_{k,m}[n]$ obtained during the last iteration. In the λth iteration, $C3''$ is written as

$$C3'': 0 \leq \sum_{m=1}^{M}\frac{1}{g_{\Psi(i,n,m),m}[n]}\times\left[y\left(x_{\Psi(i,n,m),m}[n]\right) - \tilde{y}\left(x_{\Psi(i,n,m)-1,m}[n]\middle|\hat{x}^{\lambda-1}_{\Psi(i,n,m)-1,m}[n]\right)\right] \leq \frac{p^i_{\max}}{BN_0} \quad \forall i\in\mathcal{U}, n\in\mathcal{N} \tag{27}$$

where $\hat{x}^{\lambda-1}_{k,m}[n]$ is the solution to $x_{k,m}[n]$ in the $(\lambda-1)$th iteration. Being the difference of an exponential function and an affine function makes $C3''$ a convex function, thereby ensuring the convexity of $\mathbf{SP_1''}$.

$\mathbf{SP_1''}$ as a convex optimization problem can be solved by standard methods, such as the interior-point method (IPM) [48]. The suboptimal solution of $\mathbf{SP_1}$ can be found when convergence/stop condition is satisfied when iteratively solving $\mathbf{SP_1''}$, as summarized in Algorithm 1.

Algorithm 1 Optimization of Radio and Computation Resource Allocation for Multi-UAV MEC Networks

Input: coordinates of users $\mathbf{r}^{\text{user}}$, trajectories of UAVs $\mathbf{r}^{\text{uav}}$.
Output: transmit power $\mathbf{p}$, CPU frequencies of SMDs $\mathbf{f}_{\text{user}}$, CPU frequencies of MEC servers $\mathbf{f}_{\text{uav}}$.

```
Given positions of SMDs r^user and UAV trajectories r^uav, compute the channel gain of links between each SMDs and UAVs in any time slot.
Rank the users in ascending order of channel gain for each MEC server in each time slot.
Set x_0 = 0, initial utility ζ_0 = 0 and the number of iterations λ = 1.
while |ζ_λ − ζ_{λ−1}| ≥ ε do
    Solve SP_1'' using CVX.
    λ = λ + 1.
end while
return p, f_user, f_uav.
```

B. Trajectory Scheduling of UAVs

We now consider the trajectory planning for UAVs when the resource allocation strategy is fixed. This subproblem can be formulated as

$$\mathbf{SP_2}: \min_{\mathbf{r}_u} \ \omega_{\text{user}}\left(E^{\text{user}}_{\text{comp}} + E_{\text{off}}\right) + \omega_{\text{uav}}\left(E^{\text{uav}}_{\text{comp}} + \gamma E_{\text{mov}}\right) \quad \text{s.t.} \ C6 \text{ and } C7. \tag{28}$$

As the computation resources are fixed in $\mathbf{SP_2}$, it is unnecessary to take into account the energy cost of computation. Not only do UAV trajectories determine the energy consumption of motion but they also affect channel gain, which will then alter the transmit power. Define a function

$$\chi(k,n,m) = BN_0\left[\exp\left(Jx_{k,m}[n]\right) - \exp\left(Jx_{k-1,m}[n]\right)\right] \tag{29}$$

and then the sum of transmit power is re-expressed as

$$E_{\text{off}} = \sum_{k=1}^{U}\sum_{n=1}^{N}\sum_{m=1}^{M}\chi(k,n,m)\bar{L}_{r_k,m}[n]. \tag{30}$$

In $\mathbf{SP_2}$, the value of $\chi(k,n,m)$ can be viewed as a constant. Therefore, E_{off} is a weighted sum of the path loss of all links. However, as shown in (4), the adopted path loss is the expectation of path loss in both LoS and NLoS scenarios. The probability of both cases given in (2) is a nonconvex function of UAVs coordinates and unfortunately the optimal trajectories are intractable at the moment.

An alternative is to approximate the path-loss function with a convex function [42]. Quadratic functions are selected to fit the curve with satisfactory error margin, i.e., $\bar{L}_{i,m}[n] \approx \tilde{L}_{i,m}[n] = p_1 d_{i,m}[n]^2 + p_2 d_{i,m}[n] + p_3$, where p_1, p_2, and p_3 are coefficients to be determined. Note that we constrain $p_1, p_2 \geq 0$ such that the fit curve is convex. Two examples are illustrated in Fig. 2, where h_{uav} is set to be 50 and 75 m, respectively. The goodness of fitting is also displayed. It could be found that we have a rather large error when the horizontal distance between nodes is small. However, we manage to minimize the average fitting error throughout all possible distances within the coverage area. When the distance is more than twice the UAV's altitude, the error drops to a marginal level ($\leq 1\%$). In conclusion, this method gives a reliable approximation of path loss in most cases.

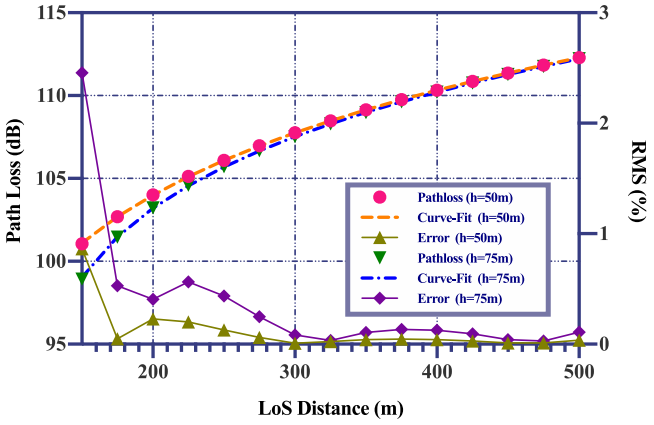

Fig. 2. Fit the path loss with a quadratic function.

As a result, $\mathbf{SP_2}$ is reformulated into $\mathbf{SP_2'}$

$$\mathbf{SP_2'}: \min_{\mathbf{r}_u} \ \omega_{\text{user}} \sum_{k=1}^{U}\sum_{n=1}^{N}\sum_{m=1}^{M} \chi(k,n,m)\tilde{L}_{r_k,m}[n] + \omega_{\text{uav}}\gamma \frac{W_m\tau}{2}\sum_{m=1}^{M}\sum_{n=1}^{N}\left(\left\|\mathbf{r}_m^{\text{uav}}[n]-\mathbf{r}_m^{\text{uav}}[n-1]\right\|/\tau\right)^2$$
$$\text{s.t. } C6 \text{ and } C7. \tag{31}$$

Since $\tilde{L}_{r_k,m}[n]$ is a quadratic function, $\mathbf{SP_2'}$ is a convex problem, which can be solved by existing methods like CVX [48], [49].

C. Joint Optimization Algorithm for Management of Radio and Computation Resources and Trajectory Planning

The joint optimization of computation and radio resources and UAV trajectory scheduling can be realized by iteratively solving subproblems 1 and 2, which is concluded in Algorithm 2.

Remark 1 (Convergence Analysis): The problem $\mathbf{P_0}$ is solved by two layers of iterations. The inner layer (Algorithm 1) is to find the optimal allocation of radio and computation resources by iteratively solving $\mathbf{SP_1''}$, while the outer layer (Algorithm 2) gives the alternate solution to $\mathbf{SP_1'}$ and $\mathbf{SP_2'}$. We first discuss the convergence properties of the inner layer. It could be validated that $\zeta_{\lambda+1} \le \zeta_\lambda$ for all values of λ. As $\mathbf{SP_1''}$ is strictly convex, ζ_λ is a monotonically nonincreasing sequence of λ. Besides, as ζ_λ is lower bounded, the convergence of Algorithm 1 is ensured [50]. Likewise, due to the strict convexity of $\mathbf{SP_2}$ and nonincreasing monotone of ζ_λ in Algorithm 1, the convergence of the lower bounded utility ρ_l in $\mathbf{P_0}$ is also guaranteed.

Remark 2 (Computational Complexity): The computational complexity of an algorithm mostly depends on the number of decision variables and constraints [48], [50]. In general, linear matrix inequality (LMI) constraints and second-order cone (SOC) constraints are considered. The original problem $\mathbf{P_0}$ has $2M(N-1)+2MNU+NU$ decision variables, $2NU+MN+U+MNU$ LMI constraints of size 1, and MN SOC constraints of size 2. As shown previously, $\mathbf{P_0}$ is divided into two subproblems and its solution is found by iteratively solving $\mathbf{SP_1'}$ and $\mathbf{SP_2'}$. Thus, the computation of its complexity involves two stages: 1) the complexity of solutions to $\mathbf{SP_1''}$ and $\mathbf{SP_2'}$, respectively, and 2) the overall complexity including all iterations.

When IPM is adopted, the computational complexity of the $\tilde{\epsilon}$-optimal solution for a convex problem is given as $\mathcal{O}(\ln(1/\tilde{\epsilon})\tilde{n}^3)$, where $\tilde{n}$ is the number of decision variables [51]. $\mathbf{SP_1''}$ includes $\tilde{n}_1 = 2MNU+NU$ decision variables and the corresponding complexity is $\mathcal{O}(\ln(1/\tilde{\epsilon})\tilde{n}_1^3)$. On the other hand, there are $2M(N-1)$ decision variables in $\mathbf{SP_2'}$ and its complexity is $\mathcal{O}(\ln(1/\tilde{\epsilon})\tilde{n}_2^3)$.

Suppose the numbers of iterations required in Algorithms 1 and 2 are L_1 and L_2, respectively, and the overall complexity of Algorithm 2 is $\tilde{C}_0 = L_2(L_1\mathcal{O}(\ln(1/\tilde{\epsilon})\tilde{n}_1^3) + \mathcal{O}(\ln(1/\tilde{\epsilon})\tilde{n}_2^3))$. It can be found that the algorithm is of high complexity, making its implementation difficult especially when the scale of the network is extremely large. However, this issue can be alleviated by choosing a larger threshold ϵ to cut down the number of iterations. This involves the tradeoff between accuracy and feasibility of the algorithm, which will be further addressed in Section IV.

Algorithm 2 Joint Optimization Algorithm for Management of Radio and Computation Resources and Trajectory Planning

Input: coordinates of users $\mathbf{r}^{\text{user}}$, initial trajectories of UAVs $\mathbf{r}^{\text{uav}}$, computation tasks $\mathbf{D}$.

Output: transmit power $\mathbf{p}$, CPU frequencies of SMDs $\mathbf{f}_{\text{user}}$, CPU frequencies of MEC servers $\mathbf{f}_{\text{uav}}$, initial trajectories of UAVs $\mathbf{r}_{\text{uav}}$.

1: Obtain the coordinates of users.
2: Set up the initial trajectories for UAVs.
3: Solve $\mathbf{SP_1''}$ to find a resource allocation strategy based on initial trajectories of UAVs.
4: Compute the initial weighted system utility ρ_0.
5: Initialize the number of iteration by setting $l=0$.
6: **while** $|\rho_l - \rho_{l-1}| \ge \epsilon$ **do**
7: Solve $\mathbf{SP_2'}$ using CVX to find the optimal trajectories based on current resource allocation strategy.
8: Solve $\mathbf{SP_1''}$ using CVX to find the optimal resource allocation strategy based on current UAV trajectories.
9: Update the weighted system utility ρ_l.
10: $l = l+1$.
11: **end while**
12: Calculate transmit power $\mathbf{p}$ with sum rate $\mathbf{x}$ and channel gain $\mathbf{g}$.
13: **return** $\mathbf{p}$, $\mathbf{f}_{\text{user}}$, $\mathbf{f}_{\text{uav}}$, $\mathbf{r}_{\text{uav}}$.

IV. Numerical Results and Analysis

In this section, numerical results and analysis are provided based on simulations. First, the convergence properties of the proposed algorithm are discussed. Then, an implementation of the proposed algorithm is introduced along with a comparison with the widely used OMA counterpart [7]. Finally,

TABLE II
SIMULATION PARAMETERS

Parameters	Values
System bandwidth (B)	4 MHz
Carrier frequency (f_c)	2 GHz
PSD of noise (N_0)	10^{-17} W/Hz
Process density (C)	10^3 cycles/bit
Effective switched capacitance (κ)	10^{-28}
Maximum CPU frequency of SMDs ($f_{\max}^{\text{user}}$)	1 GHz
Maximum CPU frequency of MEC servers ($f_{\max}^{\text{uav}}$)	10 GHz
Maximum speed of UAVs ($v_{\max}$)	15 m/s
Mass of UAVs (W)	10 kg

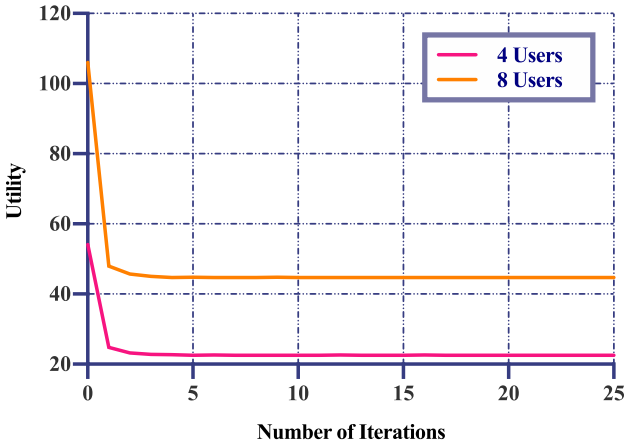

Fig. 3. Trend of utility with respect to the number of iterations with different numbers of users.

the influence of various parameters on system performance is discussed.

$U = 8$ terrestrial SMDs are uniformly distributed in a 1×1 km^2 area. Each of them has computation tasks of size 120 Mb to be finished within an interval of 100 s. Two UAVs are employed to provide edge computing for SMDs and their altitudes are fixed at $h_{\text{uav}} = 50$ m. The initial coordinates of both UAVs are $(0, 0, h_{\text{uav}})$ and the final coordinates are $(0, 1000\text{ m}, h_{\text{uav}})$ and $(1000\text{ m}, 0, h_{\text{uav}})$, respectively. Their initial paths are the straight line between initial and final coordinates. Weight factors are given as $\gamma = 10^{-4}$, $\omega_{\text{uav}} = 0.2$, and $\omega_{\text{user}} = 0.8$. The entire interval is sampled each 5 s and the rest of parameters is summarized in Table II.

Note that the process density, effective switched capacitance of CPUs at SMDs and MEC servers are all set to the values in Table II without loss of generality. These parameters can also be altered to different values according to specific application scenarios. As for the channel model, the urban scenario is adopted to obtain corresponding parameters [38]. Through S-curve fitting, we have $a = 9.61$ and $b = 0.15$. In addition, we set $\eta_{\text{LoS}} = 1$ dB and $\eta_{\text{NLoS}} = 20$ dB according to [38].

A. Convergence and Feasibility

It is imperative to first ensure the convergence of the proposed iterative algorithm during execution. Fig. 3 shows the trend of utility with respect to the number of iterations under different user numbers. The curve is the average of results in 100 rounds simulations. Note that before iteration starts, an execution of Algorithm 1 has been carried out based on initial trajectories. In both cases, the utility drops drastically at first and then gradually approaches a constant as the number of iterations increases. The fluctuation of data is below 0.1 after five iterations, which indicates the convergence of our proposed algorithm.

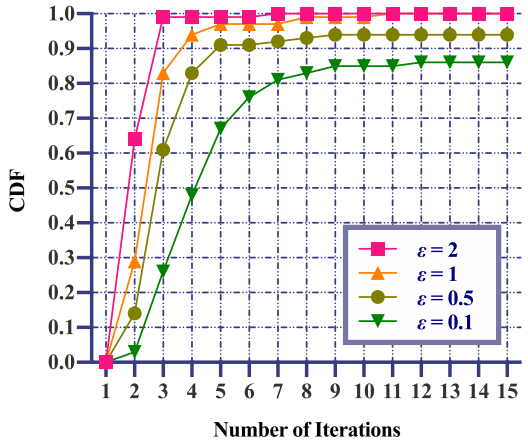

Fig. 4. Cumulative distribution function (CDF) of the number of iterations to obtain a feasible solution under different thresholds ϵ.

Recall that we consider the solution is found when the difference of utility between the current iteration and the previous one drops below ϵ, and apparently the value of threshold ϵ plays an important role in the tradeoff between accuracy and feasibility. A larger ϵ sacrifices the accuracy for a higher probability of feasibility under certain numbers of iterations and *vice versa*. Fig. 4 compares the cumulative density functions (CDFs) of the number of iterations l required to reach a solution under different thresholds, where the user number U is reset to 8. The maximum allowed number of iterations during simulation is set to $l_{\max} = 15$ because an excessive frequency leads to an unacceptable running time of the algorithm. Four curves represent the CDFs for $\epsilon = 2, 1, 0.5$, and 0.1, respectively, and all of them follow an S-curve. Increasing rates of cumulative probability first grow rapidly when l is small and then gradually drop to 0 as l becomes larger. Yet it can be found that under each number of iteration, a higher threshold ϵ always has an advantage in the probability of feasibility. This is consistent with the intuitive conclusion we have given previously.

To ensure the feasibility of algorithm during implementation, we need to stipulate a degree of confidence σ in convergence within a certain number of iterations. A common value of σ lies between 0.9 and 0.99. From Fig. 4, we can tell it is inappropriate to choose $\epsilon = 0.1$ even though it has the highest accuracy of results. The corresponding probability of feasibility is below 0.9 even after $l_{\max}$ iterations. If we assign $\sigma = 0.9$, the minimum iteration numbers required for $\epsilon = 2, 1$, and 0.5 are $3, 4$, and 6, respectively. When the confidence is raised to 0.95, only $\epsilon = 2, 1$ survive as the maximum probability of feasibility for $\epsilon = 0.5$ within $0 \le l \le l_{\max}$ iterations is 0.94. Corresponding iteration numbers required

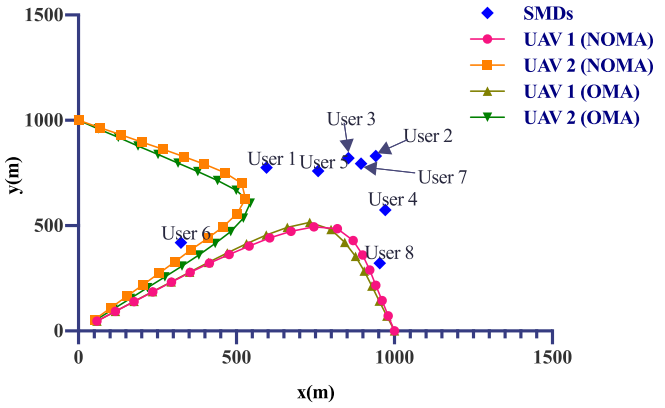

Fig. 5. Optimal trajectories of UAVs for both NOMA and OMA cases.

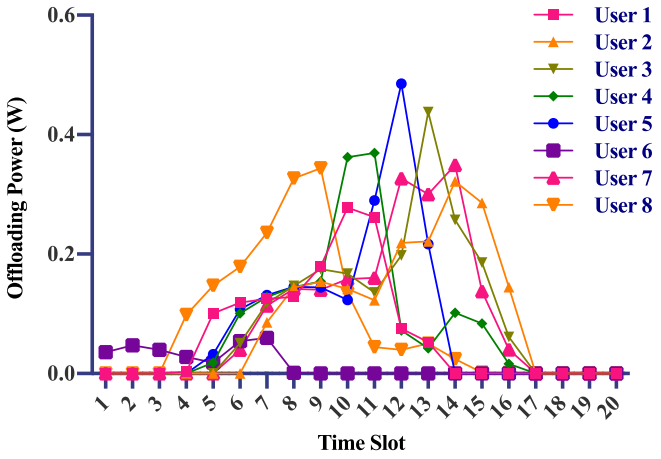

Fig. 6. Transmit power of each user in each time slot (NOMA).

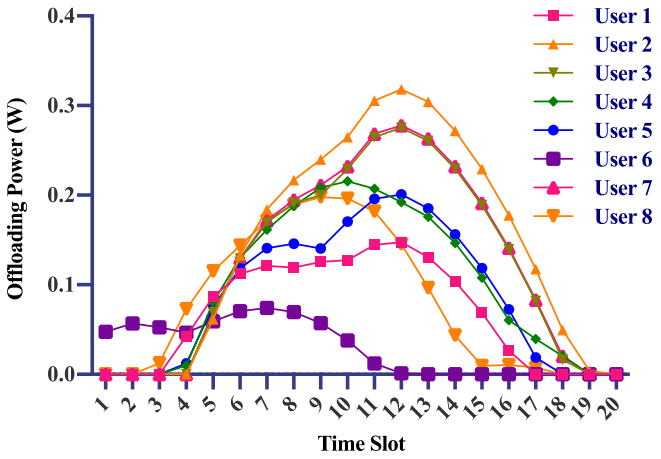

Fig. 7. Transmit power of each user in each time slot (OMA).

for $\epsilon = 2$ and 1 are 3 and 5. Although the probability climbs to 0.99 within the first three iterations for $\epsilon = 2$, this case is penalized for the lowest accuracy among four values. Thus, we conclude that $\epsilon = 1$ is more preferable as it achieves a suitable balance between rate and precision. The probability of feasibility for $\epsilon = 1$ arrives at 0.99 after the eighth iteration and reaches 1 when the tenth iteration is executed.

B. Comparison With OMA

To show the advantage of our proposed algorithm, we introduce another system as reference. This reference system is the same as the proposed system except that it adopts OMA instead of NOMA during offloading. When a single UAV is used, the reference system is similar to the one in [7]. Bandwidth for each user in the OMA system is diminished to $1/U$ of that in NOMA case, whereas no interference is considered. Thus, the relation between maximum amount of offloaded tasks and transmit power is given as

$$R_{i,m}^{\mathrm{OMA}}[n] = \frac{B \cdot \tau}{U} \log\left(1 + \frac{p_{i,m}[n] g_{i,m}[n]}{B N_0 / U}\right) \tag{32}$$

and correspondingly transmit power can be expressed as

$$p_{i,m}^{\mathrm{OMA}}[n] = \frac{B N_0}{U g_{i,m}[n]} \left[\exp\left(\frac{U \ln 2}{B\tau} R_{i,m}[n]\right) - 1\right]. \tag{33}$$

Then, the results of one realization are given in Figs. 5–8. For this specific realization of models, the system utilities for NOMA and OMA are 46.75 and 56.53 (J). Compared with the traditional OMA counterpart, our proposed algorithm reduces the weighted sum energy consumption by 17.3% when finishing the same amount of computation tasks.

Horizontal projections of the most favorable trajectories of UAVs for both NOMA and OMA cases are plotted in Fig. 5. It can be found that the optimal UAV trajectories for both cases are very similar to each other. We may further compare results in Fig. 5 with contents in Fig. 7 to find more helpful conclusions. Compared with ordinary NOMA systems, an additional degree of freedom is introduced with the deployment of UAVs. The channel condition of each link fluctuates as UAVs constantly change their positions. Each user tends to offload tasks in time slots when their channel conditions are comparatively good. For instance, the peak power of user 6 arrives in the sixth and seventh time slots, i.e., the time when UAV 1 and UAV 2 get closest to user 6, respectively. This phenomenon remains to be true for all users in both NOMA and OMA cases, as shown in Figs. 6 and 7. However, the proper time interval for task offloading can be rather short due to the constraints on UAVs' trajectories. In the OMA case, where bandwidth and power are limited, task offloading occurs even when channel conditions are not good enough in order to finish all tasks before the deadline. This to some extent leads to the ineffective utilization of energy. In contrast, all users share a much larger bandwidth in our proposed NOMA system and much less energy is required as users can complete necessary task offloading during time slots with "satisfactory" channel conditions.

As all parameters are coupled, they influence one another. Not only does NOMA improve energy efficiency in task offloading but it also reshapes the composition of the entire energy consumption. The histogram in Fig. 8 shows the composition of energy consumption for both NOMA and OMA systems. As the cost of offloading decreases in our proposed system, users are more motivated to offload tasks to MEC servers. When the same tasks are more evenly distributed between SMDs and MEC servers, the total energy consumption of computation also diminishes. It is shown that

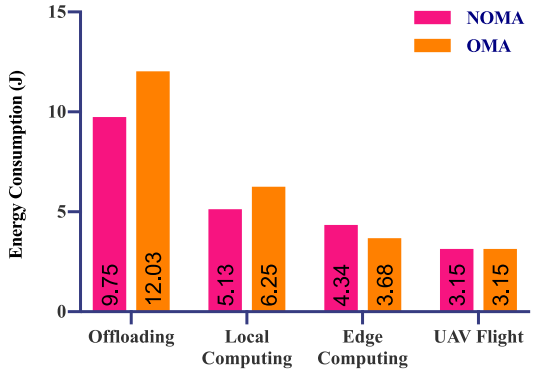

Fig. 8. Composition of system energy consumption for both NOMA and OMA cases.

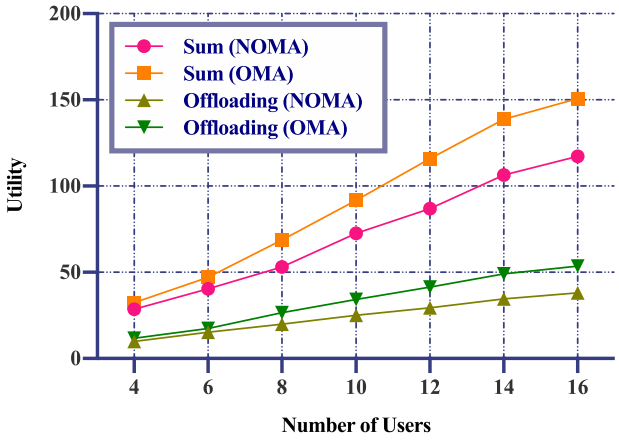

Fig. 9. Weighted system utility and energy consumption of offloading with different user numbers for both NOMA and OMA cases.

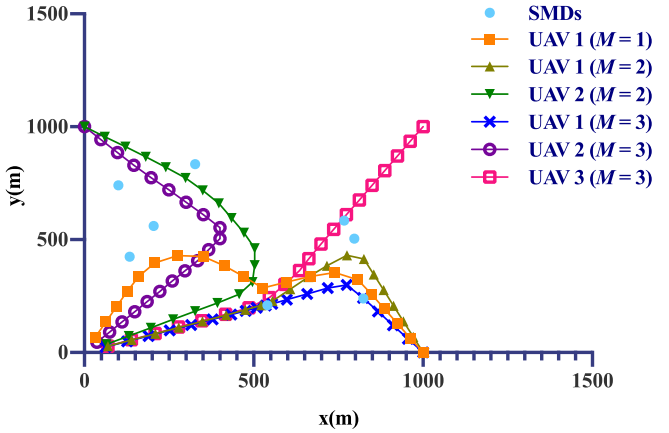

Fig. 10. Optimal UAV trajectories when different numbers of UAVs are deployed.

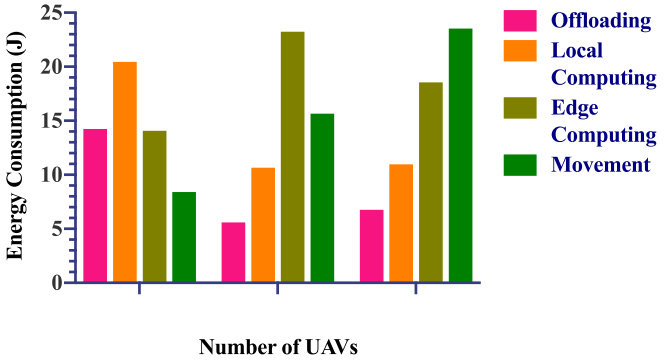

Fig. 11. Comparison of energy consumption composition between different numbers of UAVs.

computation consumes $5.13 + 4.34 = 9.47$ (J) in the NOMA case and $6.25 + 3.68 = 9.93$ (J) in the OMA case, indicating a decrease in energy cost by approximately 5%.

C. Number of Users and UAVs

In this section, the system performance with different numbers of users/UAVs is discussed. We start from the impact of the number of users. The computation capacity of MEC servers is loosed to support more users in the network and, in the meantime, the amount of computation tasks per user remains the same. How the number of users influences energy consumption is shown in Fig. 9. As the number of users increases, we can always find the advantage of the proposed NOMA system over the OMA counterpart in energy efficiency. This reward becomes more impressive as the number of users grows. The energy consumption is approximately proportional to the number of users in both NOMA and OMA cases while the gradient for NOMA is much smaller than that for OMA. Besides, we can see from Fig. 9 that the margin of NOMA in total energy consumption is much larger than that in task offloading, which implies that NOMA also brings improvement in other components of energy efficiency.

Similarly, the number of UAVs deployed in the network significantly influences the system performance. We have carried out three experiments by changing the number of UAVs from 1 to 3. The rest of the parameters remains unchanged. Fig. 10 plots trajectories of UAVs in different cases and the composition of utilities is shown in Fig. 11. When a single UAV is deployed, it has to take a long way round to approach as many users as possible. In the case of two UAVs, each UAV only needs to cover part of the SMDs. Thus, they can get closer to SMDs, which makes task offloading more energy efficient. We can see from Fig. 11 that although more tasks are offloaded to MEC servers, the energy consumption for offloading tasks is even reduced. Intuitively, increasing the number of UAVs may improve the system performance, whereas this is not always the truth. When we have $M = 3$ UAVs in the network, the additional UAV results in the unnecessary energy cost of flight. Besides, the spectrum is further divided to avoid interference between uplink transmission to different UAVs, which undermines the spectral efficiency of NOMA and causes an additional increment in transmit power. The optimal number of UAVs depends on the network scale as well as the topology of SMDs, which should be judiciously considered in the system design.

D. Weight Factors

The selection of weight factors reflects the preference of energy consumption in the system design. As referred to above, we set $\omega_{\text{user}} + \omega_{\text{uav}} = 1$. If the weighted system utility ρ is considered as a penalty function to be minimized during the

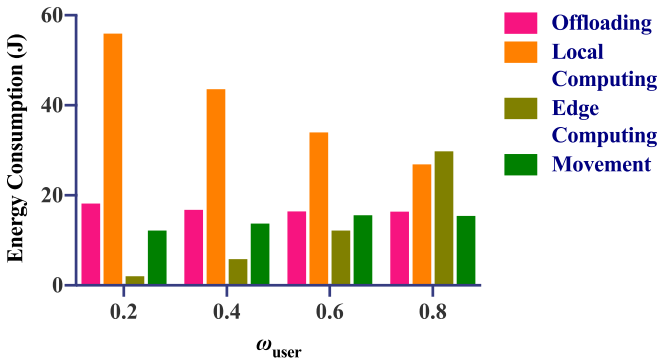

Fig. 12. Comparison of energy consumption composition between different weight factors.

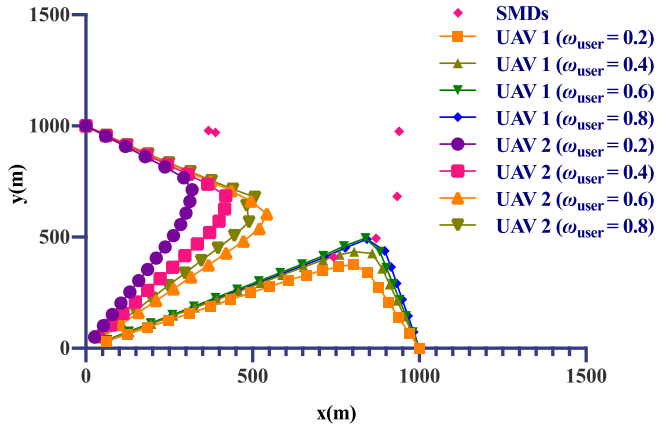

Fig. 13. Influence of weight factors ω_{user} and ω_{uav} on optimal UAV trajectories.

optimization, a larger value of ω_{user} suggests that less energy should be consumed at SMDs and *vice versa*.

Fig. 12 shows the variation in the composition of system energy consumption with respect to ω_{user}. It can be found that offloading is rarely affected by ω_{user} and ω_{uav}, while the rest is significantly dependent on the selection of weight factors. As ω_{user} grows, the optimal scheme tends to allocate more computation tasks to UAVs. Then, the energy consumption of local computing drops and the energy cost of edge computing grows dramatically. Meanwhile, the penalty caused by UAV movement is mitigated as ω_{uav} declines, and UAVs will spend more energy on flight to get closer to users for better channel conditions. Interestingly, the sum energy consumption for $\omega_{\text{user}} = 0.2, 0.4, 0.6$, and 0.8 is 88.32, 79.84, 78.20, and 88.50 (J), respectively, which does not change monotonically with weight factors. When the difference between ω_{user} and ω_{uav} gets smaller, so does the sum energy consumption. However, this may not agree with our choice in the real world, as the energy of users is, in general, more precious than that of MEC servers. SMDs are usually more compact and deployed in various sites, while UAVs can often accept larger batteries and they can be recharged more frequently. We may often sacrifice total energy consumption for extended battery life at SMDs by altering weight factors.

Fig. 13 further reveals the impact of ω_{user} and ω_{uav} on optimal UAV trajectories. As ω_{uav} decreases, the length of optimal trajectories increases and UAVs can then get closer to SMDs. The related improvement in channel conditions during time slots for task offloading is beneficial, because a reduction in transmit power at SMDs is far more critical than the energy consumption of UAV flight for $\omega_{\text{user}} > \omega_{\text{uav}}$. Interestingly, we find that the change in UAV 2's trajectory is more prominent than that in UAV 1's trajectory, since UAV 1 has already approximated its optimal trajectory (close enough to certain users). In this situation, the influence of ω_{uav} on UAV 1 is diminished. For UAV 2 whose trajectory is far from all users, extending trajectory for shorter distances to users turns out to be particularly rewarding.

V. Conclusion

In this article, we designed a multi-UAV-enabled MEC network integrating NOMA for task offloading. Given the computation tasks and deadline, the joint optimization of radio and computation resources as well as UAV trajectory planning was considered to minimize the weighted system energy consumption. The proposed algorithm splits the original problem into two subproblems. By alternately solving each of them with approximation techniques, we finally yield the optimal solution to the original problem. Furthermore, numerical results were provided to demonstrate the improvement in the energy efficiency of our proposed algorithm. Due to the incorporation of NOMA, weighted energy consumption was reduced by approximately 1/5 in a network of eight users compared with its OMA counterpart and this advantage is further enlarged as the number of users keeps growing.

However, there are still many issues worth additional discussion. The partition of computation tasks in this article is completely determined by the offloading strategy, whereas the splitting of data should consider the specific features of tasks. The arbitrary division may harm the completeness of data and a more reasonable model is preferable. Besides, with regard to implementation, the number of users sharing the same spectrum in NOMA should be restricted using user scheduling as too many users can result in the prohibitive complexity of receiver structures and error propagation in SIC. Nevertheless, the introduction of user scheduling involves integer programming, which further complicates the model. Our future work will take into account the aforementioned issues and try to seek algorithms with low complexity.

References

[1] Qualcomm Technologies, Inc. (2016). *Making 5G NR a Reality*. [Online]. Available: https://www.qualcomm.com/media/documents/files/whitepaper-making-5g-nr-a-reality.pdf

[2] T. Taleb, S. Dutta, A. Ksentini, M. Iqbal, and H. Flinck, "Mobile edge computing potential in making cities smarter," *IEEE Commun. Mag.*, vol. 55, no. 3, pp. 38–43, Mar. 2017.

[3] M. Maier and A. Ebrahimzadeh, "Towards immersive tactile Internet experiences: Low-latency FiWi enhanced mobile networks with edge intelligence (invited paper)," *IEEE/OSA J. Opt. Commun. Netw.*, vol. 11, no. 4, pp. B10–B25, Apr. 2019.

[4] L. Gupta, R. Jain, and G. Vaszkun, "Survey of important issues in UAV communication networks," *IEEE Commun. Surveys Tuts.*, vol. 18, no. 2, pp. 1123–1152, 2nd Quart., 2016.

[5] M. Mozaffari, W. Saad, M. Bennis, Y.-H. Nam, and M. Debbah, "A tutorial on UAVs for wireless networks: Applications, challenges, and open problems," *IEEE Commun. Surveys Tuts.*, vol. 21, no. 3, pp. 2334–2360, 3rd Quart., 2019.

[6] Y. Du, K. Wang, K. Yang, and G. Zhang, "Energy-efficient resource allocation in UAV based MEC system for IoT devices," in *Proc. IEEE Global Commun. Conf. (GLOBECOM)*, 2018, pp. 1–6.

[7] F. Zhou, Y. Wu, H. Sun, and Z. Chu, "UAV-enabled mobile edge computing: Offloading optimization and trajectory design," in *Proc. IEEE Int. Conf. Commun. (ICC)*, May 2018, pp. 1–6.

[8] F. Zhou, Y. Wu, R. Q. Hu, and Y. Qian, "Computation rate maximization in UAV-enabled wireless-powered mobile-edge computing systems," *IEEE J. Sel. Areas Commun.*, vol. 36, no. 9, pp. 1927–1941, Sep. 2018.

[9] Q. Hu, Y. Cai, G. Yu, Z. Qin, M. Zhao, and G. Y. Li, "Joint offloading and trajectory design for UAV-enabled mobile edge computing systems," *IEEE Internet Things J.*, vol. 6, no. 2, pp. 1879–1892, Apr. 2019.

[10] J. Zhang *et al.*, "Stochastic computation offloading and trajectory scheduling for UAV-assisted mobile edge computing," *IEEE Internet Things J.*, vol. 6, no. 2, pp. 3688–3699, Apr. 2019.

[11] M.-A. Messous, S.-M. Senouci, H. Sedjelmaci, and S. Cherkaoui, "A game theory based efficient computation offloading in an UAV network," *IEEE Trans. Veh. Technol.*, vol. 68, no. 5, pp. 4964–4974, May 2019.

[12] Y. Saito, Y. Kishiyama, A. Benjebbour, T. Nakamura, A. Li, and K. Higuchi, "Non-orthogonal multiple access (NOMA) for cellular future radio access," in *Proc. IEEE 77th Veh. Technol. Conf. (VTC Spring)*, Jun. 2013, pp. 1–5.

[13] D. Tse and P. Viswanath, *Fundamentals of Wireless Communications*. Cambridge, U.K.: Cambridge Univ. Press, 2005.

[14] S. Sardellitti, G. Scutari, and S. Barbarossa, "Joint optimization of radio and computational resources for multicell mobile-edge computing," *IEEE Trans. Signal Inf. Process. Netw.*, vol. 1, no. 2, pp. 89–103, Jun. 2015.

[15] J. Zhang *et al.*, "Energy-latency tradeoff for energy-aware offloading in mobile edge computing networks," *IEEE Internet Things J.*, vol. 5, no. 4, pp. 2633–2645, Aug. 2018.

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

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

[18] B. Cao, L. Zhang, Y. Li, D. Feng, and W. Cao, "Intelligent offloading in multi-access edge computing: A state-of-the-art review and framework," *IEEE Commun. Mag.*, vol. 57, no. 3, pp. 56–62, Mar. 2019.

[19] Q. Cui *et al.*, "Stochastic online learning for mobile edge computing: Learning from changes," *IEEE Commun. Mag.*, vol. 57, no. 3, pp. 63–69, Mar. 2019.

[20] Y. Zeng, R. Zhang, and T. J. Lim, "Wireless communications with unmanned aerial vehicles: Opportunities and challenges," *IEEE Commun. Mag.*, vol. 54, no. 5, pp. 36–42, May 2016.

[21] X. Lin *et al.*, "The sky is not the limit: LTE for unmanned aerial vehicles," *IEEE Commun. Mag.*, vol. 56, no. 4, pp. 204–210, Apr. 2018.

[22] Y. Qian *et al.*, "User association and path planning for UAV-aided mobile edge computing with energy restriction," *IEEE Wireless Commun. Lett.*, vol. 8, no. 5, pp. 1312–1315, Oct. 2019.

[23] J. Hu, M. Jiang, Q. Zhang, Q. Li, and J. Qin, "Joint optimization of UAV position, time slot allocation, and computation task partition in multiuser aerial mobile-edge computing systems," *IEEE Trans. Veh. Technol.*, vol. 68, no. 7, pp. 7231–7235, Jul. 2019.

[24] A. Asheralieva and D. Niyato, "Hierarchical game-theoretic and reinforcement learning framework for computational offloading in UAV-enabled mobile edge computing networks with multiple service providers," *IEEE Internet Things J.*, vol. 6, no. 5, pp. 8753–8769, Oct. 2019.

[25] L. Dai, B. Wang, Y. Yuan, S. Han, C. I, and Z. Wang, "Non-orthogonal multiple access for 5G: solutions, challenges, opportunities, and future research trends," in *IEEE Commun. Mag.*, vol. 53, no. 9, pp. 74–81, Sep. 2015.

[26] S. M. R. Islam, N. Avazov, O. A. Dobre, and K. Kwak, "Power-domain non-orthogonal multiple access (NOMA) in 5G systems: Potentials and challenges," *IEEE Commun. Surveys Tuts.*, vol. 19, no. 2, pp. 721–742, 2nd Quart., 2017.

[27] L. Lei, D. Yuan, C. K. Ho, and S. Sun, "Power and channel allocation for non-orthogonal multiple access in 5G systems: Tractability and computation," *IEEE Trans. Wireless Commun.*, vol. 15, no. 12, pp. 8580–8594, Dec. 2016.

[28] Z. Ding, P. Fan, and H. V. Poor, "Impact of user pairing on 5G nonorthogonal multiple-access downlink transmissions," *IEEE Trans. Veh. Technol.*, vol. 65, no. 8, pp. 6010–6023, Aug. 2016.

[29] F. Fang, H. Zhang, J. Cheng, and V. C. M. Leung, "Energy-efficient resource allocation for downlink non-orthogonal multiple access network," *IEEE Trans. Commun.*, vol. 64, no. 9, pp. 3722–3732, Sep. 2016.

[30] K. Yang, N. Yang, N. Ye, M. Jia, Z. Gao, and R. Fan, "Non-orthogonal multiple access: Achieving sustainable future radio access," *IEEE Commun. Mag.*, vol. 57, no. 2, pp. 116–121, Feb. 2019.

[31] A. Kiani and N. Ansari, "Edge computing aware NOMA for 5G networks," *IEEE Internet Things J.*, vol. 5, no. 2, pp. 1299–1306, Apr. 2018.

[32] Y. Pan, M. Chen, Z. Yang, N. Huang, and M. Shikh-Bahaei, "Energy-efficient NOMA-based mobile edge computing offloading," *IEEE Commun. Lett.*, vol. 23, no. 2, pp. 310–313, Feb. 2019.

[33] Z. Ding, D. W. K. Ng, R. Schober, and H. V. Poor, "Delay minimization for NOMA-MEC offloading," *IEEE Signal Process. Lett.*, vol. 25, no. 12, pp. 1875–1879, Dec. 2018.

[34] Z. Ding, P. Fan, and H. V. Poor, "Impact of non-orthogonal multiple access on the offloading of mobile edge computing," *IEEE Trans. Commun.*, vol. 67, no. 1, pp. 375–390, Jan. 2019.

[35] T. Hou, Y. Liu, Z. Song, X. Sun, and Y. Chen, "Multiple antenna aided NOMA in UAV networks: A stochastic geometry approach," *IEEE Trans. Commun.*, vol. 67, no. 2, pp. 1031–1044, Feb. 2019.

[36] W. Mei and R. Zhang, "Uplink cooperative NOMA for cellular-connected UAV," *IEEE J. Sel. Topics Signal Process.*, vol. 13, no. 3, pp. 644–656, Jun. 2019.

[37] Q. Wu, Y. Zeng, and R. Zhang, "Joint trajectory and communication design for multi-UAV enabled wireless networks," *IEEE Trans. Wireless Commun.*, vol. 17, no. 3, pp. 2109–2121, Mar. 2018.

[38] A. Al-Hourani, S. Kandeepan, and S. Lardner, "Optimal LAP altitude for maximum coverage," *IEEE Wireless Commun. Lett.*, vol. 3, no. 6, pp. 569–572, Dec. 2014.

[39] M. Mozaffari, W. Saad, M. Bennis, and M. Debbah, "Drone small cells in the clouds: Design, deployment and performance analysis," in *Proc. IEEE Global Commun. Conf. (GLOBECOM)*, Dec. 2015, pp. 1–6.

[40] H. Zhao, H. Wang, W. Wu, and J. Wei, "Deployment algorithms for UAV airborne networks toward on-demand coverage," *IEEE J. Sel. Areas Commun.*, vol. 36, no. 9, pp. 2015–2031, Sep. 2018.

[41] E. Kalantari, I. Bor-Yaliniz, A. Yongacoglu, and H. Yanikomeroglu, "User association and bandwidth allocation for terrestrial and aerial base stations with backhaul considerations," in *Proc. IEEE 28th Annu. Int. Symp. Pers. Indoor Mobile Radio Commun. (PIMRC)*, Oct. 2017, pp. 1–6.

[42] M. Mozaffari, W. Saad, M. Bennis, and M. Debbah, "Mobile unmanned aerial vehicles (UAVs) for energy-efficient Internet of Things communications," *IEEE Trans. Wireless Commun.*, vol. 16, no. 11, pp. 7574–7589, Nov. 2017.

[43] H. Wang, H. Zhao, W. Wu, J. Xiong, D. Ma, and J. Wei, "Deployment algorithms of flying base stations: 5G and beyond with UAVs," *IEEE Internet Things J.*, vol. 6, no. 6, pp. 10009–10027, Dec. 2019.

[44] W. Zhang, Y. Wen, K. Guan, D. Kilper, H. Luo, and D. O. Wu, "Energy-optimal mobile cloud computing under stochastic wireless channel," *IEEE Trans. Wireless Commun.*, vol. 12, no. 9, pp. 4569–4581, Sep. 2013.

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

[46] A. Filippone, *Flight Performance of Fixed and Rotary Wing Aircraft*. London, U.K.: Elsevier, 2006.

[47] S. Jeong, O. Simeone, and J. Kang, "Mobile edge computing via a UAV-mounted cloudlet: Optimization of bit allocation and path planning," *IEEE Trans. Veh. Technol.*, vol. 67, no. 3, pp. 2049–2063, Mar. 2018.

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

[49] M. C. Grant and S. P. Boyd. (Dec. 2018). *CVX: MATLAB Software for Disciplined Convex Programming*. [Online]. Available: http://web.cvxr.com/cvx/doc/CVX.pdf

[50] J. Xiong, D. Ma, H. Zhao, and F. Gu, "Secure multicast communications in cognitive satellite-terrestrial networks," *IEEE Commun. Lett.*, vol. 23, no. 4, pp. 632–635, Apr. 2019.

[51] K. Wang, A. M. So, T. Chang, W. Ma, and C. Chi, "Outage constrained robust transmit optimization for multiuser MISO downlink: Tractable approximations by conic optimization," *IEEE Trans. Signal Process.*, vol. 62, no. 21, pp. 5690–5705, Nov. 2014.

Xiaochen Zhang received the B.S. degree from the National University of Defense Technology, Changsha, China, in 2018, where he is currently pursuing the graduation degree with the College of Electronic Science and Engineering.

His research interests include resource allocation, multiaccess edge computing, machine learning, and channel modeling.

Jiao Zhang received the B.S. degree from Xiangtan University, Xiangtan, China, in 2013, and the M.S. degree from Xidian University, Xi'an, China, in 2016. She is currently pursuing the Ph.D. degree with the College of Electronic Science and Engineering, National University of Defense Technology, Changsha, China.

Her research interests include mobile-edge computing and resource allocation in heterogeneous networks.

Jun Xiong received the B.S. and Ph.D. degrees from the National University of Defense Technology (NUDT), Changsha, China, in 2009 and 2014, respectively.

He is currently an Associate Professor with the College of Electronic Science and Engineering, NUDT. His research interests include cooperative communications, physical-layer security, and resource allocation.

Li Zhou received the B.S., M.S., and Ph.D. degrees from the National University of Defense Technology (NUDT), Changsha, China, in 2009, 2011, and 2015, respectively.

From September 2013 to September 2014, he worked as a Visiting Scholar with the University of British Columbia, Vancouver, BC, Canada. He is currently an Associate Professor with the College of Electronic Science and Engineering, NUDT. His research contributions have been published and presented in more than 40 prestigious journals and conferences. His research interests are in the area of wireless networks, software-defined systems, and machine learning.

Dr. Zhou was invited as a Keynote Speaker in China Satellite 2017 and ICWCNT 2016. He served as a TPC Member of IEEE ICC 2020/2019 and IEEE CIT 2018/2017, and the Co-Chair of ITA 2016.

Jibo Wei received the B.S. and M.S. degrees in electronic engineering from the National University of Defense Technology (NUDT), Changsha, China, in 1989 and 1992, respectively, and the Ph.D. degree in electronic engineering from Southeast University, Nanjing, China, in 1998.

He is currently the Director and a Professor with the Department of Communication Engineering, NUDT. His research interests include wireless network protocol and signal processing in communications, more specifically, the areas of MIMO, multicarrier transmission, cooperative communication, and cognitive network.

Prof. Wei is an editor of the *Journal on Communications* and the Senior Member of the China Institute of Communications and Electronics. He is a member of the IEEE Communications Society and IEEE Vehicular Technology Society.