Joint Multi-domain Resource Allocation and Trajectory Optimization in UAV-assisted M-LoT Networks

This paper was downloaded from TechRxiv (<https://www.techrxiv.org>).

LICENSE

CC BY 4.0

SUBMISSION DATE / POSTED DATE

30-12-2021 / 05-01-2022

CITATION

Qian, Liping (2022): Joint Multi-domain Resource Allocation and Trajectory Optimization in UAV-assisted M-LoT Networks. TechRxiv. Preprint. <https://doi.org/10.36227/techrxiv.17704463.v1>

DOI

[10.36227/techrxiv.17704463.v1](https://doi.org/10.36227/techrxiv.17704463.v1)

Joint Multi-domain Resource Allocation and Trajectory Optimization in UAV-assisted M-IoT Networks

Li Ping Qian, *Senior Member, IEEE*, Hongsen Zhang, Qian Wang, *Member, IEEE*,
Yuan Wu, *Senior Member, IEEE*, and Bin Lin, *Senior Member, IEEE*

Abstract—The integration of Maritime Internet of Things (M-IoT) technology and unmanned aerial/surface vehicles (UAVs/USVs) has been emerging as a promising navigational information technique in intelligent ocean systems. With the unprecedented increase of computation-intensive yet latency-sensitive marine mobile Internet services, mobile edge computing (MEC) and non-orthogonal multiple access (NOMA) have been envisioned as promising approaches to providing with the low-latency as well as reliable computing services and ultra-dense connectivity. In this paper, we investigate the energy consumption minimization based energy-efficient MEC via cooperative NOMA for the UAV-assisted M-IoT networks. We consider that USVs offload their computation-workload to the UAV equipped with the edge-computing server subject to the UAV mobility. To improve the energy efficiency of offloading transmission and workload computation, we focus on minimizing the total energy consumption by jointly optimizing the USVs' offloaded workload, transmit power, computation resource allocation as well as the UAV trajectory subject to the USVs' latency requirements. Despite the nature of mixed discrete and non-convex programming of the formulated problem, we exploit the vertical decomposition and propose a two-layered algorithm for solving it efficiently. Specifically, the top-layered algorithm is proposed to solve the problem of optimizing the UAV trajectory based on the idea of Deep Reinforcement Learning (DRL), and the underlying algorithm is proposed to optimize the underlying multi-domain resource allocation problem based on the idea of the Lagrangian multiplier method. Numerical results are provided to validate the effectiveness of our proposed algorithms as well as the performance advantage of NOMA-enabled computation offloading in terms of overall energy consumption.

Index Terms—Maritime Internet of Things (M-IoT), Non-orthogonal multiple access (NOMA), Mobile edge computing (MEC), Multi-domain resource allocation, UAV trajectory optimization

I. INTRODUCTION

THE Internet of Things (IoT) has promoted the rapid development of various applications, such as smart metering, smart manufacturing, smart city/ocean, automatic driving, health monitoring, and many others (see a recent survey in [1]). As a representative IoT application, the Maritime Internet of

Things (M-IoT) has been emerging as a promising navigational information technique for maritime environmental monitoring, underwater biological monitoring, tactical monitoring, and aquaculture monitoring [2]. With the advancement of unmanned surface vehicles (USVs), it has been explored as a type of M-IoT devices for ever-increasing maritime activities [3]. Along with ever-growing deployment of USVs, the M-IoT poses serious challenges for dealing with a massive amount of computation-intensive yet latency-sensitive maritime data in an energy-efficient and low-latency way due to the limited computation units of USVs.

To cope with these challenges, the integration of unmanned aerial vehicle communication (UAV-C), mobile edge computing (MEC), and non-orthogonal multiple access (NOMA) have been envisioned to bring enormous potential benefits to the M-IoT. MEC, which deploys a sufficient amount of computation-resources at the edge of wireless networks (e.g., macro/micro base stations or access points), can offer cloud computing capabilities to the computation-intensive maritime services [4]. However, considering the limited computation-resources and energy of USVs, the computation-intensive maritime services might suffer severe transmission latency when offloaded to the onshore MEC units far from USVs. To this end, the unmanned aerial vehicle (UAV) can be exploited as portable MEC units in close proximity to USVs due to the benefits of mobility, flexibility and manoeuvrability [5]. Through offloading parts of intensive computation tasks from USVs to MEC units equipped in the UAV, the latency in completing the tasks as well as the transmission bandwidth of sending the tasks can be efficiently reduced for USVs. However, the scarcity of spectrum resource leads to a heavy pressure on the efficient data transmission between the UAV and massive USVs, and a degraded quality of experience consequently. NOMA, which serves multiple USVs on the same frequency-time resource simultaneously through the non-orthogonal resource allocation in the power domain or code domain, has provided a promising paradigm to address this issue [6]. Thanks to the great potentials, the paradigms of MEC, NOMA and UAV-C has attracted lots of research efforts in the past years [7]–[12].

Recently, the advantages of NOMA have also motivated the studies on the offloading of MEC [7], [8]. For example, Huang et al. [7] investigated the joint task offloading and resource allocation for achieve both channel diversity and computing diversity faster in NOMA-based MEC networks. Ding et al. [8] considered a NOMA-assisted MEC system, in which the

Li Ping Qian, Hongsen Zhang, and Qian Wang are with the College of Information Engineering, Zhejiang University of Technology, Hangzhou 310023, China (e-mail: lpqian@zjut.edu.cn; wang.qian@u.nus.edu).

Yuan Wu is with the State Key Laboratory of Internet of Things for Smart City, University of Macau, Macau, China, and also with the Department of Computer Information Science, University of Macau, Macau, China (e-mail: yuanwu@um.edu.mo).

Bin Lin is with the Department of Communication Engineering, Dalian Maritime University, Dalian 116026, China (e-mail: binlin@dlmu.edu.cn).

power and time allocation was jointly optimized for decreasing the energy consumption of computation offloading. Due to the manoeuvrability of UAVs, the integration of UAV-C and MEC can further improve the computation offloading performance [9], [10]. Yang et al. [9] minimized the total energy consumption by jointly optimizing user association, the offloading factor, power control, computation resource allocation, and UAVs' locations for UAV-enabled MEC networks. Zhang et al. [10] utilized the UAV as a computing node as well as a relay node to improve the average user latency in the UAV-aided MEC network, minimized the average latency of all users by jointly optimizing nnnnnnnnnn.

As the aerial base station, the UAV is subject to the additional propulsion power consumption for maintaining its suspension and motion, and thus it is of practical meaning to improve the system performance by optimizing the UAV trajectory for UAV-assisted MEC networks. For example, Zhang et al. [11] maximized the energy efficiency by iteratively optimizing the UAV trajectory. Hu et al. [12] proposed the penalty dual decomposition based algorithm to minimize the maximum delay among all users by jointly optimizing the UAV trajectory, the offloading ratio of task, and the user scheduling.

Despite the tremendous efforts have been devoted to UAV-assisted MEC networks [7]–[12], the minimization of total energy consumption has been not yet studied for UAV-assisted M-IoT networks, in the optimization framework of transmit power, task offloading ratio, computation resource allocation, and UAV trajectory. Considering the limited energy provisioning of UAVs and USVs, it is crucial to prolong their lifetime by minimizing the total energy consumption. Motivated by the above considerations, in this work, we aim to minimize the total energy consumption of UAVs and USVs by jointly optimizing the communication and computation resource allocation, and the UAV trajectory as well. Our contribution can be summarized as follows.

- **Problem formulation:** We first propose a NOMA-based MEC model for the UAV-assisted M-IoT systems that provides UAVs with the low-latency and reliable computing services. We then formulate the total energy consumption minimization problem based on the proposed system model, which jointly optimizes USVs' transmit power, task offloading ratio, computation resource allocation and the UAV trajectory as well.
- **Efficient algorithm design:** Despite the strict non-convexity of the formulated joint optimization problem, we identify the hidden convexity of resource allocation under the given UAV trajectory, and analytically characterize the consequently optimal USVs' resource allocation decision. Exploiting this feature, we propose an efficient layered algorithm for solving the joint optimization problem, in which the resource allocation and UAV trajectory are alternatively optimized. Specifically, we achieve the optimal resource allocation decision in the closed-form expression based on the Lagrange dual method and the idea of gradient descent. Furthermore, we propose the deep deterministic policy gradient (DDPG) algorithm to find the optimal UAV trajectory on the

basis of the proposed resource allocation optimization algorithm.

- **Performance evaluation:** Extensive simulation results are provided to validate our proposed algorithms as well as the performance of our proposed UAV-assisted M-IoT system. Specifically, our proposed joint transmit power and computation resource allocations for NOMA transmission can significantly reduce the energy consumption, in comparison with the conventional OMA-based transmission scheme. In addition, a series of numerical results demonstrate that the proposed UAV trajectory optimization algorithm as well as the resource allocation optimization can effectively reduce the whole M-IoT networks system's total energy consumption.

The remainder of this paper is organized as follows. We review the related studies in Section II. We present the UAV-assisted M-IoV network model and problem formulation in Section III. Section IV presents a two-layered algorithm to solve the proposed optimization problem efficiently. The performance of the proposed algorithm is evaluated through extensive simulations in Section V. Finally we conclude this work and discuss the future direction.

II. RELATED WORK

Driven by the potential benefits, many research efforts have been spent on the studies of NOMA, MEC, and UAV-C recently. Considering this work focused on the NOMA-based MEC model for the UAV-assisted M-IoT system, we elaborate on the related studies on NOMA-enabled MEC and UAV trajectory optimization in this section.

Studies on NOMA-MEC: A proper joint management of NOMA and MEC has been explored to improve the performance of computation and communication. There have been many studies on investigating the integration of MEC and NOMA [13]–[16]. For example, Qian et al. [13] proposed an optimal algorithm for the NOMA-MEC-assisted IoT network to obtain the optimal SIC ordering and computation resource allocation by the convex computation resource allocation optimization followed by the combinatorial SIC ordering optimization. Fang et al. [14] studied a multi-user NOMA-MEC network, and proposed a low complexity algorithm to solve the task delay minimization by optimizing the task partition ratio and transmit power. Li et al. [15] investigated how to exploit the cooperative mechanism between NOMA user pairs to enhance the security of the NOMA-MEC system. Wu et al. [16] maximized the minimum anti-eavesdropping ability for uplink NOMA users in the context of the worst-case secrecy rate requirements and limited transmit power budgets for the uplink NOMA-MEC network. Fang et al. [17] proposed a low complexity algorithm to minimize the total energy consumption by the task assignment, power allocation and user association for the multi-user NOMA-MEC network. With the rapid development of artificial intelligence, many studies have been on spent for exploiting the potentials of NOMA and MEC by designing disparate learning schemes [18]–[20]. Tuong et al. [18] utilized the deep Q-network and actor-critic network to reduce effectively the computational

overhead by jointly optimizing the computation offloading policy and channel resource allocation in a NOMA-MEC network. Qian et al. [19] proposed a cross-entropy algorithm based on the probabilistic learning, to find the optimal pairing of vehicular computing-users in NOMA aided vehicular edge computing networks. Li et al. [20] proposed a deep learning-based intelligent offloading scheme to minimize the total energy consumption in dual-connectivity and NOMA-assisted computation offloading systems.

Studies on UAV trajectory optimization: There have been extensive studies recently done to address a variety of challenges in UAV-C networks. Designing the optimal trajectory of the UAV is not only one of the research challenges, but also a significant problem which we have to overcome. In particular, designing the optimal trajectory of UAVs need to consider various design aspects. For example, due to the attractive advantages of UAVs in terms of their flexible movement and easy deployment, the integration of UAV-C with MEC can further enhance the computational performance [21]–[26]. Yu et al. [21] studied an UAV-enabled MEC system and proposed an efficient algorithm based on the successive convex approximation to minimize the weighted sum of the service delay of all IoT devices and UAV energy consumption by jointly optimizing UAV position, communication and computing resource allocation, and task splitting decisions. Ji et al. [22] developed alternating iterative algorithms based on the block alternating descent method to minimize the weighted-sum energy consumption of the UAV and user devices by jointly optimizing the UAV trajectory and computation resource allocation subject to the number of computation bits. The UAV's velocity has an enormous influence on its own energy consumption. Thus, Zeng et al. [23] investigated a point-to-point communication link between the UAV and a ground user, and the UAV's trajectory was optimized under a UAV's energy consumption model at different speeds. Mei et al. [24] proposed a block coordinate descent algorithm to minimize the overall energy consumption of the UAV by optimizing the resource allocation and UAV trajectory. Hu et al. [25] studied a UAV-assisted MEC system and proposed a novel penalty dual decomposition-based algorithm to minimize the sum of the maximum delay among all the users by jointly optimizing the UAV trajectory, task offloading ratio, and user scheduling. Zhang et al. [26] minimized the total energy consumption by optimizing the bits allocation, time slot scheduling, and power allocation as well as UAV trajectory design, via the Lagrangian duality method for a UAV-assisted MEC system. Considering Zhang et al. [27] investigated an energy-efficient UAV-assisted MEC framework in which multiple UAVs are deployed as edge servers to provide computation offloading to terrestrial users in NOMA. Also, the artificial intelligence has drawn lots research interests to exploit potentials of NOMA and MEC in the UAV communication system [28], [29]. For example, a DRL algorithm was proposed in [28] to improve the performance of UAV communication by jointly optimizing the UAV trajectory, transmit power and power reflection coefficients. Yin et al. [29] proposed the multi-agent reinforcement learning algorithm to optimize both overall and fairness throughput.

These previous work mainly focused on optimizing the

Fig. 1. A NOMA based UAV-assisted M-IoT network consisting of a UAV flying in the air and a group of USVs sailing on the sea, in which all USVs can offload their computation-intensive tasks to the UAV for the lower task execution latency in NOMA.

multi-domain resource allocation for MEC-enabled OMA communication systems with/without the aid of UAVs. To the best of our knowledge, the total energy consumption minimization has been studied for the UAV-assisted M-IoT network in which USVs offload the computation-intensive tasks to the hovering UAV in the NOMA.

III. SYSTEM MODEL AND PROBLEM FORMULATION

A. System Model

In this paper, we consider a NOMA based UAV-assisted M-IoV MEC system as illustrated in Fig.1, which consists of a UAV and a group of USVs sailing on the sea denoted as $\mathcal{N} = \{1, \dots, N\}$. Every USV has a computing processor which can compute simple tasks locally. The UAV equipped with powerful MEC units is deployed in the air as an aerial base station to collect and process the computation-intensive tasks offloaded from USVs. All USVs associated with the UAV can offload their computation-intensive tasks to the UAV for the lower task execution latency in NOMA, and then the task signals of all USVs are decoded by successively canceling the co-channel interference at the UAV. In order to execute the offloaded computation-intensive tasks more efficiently, the UAV flies at a fixed altitude level height H above the sea surface for the task collection during the time duration T . To represent the UAV trajectory, we divide the time duration T into M equal-length time slots, each denoted by $m \in \mathcal{M} = \{1, \dots, M\}$. We use the 3-D Cartesian coordinate system to represent the locations of USVs and UAV, and set the z-coordinate of sea surface to be 0. We define the coordinate of the n -th USV in the m th time slot as $q_{nm} = (x_{nm}, y_{nm}, z_{nm})$, where x_{nm} , y_{nm} , and z_{nm} mean the x-coordinate, y-coordinate, and z-coordinate in the m th time slot, respectively. The location of UAV in the m th time slot is denoted by $U_m = (X_m, Y_m, H)$, where X_m and Y_m mean the x-coordinate and y-coordinate of UAV in the m th time slot, respectively.

B. Problem Formulation

Due to the air-to-ocean channel, the propagation between the UAV and USA can be considered as the line-of-sight (LoS)

propagation. Thus, the channel gain between the n -th USV and the UAV in the m th time slot can be represented as

$$h_{nm} = \frac{h_0}{\|U_m - q_{nm}\|^2 + H^2}, \forall n \in \mathcal{N}, \quad (1)$$

where h_0 denotes the channel gain of LoS propagation at the reference distance $d_0 = 1$ m. Considering the NOMA, we adopt the successive cancellation ordering in the descending ordering of channel gains, i.e.,

$$h_{1m} \geq h_{2m} \geq \dots \geq h_{nm} \geq \dots \geq h_{Nm}, \forall n \in \mathcal{N}. \quad (2)$$

Let p_{nm} be the USV n 's transmit power in the m th time slot. Based on the Shannon capacity formula, the data rate of the USV n in the m th time slot is expressed as

$$R_{nm} = W_m \log \left(1 + \frac{h_{nm} p_{nm}}{\sum_{j=n+1}^N h_{jm} p_{jm} + n_0} \right), \forall n \in \mathcal{N}, \quad (3)$$

where W_m denotes the bandwidth used in the m th time slot, and n_0 denotes the power of additive Gaussian noise at the UAV. Suppose that every USV n has a B_n -bit computation-intensive tasks, which can be computed locally or offloaded to the UAV for the computation. Let γ_n denote the ratio of computation-intensive task offloaded from the USV n to the UAV. We use B_{nm} to indicate the number of bits offloaded by the USV n in the m th time slot, which satisfies

$$\gamma_n B_n = \sum_{m=1}^M B_{nm}. \quad (4)$$

By (3) and (4), we have

$$\sum_{m=1}^M W_m \log \left(1 + \frac{h_{nm} p_{nm}}{\sum_{j=n+1}^N h_{jm} p_{jm} + n_0} \right) = \frac{M}{T} \gamma_n B_n. \quad (5)$$

The USVs and UAV will execute the task data computing individually. Let C_{sn} denote the USV n 's computing capability. The local computation time of the USV n can be expressed as :

$$T_{sn} = \frac{(1 - \gamma_n) s_n B_n}{C_{sn}}, \forall n \in \mathcal{N}, \quad (6)$$

where s_n means the number of cycles needed by the USV n for computing each bit. Let C_{an} denote the UAV's computation resource allocated to compute the task from the USV n , which is subject to the UAV's computing capacity $C_{\max}$, i.e.,

$$\sum_{n=1}^N C_{an} \leq C_{\max}. \quad (7)$$

Correspondingly, the computation time needed to compute the task from the USV n satisfies

$$T_{ac} = \frac{\gamma_n s_a B_n}{C_{an}}, \forall n \in \mathcal{N}, \quad (8)$$

where s_a means the number of cycles of computing each bit in the UAV. Let E_{tn} and E_{cn} indicate the transmit energy

computation and computation energy consumption of the USV n , satisfying

$$E_{tn} = \sum_{m=1}^M p_{nm} T / M, \forall n \in \mathcal{N}, \quad (9)$$

and

$$E_{cn} = l_n (1 - \gamma_n) s_n C_{sn}^2 B_n, \forall n \in \mathcal{N}, \quad (10)$$

respectively. Here, l_n means the effective switched capacitance of USV n [32]. Let $\hat{E}_c$ denote the UAV's computation energy consumption, satisfying

$$\hat{E}_c = l_v \sum_{n=1}^N \gamma_n s_a B_n C_{an}^2, \quad (11)$$

where l_v means the UAV's effective switched capacitance. Let v_m denote the UAV's velocity in the m th time slot, i.e.,

$$v_m = \frac{\|U_m - U_{m-1}\| M}{T}, \forall m \in \mathcal{M}. \quad (12)$$

We use $\hat{E}_f$ to indicate the UAV's propulsion energy consumption, i.e.,

$$\begin{aligned} \hat{E}_f &= \sum_{m=1}^M (\rho_1 v_m^3 + \frac{\rho_2}{v_m}) \\ &= \sum_{m=1}^M (\rho_1 (\frac{\|U_m - U_{m-1}\| M}{T})^3 + \frac{T \rho_2}{\|U_m - U_{m-1}\| M}), \end{aligned} \quad (13)$$

where ρ_1 and ρ_2 are the parameters related to the aircraft's weight, wing area, air density, and etc [33]. We aim to minimize the total energy consumption of the considered UAV-assisted M-LoT system subject to the execution latency constraint and computation resource capacity. Mathematically, we formulate the joint optimization problem in the following form, denoted by OEMC.

(OECM:)

$$\min_{\gamma, \mathbf{p}, \mathbf{C}, \mathbf{U}} \hat{E}_c + \sum_{n=1}^N (E_{tn} + E_{cn}) + \hat{E}_f \quad (14a)$$

$$\text{s.t. constraints (5), (7),} \quad (14b)$$

$$0 \leq \gamma_n \leq 1, \forall n \in \mathcal{N}, \forall m \in \mathcal{M}, \quad (14c)$$

$$0 \leq \frac{\|U_m - U_{m-1}\| M}{T} \leq v_{\max}, \forall m \in \mathcal{M}, \quad (14d)$$

$$0 \leq p_{nm} \leq P_n^{\max}, \forall n \in \mathcal{N}, \quad (14e)$$

$$\max\{T_{sn}, T + T_{ac}\} \leq T_{\max}, \forall n \in \mathcal{N}, \quad (14f)$$

$$C_{an} \geq 0, \forall n \in \mathcal{N}, \quad (14g)$$

where γ , $\mathbf{p}$, $\mathbf{C}$, and $\mathbf{U}$ denote the vectors of γ_n^s , p_{nm}^s , C_{an}^s , and U_m^s , respectively. We explain the formulation of Problem (OECM) as follows. The constraint (14c) denote the scope of the ratio of computation-intensive task offloaded from the USV n to the UAV. Constraints (14d) and (14e) mean the UAV's flight velocity and the transmit power of the USV n are subject to the maximum allowable value $v_{\max}$ and $P_n^{\max}$, respectively. The constraint (14f) means that to perform the efficient task computation, each USV must meet the maximum execution latency requirement $T_{\max}$. The constraint (14g) limits the

scope of the UAV's computing capacity. Note that due to the UAV trajectory optimization, Lemma 1 shows the NP-hardness of the optimization problem (OECM).

Theorem 1: The optimization problem (14) is NP-hard.

Proof: The optimization problem jointly optimizes the UAV trajectory and multi-dimensional resource allocation. Given the UAV trajectory, the optimization of resource allocation is to find the root of polynomial equations, and thus it is convex. However, the UAV trajectory optimization is equivalent to the Traveling Saleman Problem, which is NP-hard. Thus, it follows that the optimization problem in (14) is NP-hard. ■

Considering the NP-hardness, Problem (OECM) is challenging to solve in general. To address this difficulty, we explore the layered feature of Problem (OECM) and adopt a decomposition approach for solving it in the following section.

IV. PROBLEM TRANSFORMATION AND DECOMPOSITION

Due to its NP-hardness, Problem (OECM) is challenging to solve due to the existence of the UAV trajectory optimization. In this section, we first transform Problem (OECM) into an equivalent optimization problem through reparameterization. Then, we propose a double-layered decomposition approach for solving Problem (OECM-E).

A. Equivalent Transformations

To make Problem (OECM) tractable, we first transform it into an equivalent optimization problem by introducing a new variable x_{nm} . Specifically, the variable x_{nm} means the SINR received from the USV n , satisfying

$$x_{nm} = \frac{h_{nm}p_{nm}}{\sum_{j=n+1}^N h_{jm}p_{jm} + n_0}. \quad (15)$$

By (15), p_{nm} can be rewritten as

$$p_{nm} = \frac{n_0 x_{nm} \prod_{j=n+1}^N (1 + x_{jm})}{h_{nm}}. \quad (16)$$

It implies that the constraint (14d) can be rewritten as

$$0 \leq \frac{n_0 x_{nm} \prod_{j=n+1}^N (1 + x_{jm})}{h_{nm}} \leq P_n^{\max}, \quad (17)$$

and the constraint (14e) can be rewritten as

$$\max_n \left\{ \frac{(1 - \gamma_n)S_n B_n}{C_{sn}}, T + \frac{\gamma_n S_a B_n}{C_{an}} \right\} \leq T_{\max}. \quad (18)$$

Since the transmit power of USV n increases with the increase of x_{jm}^{ts} over j , the constraint (5) can be rewritten as

$$\sum_{m=1}^M W_m \log(1 + x_{nm}) \geq \frac{M}{T} \gamma_n B_n, \forall n \in \mathcal{N}. \quad (19)$$

With the above transformations, we can transform the original Problem (OECM) into the following equivalent form:

(OECM-E):

$$\begin{aligned} \min_{\gamma, \mathbf{x}, \mathbf{C}, \mathbf{U}} \quad & \sum_{n=1}^N \left(\frac{T}{M} \sum_{m=1}^M \frac{n_0 x_{nm} \prod_{j=n+1}^N (1 + x_{jm})}{h_{nm}} + E_{cn} \right) \\ & + \hat{E}_c + \hat{E}_f, \\ \text{s.t.} \quad & \text{constraints (17) - (19), (14d),} \\ & \sum_{n=1}^N C_{an} \leq C_{\max}, \\ & 0 \leq \gamma_n \leq 1, x_{nm} \geq 0, C_{an} \geq 0, \forall n \in \mathcal{N}, \forall m \in \mathcal{M}, \end{aligned} \quad (20)$$

where $\mathbf{x}$ means the vector of x_{nm}^{ts} .

B. Layered Approach for Solving Problem (OECM-E)

We propose a double-layered decomposition approach to solve Problem (OECM-E), which includes i) a subproblem for optimizing $(\mathbf{x}, \mathbf{C})$ under given UAV trajectory $\mathbf{U}$ and offloading ratio vector γ , and ii) a top-problem for further optimizing $\mathbf{U}$ and γ based on the optimal value of the subproblem.

1) **Sub-problem for optimizing $(\mathbf{x}, \mathbf{C})$:** Supposing that the UAV trajectory $\mathbf{U}$ and offloading ratio vector γ are given in advance, we firstly consider the following subproblem which optimizes $(\mathbf{x}, \mathbf{C})$.

(OECM-E-Sub):

$$\begin{aligned} E(\gamma, \mathbf{U}) = \min_{\mathbf{x}, \mathbf{C}} \quad & \sum_{n=1}^N \left(\frac{T}{M} \sum_{m=1}^M \frac{n_0 x_{nm} \prod_{j=n+1}^N (1 + x_{jm})}{h_{nm}} \right) + \hat{E}_c \\ \text{s.t.} \quad & \text{constraints (17), (19),} \\ & T + \frac{\gamma_n S_a B_n}{C_{an}} \leq T_{\max}, \\ & \sum_{n=1}^N C_{an} \leq C_{\max}, \\ & x_{nm} \geq 0, C_{an} \geq 0, \forall n \in \mathcal{N}, \forall m \in \mathcal{M}. \end{aligned} \quad (21)$$

We emphasize that since the trajectory $\mathbf{U}$ and offloading ratio vector γ are fixed in Problem (OECM-E), the propulsion energy consumption $\hat{E}_f$ and the USV n 's computation energy consumption E_{cn} are fixed and the objective function of Problem (OECM-E-Sub) stems from the previous two terms of (14a) before. The following Theorem 1 shows the optimization problem (OECM-E-Sub) can be transformed into a convex optimization problem.

Theorem 2: The optimization problem (OECM-E-Sub) can be transformed into a convex optimisation problem through the one-to-one logarithmic domain and range transformation.

Proof: According to the geometric programming, we use logarithmic function to transform the objective function and constraint conditions to transform the optimization problem (21) into a convex optimization problem. We use $\tilde{x}_{nm}$ and $\tilde{C}_{an}$ to denote the logarithmic transformation of x_{nm} and C_{an} ,

respectively, i.e., $\tilde{x}_{nm} = \log x_{nm}$ and $\tilde{C}_{an} = \log C_{an}$. Therefore, given the UAV trajectory $\mathbf{U}$ and offloading ratio vector $\boldsymbol{\gamma}$, Problem (OECM-E-Sub) can be equivalently transformed into the following Problem (OECM-E-Sub-Log).

(OECM-E-Sub-Log):

$$E(\boldsymbol{\gamma}, \mathbf{U}) = \min_{\tilde{\mathbf{x}}, \tilde{\mathbf{C}}} \log \sum_{n=1}^N \left(\frac{T}{M} \sum_{m=1}^M \frac{n_0 e^{\tilde{x}_{nm}} \prod_{j=n+1}^N (1 + e^{\tilde{x}_{jm}})}{h_{nm}} + l_v \gamma_n B_n s_a e^{2\tilde{C}_{an}} \right), \quad (22a)$$

$$\text{s.t. } \log \left(\sum_{n=1}^N e^{\tilde{C}_{an}} \right) \leq \log C_{\max}, \quad (22b)$$

$$\log \left(\frac{\gamma_n S_a B_n}{T_{\max} - T} \right) \leq \tilde{C}_{an}, \quad (22c)$$

$$\tilde{x}_{nm} + \sum_{j=n+1}^N \log(1 + e^{\tilde{x}_{jm}}) \leq \log \frac{h_{nm} P_n^{\max}}{n_0}, \quad (22d)$$

$$\log \left(\frac{M}{T} \gamma_n B_n \right) \leq \log \left(\sum_{m=1}^M W_m \log(1 + e^{\tilde{x}_{nm}}) \right). \quad (22e)$$

By the property of the geometric programming, the transformed objective function (22a) and constraints (22b), (22c) and (22d) are convex. Therefore, Theorem 1 follows. Further, the function $\log \left(\sum_{m=1}^M W_m \log(1 + e^{\tilde{x}_{nm}}) \right)$ is concave, and thus the constraint (22e) is convex. Consequently, the Problem (OECM-E-Sub-Log) is convex. ■

2) **Top-problem for optimizing $(\boldsymbol{\gamma}, \mathbf{U})$:** With $E(\boldsymbol{\gamma}, \mathbf{U})$ output from the subproblem before, we further optimize the UAV trajectory $\mathbf{U}$ and offloading ratio vector $\boldsymbol{\gamma}$, which corresponds to the following Top-problem.

(OECM-E-Top):

$$\begin{aligned} \min_{\boldsymbol{\gamma}, \mathbf{U}} \quad & E(\boldsymbol{\gamma}, \mathbf{U}) + \sum_{n=1}^N E_{cn} + \hat{E}_f \\ \text{s.t.} \quad & 0 \leq \gamma_n \leq 1, \\ & \frac{(1 - \gamma_n) S_n B_n}{C_{sn}} \leq T_{\max}, \\ & 0 \leq \frac{\| \mathbf{U}_m - \mathbf{U}_{m-1} \| M}{T} \leq v_{\max}, \forall m \in \mathcal{M}. \end{aligned} \quad (23)$$

The advantage of the above layered approach is as follows. By exploiting the convex nature of Problem (OECM-E-Sub), we can derive its optimal solution $(\mathbf{x}, \mathbf{C})$ with the convex optimization arguments. Considering the NP-hardness of UAV trajectory and offloading ratio $(\boldsymbol{\gamma}, \mathbf{U})$ optimization, we propose a DRL-based algorithm for solving Problem (OECM-E-Top). In summary, Fig.2 shows the process of the equivalent transformation and double-layered decomposition about Problem (OECM).

Fig. 2. Problem Transformation and Decomposition. (a) Problem (OECM) is converted into an equivalent optimization problem (OECM-E) through reparameterization. (b) The double-layered decomposition approach is proposed to solve the convex sub-problem for optimizing $(\tilde{\mathbf{x}}, \tilde{\mathbf{C}})$, and non-convex top-problem for optimizing $(\boldsymbol{\gamma}, \mathbf{U})$.

V. ALGORITHM DESIGN

In this section, we aim to find the optimal transmit power, task offloading ratio, computation resource allocation, and UAV trajectory by optimizing multiple mutual-coupling variables in Problem (OECM-E) sequentially in the double-layered decomposition framework. We first develop an efficient optimal algorithm to solve Problem (OECM-E-Sub) based on the primal-dual arguments, and further find the optimal UAV trajectory and offloading ratio vector of Problem (OECM-E-Top) in the DRL framework.

A. Multi-dimensional Resource Optimization (Sub-Algorithm)

Due to the convex nature (Theorem 1), the resource allocation variables $(\tilde{\mathbf{x}}, \tilde{\mathbf{C}})$ can be optimized by the KKT method. Specifically, we first have the Lagrangian of the problem (OECM-E-Sub-Log) as (24).

$$\begin{aligned} \mathcal{L} = \mathcal{L}(\tilde{\mathbf{x}}, \tilde{\mathbf{C}}, \boldsymbol{\varphi}, \xi, \boldsymbol{\varrho}, \boldsymbol{\psi}) \\ = \log \sum_{n=1}^N \left(\frac{T}{M} \sum_{m=1}^M \frac{n_0 e^{\tilde{x}_{nm}} \prod_{j=n+1}^N (1 + e^{\tilde{x}_{jm}})}{h_{nm}} + l_v \gamma_n B_n s_a e^{2\tilde{C}_{an}} \right) \\ + \sum_{n=1}^N \varphi_{nm} \left(\tilde{x}_{nm} + \sum_{j=n+1}^N \log(1 + e^{\tilde{x}_{jm}}) - \log \frac{h_{nm} P_n^{\max}}{n_0} \right) \\ + \xi \left(\log \left(\sum_{n=1}^N e^{\tilde{C}_{an}} \right) - \log C_{\max} \right) \\ + \sum_{n=1}^N \varrho_{nm} \left(\log \left(\frac{M}{T} \gamma_n B_n \right) - \log \left(\sum_{m=1}^M W_m \log(1 + e^{\tilde{x}_{nm}}) \right) \right) \\ + \sum_{n=1}^N \psi_{nm} \left(\log \left(\frac{\gamma_n S_a B_n}{T_{\max} - T} \right) - \tilde{C}_{an} \right) \end{aligned} \quad (24)$$

where φ , ξ , ϱ and ψ represent the vector of φ_{nm} , ξ , ϱ_{nm} and ψ_{nm} . Also, $\varphi_{nm} \geq 0$, $\xi \geq 0$, $\varrho_{nm} \geq 0$, $\psi_{nm} \geq 0$ are the Lagrange multipliers corresponding to the constraints (22b)-(22e), respectively. Then, we minimize the Lagrangian function $\mathcal{L}$ (24) and maximize the Lagrangian dual function $g(\lambda, \varphi, \xi, \varrho, \psi)$,

$$g(\varphi, \xi, \varrho, \psi) = \min_{\tilde{\mathbf{x}}, \tilde{\mathbf{C}}} \mathcal{L}(\tilde{\mathbf{x}}, \tilde{\mathbf{C}}, \varphi, \xi, \varrho, \psi) \quad (25)$$

and

$$\begin{aligned} & \max_{\varphi, \xi, \varrho, \psi} g(\varphi, \xi, \varrho, \psi) \\ & \text{s.t. } \varphi_{nm} \geq 0, \xi \geq 0, \varrho_{nm} \geq 0, \psi_{nm} \geq 0. \end{aligned} \quad (26)$$

Considering the convex nature, we propose the optimal allocation algorithm to solve (25) and (26) on the basis of gradient descent method and subgradient method. In particular, the proposed algorithm has three key ingredients at each iteration j as follows.

- Determine a descent direction of the optimisation variables in (25);
- Update the optimisation variables in (25) via the linear search;
- Update the multiplier variables in (26) by the subgradient method.

In the following, we would introduce the details of the proposed resource allocation algorithm.

First, the objective function (22a) in the optimization problem (OECM-E-Sub-Log) is expressed as $\log \Theta(\tilde{\mathbf{x}}, \tilde{\mathbf{C}})$. We have the first-order derivatives of the Lagrangian function $\mathcal{L}$ over $\tilde{\mathbf{x}}$ and $\tilde{\mathbf{C}}$ as follows (27-28):

$$\begin{aligned} \frac{\partial \mathcal{L}}{\partial \tilde{x}_{nm}} &= \frac{n_0 T}{M \Theta(\tilde{\mathbf{x}}, \tilde{\mathbf{C}})} \sum_{m=1}^M \left(\frac{e^{\tilde{x}_{nm}} \prod_{j=n+1}^N (1 + e^{\tilde{x}_{jm}})}{h_{nm}} \right. \\ & \quad \left. + \sum_{i=1}^{n-1} \frac{e^{\tilde{x}_{im} + \tilde{x}_{nm}} \prod_{j=i+1, j \neq n}^N (1 + e^{\tilde{x}_{jm}})}{h_{nm}} \right) \\ & \quad + \frac{\varrho_{nm} e^{\tilde{x}_{nm}}}{W_m \log(1 + e^{\tilde{x}_{nm}})(1 + e^{\tilde{x}_{nm}})} \\ & \quad + \varphi_{nm} + \frac{e^{\tilde{x}_{nm}}}{(1 + e^{\tilde{x}_{nm}})} \sum_{i=1}^{n-1} \varphi_{im} \end{aligned} \quad (27)$$

$$\frac{\partial \mathcal{L}}{\partial \tilde{C}_{an}} = \frac{2l_v \gamma_n s_a B_n e^{2\tilde{C}_{an}}}{\Theta(\tilde{\mathbf{x}}, \tilde{\mathbf{C}})} + \xi \frac{e^{\tilde{C}_{an}}}{\sum_{i=1}^N e^{\tilde{C}_{ai}}} - \psi_{nm} \quad (28)$$

Second, let $\tilde{\mathbf{x}}^{(j-1)}$ and $\tilde{\mathbf{C}}^{(j-1)}$ represent the updated variables of (25) at the $(j-1)$ th iteration, and $\varphi^{(j-1)}$, $\xi^{(j-1)}$, $\varrho^{(j-1)}$, and $\psi^{(j-1)}$ represent the updated Lagrangian multiplier number of (25) at the $(j-1)$ th iteration. Therefore, the decreasing directions of the optimized variables in (25) can be expressed as:

$$\begin{aligned} \Delta_{\tilde{x}_{nm}}^{(j)} &= -\frac{\partial \mathcal{L}}{\partial \tilde{x}_{nm}} \Big|_{\tilde{\mathbf{x}}^{(j-1)}, \tilde{\mathbf{C}}^{(j-1)}, \varphi^{(j-1)}, \varrho^{(j-1)}} \\ \Delta_{\tilde{C}_{an}}^{(j)} &= -\frac{\partial \mathcal{L}}{\partial \tilde{C}_{an}} \Big|_{\tilde{\mathbf{x}}^{(j-1)}, \tilde{\mathbf{C}}^{(j-1)}, \xi^{(j-1)}, \psi^{(j-1)}} \end{aligned} \quad (29)$$

Third, we update the optimization variables in (25) as $\tilde{\mathbf{x}}^{(j)}$ and $\tilde{\mathbf{C}}^{(j)}$ via linear search at the j th iteration, i.e., (30).

$$\begin{aligned} \tilde{x}_{nm}^{(j)} &= \tilde{x}_{nm}^{(j-1)} + \kappa^{(j)} \Delta_{\tilde{x}_{nm}}^{(j)}, \forall n \in N, \\ \tilde{C}_{an}^{(j)} &= \tilde{C}_{an}^{(j-1)} + \kappa^{(j)} \Delta_{\tilde{C}_{an}}^{(j)}, \forall n \in N. \end{aligned} \quad (30)$$

Here $\kappa^{(j)}$ means the step size at the j th iteration, which is allowed to be different when updating $\tilde{\mathbf{x}}^{(j)}$ and $\tilde{\mathbf{C}}^{(j)}$.

Fourth, the multiplier variables are updated to $\varphi^{(j)}$, $\xi^{(j)}$, $\varrho^{(j)}$, and $\psi^{(j)}$ through the sub-gradient method at the j th iteration. Correspondingly, the updated multiplier variables are expressed as follows:

$$\begin{aligned} \varphi_{nm}^{(j)} &= \left[\varphi_{nm}^{(j-1)} + \nu^{(j)} \left(\tilde{x}_{nm}^{(j)} + \log(1 + e^{\tilde{x}_{nm}^{(j)}}) \right. \right. \\ & \quad \left. \left. - \log \frac{P_n^{max} h_{nm}}{n_0} \right) \right]_0 \\ \xi^{(j)} &= \left[\xi^{(j-1)} + \nu^{(j)} \left(\log \left(\sum_{n=1}^N e^{\tilde{C}_{an}^{(j)}} \right) - \log C_{max} \right) \right]_0 \\ \varrho_{nm}^{(j)} &= \left[\varrho_{nm}^{(j-1)} + \nu^{(j)} \left(\log \left(\frac{M}{T} \gamma_n B_n \right) \right. \right. \\ & \quad \left. \left. - \log \left(\sum_{m=1}^M W_m \log(1 + e^{\tilde{x}_{nm}^{(j)}}) \right) \right) \right]_0 \\ \psi_{nm}^{(j)} &= \left[\psi_{nm}^{(j-1)} + \nu^{(j)} \left(\log \left(\frac{\gamma_n S_a B_n}{T^{max} - T} \right) - \tilde{C}_{an}^{(j)} \right) \right]_0. \end{aligned} \quad (31)$$

Here, $\nu^{(j)}$ means the step size at the j th iteration, which is allowed to be different when updating $\varphi^{(j)}$, $\xi^{(j)}$, $\varrho^{(j)}$, and $\psi^{(j)}$.

Finally, we can repeat the procedure above until the stopping criterion, defined as $\|[\Delta_{\tilde{\mathbf{x}}}^{(j)}, \Delta_{\tilde{\mathbf{C}}}^{(j)}]\|_2 \leq \epsilon$, is satisfied. Here, ϵ is a small enough positive constant, as well as $\Delta_{\tilde{\mathbf{x}}}^{(j)}$, $\Delta_{\tilde{\mathbf{C}}}^{(j)}$ are the vectors of $\Delta_{\tilde{x}_{nm}}^{(j)}$ and $\Delta_{\tilde{C}_{an}}^{(j)}$, respectively.

According to the steps above, we would propose the following Sub-Algorithm (i.e., Algorithm 1) to solve the underlying optimization problem (OECM-E-Sub-log), and obtain the optimal value of $\tilde{\mathbf{x}}^{(*)}$ and $\tilde{\mathbf{C}}^{(*)}$. Accordingly, we can obtain the optimal power allocation $\mathbf{p}^*$ by replacing $e^{\tilde{\mathbf{x}}^{(*)}}$ with $\mathbf{x}$ in (16), and the optimal computation resource allocation $\mathbf{C}^*$ satisfying $\mathbf{C}^* = e^{\tilde{\mathbf{C}}^{(*)}}$.

Algorithm 1 Algorithm for solving the underlying optimization problem (OECM-E-Sub-log) (Sub-Algorithm)

- 1: **Initialization:** Randomly choose feasible primal variables (i.e., $\tilde{\mathbf{x}}^{(0)}$ and $\tilde{\mathbf{C}}^{(0)}$) and feasible Lagrange multiplier vectors (i.e., $\varphi^{(0)}$, $\xi^{(0)}$, $\varrho^{(0)}$, $\psi^{(0)}$). Set $j = 1$.
- 2: **repeat**
- 3: Update the descent direction of the primal variables by (29).
- 4: Update $\tilde{\mathbf{x}}^{(j)}$ and $\tilde{\mathbf{C}}^{(j)}$ by (30).
- 5: Update $\varphi^{(j)}$, $\xi^{(j)}$, $\varrho^{(j)}$, and $\psi^{(j)}$ by (31).
- 6: Set $j = j + 1$.
- 7: **until** Meet the stopping criterion, i.e., $\|[\Delta_{\tilde{\mathbf{x}}}^{(j)}, \Delta_{\tilde{\mathbf{C}}}^{(j)}]\|_2 \leq \epsilon$.
- 8: Obtain the optimal solution $[\tilde{\mathbf{x}}^*, \tilde{\mathbf{C}}^*]$ to problem (OECM-E-Sub-log), i.e., $[\tilde{\mathbf{x}}^{(j-1)}, \tilde{\mathbf{C}}^{(j-1)}]$.

B. UAV Trajectory and Offloading Ratio ratio Optimization (Top-Algorithm)

We solve Problem (OECM-E-Top) based on the solution of Problem (OECM-E-Sub). Specifically, after the resource allocation $(\tilde{x}, \tilde{C})$ has been optimized by the proposed Sub-Algorithm, we would propose a DRL-based algorithm (i.e., Top-Algorithm) to solve the problem (OECM-E-Top), whose key idea mainly comes from the Deep Deterministic Policy Gradient (DDPG) algorithm.

The proposed Top-Algorithm can be expressed as a triple $\langle \mathcal{S}, \mathcal{A}, r \rangle$. In particular, the symbol $\mathcal{S}$ means the state space, in which the state at the k th-round trajectory update can be expressed as $s_k = \{\tilde{x}_k^*, \tilde{C}_k^*, V_k\}$. Here, $(\tilde{x}_k^*, \tilde{C}_k^*)$ means the optimal solution to Problem (OECM-E-Sub-Log) obtained by the Sub-Algorithm when the $(k-1)$ th-round trajectory and offloading ratio (i.e., U_{k-1} and γ_{k-1}) is given, and V_k means the locations of all USVs at the k th-round trajectory update. The symbol $\mathcal{A}$ means the action space, in which the action in k th-round trajectory update can be expressed as $a_k = \{U_k, \gamma_k\}$ in k th-round trajectory update. The symbol r means the reward function, which can be calculated as the energy consumption, i.e., $r_k(a_k, s_k) = E(\gamma_k, U_k) + \sum_{n=1}^N E_{cn} + \hat{E}_f$ in k th-round trajectory update, due to the total energy consumption minimization.

As shown in Fig. 3, the framework of the proposed algorithm consists of the environment entity, online network, target network, and replay buffer R . The environment entity is to sense the environment state s_k in k th-round trajectory update. The online network and target network are composed of the actor network and critic network, respectively. The online actor network is to obtain the optimal policy μ , and the action a_k accordingly. The online critic network is to obtain the action value function, which is used to update the weights of online network. The goal of target network is to update the weights of online network. Notably, the online network and target network have the same structures. The replay buffer is used to store the transition (s_k, a_k, r_k, s_{k+1}) . Therefore, there are three key ingredients at each trajectory update as follows for the proposed algorithm.

1) **Action selection:** Find the optimal action a_k by the online actor network with the weights θ^μ in k th-round trajectory update.

2) **Action-value update:** Update the evaluated action-value function $Q(s_k, a_k | \theta^Q)$ by the online critic network with the weights θ^Q in k th-round trajectory update.

3) **Gradient update and weight update:** Update the weight gradients of actor and critic in order to obtain a better action selection policy.

In the following, we introduce every key ingredient of the proposed optimization algorithm over the actor-critic architecture.

1) **Action selection:** The deterministic policy μ generated by the online actor network determines the current action a_k :

$$a_k = \mu(s_k | \theta^\mu), \quad (32)$$

Fig. 3. The framework of proposed algorithm

where θ^μ represents the weights of the online actor network. That is, we input the state s_k in the online actor network, and we have the a_k as the output.

2) **Action-value update:** After the action a_k is determined, the environmental state is moved to s_{k+1} , and the real immediate reward value obtained is r_k . Then, we can obtain the action value function as follows:

$$Q(s_k, a_k | \theta^Q) = \mathbb{E}_{s_{k+1} \sim \rho^\mu} [r_k(s_k, a_k) + \varsigma Q(s_{k+1}, a_{k+1} | \theta^Q)], \quad (33)$$

by the online critic network with the weights θ^Q . Here, $\varsigma \in [0, 1)$ is the discounting factor in reinforcement learning that represents the uncertainty of future revenue, ρ^μ is the distribution of s_{k+1} under the policy μ , and $\mathbb{E}[\cdot]$ means the expectation.

3) **Gradient update and weight update:** The weights θ^μ of online actor network are updated by the gradients of action value function $Q(s, a | \theta^Q)$ and action policy $\mu(s | \theta^\mu)$ (i.e., $\nabla_a Q(s, a | \theta^Q)$ and $\nabla_{\theta^\mu} \mu(s | \theta^\mu)$) based on the gradient method. Specifically, the gradient of θ^μ can be expressed as

$$\nabla_{\theta^\mu} J = \frac{1}{G} \sum_{i=1}^N [\nabla_a Q(s, a | \theta^Q) |_{s_i, \mu(s_i)} \nabla_{\theta^\mu} \mu(s | \theta^\mu) |_{s_i}], \quad (34)$$

where s_i' and a_i' mean the states and actions stored in the replay buffer R , and G means the number of transitions selected from replay buffer. Correspondingly, the weights θ^μ is updated as

$$\theta^\mu = \theta^\mu + \alpha_\mu \nabla_{\theta^\mu} J, \quad (35)$$

where α_μ means the learning rate of online actor network. The online critic network θ^Q is updated based on the gradient method by minimizing the loss, which is expressed as

$$L(\theta^Q) = E_{s_i \sim \varrho^\mu} [(Q(s_i, a_i | \theta^Q) - y_m)^2], \quad (36)$$

where ϱ^μ represents the distribution of s_i under the policy μ . Besides, the target function y_m is defined as follows:

$$y_m = r_i(s_i, a_i) + \varsigma \hat{Q}(s_{i+1}, \hat{\mu}(s_{i+1} | \theta^\mu) | \theta^Q), \quad (37)$$

where r_i^s mean the rewards stored in the replay buffer (need discuss) after K trajectory updates. The symbols $\theta^{\hat{Q}}$ and $\theta^{\hat{\mu}}$ mean the weights of the target critic network and target actor network, respectively. Besides, $\hat{Q}(\cdot|\theta^{\hat{Q}})$ and $\hat{\mu}(\cdot|\theta^{\hat{\mu}})$ mean the action value function and action policy obtained by the target critic network and actor network, respectively. Correspondingly, the weights of online critic network are updated as

$$\theta^Q = \theta^Q + \alpha_Q \nabla_{\theta^Q} L(\theta^Q), \quad (38)$$

where α_Q means the learning rate of online critic network. The target network has the same structure as the online network, and thus the weights of target actor network and critic network are updated as

$$\theta^{\hat{\mu}} \leftarrow \delta \theta^{\hat{\mu}} + (1 - \delta) \theta^{\hat{\mu}}, \quad (39)$$

and

$$\theta^{\hat{Q}} \leftarrow \delta \theta^{\hat{Q}} + (1 - \delta) \theta^{\hat{Q}}, \quad (40)$$

respectively. Here, δ is a hyperparameter between 0 and 1 (usually close to 1).

Having introduced all key ingredients, we present the Top-Algorithm as shown in Algorithm 2. In the following, we discuss the computational complexity of Top-Algorithm. Due to the proposed algorithm is based on the reinforcement learning, and the actor network and critic network need to be respectively updated for once in each episode for the UAV-aided NOMA communication network. Thus, the computational complexity of the proposed algorithm is $O(K(F_a L_a + F_a) + K(F_c L_c + F_c))$, where K means the number of episodes and F_a , F_c , L_a , and L_c mean the number of units in each hidden layer of the actor network and critic network and the number of hidden layers of the actor network and critic network, respectively. Note that the optimal solution to Problem (OECM) can be obtained by performing the Sub-Algorithm and Top-Algorithm alternatively.

VI. NUMERICAL RESULTS

In this section, we validate the performance of the proposed algorithms through a set of simulations. Referring to the parameter settings in [34], we set the simulation parameters in Table I. In the following simulations, we consider a set of UAV-assisted M-IoT network topologies as shown in Fig. 4, where all USVs are randomly deployed in a plane with the center point being (1000m, 1000m, 0m) and the radius being 1000m, and the starting point of UAV is (1000m, 1000m, 100m).

A. Global Optimality and Time Complexity

Example 1: In this simulation example, we want to verify the global optimality and time complexity of the proposed Sub-Algorithm and the proposed Top-Algorithm, when the number of USVs is set to be 4 in Fig. 4. Specifically, all four USVs float in four 50m-radius planes with the center points being (500m, 1000m, 0m), (1000m, 1500m, 0m), (1500m, 1000m, 0m) and (1000m, 500m, 0m), respectively.

Fig. 5(a) reveals that the energy consumption approaches a constant with the increase of the number of iterations. It

Algorithm 2 Algorithm for solving the top-level problem (OECM-Top) (Top-Algorithm)

- 1: **Input:** Training epoch length T , training sample length K ; learning rate for actor network α_μ , learning rate for critic network α_Q ; discount factor ζ , soft update factor δ ; replay buffer R , mini-batch size N .
- 2: **Output:** the UAV trajectory U and the ratio of tasks offloaded γ .
- 3: Initialize the replay buffer R .
- 4: Randomly initialize online critic network and actor with the weights θ^Q and θ^μ , respectively.
- 5: Initialize the target critic network and actor network with the weights $\theta^{\hat{Q}} \leftarrow \theta^Q$, $\theta^{\hat{\mu}} \leftarrow \theta^\mu$, respectively.
- 6: Receive initial observation state s_1 .
- 7: **for** each episode $k=1, \dots, K$ **do**
- 8: Obtain the state $s_k = \{\tilde{x}_k^*, \tilde{C}_k^*, V_k\}$.
- 9: Select the action $a_k = \mu(s_k|\theta^\mu)$ by the online actor network.
- 10: Update $\tilde{x}_{k+1}^*$ and $\tilde{C}_{k+1}^*$ by the Sub-Algorithm based on the action a_k , obtain the reward r_k , and observe the new state s_{k+1} .
- 11: **if** the replay buffer is not full **then**
- 12: Store the transition (s_k, a_k, r_k, s_{k+1}) in the buffer.
- 13: **else**
- 14: Randomly replace a transition in the buffer with (s_k, a_k, r_k, s_{k+1}) .
- 15: **end if**
- 16: Sample a random minibatch of transitions (s_i, a_i, r_i, s_{i+1}) from R .
- 17: Update the weights θ^μ and θ^Q by (35) and (38), respectively.
- 18: Update the weights $\theta^{\hat{\mu}}$ and $\theta^{\hat{Q}}$ by (39) and (40), respectively.
- 19: **end for**

Fig. 4. The network topology used for Example 1.

implies that the Sub-Algorithm can converge until the stopping criterion of $\|[\Delta_{\tilde{x}}^{(j)}, \Delta_{\tilde{C}}^{(j)}]\|_2 \leq \epsilon$ is satisfied. Also, we can find from Fig. 5(a) that the number of iterations needed for the convergence increases with the decrease of ϵ , while the minimum energy consumption decreases. It implies that we can strike a tradeoff between the time complexity and optimality by tuning the parameter ϵ .

Fig. 5(b) reveals the convergence performance of Top-Algorithm under the different setting of minibatch. We can

TABLE I
THE PARAMETERS OF THE SYSTEM.

Simulation parameters	Value chosen
Radius of Sea area	2km
Moving Radius of USVs	50m
Channel Bandwidth Capacity, W	1MHz
Noise power spectral density, N_0	-110dB
UAV Altitude, H	100
The number of UAV	1
The number of USVs	4~16
Upper limit of transmitted power(USV)	1W
Aerial computing capacity, $C_{\max}$	500MHz
Local computing capacity, C_{sn}	10~20MHz
The reference channel gain, h_0	-30dB
The computation-intensive tasks, B_n	10Mbit
Coefficient, δ_1 and δ_2	10^{-12}
The time duration, T	50s
The maximum time latency, $T_{\max}$	60s
The length of time slot, M	50
Error tolerance, ϵ	0.01
Step size at the sth iteration, κ	10^{-3}
Step size at the sth iteration, v	10^{-4}
Discount coefficient, ς	0.95
Soft replacement Value, δ	0.01
Learning rate of online actor network, α_μ	0.001
Earning rate of online critic network, α_Q	0.002
Iterative rounds, K	3000

find that the convergence rate of Top-Algorithm, increases with the decrease of minibatch. For example, when we set the minibatch to be 32, the Top-Algorithm converges almost in the 500th episode. It implies that we should carefully set the minibatch for ensuring the convergence rate of Top-Algorithm.

B. Trajectory Optimization

Example 2: In this simulation example, we want to show the optimal trajectory obtained by alternatively running the Sub-Algorithm and Top-Algorithm for the network topology considered in Example 1.

We plot the UAV trajectories obtained at episode 1, episode 150, episode 300, and episode 600 in Fig. 6. As shown in Fig. 6(a), the UAV flies randomly to seek a trajectory for less energy consumption in the first episode. Thus, the UAV might fly over a certain USV, i.e., USV 3. When running the algorithms continuously, the UAV would fly over more USVs, e.g., USV 3 and USV 4 in episode 300, and all USVs in episode 600, which can be seen in Fig. 6(c) and Fig. 6(d). These observations imply that the optimal UAV trajectory is nearly a closed loop covering all USVs.

C. Performance Comparison

To the best of our knowledge, there is no algorithm proposed for the same target in the literature. For the comparison with our proposed joint optimization algorithm, we introduce three baseline schemes: communication and computation resource allocation with fixed UAV trajectory or offloading ratio, and frequency division multiple access (FDMA). The first baseline scheme means that the energy consumption is minimized when the UAV is hovering at the starting point as a base station to collect and process the computation-intensive tasks offloaded from the USVs. The second baseline scheme means the energy consumption is minimized when the offloading of each USV

(a) The time complexity and optimality of the proposed Sub-Algorithm.

(b) The convergence rate of the proposed Top-Algorithm.

Fig. 5. The convergence performance of the proposed algorithms.

is given. The last baseline scheme means we compare the NOMA performance with other OMA transmission scheme, e.g. FDMA. In the simulation, we evaluate the obtained total energy consumption as the performance criteria of the three schemes. In the following performance comparison, we first compare our proposed algorithm with the former two baseline schemes in which the UAV trajectory or offloading ratio is given. Then, we compare our proposed algorithm with the FDMA scheme.

Example 3 (Performance comparison with the former two schemes): We consider a set of UAV-assisted M-IoT network topologies, where we randomly deploy 4~16 USVs in a plane with the center point being (1000m, 1000m, 0m) and the radius being 1000m, and each USV floats in a 50m-radius plane. We vary the computing capacity of each USV from 10 MHz to 20 MHz. In the first baseline scheme, the location of the UAV is fixed at the starting point (1000m, 1000m, 100m). In the second baseline scheme, we set the offloading ratio of each USV to be 0.5. Other simulation parameters are the same as those in Example 1.

Fig. 7 shows that as the number of USVs increases, the obtained total energy consumption increases for the three schemes. It can also be seen that the proposed algorithm

(a) the 1st episode

(b) the 150th episode

(c) the 300th episode

(d) the 600th episode

Fig. 6. The UAV trajectory varies with time for the network topology in Fig. 4 when the number of batch is 32.

Fig. 7. The minimum total energy consumption obtained by the former two schemes and the proposed algorithm at different densities of UAVs, in which the computing capacity is set to be 10 MHz for each USV.

Fig. 8. The minimum total energy consumption obtained by the former two schemes and the proposed algorithm under different computing capacity for the 4-USV network topology.

always outperforms the other two baseline schemes. Compared with the other two baseline schemes, our proposed algorithm can reduce the total energy consumption by 14.8% and 39.6% on average, respectively. Fig. 8 shows that with the increase of local computing capacity, the total energy consumption obtained by the three schemes increases. Also, our proposed algorithm can achieve the minimum total energy consumption. Compared with the other two baseline schemes, our proposed algorithm can reduce the total energy consumption by 37.4% and 51.3% on average, respectively. These observations imply that for minimizing the total energy consumption of NOMA based UAV-assisted M-LoV MEC system, it is of practical meaning to jointly optimize the jointly optimizing the USVs' offloaded workload, transmit power, computation resource

allocation as well as the UAV trajectory.

Example 4 (Performance comparison with the FDMA):

In this simulation example, we want to verify the efficiency of NOMA-enabled computation offloading, in comparison with the FDMA, when varying the number of USVs and USVs' computing capacity. When adopting FDMA, the dedicated bandwidth (1MHz) is equally allocated to all USVs, and all optimization variables are the same as these considered in our system model. Other simulation parameters are the same as those in Example 3.

From Fig. 9, we can see that the total energy consumption increases with the increase of the number of USVs for the NOMA and FDMA. Also, it is straightforward that more energy is consumed for computing more workload for both

Fig. 9. The minimum total energy consumption obtained in the NOMA and FDMA manners at different densities of UAVs.

Fig. 10. The minimum total energy consumption obtained by the two multiple access schemes when varying the computing capacity of each USV from 10 MHz to 20 MHz.

multiple access schemes. Noticeably, the NOMA always outperforms the FDMA in terms of total energy consumption. Compared with FDMA, the NOMA can reduce the total energy consumption by 37.5% on average. Fig. 10 shows the minimum total energy consumption obtained by the two multiple access schemes when varying the computing capacity of each USV from 10MHz to 20MHz. From Fig. 10, we find that the total energy consumption increases as the USVs' computing capacity and the number of USVs increase for the two multiple access schemes. Moreover, in comparison with FDMA, the total energy consumption in NOMA is reduced by 15.2% on average. These observations imply the NOMA would be an efficient multiple access scheme that can applied to the UAV-assisted M-IoV MEC system.

VII. CONCLUSION

In this paper, we have studied the joint optimization of power control, task offloading ratio, computation resource allocation and UAV trajectory that minimizes the total energy consumption for the UAV-assisted M-IoT network in the NOMA manner. By exploiting the hidden convexity of this optimization problem, we have proposed an efficient double-layered decomposition algorithm to obtain the optimal solution. Specifically, in our proposed algorithms, we first designed the primal-dual-based Lagrangian minimization algorithm to obtain the optimal solution to the multi-dimensional resource optimization sub-problem. Then, we designed the DRL-based algorithm to obtain the optimal UAV trajectory and USVs' task

offloading ratios. The simulation results have been provided to validate the effectiveness of the proposed algorithms. In the future work, we will study the secure computation offloading. Specifically, we will apply the cooperative jamming to improve the communication security when offloading the computation-intensive tasks.

REFERENCES

- [1] S. Verma, Y. Kawamoto, Z. M. Fadlullah et al., "A survey on network methodologies for real-time analytics of massive iot data and open research issues," *IEEE Commun. Surveys Tuts.*, vol. 19, no. 3, pp. 1457-1477, Third Quarter 2017.
- [2] M. M. Wang, J. Zhang, and X. You, "Machine-type communication for maritime internet of things: a design," *IEEE Commun. Surveys Tuts.*, vol. 22, no. 4, pp. 2550-2585, Fourth Quarter 2020.
- [3] T. Yang, H. Feng, S. Gao, et al, "Two-stage offloading optimization for energy-latency tradeoff with mobile edge computing in maritime internet of things," *IEEE Internet Things J.*, vol. 7, no. 7, pp. 5954-5963, Jul. 2020.
- [4] C. Wang, Y. He, F.R. Yu, Q. Chen, and L. Tang, "Integration of networking, caching, and computing in wireless systems: a survey, some research issues, and challenges," *IEEE Commun. Surveys Tuts.*, vol.20, no. 1, pp. 7-38, First Quarter 2018.
- [5] Y. Liu, S. Xie, and Y. Zhang, "Cooperative offloading and resource management for uav-enabled mobile edge computing in power iot system," *IEEE Trans. Veh. Tech.*, vol. 69, no. 10, pp. 12229-12239, Oct. 2020.
- [6] 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.
- [7] Y. Huang, Y. Liu and F. Chen, "NOMA-aided mobile edge computing via user cooperation," *IEEE Trans. Commun.*, vol. 68, no. 4, pp. 2221-2235, Apr. 2020.
- [8] Z. Ding, J. Xu, O. A. Dobre and H. V. Poor, "Joint power and time allocation for noma-mec offloading," *IEEE Trans. Veh. Tech.*, vol. 68, no. 6, pp. 6207-6211, Jun. 2019.
- [9] Z. Yang, C. Pan, K. Wang and M. Shikh-Bahaei, "Energy efficient resource allocation in uav-enabled mobile edge computing networks," *IEEE Trans. Wireless Commun.*, vol. 18, no. 9, pp. 4576-4589, Sept. 2019.
- [10] L. Zhang and N. Ansari, "Latency-aware iot service provisioning in uav-aided mobile-edge computing networks," *IEEE Internet Things J.*, vol. 7, no. 10, pp. 10573-10580, Oct. 2020.
- [11] 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.
- [12] 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.
- [13] L. P. Qian, A. Feng, Y. Huang, Y. Wu, B. Ji and Z. Shi, "Optimal sic ordering and computation resource allocation in mec-aware noma nb-iot networks," *IEEE Internet Things J.*, vol. 6, no. 2, pp. 2806-2816, Apr. 2019.
- [14] F. Fang, Y. Xu, Z. Ding, C. Shen, M. Peng and G. K. Karagiannis, "Optimal resource allocation for delay minimization in noma-mec networks," *IEEE Trans. Wireless Commun.*, vol. 68, no. 12, pp. 7867-7881, Dec. 2020.
- [15] B. Li, W. Wu, W. Zhao and H. Zhang, "Security enhancement with a hybrid cooperative noma scheme for mec system," *IEEE Trans. Veh. Technol.*, vol. 70, no. 3, pp. 2635-2648, Mar. 2021.
- [16] W. Wu, X. Wang, F. Zhou, K. Wong, C. Li and B. Wang, "Resource allocation for enhancing offloading security in noma-enabled mec networks," *IEEE Internet System J.*, vol. 15, no. 3, pp. 3789-3792, Sept. 2021.
- [17] F. Fang, K. Wang, Z. Ding and V. C. M. Leung, "Energy-efficient resource allocation for noma-mec networks with imperfect csi," *IEEE Trans. Wireless Commun.*, vol. 69, no. 5, pp. 3436-3449, May 2021.
- [18] V. D. Tuong, T. P. Truong, T. -V. Nguyen, W. Noh and S. Cho, "Partial computation offloading in noma-assisted mobile edge computing systems using deep reinforcement learning," *IEEE Internet Things J.*, vol. 8, no. 17, pp. 13196-13208, Sept. 2021.
- [19] L. Qian, Y. Wu, N. Yu, F. Jiang, H. Zhou and T. Q. S. Quek, "Learning driven noma assisted vehicular edge computing via underlay spectrum sharing," *IEEE Trans. Veh. Technol.*, vol. 70, no. 1, pp. 977-992, Jan. 2021.

- [20] C. Li, H. Wang and R. Song, "Intelligent offloading for noma-assisted mec via dual connectivity," *IEEE Internet Things J.*, vol. 8, no. 4, pp. 2802-2813, Feb. 2021.
- [21] 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.
- [22] J. Ji, K. Zhu, C. Yi and D. Niyato, "Energy consumption minimization in uav-assisted mobile-edge computing Systems: joint resource allocation and trajectory design," *IEEE Internet Things J.*, vol. 8, no. 10, pp. 8570-8584, May 2021.
- [23] Y. Zeng and R. Zhang, "Energy-efficient uav communication with trajectory optimization," *IEEE Trans. Wireless Commun.*, vol. 16, no. 6, pp. 3747-3760, Jun. 2017.
- [24] H. Mei, K. Yang, Q. Liu and K. Wang, "Joint trajectory-resource optimization in uav-enabled edge-cloud system with virtualized mobile clone," *IEEE Internet Things J.*, vol. 7, no. 7, pp. 5906-5921, Jul. 2020.
- [25] 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.
- [26] T. Zhang, Y. Xu, J. Loo, D. Yang and L. Xiao, "Joint computation and communication design for uav-assisted mobile edge computing in iot," *IEEE Trans. Wireless Commun. Industrial Infor.*, vol. 16, no. 8, pp. 5505-5516, Aug. 2020.
- [27] X. Zhang, J. Zhang, J. Xiong, L. Zhou and J. Wei, "Energy-efficient multi-uav-enabled multiaccess edge computing incorporating noma," *IEEE Internet Things J.*, vol. 7, no. 6, pp. 5613-5627, Jun. 2020.
- [28] Y. Li, S. Zhang, F. Ye, T. Jiang and Y. Li, "A uav path planning method based on deep reinforcement learning," 2020 IEEE USNC-CNC-URSI North American Radio Science Meeting (Joint with AP-S Symposium), pp. 93-94, 2020.
- [29] S. Yin and F. R. Yu, "Resource allocation and trajectory design in uav-aided cellular networks based on multi-agent reinforcement learning," *IEEE Internet Things J.*, Jul. 2021.
- [30] Q. Wang, H. Dai, Q. Wang, M. K. Shukla, W. Zhang and C. G. Soares, "On connectivity of uav-assisted data acquisition for underwater internet of things," *IEEE Internet Things J.*, vol. 7, no. 6, pp. 5371-5385, Jun. 2020.
- [31] M. Li, N. Cheng, J. Gao, Y. Wang, L. Zhao and X. Shen, "Energy-efficient uav-assisted mobile edge computing: resource allocation and trajectory optimization," *IEEE Trans. Veh. Technol.*, vol. 69, no. 3, pp. 3424-3438, Mar. 2020.
- [32] L. Kuang, X. Chen, C. Jiang, H. Zhang and S. Wu, "Radio resource management in future terrestrial-satellite communication networks," *IEEE Trans. Commun.*, vol. 24, no. 5, pp. 81-87, Oct. 2017.
- [33] Y. Zeng, J. Xu and R. Zhang, "Energy minimization for wireless communication with rotary-wing uav," *IEEE Trans. Wireless Commun.*, vol. 18, no. 4, pp. 2329-2345, Apr. 2019.
- [34] Y. Wang et al., "Trajectory design for uav-based internet-of-things data collection: a deep reinforcement learning approach," *IEEE Internet Things J.*, Jul. 2021.