Age of Information Minimization for UAV-Assisted Internet of Things Networks: A Safe Actor-Critic With Policy Distillation Approach

Fang Fu, Xianpeng Wei, Zhicai Zhang, *Member, IEEE*, Laurence T. Yang, *Fellow, IEEE*, Lin Cai, *Fellow, IEEE*, Jia Luo, Zhe Zhang, *Member, IEEE*, and Chenmeng Wang

***Abstract*—Thanks to smart manufacturing and artificial intelligence technologies, unmanned aerial vehicles (UAVs) are envisioned to play a critical role in future Internet of things (IoT) networks to execute data collection tasks. In this article, we leverage age of information (AoI) to measure the freshness of data packets received by the UAV from IoT sensors. Considering the heterogeneity of IoT devices, we aim to minimize the weighted sum AoI by jointly optimizing the UAV's trajectory and IoT devices association in UAV-assisted IoT networks, where the UAV's cumulative propulsion energy cost is limited by the battery capacity. Since the optimization object is confined by a set of short-term constraints and a long-term constraint, this problem is modeled as a constrained Markov decision process (CMDP). We leverage safe actor-critic (Safe-AC) to solve the CMDP. To satisfy the mixed constraints, the safe policy set of Safe-AC is induced by a Lyapunov function, thereafter, a policy distillation technology is leveraged to search the optimal policy. Experimental results indicate that our proposed scheme can strictly satisfy the propulsion energy cost budget requirement at the expense of around 2% loss of the reward compared to baseline methods.**

***Index Terms*—Age of information, unmanned aerial vehicles, Internet of Things, safe actor-critic, policy distillation.**

Manuscript received 6 May 2023; revised 12 August 2023; accepted 27 September 2023. Date of publication 3 October 2023; date of current version 8 January 2024. This work was supported in part by the National Key R&D Program of China under Grant 2022ZD0118300, in part by the Fundamental Research Program of Shanxi Province under Grants 202103021224024 and 202103021223021, and in part by the Key R&D Program of Shanxi Province under Grant 202202020101004. Recommended for acceptance by Dr. Dejun Yang. *(Corresponding author: Zhicai Zhang.)*

Fang Fu and Zhicai Zhang are with the School of Computer Science and Technology, Hainan University, Haikou 570228, China, and also with the School of Physics and Electronic Engineering, Shanxi University, Taiyuan 030006, China (e-mail: fufang0621@hainanu.edu.cn; zzcai@hainanu.edu.cn).

Xianpeng Wei is with the School of Physics and Electronic Engineering, Shanxi University, Taiyuan 030006, China (e-mail: WeixpSXU@gmail.com).

Laurence T. Yang is with the School of Computer Science and Technology, Hainan University, Haikou 570228, China, and with the School of Computer Science and Technology, Huazhong University of Science and Technology, Wuhan 430030, China, and also with the Department of Computer Science, St. Francis Xavier University, Antigonish, NS B2G 2W5, Canada (e-mail: ltyang@ieee.org).

Lin Cai is with the Department of Electrical and Computer Engineering, University of Victoria, Victoria, BC V8W 3Y1, Canada (e-mail: cai@ece.uvic.ca).

Jia Luo is with the School of Cyber Security and Information Law, Chongqing University of Posts and Telecommunications, Chongqing 400065, China (e-mail: luojia@cqupt.edu.cn).

Zhe Zhang is with the School of Communications and Information Engineering, Nanjing University of Posts and Telecommunications, Nanjing 210049, China (e-mail: zhezhang@njupt.edu.cn).

Chenmeng Wang is with the School of Computer Science and Technology, Hainan University, Haikou 570228, China (e-mail: wangc@hainanu.edu.cn).

Digital Object Identifier 10.1109/TNSE.2023.3321764

I. Introduction

INTERNET of things (IoT) devices have been extensively used in smart agriculture [1], intelligent fishery [2], forest monitoring [3], [4] etc. to collect real-time data from surrounding environment. To support these real-time applications, the generated and perceived data by IoT sensors are expected to be transmitted to the receiver as new as possible. Take marine fisheries as an example, the data on temperature, salinity, power of hydrogen, and dissolved oxygen, are time-sensitive and obsolete data may cause fish disease, even causing disastrous loss. To measure the freshness of the data from the receiver's perspective, age of information (AoI) as an effective performance metric was proposed [5], which is defined as the elapsed time since the latest received update was generated, i.e., the recently received packet has a smaller value of age. Traditionally, throughput [6], coverage ratio [7], [8], and latency [9] are the main metrics to evaluate the performance of IoT networks. Nevertheless, these performance metrics may not quantify the freshness of the received data. For example, the delay metric represents the amount of the time spent from the source to the destination, which may not be able to character the age of the received data. Therefore, we can keep the received data fresh by minimizing the AoI.

Traditionally, the collected data are delivered to the receivers via terrestrial communication networks; however, ground infrastructures can be too costly to deploy in non-populated regions. Designing, developing, and deploying novel communication facilities is extremely urgent. Unmanned aerial vehicle (UAV)-assisted IoT network, a new paradigm of wireless communications is emerged as a promising solution for real- time data collection in agriculture management and other scenarios. Compared to terrestrial base stations, UAVs are able to overcome ground obstacles and make a flexible flying path to communicate with sensors more economically and efficiently. Besides, UAVs can leverage Line-of-Sight (LoS) links to serve IoT sensors. However, UAVs' on-board energy is limited and UAVs have to reserve sufficient energy to return to their bases or charge stations, which confines the flying path and data collection time. In addition, IoT sensors are often deployed dispersedly

and irregularly, which further increased trajectory designing difficulty. Therefore, how to guarantee the timeliness of the received data by optimizing UAV's trajectory while considering its limited energy budget, is a challenging task for UAV-assisted IoT networks. In the following statement, the terms "IoT sensors" and "IoT devices" are used interchangeable.

We investigate a UAV-enabled IoT system, where IoT sensors generate time-sensitive data and a battery-limited UAV cruises around to gather the information as fresh as possible. Considering the timeliness requirement and heterogeneity of devices, the optimization object is formulated to minimize the weighted sum AoI while taking UAV's limited on-board energy into account. Our main contributions can be summarized in the following points.

- We minimize the long-term weighted sum AoI of the network by jointly considering the UAV's path and IoT devices association, where the UAV cumulative propulsion energy cost is limited by the energy budget.
- We leverage safe actor-critic (Safe-AC) with policy distillation approach [10] to deal with the aforementioned problem. Since the optimization object is confined by a set of short-term constraints and a long-term energy constraint, we model the problem as a constrained Markov decision process (CMDP). Moreover, we leverage Safe-AC to deal with the CMDP. To ensure the safety of the policy, i.e., to satisfy the mixed constraints, the policy of Safe-AC is calculated in a safe policy set generated by a Lyapunov function. Furthermore, the optimal policy is obtained by employing a policy distillation technology to distill multiple trajectories' policies knowledge into a single one.
- A Python-based simulator is developed to implement the proposed algorithm. Extensive experimental results illustrate that our proposed scheme can take full advantages of available energy and avoid exceeding energy budget compared to baseline methods while effectively capturing IoT devices' status data.

The remainder of this article is organized as follows. We present the literature review of existing works in Section II. We introduce the system model and formulate the optimization problem in Sections III and IV, respectively. The Safe-AC approach is introduced to solve the problem in Section V. Simulation results are discussed in Section VI. At last, we present the conclusion and future work in Section VII.

II. Related Work

We briefly introduce the recent works in AoI-driven UAV resource allocation and deep reinforcement learning (DRL)-based UAV trajectory designing, respectively.

A. AoI-Driven UAV Resource Allocation

There are some excellent works on AoI-driven resource allocation in UAV-assisted IoT networks [11], [12], [13], [14], [15], [16], [17], [18]. For example, reference [11] minimized the total AoI of the network by optimizing the UAVs' path, UAVs' transmission and sensing time. Gu et al. derived the analytical solution of the average peak AoI based on the research of the status updating model of IoT devices [12]. Hu et al. proposed a data collection and wireless power transfer scheme for UAV-enabled IoT networks to minimize the average AoI by jointly optimizing the UAV's trajectory and the time assignment [13]. The above works try to optimize the freshness of the received data by designing UAVs' trajectory while satisfying a series of constraints. As we know, the UAV's cruise usually requires a large amount of propulsion energy, which has great influence on the UAV's path designing, especially for the energy-limited UAV systems. However, these works [11], [12], [13] overlook the propulsion energy cost. Considering the limitation of UAV's on-board energy, reference [14] proposed a UAV path designing scheme to reduce the UAV energy cost. A multi-objective optimization method was presented to minimize both AoI and the UAV energy cost in [15]. Reference [16] aimed to minimize the average peak AoI together with energy cost of both UAVs and sensors. Sun et al. tried to find a balance between AoI and the flying energy cost by optimizing UAV's flight path and spectrum allocation [17]. Fang et al. proposed a novel adaptive time slot and power control scheme for next generation multiple access systems to minimize the average peak AoI and energy consumption [18]. The above works may effectively decrease energy cost, however, which may not guarantee that the UAV's long-term propulsion energy cost never exceed the total energy budget. Furthermore, the UAV available energy is often underutilized in these schemes, which induces sub-optimal solutions on trajectory designing and results in high AoI consequently. Therefore, how to take full advantages of the UAV energy to make more reasonable decisions is a problem deserved to study. This article tries to investigate the UAV's trajectory designing and user association problem with the aim of minimizing the weighted sum AoI while satisfying the long-term propulsion energy constraint.

B. DRL Based UAV Trajectory Designing

DRL algorithms are considered as effective methods to deal with UAV trajectory designing problems [19], [20], [21], [22], where UAVs are treated as robots to search the optimal trajectory by interacting with the environment directly. Many excellent works have been done in academia recently. For instance, Wang et al. considered a UAV trajectory designing approach based on deep Q-learning network (DQN) algorithm to maximize the amount of user equipments (UEs) served in drone-enabled emergence communication systems [19]. To make the UAV exploring environment more efficiently, a curiosity-driven DQN (C-DQN) based trajectory designing method was proposed in [20]. To reduce energy cost of UEs in a drone-assisted edge computing system, reference [21] presented a deep deterministic policy gradient (DDPG) based algorithm for the UAV path design. Samir et al. [22] introduced a UAV altitude control and IoT devices association scheme based on proximal policy optimization (PPO). The optimization problems formulated in these works [19], [20], [21], [22] are with a set of short-term constraints, which are generally reformulated as a Markov decision process (MDP) and solved by the traditional DRL methods, such as, DQN, actor-critic, PPO, etc. Nevertheless,

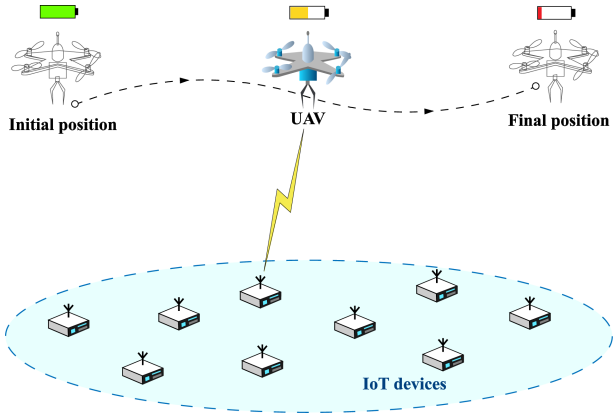


Fig. 1. UAV-assisted IoT networks.

since the optimization problem considered in this study is confined by a set of short-term constraints and a long-term constraint, how to transform the long-term constraint into short-term constraints for these traditional DRL algorithms is a big challenge.

Lagrangian-based DRL algorithms are useful tools to deal with this kind of problem [23], [24], [25], the CMDP is reformulated as an MDP by transforming the long-term constraint into a penalty reward, thereafter, traditional DRL methods are leveraged to solve the MDP. Reference [25] proposed a dueling double deep Q-learning network (DDQN) based UAV trajectory designing scheme to minimize the outage duration, where the long-term energy constraint is transformed as a short-term constraint by adding a penalty item to the reward function. However, these Lagrangian-based DRL methods [23], [24], [25] may result in either energy is underutilized or dead battery. Therefore, how to construct a safe and feasible policy set that satisfies the mixed constraints for CMDP is a challenging task.

In this study, we employ a novel DRL method, namely, safe actor-critic with policy distillation, to deal with the long-term weighted sum AoI minimization problem, which satisfies a set of short-term constraints and a long-term energy constraint.

III. System Model and Assumptions

Fig. 1 depicts the overview of the considered UAV-enabled IoT networks. In an outdoor area, K IoT devices are randomly deployed and a UAV is cruising at a fixed altitude H to collect IoT devices' status information from the start spot $\boldsymbol{p}[0] = (u_{\text{start}}, v_{\text{start}}, H)$ to the destination $\boldsymbol{p}[N] = (u_{\text{dest}}, v_{\text{dest}}, H)$ as fresh as possible. The UAV's whole cruising duration is split into N time slots equally and each slot has τ seconds. Let $\mathcal{N} = [1, \ldots, N]$ denote the collection of slots. The set of IoT devices is represented by $\mathcal{K} = \{1, \ldots, K\}$. Let $u[n]$ and $v[n]$ be the horizontal coordinate and vertical coordinate, respectively, therefore, the UAV's position in the n-th time interval can be denoted by $\boldsymbol{p}[n] = (u[n], v[n], H)$ $(\forall n \in \mathcal{N})$. The location of device k is denoted by $\boldsymbol{p}_k = (u_k, v_k, 0), \forall k \in \mathcal{K}$.

Each IoT device updates its status data at the start of each time slot, which is transmitted to the UAV directly when the device is scheduled. It is assumed that time division multiple access (TDMA) technology is used, i.e., at most one device can be associated with the UAV in a time slot. Let the binary variable s_k^n denote whether device k is scheduled by the UAV or not and $s_k^n = 1$ means device k is associated with the UAV and $s_k^n = 0$ otherwise. Therefore, we have

$$s_k^n \in \{0, 1\}, \forall k \in \mathcal{K}, n \in \mathcal{N}, \tag{1}$$

$$\sum_{k=1}^{K} s_k^n \leq 1, \forall k \in \mathcal{K}, n \in \mathcal{N}. \tag{2}$$

Let $\boldsymbol{s}[n] = \{s_1^n, \ldots, s_k^n, \ldots, s_K^n\}$ denote the UAV's scheduling strategy within time slot n.

A. The Propulsion Energy Model

According to references [26], [27], [28], [29], the UAV's propulsion power is calculated by

$$P_{\text{fly}}[n] = P_0 \left(1 + \frac{3\boldsymbol{V}^2[n]}{U_{tip}^2}\right) + P_1 \left(\sqrt{1 + \frac{\boldsymbol{V}^4[n]}{4V_0^4}} - \frac{\boldsymbol{V}^2[n]}{2V_0^2}\right)^{1/2} + \frac{1}{2} z_0 \rho \mu \xi \boldsymbol{V}^3[n], \tag{3}$$

where P_0 is the blade profile power, P_1 and V_0 are induced power and the average rotor induced velocity in hover, respectively. U_{tip} denotes the tip speed of the rotor blade and $\boldsymbol{V}[n]$ is computed by $\boldsymbol{V}[n] = \|\boldsymbol{p}[n] - \boldsymbol{p}[n-1]\|/\tau$ [14]. z_0 is the fuselage drag rate, μ is the rotor solidity, ρ is the air density, and ξ is the rotor disc area.

To reserve enough energy for the UAV to execute other functionalities, such as flying back to its bases or charging stations safely, the cumulative propulsion energy cost $E_{\text{fly}}[N]$ should satisfy

$$E_{\text{fly}}[N] = \sum_{n=1}^{N} P_{\text{fly}}[n]\tau \leq E_{\max}, \tag{4}$$

where $E_{\max}$ is the UAV's maximal available propulsion energy [30].

B. The Channel Model

G_{k2U} denotes the channel gain between device k and the UAV with position $\boldsymbol{p}[n]$, which is the average value over two transmission channels, i.e., LoS and non-LoS (NLoS) links and calculated by

$$G_{k2U}(\boldsymbol{p}[n]) = 20\log(4\pi f_c d_{k2U}(\boldsymbol{p}[n])\iota^{-1}) + \eta_{LoS}\Lambda_{LoS}(\boldsymbol{p}[n]) + \eta_{NLoS}(1 - \Lambda_{LoS}(\boldsymbol{p}[n])), \tag{5}$$

where ι is the speed of light, f_c is the carrier frequency, and $d_{k2U}(\boldsymbol{p}[n])$ represents the distance between device k and the UAV calculated by

$$d_{k2U}(\boldsymbol{p}[n]) = \sqrt{(u[n] - u_k)^2 + (v[n] - v_k)^2 + H^2}. \tag{6}$$

In (5), η_{LoS} and η_{NLoS} denote the additional mean losses of LoS and NLoS channels, respectively [31]. The probability of LoS Λ_{LoS} can be expressed as $\Lambda_{LoS}(\varphi) =$

$[1+\delta\exp(-\beta(\varphi-\delta))]^{-1}$, where δ and β are S-curve parameters [32], and φ is given by

$$\varphi(\boldsymbol{p}[n])=\frac{180}{\pi}\arctan\left(H^{-1}\sqrt{(u[n]-u_k)^2+(v[n]-v_k)^2}\right). \tag{7}$$

C. The AoI Model

Instead of leveraging throughput or latency, we adopt AoI to quantify the freshness of data from the receiver's view. Let A_k^n denote the AoI value of device k's status information saved on the UAV within time slot n.

According to Shannon theory, the achievable uplink data rate from device k to the UAV is written as

$$R_{k2U}^n(\boldsymbol{p}[n]) = B\log_2(1+P_{k2U}G_{k2U}(\boldsymbol{p}[n])/\sigma^2), \tag{8}$$

where P_{k2U} denotes the transmission power of device k, B denotes the bandwidth, σ^2 is the additive white Gaussian noise power, and $G_{k2U}(\boldsymbol{p}[n])$ is the channel gain discussed in Section III-B. We assume both B and P_{k2U} are constant, R_{k2U}^n can be regarded as a function of $\boldsymbol{p}[n]$ consequently. Considering (2) and (8), the amount of data received by the UAV from device k in time slot n can be calculated as

$$D_k^n(\boldsymbol{p}[n],\boldsymbol{s}[n]) = s_k^n\cdot R_{k2U}^n(\boldsymbol{p}[n])\cdot\tau. \tag{9}$$

Let $D_{\min}$ denote the minimum data size required to recover or decode the received data successfully [33], [34]. If $D_k^n(\boldsymbol{p}[n],\boldsymbol{s}[n])\geq D_{\min}$, we have $A_k^n=1$ that means the current status data of device k is transmitted to the UAV successfully; otherwise, we have $A_k^n=A_k^{n-1}+1$ that means device k's status data saved on the UAV is not updated within time slot n and AoI A_k^n becomes older. Obviously, A_k^n is a mapping from the UAV's path plan $\boldsymbol{p}[n]$ and UAV-IoT device association strategy $\boldsymbol{s}[n]$ to AoI, i.e.,

$$A_k^n(\boldsymbol{p}[n],\boldsymbol{s}[n]) = \begin{cases} 1; \mathrm{D}_k^n(\boldsymbol{p}[n],\boldsymbol{s}[n])\geq D_{\min}, \\ A_k^{n-1}+1; \text{otherwise}. \end{cases} \tag{10}$$

IV. Problem Formulation

Since the heterogeneity of IoT devices, the initial AoI values of devices are not equal. Considering the different priority of devices, we use the weight ω_k to indicate the relative importance of device k's information. Therefore, we aim to minimize the long-term weighted sum AoI by optimizing the UAV's path $\boldsymbol{p}[n]$ and the association strategy $\boldsymbol{s}[n]$. The optimization problem is described as:

$$\min_{\boldsymbol{p}[n],\boldsymbol{s}[n]}\sum_{n=1}^{N}\sum_{k=1}^{K}\omega_k A_k^n(\boldsymbol{p}[n],\boldsymbol{s}[n]) \tag{11a}$$

$$\text{s.t. } E_{\text{fly}}[N]=\sum_{n=1}^{N}P_{\text{fly}}[n]\tau\leq E_{\max}, \tag{11b}$$

$$\boldsymbol{p}[0]=(u_{\text{start}},v_{\text{start}},H), \tag{11c}$$

$$\boldsymbol{p}[N]=(u_{\text{dest}},v_{\text{dest}},H), \tag{11d}$$

$$\|\boldsymbol{p}[n]-\boldsymbol{p}[n-1]\|\leq v_{\max}\tau, \tag{11e}$$

$$s_k^n\in\{0,1\},\forall k\in\mathcal{K},n\in\mathcal{N}, \tag{11f}$$

$$\sum_{k=1}^{K}s_k^n\leq 1,\forall k\in\mathcal{K},n\in\mathcal{N}, \tag{11g}$$

where (11b) is the propulsion energy cost constraint. (11c) and (11d) represent the UAV's start location and destination, respectively. (11e) is the UAV's mobility constraint. (11f) and (11g) ensure that at most one device can be associated with the UAV in each time slot.

We find that the constraint conditions of problem (11) are short-term constraints (11c)–(11g) mixed with a long-term constraint condition (11b). The challenge is how to transform the long-term constraint as a short-term constraint. To solve the above problem, (11) is reformulated as a CMDP, which is subsequently solved by Safe-AC [10].

A. Problem Reformulation Based on CMDP

The problem (11) is reformulated as a CMDP that is described by a tuple $\langle\mathcal{X},\mathcal{A},\mathcal{P},x_0,r,c,c_0\rangle$. The details are given as follows.

- $\mathcal{X}=\mathcal{X}'\cup\mathcal{X}_{\text{dest}}$ represents the environment state feature space, where $\mathcal{X}'$ is the transient state space and $\mathcal{X}_{\text{dest}}$ is the final state space. $\mathcal{X}'$ contains three parts: (a) The UAV's position at the beginning of n-th time slot, i.e., $\boldsymbol{p}[n-1]=(u[n-1],v[n-1],H),\forall n\in\mathcal{N}$, where $(u[n-1],v[n-1])\neq(u_{\text{dest}},v_{\text{dest}})$. (b) The IoT devices' position $\boldsymbol{p}_k=(u_k,v_k,0),\forall k\in\mathcal{K}$. (c) The IoT devices' AoI values $\{A_k^n|\forall k\in\mathcal{K},n\in\mathcal{N}\}$. $\mathcal{X}_{\text{final}}$ includes $\{\boldsymbol{p}[n]=(u_{\text{dest}},v_{\text{dest}},H)|\forall n\in\mathcal{N}\}$.
- $\mathcal{A}$ denotes the action space that consists of the UAV's coordinates $\boldsymbol{p}[n]=(u[n],v[n],H)$ that satisfies (11e) and the scheduling strategy $\boldsymbol{s}[n]$ that satisfies (11f) and (11g).
- $\mathcal{P}$ is the state feature transition function. Considering that the state features include the UAV's position, the IoT devices' position, and AoI values, the corresponding transition functions are set as follows. First, since the UAV's position at the beginning of n-th time slot is determined by the action $\boldsymbol{p}[n-1]$, therefore, the UAV's position state transits according to $\widehat{\boldsymbol{p}}[n]=\boldsymbol{p}[n-1]$, where $\widehat{\boldsymbol{p}}[n]$ is the UAV's coordinates at the start of time slot n. Second, the IoT devices' position is $\boldsymbol{p}_k=(u_k,v_k,0)$. Third, AoI value A_k^n transits according to (10).
- $x_0\in\mathcal{X}'$ denotes the start state feature, which includes $\boldsymbol{p}[0]=(u_{\text{start}},v_{\text{start}},H)$ and A_k^0 $(\forall k\in\mathcal{K})$.
- r represents the immediate reward function. According to (11a) and optimality theory, r is given by

$$r=\begin{cases}-\sum_{k=1}^{K}\omega_k A_k^n, & \text{if } \boldsymbol{p}[n]\neq(u_{\text{dest}},v_{\text{dest}},H),\\ -\sum_{k=1}^{K}\omega_k A_k^n+\Omega, & \text{otherwise},\end{cases} \tag{12}$$

  where Ω is a positive constant that is used to induce the UAV to the final spot.
- c is the constraint cost, which can be described as: $c(x,a)=P_{\text{fly}}[n]\tau$ based on (11b).
- c_0 denotes the upper bound of the long-term constraint cost, according to (11b), we have $c_0=E_{\max}$.

Let $\Pi(x)=\{\pi(\cdot|x)|\sum_{a\in\mathcal{A}}\pi(a|x)=1\}$ denote the policy set. Given x_0 and π $(\forall\pi\in\Pi(x))$, the long-term reward of the

UAV is calculated by

$$\mathcal{R}^{\pi}(x_0) = \mathbb{E}\left\{\sum_{n=0}^{N^*-1} r(x_n, a_n)|x_0, \pi\right\}, \tag{13}$$

where N^* is the first arriving time from the start state x_0 to the destination. The long-term energy cost is written as

$$\mathcal{C}^{\pi}(x_0) = \mathbb{E}\left\{\sum_{n=0}^{N^*-1} c(x_n)|x_0, \pi\right\}, \tag{14}$$

which satisfies $\mathcal{C}^{\pi}(x_0) \leq c_0$. Above all, the optimization problem of the CMDP is formulated as

$$\pi^*(\cdot|x) = \arg\max_{\pi\in\Pi}\{\mathcal{R}^{\pi}(x_0)|\mathcal{C}^{\pi}(x_0) \leq c_0\}. \tag{15}$$

How to transfer the long-term constraint $\mathcal{C}^{\pi}(x_0)$ as a feasible single-step policy set is a critical issue to solve the CMDP. In the next section, we will take advantages of the Lyapunov function theory to construct a feasible policy set for the UAV to guarantee the obtained policy is safe, i.e., satisfying all constraints (11b)–(11g).

V. Safe Actor-Critic With Policy Distillation Approach

In this section, we leverage Safe-AC method to deal with the CMDP. We first introduce the construction of safe policy set induced by a Lyapunov function, based on which the critic and actor parts are presented, respectively. To fully utilize the past experiences when searching the optimal policy, policy distillation technology is leveraged. At last, the pseudo-code of Safe-AC based algorithm is presented.

A. The Safe Policy Set

In this part, the Lyapunov function theory is leveraged to build the safe policy set. To start with, it is assumed that we can obtain a baseline feasible policy[1] of Problem (15) denoted by $\pi_b(\cdot|x) \in \Pi$.

Definition 1: Given constraint threshold c_0 and the initial state x_0, the set of Lyapunov functions can be represented as

$$\Gamma_{\pi_b}(x_0, c_0) = \{\ell(x) : B_{\pi_b,c}[\ell](x) \leq \ell(x), \forall x \in \mathcal{X}';$$
$$\ell(x) = 0, \forall x \in \mathcal{X}\backslash\mathcal{X}'; \ell(x_0) \leq c_0\}, \tag{16}$$

where $B_{\pi_b,c}[\ell](x)$ is calculated by Bellman function operator, i.e., $B_{\pi_b,c}[\ell](x) = \sum_{a\in\mathcal{A}} \pi_b(a|x)[c(x,a) + \gamma \sum_{x'\in\mathcal{X}'} \mathcal{P}(x'|x,a)\ell(x')], \forall x \in \mathcal{X}, \pi_b \in \Pi$. For $\forall \ell(x) \in \Gamma_{\pi_b}(x_0, c_0)$, the Lyapunov function ℓ-induced safe policy set is written as

$$F_{\ell}(x) = \{\pi(\cdot|x) \in \Pi(x) : B_{\pi,c}[\ell](x) \leq \ell(x)\}. \tag{17}$$

Considering the contraction features of $B_{\pi,c}[\ell](x)$ and $\ell(x_0) \leq c_0, \forall \pi(\cdot|x) \in F_{\ell}(x)$ is a feasible policy of (15). From (17), it is observed that the larger ℓ means that the larger set $F_{\ell}(x)$ can be obtained and we have more opportunities to acquire π^* in $F_{\ell}(x)$ correspondingly. Hence, the critical job in the following is to construct a suitable Lyapunov function ℓ.

[1]For example, $\pi_b(\cdot|x) \in \arg\min_{\pi\in\Pi(x)} \mathcal{C}^{\pi}(x)$ is a baseline feasible policy of Problem (15).

We can transform the long-term constraint $\mathcal{C}^{\pi*}(x)$ w.r.t. π^* into a Lyapunov function induced by π_b, which is written as

$$\ell_{\Delta}(x) = \mathcal{C}^{\pi*}(x) = \mathbb{E}\left\{\sum_{n=0}^{N^*-1} [c(x_n) + \Delta(x_n)]\,|\pi_b, x\right\},$$
$$\forall x \in \mathcal{X}', \text{ and } \ell_{\Delta}(x) = 0, \forall x \in \mathcal{X}\backslash\mathcal{X}', \tag{18}$$

where $\Delta(x_n)$ is an additional constraint cost available at each step, which is utilized to expand the feasible action space and improve the policy consequently. Nevertheless, it is challenging to build $\Delta(x_n)$ without the priori knowledge of π^*. To reduce the computational complexity, $\Delta(x_n)$ is approximated by

$$\Delta = \Delta(x_n) = (c_0 - \mathcal{C}^{\pi_b}(x_0))/\mathbb{E}[N^*|x_0, \pi_b], \tag{19}$$

where $c_0 - \mathcal{C}^{\pi_b}(x_0)$ is the total auxiliary constraint cost available from x_0 to the final state and $\mathbb{E}[N^*|x_0, \pi_b]$ is the UAV's expected first-arriving time from the start position to the destination. In such a manner, we can take full advantages of the UAV's propulsion energy budget while planning the trajectory.

According to (18), $\ell_{\Delta}(x)$ can be calculated by

$$\ell_{\Delta}(x) = \sum_{a\in\mathcal{A}} \{\pi(a|x) Q_{\ell_{\Delta}}(x, a)\}, \tag{20}$$

where $Q_{\ell_{\Delta}}(x,a) = Q_C(x,a) + \Delta(x) Q_N(x,a)$ is the state-action value of ℓ_{Δ}, $Q_C(x,a)$ is the constraint value, $Q_N(x,a)$ is the residual steps from x to the final state, and $\Delta(x)Q_N(x,a)$ presents the rest of constraint cost, respectively. To guarantee the policy $\pi(a|x)$ is safe, the following inequation should be satisfied

$$[\pi(a|x) - \pi_b(a|x)]^{\top} Q_{\ell_{\Delta}}(x, a) \leq \Delta(x), \tag{21}$$

which means the extra costs $[\pi(a|x) - \pi_b(a|x)]^{\top} Q_{\ell_{\Delta}}(x,a)$ caused by $\pi(a|x)$ cannot exceed $\Delta(x)$. Then, the safe policy set (17) induced by $\ell_{\Delta}(x)$ can be rewritten as

$$F_{\ell_{\Delta}}(x) = \{\pi(\cdot|x) \in \Pi(x) :$$
$$[\pi(a|x) - \pi_b(a|x)]^{\top} Q_{\ell_{\Delta}}(x, a) \leq \Delta(x)\}. \tag{22}$$

B. The Critic Part

We adopt actor-critic framework to solve Problem (15) in following sections. In the critic part, we employ deep neural network (DNN) to evaluate $Q(x,a)$, $Q_C(x,a)$, and $Q_N(x,a)$, respectively. $Q(x,a)$ is evaluated by $Q(x,a) \doteq Q(x,a;\boldsymbol{\vartheta})$, where $\boldsymbol{\vartheta} = \{\vartheta_1, \vartheta_2, \cdots \vartheta_W\}$. At each step, the newly generated data is saved in an experience replay memory, i.e., $\mathcal{D} \leftarrow (x,a,r,c,x') \cup \mathcal{D}$. The DNN is trained by randomly sampling a batch of samples (x,a,r,c,x') from the replay memory and the parameter $\boldsymbol{\vartheta}$ is renewed by [35]

$$Loss(\boldsymbol{\vartheta}) = \mathbb{E}[(y - Q(x,a;\boldsymbol{\vartheta}))]^2, \tag{23}$$

where the target value is $y = r(x,a) + \max_{a\in\mathcal{A}} \widehat{Q}(x', a; \widehat{\boldsymbol{\vartheta}})$ and $Q(x,a;\boldsymbol{\vartheta})$ is the current Q value with parameter $\boldsymbol{\vartheta}$. The parameter $\boldsymbol{\vartheta}$ is renewed by

$$\boldsymbol{\vartheta} = \boldsymbol{\vartheta} - \alpha_{c,n}(y - Q(x,a;\boldsymbol{\vartheta})) \cdot \nabla_{\boldsymbol{\vartheta}} Q(x,a;\boldsymbol{\vartheta}), \tag{24}$$

where $\alpha_{c,n}$ is the critic's learning rate. The parameter $\widehat{\boldsymbol{\vartheta}}$ of the target Q value $\widehat{Q}(x', a; \widehat{\boldsymbol{\vartheta}})$ is updated by $\widehat{\boldsymbol{\vartheta}} = \boldsymbol{\vartheta}$ after several steps.

Algorithm 1: The Safe Actor-Critic Based Algorithm.

begin
  Initialize $\pi_b(\cdot|x)$, $\boldsymbol{\vartheta}$, $\boldsymbol{\vartheta}_C$, α, γ, $\boldsymbol{p}[0] = (u_{\text{start}}, v_{\text{start}})$, and $\boldsymbol{p}[N] = (u_{\text{end}}, v_{\text{end}})$;
  for $m \in \{0, 1, \cdots, \}$ **do**
    for $n = 1$ *to* $n = N$ **do**
      Select $a_n \sim \pi_m(\cdot|x)$ based on x_n, execute a_n, observe cost (r_n, c_n) and x_{n+1} orderly;
      Renew $\mathcal{D} \leftarrow (x_n, a_n, r_n, c_n, x_{n+1}) \cup \mathcal{D}$;
      `// The critic part`
      Sample $\mathcal{I} = \{(x_i, a_i, r_i, c_i, x_i')\}_{i=1}^{|I|}$ from $\mathcal{D}$;
      Update $\boldsymbol{\vartheta}$ and $\boldsymbol{\vartheta}_C$ based on (24) and (25) respectively;
      Update $Q_N(x, a) = N - n$ and $\Delta(x_n)$ according to (27);
      Update $Q_{\ell_\Delta}(x, a) = Q_C(x, a) + \Delta(x) Q_N(x, a)$;
      `// The actor part`
      Policy distillation:
      Extract the trajectories $\{x_{0,i}', \cdots, x_{N-1,i}'\}_{i=1}^{|I|}$ from $\mathcal{D}$ based on the set $\mathcal{I}$;
      Calculate the policy vector $\{\pi(\cdot|x_{0,i}'), \cdots, \pi(\cdot|x_{N-1,i}')\}_{i=1}^{|I|}$ according to (28);
      Calculate $\boldsymbol{\phi}^*$ according to (29);
      Update $\pi_{m+1}(\cdot|x) = \pi_{\boldsymbol{\phi}^*}(\cdot|x)$;
    end
    Update $\widehat{\boldsymbol{\vartheta}} = \boldsymbol{\vartheta}$ and $\widehat{\boldsymbol{\vartheta}}_C = \boldsymbol{\vartheta}_C$ after several episodes;
  end
end

Similarly, $Q_C(x, a)$ and $Q_N(x, a)$ are also evaluated by DNN approximators $Q_C(x, a; \boldsymbol{\vartheta}_C)$ and $Q_N(x, a; \boldsymbol{\vartheta}_N)$, respectively. The parameters $\boldsymbol{\vartheta}_C$ and $\boldsymbol{\vartheta}_N$ are updated by

$$\boldsymbol{\vartheta}_C = \boldsymbol{\vartheta}_C - \alpha_{c,n}(y_C - Q_C(x, a; \boldsymbol{\vartheta}_C)) \cdot \nabla_{\boldsymbol{\vartheta}} Q_C(x, a; \boldsymbol{\vartheta}_C), \tag{25}$$

$$\boldsymbol{\vartheta}_N = \boldsymbol{\vartheta}_N - \alpha_{c,n}(y_N - Q_N(x, a; \boldsymbol{\vartheta}_N)) \cdot \nabla_{\boldsymbol{\vartheta}} Q_N(x, a; \boldsymbol{\vartheta}_N), \tag{26}$$

where $y_C = c(x) + \pi(a|x')^\top \widehat{Q}_C(x', a; \widehat{\boldsymbol{\vartheta}}_C)$ and $y_N = 1 + \pi(a|x')^\top \widehat{Q}_N(x', a; \widehat{\boldsymbol{\vartheta}}_N)$, respectively. Consequently, (19) is transformed as

$$\Delta(x) = \frac{(c_0 - \pi_b(\cdot|x_0)^\top Q_C(x_0, \cdot; \boldsymbol{\vartheta}_C))}{\pi_b(\cdot|x_0)^\top Q_N(x_0, \cdot; \boldsymbol{\vartheta}_N)}. \tag{27}$$

C. The Actor Part

Based on the values $Q_C(x, a)$ and $Q_N(x, a)$ obtained in Section V-B, the safe policy set (22) is constructed. Then, the optimal action probabilities $\pi'(a|x)$ of (15) is calculated by

$$\pi'(a|x) = \arg\max_{\pi \in \Delta} \{\mathcal{R}^\pi(x) : [\pi(a|x) - \pi_b(a|x)]^\top Q_{\ell_\Delta}(x, a) \le \Delta(x)\}, \tag{28}$$

where $\mathcal{R}^\pi(x) = \pi(a|x)^\top Q(x, a)$ and $Q(x, a)$ is the Q-value of the reward. However, since the safe policy set $F_{\ell_\Delta}(x)$ is not stable at the beginning of the training, it is easy to found that $\pi'(a|x)$ cannot be used directly.

Policy distillation is a famous method for model compression [38], which has following advantages. a) Multiple teacher policies can be combined into a single student policy which has better performance than teachers. b) Policy distillation can be applied as an online learning process, which is able to continually distill the best policy for the actor. In this study, to make the best benefits of the past experiences when searching the optimal policy, policy distillation is leveraged to distill multi-trajectory policies knowledge into a single one [39]. First, we sample a batch of state trajectories $\{x_{0,j}', \ldots, x_{N-1,j}'\}_{j=1}^{|J|}$ from experience replay memory $\mathcal{D}$. Second, action probabilities $\{\pi(\cdot|x_{0,j}'), \ldots, \pi(\cdot|x_{N-1,j}')\}_{j=1}^{|J|}$ of these trajectories that are calculated by (28) are sent to the policy distillation part of Fig. 2. Third, by minimizing the average Jensen-Shannon (JS) divergence between the parameterized policy DNN $\pi_{\boldsymbol{\phi}}(\cdot|x)$ and action probabilities $\{\pi(\cdot|x_{0,j}'), \ldots, \pi(\cdot|x_{N-1,j}')\}_{j=1}^{|J|}$, the optimal policy parameter $\boldsymbol{\phi}^*$ is renewed by

$$\boldsymbol{\phi}^* \in \arg\min_{\boldsymbol{\phi}} \frac{1}{J} \sum_{j=1}^{J} \sum_{n=0}^{N-1} D_{JS}(\pi_{\boldsymbol{\phi}}(\cdot|x_{n,j}) || \pi'(\cdot|x_{n,j})), \tag{29}$$

where $D_{JS}(Y||Z) = \frac{1}{2} D_{KL}(Y||\frac{1}{2}(Y + Z)) + \frac{1}{2} D_{KL}(Z||\frac{1}{2}(Y + Z))$ and $D_{KL}(Y||Z)$ is Kullback-Leibler (KL) divergence that is used to measure the difference of distributions Y and Z.

D. Safe Actor-Critic Based Algorithm

Fig. 2 shows the framework of Safe-AC based algorithm and the pseudo-code is given in Algorithm 1, the convergence performance analysis of which can be found in [10]. Considering the whole flying time of the UAV is slotted as N, we have $\mathbb{E}[N^*|x_0, \pi_b] = N$ in (19) and $Q_N(x_j, a_j) = N - N'$, respectively, where N' is the experienced steps from x_0 to x_j. The learning rates $\alpha_{c,n}$ and $\alpha_{a,n}$ satisfy [40]

$$\sum_{n=0}^{\infty} \alpha_{c,n} = \infty, \sum_{n=0}^{\infty} \alpha_{c,n}^2 < \infty,$$
$$\sum_{n=0}^{\infty} \alpha_{a,n} = \infty, \sum_{n=0}^{\infty} \alpha_{a,n}^2 < \infty, \lim_{n \to \infty} \frac{\alpha_{a,n}}{\alpha_{c,n}} = 0. \tag{30}$$

VI. Simulation Results and Analysis

In this section, the proposed scheme and other baseline methods are implemented on a Python-based simulator. The simulation environment and parameters are described as follows. IoT devices are deployed on a 500×120 m^2 area as shown in Fig. 3, where some devices are located far away from the start and final spots [41]. The number of IoT devices K ranges

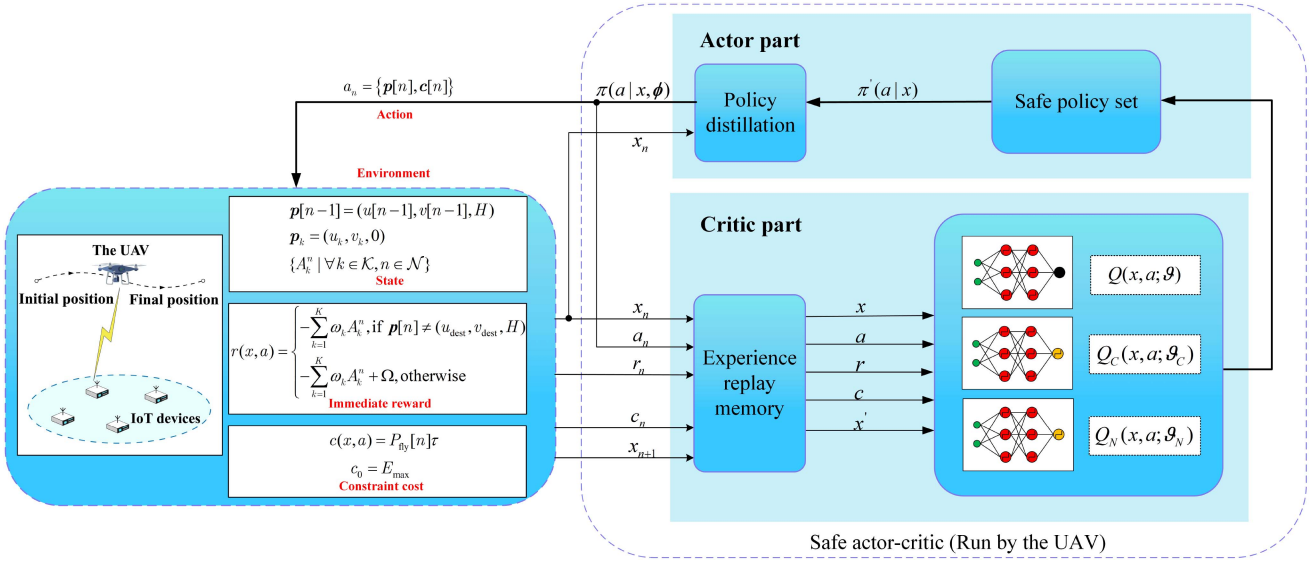


Fig. 2. Framework of safe actor-critic with policy distillation.

TABLE I
DEFINITIONS OF ABBREVIATIONS

Abbreviations	Definition
UAVs	Unmanned aerial vehicles
IoT	Internet of things
AoI	Age of information
MDP	Markov decision process
CMDP	Constrained Markov decision process
Safe-AC	Safe actor-critic
LoS	Line-of-Sight
NLoS	non-LoS
DRL	Deep reinforcement learning
DQN	Deep Q-learning network
C-DQN	Curiosity-driven DQN
UEs	User equipments
DDPG	Deep deterministic policy gradient
PPO	Proximal policy optimization
DDQN	Dueling double deep Q-learning network
TDMA	Time division multiple access
DNN	Deep neural network
JS	Jensen-Shannon
KL	Kullback-Leibler
NAC	Nature actor-critic
LAC	Lagrangian-based actor-critic

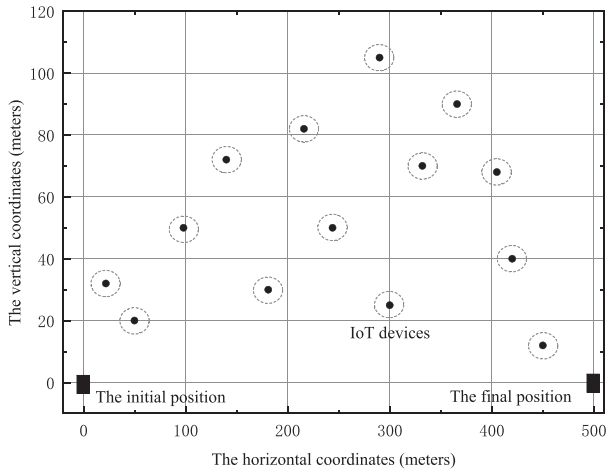


Fig. 3. Diagram of IoT devices' locations.

from 3 to 10. A UAV is cruising above the area to receive the data generated by IoT devices, the hovering altitude of which is fixed as $H = 100$ m [42]. The parameters of (12) are set as: $\Omega = 30$, $K = 6$, $\omega_1 = 4 \times 10^{-3}$, $\omega_k = 2 \times 10^{-3}$ ($\forall k \in \mathcal{K}$, $k \neq 1$). To make the UAV's path more regular and sleek, the initial AoI values are set as $A_1^0 = 10$ and $A_k^0 \geq A_{k+1}^0$ ($\forall k \in \mathcal{K}$). The detailed simulation parameters are listed in Table III.

In Figs. 4 and 5, the darker lines represent the average values and the shaded area represents the average values $\pm$ the standard error that reflects the variance of the curves. Fig. 4 demonstrates the convergence performance comparison of the proposed Safe-AC based algorithm w.r.t. different actor's learning rates, which satisfies (30) and are set by *trial-and-error*. The reward per episode is calculated according to (11a). In this part, the critic's

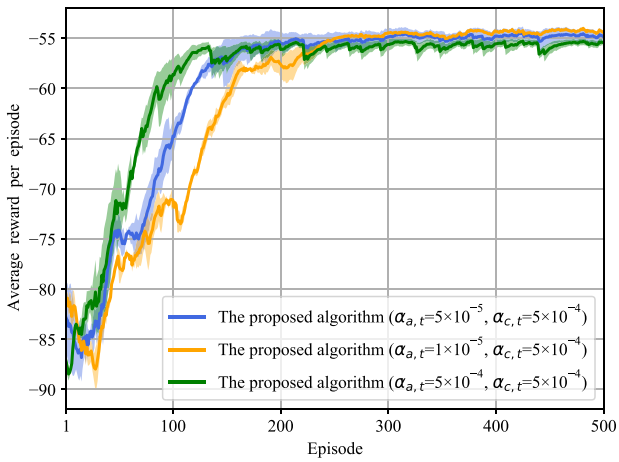


Fig. 4. Reward performance comparison w.r.t. different actor's learning rates.

TABLE II
DEFINITIONS OF NOTATION

Notation	Definition
$\mathcal{N}=[1,\cdots,N]$	The collection of time slots
$\mathcal{K}=\{1,\cdots,K\}$	The set of IoT-devices
$\boldsymbol{p}[n]=(u[n],v[n],H)$	The UAV's position
$\boldsymbol{p}_k=(u_k,v_k,0)$	The location of device k
$\boldsymbol{s}[n]$	The UAV's scheduling strategy
$P_{\text{fly}}[n]$	The UAV's propulsion power
$\boldsymbol{V}[n]$	The UAV's speed
$E_{\text{fly}}[N]$	The total propulsion energy cost
G_{k2U}	The channel gain
ι	The speed of light
$d_{k2U}(\boldsymbol{p}[n])$	The distance between device k and the UAV
$R^n_{k2U}(\boldsymbol{p}[n])$	The uplink data rate
$D^n_k(\boldsymbol{p}[n],\boldsymbol{s}[n])$	The amount of data received by the UAV
$A^n_k(\boldsymbol{p}[n],\boldsymbol{s}[n])$	The value of AoI
ω_k	The weight of AoI
r	The immediate reward function
c	The constraint cost
c_0	The budget of propulsion energy cost
$\Pi(x)$	The policy set
π	The UAV's policy
$\mathcal{R}^\pi(x_0)$	The long-term reward
$\mathcal{C}^\pi(x_0)$	The long-term energy cost
$\Gamma_{\pi_b}(x_0,c_0)$	The set of Lyapunov functions
$\pi_b(\cdot\|x)$	A baseline policy
$F_\ell(x)$	Lyapunov function ℓ induced safe policy set
$\Delta(x_n)$	An additional constraint cost at each step
$\ell_\Delta(x)$	Lyapunov function induced by π_b
$Q_{\ell_\Delta}(x,a)$	The state-action value of ℓ_Δ
$Q_C(x,a)$	The constraint value
$Q_N(x,a)$	The residual steps from x to the final state
ϑ	The hyper-parameters of $Q(x,a)$
$\mathcal{D}$	The experience replay memory
ϑ_C	The hyper-parameters of $Q_C(x,a)$
ϑ_N	The hyper-parameters of $Q_N(x,a)$
ϕ	The hyper-parameters of the policy network
$\alpha_{c,n}$	The critic's learning rate
$\alpha_{a,n}$	The actor's learning rate

TABLE III
SIMULATION PARAMETERS

Parameter	Value
Carrier frequency f_c	5.9 GHz [36]
Bandwidth B	1 MHz [36]
Transmission power p_{k2U}	0.1 W [37]
The additive white Gaussian noise power σ^2	-110 dBm
The additional mean losses of LoS channels η_{LoS}	1 dB [32]
The additional mean losses of NLoS channels η_{NLoS}	20 dB [32]
S-curve parameter δ	9.61 [32]
S-curve parameter β	0.16 [32]
The blade profile power P_0	3.4 W [26]
The induced power P_1	118 W [26]
The average rotor induced velocity in hover V_0	5.4 m/s [26]
The maximal speed $v_{\max}$	30 m/s [26]
The tip speed of the rotor blade U_{tip}	60 m/s [26]
The fuselage drag rate z_0	0.3 [26]
The rotor solidity μ	0.03 [26]
The air density ρ	1.225 kg/m^3
The rotor disc area ξ	0.28 m^2 [26]

learning rate is set as $\alpha_{c,t}=5\times10^{-4}$. The algorithm runs 500 episodes totally and each episode includes 100 steps. We find that the curve reaches convergence after around 150 episodes while suffering a high variance and low reward when $\alpha_{a,t}=5\times10^{-4}$. That is because the high learning rate always result in over-shooting. Nevertheless, the learning speed becomes slower when the learning rate is dropped to $\alpha_{a,t}=1\times10^{-5}$. Compared to

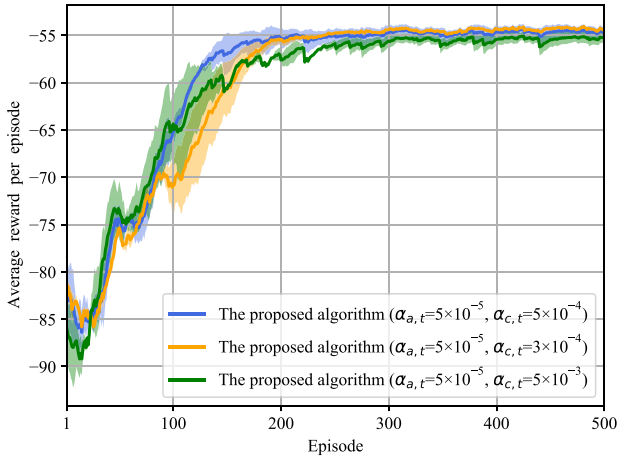


Fig. 5. Reward performance comparison w.r.t. different critic's learning rates.

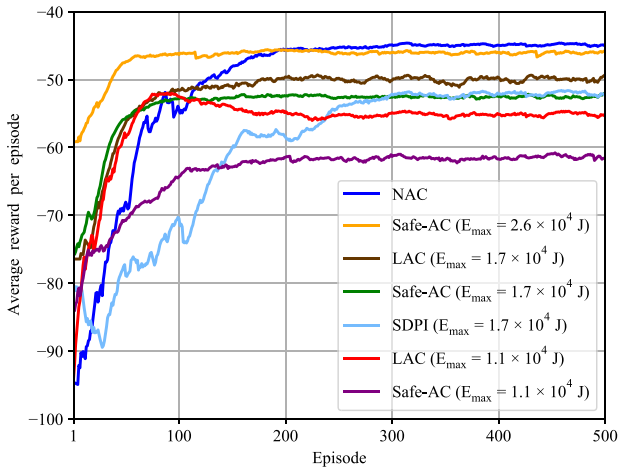


Fig. 6. Reward performance per episode w.r.t. different total energy budget.

$\alpha_{a,t}=1\times10^{-5}$ and $\alpha_{a,t}=5\times10^{-4}$ cases, $\alpha_{a,t}=5\times10^{-5}$ is the best learning rate, which has excellent performances in terms of average return and variance. Fig. 5 shows the convergence properties of the proposed algorithm w.r.t. the critic's learning rate $\alpha_{c,t}$, where $\alpha_{a,t}$ is set as 5×10^{-5}. We also find that the convergence performance is sensitive to learning rates, to be specific, the learning rate $\alpha_{c,t}=5\times10^{-3}$ results in significant variances while $\alpha_{c,t}=3\times10^{-4}$ causes the longer learning time. We observe that the best learning rate of the critic is $\alpha_{c,t}=5\times10^{-4}$. Hence, in the following part, $\alpha_{a,t}$ and $\alpha_{c,t}$ are set as $\alpha_{a,t}=5\times10^{-5}$ and $\alpha_{c,t}=5\times10^{-4}$, respectively.

To show the high efficiency of the proposed Safe-AC based algorithm, a nature actor-critic based algorithm (NAC) [43], a Lagrangian-based actor-critic algorithm (LAC) [23], [24], [25], and safe deep policy improvement based algorithm (SDPI) [10] are also simulated. The key performance comparisons among these algorithms are summarized in Table IV. Fig. 6 is the reward performance per episode w.r.t. different total energy budget of Safe-AC, LAC, and NAC. From the figure, we find that NAC gets the highest reward among the algorithms. Since NAC does not consider the propulsion energy limitation, therefore, the policy of NAC is not limited by $E_{\max}$ and NAC can make the flying path

TABLE IV
COMPARISONS AMONG THE SAFE-AC BASED ALGORITHM AND OTHER BASELINE METHODS

Methods	Base algorithm	MDP/CMDP	The safety policy set	Satisfying the propulsion energy budget?	Time complexity*
The Safe-AC based algorithm	Safe actor-critic	CMDP	✓	✓	$O(MN\|X\|^2\|A\|^2 + MN\|X\|^3\|A\|^3)$
The NAC-based algorithm	Nature actor-critic	MDP	✗	✗	$O(MN\|X\|^2\|A\|^2)$
The LAC-based algorithm	Lagrangian-based actor-critic	MDP	✗	✗	$O(MN\|X\|^2\|A\|^2)$

* M is the number of episodes and N is the number of steps in an episode. $|X|$ and $|A|$ denote the dimensions of $\mathcal{X}$ and $\mathcal{A}$, respectively.

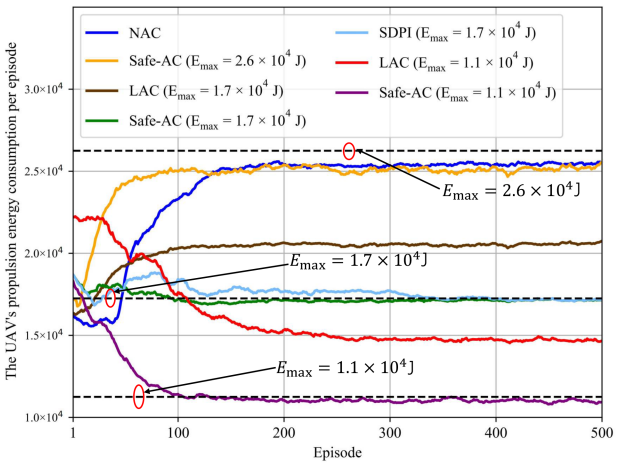


Fig. 7. UAV's cumulative propulsion energy cost per episode w.r.t. different total energy budget.

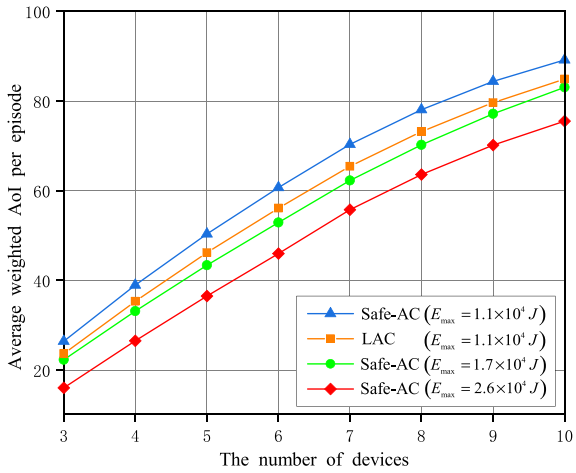


Fig. 8. Weighted sum AoI of all devices per episode w.r.t. different total energy budget.

decision more flexible and obtain a higher reward. Therefore, NAC is considered as a baseline of the reward in the following analysis. From Fig. 6, we notice that the reward of Safe-AC is obviously increased when $E_{\max}$ rises from 1.1×10^4 J (the purple curve) to 2.6×10^4 J (the yellow curve). Because the larger $E_{\max}$ results in a bigger feasible action space, which means the UAV have more opportunities to obtain the optimal strategy and a higher reward. We also find that LAC (the red curve) has a slight high reward compared to Safe-AC (the purple line) when $E_{\max} = 1.1 \times 10^4$ J. That is because the trajectory designing of LAC is more flexible than Safe-AC case, due to the fact that the policy of LAC is not seriously confined by the energy cost limit that can be seen in Fig. 7. At last, Safe-AC (the yellow curve) has similar reward with NAC (the deep blue curve) when the energy budget is sufficient.

Fig. 7 is the UAV's cumulative propulsion energy cost per episode of Safe-AC, NAC, and LAC w.r.t. different total energy budget. From Fig. 7, we observe that the total propulsion energy cost of Safe-AC (the purple curve) is decreasing from around 1.6×10^4 J to less than 1.1×10^4 J after convergence when $E_{\max} = 1.1 \times 10^4$ J. On the contrary, we observe that the energy cost of LAC (the red curve) is around 1.5×10^4 J when $E_{\max} = 1.1 \times 10^4$ J. The reason is that Safe-AC constructs a safe policy set for the UAV based on the energy budget $E_{\max}$, therefore, the total propulsion energy cost do not exceed the budget $E_{\max}$ consequently; while the policy of LAC cannot be seriously limited by the long-term energy constraint, i.e., the UAV's total propulsion energy cost per episode may exceed the total energy budget. By comparison, the energy cost of Safe-AC (the yellow curve) increases significantly from around 2.1×10^4 J to around 2.5×10^4 J when $E_{\max} = 2.6 \times 10^4$ J. That is because Safe-AC tries to fully utilize available energy while not exceeding the energy budget. Besides, it is observed from Fig. 7 that SDPI can satisfy the energy budget when $E_{\max} = 1.7 \times 10^4$ J, however, which suffers lower learning speed than the proposed algorithm. The main reason is that SDPI is lack of a value estimation network and updates its policy parameters at the end of each episode, while the proposed algorithm updates the policy parameters immediately after the action has been taken according to the evaluated state-action values. Hence, based on Figs. 6 and 7, we draw a conclusion that our proposed Safe-AC strictly satisfy the propulsion energy cost budget requirement at the expense of slight loss (around 2%) of the reward compared to NAC and LAC.

Fig. 8 shows the AoI values of IoT devices versus the number of devices when $\omega_1 = 4 \times 10^{-3}$, $\omega_k = 2 \times 10^{-3}$ ($\forall k \in \mathcal{K}$, $k \neq 1$), and $A_k^0 = 10$ ($\forall k \in \mathcal{K}$). The values in the figure are averaged over the last 50 episodes after convergence. Similar with Fig. 3, some devices are deployed far away from the start and final positions in this part. From the figure, we find that when the number of devices is rising, the weighted sum AoI keeps increasing observably. That is because at most one device is connected by the UAV at each time slot, more devices deployed means each device enjoys less services and the

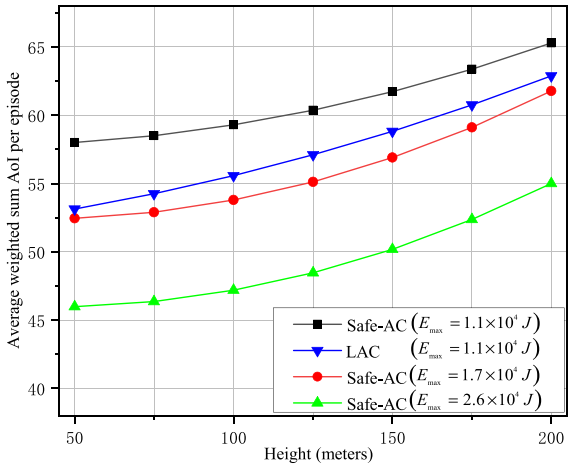


Fig. 9. Reward performance versus height.

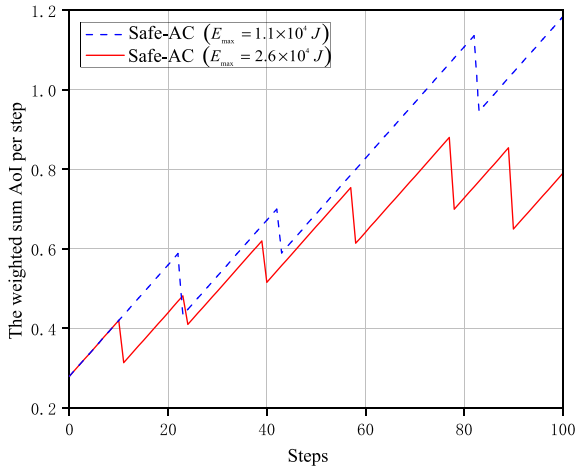


Fig. 10. Weighted sum AoI per step.

sum of AoI increases consequently. Besides, when the energy budget is rising, the AoI is decreasing for the fixed number of devices. The reason is that more propulsion energy budget is available, the UAV can make more flexible trajectory planning to receive the data of devices with higher AoI values. At last, LAC has lower AoI than Safe-AC when $E_{\max} = 1.1 \times 10^4$ J, which can also be observed in Fig. 9. Since LAC may exceed the energy budget and reaches the farther device than Safe-AC does.

Fig. 9 shows the weighted sum AoI per episode versus the flying height of the UAV. The values in the figure are averaged over the last 50 episodes after convergence. We notice that the AoI value rises when the UAV's flight altitude is increasing. Since the channel gains between the IoT device and the UAV are mainly determined by the distance between them, therefore, the higher flying altitude causes the weaker channel conditions and the lower transmit rate consequently when the bandwidth and transmit power are given. Furthermore, from (10), we find that the lower transmit rate will cause the higher AoI value. Therefore, the AoI value increases when the UAV's flight altitude is rising.

Fig. 10 shows the weighted sum AoI per step in an episode, which is calculated by $\sum_{k=1}^{K} \omega_k A_k^n(\boldsymbol{p}[n], \boldsymbol{s}[n])$ $(\forall n \in \mathcal{N})$, where $K = 6, N = 100, \omega_1 = 4 \times 10^{-3}, \omega_k = 2 \times 10^{-3}$ $(\forall k \in \mathcal{K}, k \neq 1)$, and $A_k^0 = 10$ $(\forall k \in \mathcal{K})$. We find that the curves are in saw-toothed. That is because the AoI of a device is set as 1 if the current generated data by the device is transmitted to the UAV successfully, otherwise the AoI value adds 1 that can be found from (10). On the other hand, we find from the figure that the AoI value of the curve changes 6 times when $E_{\max} = 2.6 \times 10^4$ J, which means $K = 6$ devices are traversed by the UAV, while the curve when $E_{\max} = 1.1 \times 10^4$ J only changes 3 times. The reason is that more energy available can make more flexible flying path decision to capture all devices' data.

Fig. 11 shows the UAV's trajectories w.r.t. different energy budget. The weight ω_1 is set higher than that of other devices, which is used to stimulate the UAV to capture the status data of device 1 firstly. To demonstrate the efficiency of Safe-AC, some

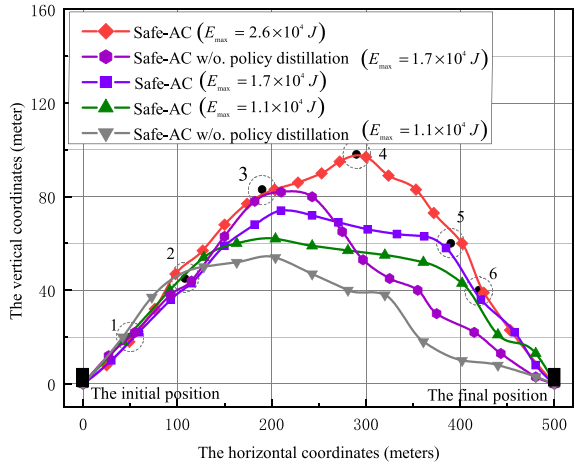


Fig. 11. UAV's trajectories w.r.t. different total energy budget.

IoT devices, e.g. devices 3, 4, and 5 in Fig. 11, are deployed far away from the start and final spots. According to (10), since the AoI value is set to 1 when the status information is collected successfully, the UAV is likely to move towards the next device with high AoI value. In such a manner, the UAV is encouraged to traverse each device in turn. From Fig. 11, we find that the UAV's path can pass through all IoT devices when $E_{\max} = 2.6 \times 10^4$ J, while only a part of devices are served when $E_{\max} = 1.1 \times 10^4$ J and $E_{\max} = 1.7 \times 10^4$ J. For instance, device 3 and 4 are not served by the UAV when $E_{\max} = 1.7 \times 10^4$ J. This is due to the fact that these devices are dispatched far away from the destination, which may consume a large amount of the UAV's energy to cover them. To guarantee the UAV's accumulated energy cost is no more than the total energy budget, these devices are ignored. At last, from Fig. 11, we find that the trajectories of Safe-AC without (w/o.) policy distillation cover less devices than the case of Safe-AC no matter when $E_{\max} = 1.7 \times 10^4$ J or when $E_{\max} = 1.1 \times 10^4$ J. That is because policy distillation employed by Safe-AC can combine multiple expert polices into a single policy, which outperforms the original polices. Therefore,

the proposed Safe-AC can give a reasonable trajectory to served IoT devices while satisfying the long-term energy constraint.

VII. Conclusion

This article focused on minimizing the long-term weighted sum AoI for UAV-aided IoT networks by optimizing the UAV's trajectory designing and IoT devices association strategy. Considering the limitation of UAV on-board energy, we take the UAV's long-term flight energy cost into account. Since the optimization object was confined by a set of short-term constraints and a long-term constraint, the problem was formulated as a CMDP, which was subsequently solved by Safe-AC approach with policy distillation. Finally, experimental results showed that our proposed Safe-AC based scheme could fully utilize available energy and avoid exceeding energy budget compared to baseline methods while effectively capturing IoT devices' status data. In the future work, we will discuss AoI minimization problem in multiple UAVs scenarios by leveraging multi-agent DRL methods.

References

[1] H. Yang, R. Ruby, Q.-V. Pham, and K. Wu, "Aiding a disaster spot via multi-UAV-based IoT networks: Energy and mission completion time-aware trajectory optimization," *IEEE Internet Things J.*, vol. 9, no. 8, pp. 5853–5867, Apr. 2022.

[2] A. Prayudi, I. A. Sulistijono, A. Risnumawan, and Z. Darojah, "Surveillance system for illegal fishing prevention on UAV imagery using computer vision," in *Proc. IEEE Int. Electron. Symp.*, 2020, pp. 385–391.

[3] N. Lin, Y. Liu, L. Zhao, D. O. Wu, and Y. Wang, "An adaptive UAV deployment scheme for emergency networking," *IEEE Trans. Wireless Commun.*, vol. 21, no. 4, pp. 2383–2398, Apr. 2022.

[4] L. Sun, L. Wan, and X. Wang, "Learning-based resource allocation strategy for industrial IoT in UAV-enabled MEC systems," *IEEE Trans. Ind. Informat.*, vol. 17, no. 7, pp. 5031–5040, Jul. 2021.

[5] S. Kaul, R. Yates, and M. Gruteser, "Real-time status: How often should one update?," in *Proc. IEEE INFOCOM*, 2012, pp. 2731–2735.

[6] L. Liu, A. Wang, G. Sun, and J. Li, "Multi-objective optimization for improving throughput and energy efficiency in UAV-enabled IoT," *IEEE Internet Things J.*, vol. 9, no. 20, pp. 20763–20777, Oct. 2022.

[7] T. Li, W. Liu, Z. Zeng, and N. N. Xiong, "DRLR: A deep reinforcement learning based recruitment scheme for massive data collections in 6G-based IoT networks," *IEEE Internet Things J.*, vol. 9, no. 16, pp. 14595–14609, Aug. 2022.

[8] N. Gupta, S. Agarwal, and D. Mishra, "Joint trajectory and velocity-time optimization for throughput maximization in energy constrained UAV," *IEEE Internet Things J.*, vol. 9, no. 23, pp. 24516–24528, Dec. 2022.

[9] C. Deng, X. Fang, and X. Wang, "UAV-enabled mobile edge computing for AI applications: Joint model decision, resource allocation and trajectory optimization," *IEEE Internet Things J.*, vol. 10, no. 7, pp. 5662–5675, Apr. 2023.

[10] Y. Chow, O. Nachum, A. Faust, E. D. Guzmn, and M. Ghavamzadeh, "Lyapunov-based safe policy optimization for continuous control," in *Proc. 36th Int. Conf. Mach. Learn. Workshops*, 2019, pp. 1–22.

[11] S. Zhang, H. Zhang, Z. Han, H. V. Poor, and L. Song, "Age of information in a cellular Internet of UAVs: Sensing and communication trade-off design," *IEEE Trans. Wireless Commun.*, vol. 19, no. 10, pp. 6578–6592, Oct. 2020.

[12] Y. Gu, H. Chen, Y. Zhou, Y. Li, and B. Vucetic, "Timely status update in Internet of Things monitoring systems: An age-energy tradeoff," *IEEE Internet Things J.*, vol. 6, no. 3, pp. 5324–5335, Jun. 2019.

[13] H. Hu, K. Xiong, G. Qu, Q. Ni, P. Fan, and K. B. Letaief, "AoI-minimal trajectory planning and data collection in UAV-assisted wireless powered IoT networks," *IEEE Internet Things J.*, vol. 8, no. 2, pp. 1211–1223, Jan. 2021.

[14] X. Hu, K.-K. Wong, K. Yang, and Z. Zheng, "UAV-assisted relaying and edge computing: Scheduling and trajectory optimization," *IEEE Trans. Wireless Commun.*, vol. 18, no. 10, pp. 4738–4752, Oct. 2019.

[15] Y. Liao and V. Friderikos, "Energy and age Pareto optimal trajectories in UAV-assisted wireless data collection," *IEEE Trans. Veh. Technol.*, vol. 71, no. 8, pp. 9101–9106, Aug. 2022.

[16] X. Diao, X. Guan, and Y. Cai, "Joint offloading and trajectory optimization for complex status updates in UAV-assisted Internet of Things," *IEEE Internet Things J.*, vol. 9, no. 23, pp. 23881–23896, Dec. 2022.

[17] M. Sun, X. Xu, X. Qin, and P. Zhang, "AoI-energy-aware UAV-assisted data collection for IoT networks: A deep reinforcement learning method," *IEEE Internet Things J.*, vol. 8, no. 24, pp. 17275–17289, Dec. 2021.

[18] Z. Fang, J. Wang, Y. Ren, Z. Han, H. V. Poor, and L. Hanzo, "Age of information in energy harvesting aided massive multiple access networks," *IEEE J. Sel. Areas Commun.*, vol. 40, no. 5, pp. 1441–1456, May 2022.

[19] L. Wang, K. Wang, C. Pan, X. Chen, and N. Aslam, "Deep Q-network based dynamic trajectory design for UAV-aided emergency communications," *J. Commun. Inf. Netw.*, vol. 5, no. 4, pp. 393–402, Dec. 2020.

[20] F. Fu, Q. Jiao, F. R. Yu, Z. Zhang, and J. Du, "Securing UAV-to-vehicle communications: A curiosity-driven deep Q-learning network (C-DQN) approach," in *Proc. IEEE Int. Conf. Commun. Workshops*, 2021, pp. 1–6.

[21] L. Wang, K. Wang, C. Pan, W. Xu, N. Aslam, and A. Nallanathan, "Deep reinforcement learning based dynamic trajectory control for UAV-assisted mobile edge computing," *IEEE Trans. Mobile Comput.*, vol. 21, no. 10, pp. 3536–3550, Oct. 2022.

[22] M. Samir, C. Assi, S. Sharafeddine, and A. Ghrayeb, "Online altitude control and scheduling policy for minimizing AoI in UAV-assisted IoT wireless networks," *IEEE Trans. Mobile Comput.*, vol. 21, no. 7, pp. 2493–2505, Jul. 2022.

[23] P. Geibel and F. Wysotzki, "Risk-sensitive reinforcement learning applied to control under constraints," *J. Artif. Intell. Res.*, vol. 24, pp. 81–108, 2005.

[24] Z. Zhang et al., "Energy-efficient secure video streaming in UAV-enabled wireless networks: A safe-DQN approach," *IEEE Trans. Green Commun. Netw.*, vol. 5, no. 4, pp. 1892–1905, Dec. 2021.

[25] Y. Gao, L. Xiao, F. Wu, D. Yang, and Z. Sun, "Cellular-connected UAV trajectory design with connectivity constraint: A deep reinforcement learning approach," *IEEE Trans. Green Commun. Netw.*, vol. 5, no. 3, pp. 1369–1380, Sep. 2021.

[26] 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.

[27] S. Hu, W. Ni, X. Wang, A. Jamalipour, and D. Ta, "Joint optimization of trajectory, propulsion, and thrust powers for covert UAV-on-UAV video tracking and surveillance," *IEEE Trans. Inf. Forensics Secur.*, vol. 16, pp. 1959–1972, 2021.

[28] J. Wang, C. Jiang, Z. Wei, C. Pan, H. Zhang, and Y. Ren, "Joint UAV hovering altitude and power control for space-air-ground IoT networks," *IEEE Internet Things J.*, vol. 6, no. 2, pp. 1741–1753, Apr. 2019.

[29] T. Bai, C. Pan, H. Ren, Y. Deng, M. Elkashlan, and A. Nallanathan, "Resource allocation for intelligent reflecting surface aided wireless powered mobile edge computing in OFDM systems," *IEEE Trans. Wireless Commun.*, vol. 20, no. 8, pp. 5389–5407, Aug. 2021.

[30] K. Xu, M. M. Zhao, Y. Cai, and L. Hanzo, "Low-complexity joint power allocation and trajectory design for UAV-enabled secure communications with power splitting," *IEEE Trans. Commun.*, vol. 69, no. 3, pp. 1896–1911, Mar. 2020.

[31] T. Bai, J. Wang, Y. Ren, and L. Hanzo, "Energy-efficient computation offloading for secure UAV-edge-computing systems," *IEEE Trans. Veh. Technol.*, vol. 68, no. 6, pp. 6074–6087, Jun. 2019.

[32] 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.

[33] M. Samir, C. Assi, S. Sharafeddine, D. Ebrahimi, and A. Ghrayeb, "Age of information aware trajectory planning of UAVs in intelligent transportation systems: A deep learning approach," *IEEE Trans. Veh. Technol.*, vol. 69, no. 11, pp. 12382–12395, Nov. 2020.

[34] O. S. Oubbati, M. Atiquzzaman, H. Lim, A. Rachedi, and A. Lakas, "Synchronizing UAV teams for timely data collection and energy transfer by deep reinforcement learning," *IEEE Trans. Veh. Technol.*, vol. 71, no. 6, pp. 6682–6697, Jun. 2022.

[35] F. Fu et al., "Live traffic video multicasting services in UAVs-assisted intelligent transport systems: A multi-actor attention critic approach," *IEEE Internet Things J.*, vol. 10, no. 22, pp. 19740–19752, Nov. 2023.

[36] Z. Zhang, R. Wang, F. R. Yu, F. Fu, and Q. Yan, "QoS aware transcoding for live streaming in edge-clouds aided HetNets: An enhanced actor-critic approach," *IEEE Trans. Veh. Technol.*, vol. 68, no. 11, pp. 11295–11308, Nov. 2019.
[37] J. Du et al., "Resource pricing and allocation in MEC enabled blockchain systems: An A3C deep reinforcement learning approach," *IEEE Trans. Netw. Sci. Eng.*, vol. 9, no. 1, pp. 33–44, Jan./Feb. 2022.
[38] Y. Sun and Q. Zhang, "Ensemble policy distillation with reduced data distribution mismatch," in *Proc. Int. Joint Conf. Neural Netw.*, 2022, pp. 1–8.
[39] A. A. Rusu et al., "Policy distillation," in *Proc. Int. Conf. Learn. Representations*, 2016, pp. 1–13.
[40] F. Fu, Y. Kang, Z. Zhang, F. R. Yu, and T. Wu, "Soft actor–critic DRL for live transcoding and streaming in vehicular fog-computing-enabled IoV," *IEEE Internet Things J.*, vol. 8, no. 3, pp. 1308–1321, Feb. 2021.
[41] J. Luo, L. Tang, Q. Chen, and Z. Zhang, "Trajectory design and bandwidth allocation considering power-consumption outage for UAV communication: A machine learning approach," *IEEE Trans. Ind. Informat.*, to be published, doi: 10.1109/TII.2023.3292522.
[42] X. Hou, J. Wang, C. Jiang, X. Zhang, Y. Ren, and M. Debbah, "UAV-enabled covert federated learning," *IEEE Trans. Wireless Commun.*, vol. 22, no. 10, pp. 6793–6809, Oct. 2023.
[43] R. S. Sutton and A. G. Barto, *Reinforcement Learning: An Introduction*. Cambridge, MD, USA: MIT Press, 2017.

Fang Fu received the Ph.D. degree from the Department of Information and Communication Engineering, Harbin Engineering University, Harbin, China, in 2014. She is currently a Postdoctoral Fellow under the supervision of Prof. Laurence T. Yang at Hainan University, Haikou, Hainan, China. Her research interests include UAVs, federated learning, and deep reinforcement learning. She was the recipient of the Best Paper Award at IEEE Globecom'20.

Xianpeng Wei received the B.S. degree from Qingdao University, Qingdao, China, in 2020. He is currently working toward the master's degree with the School of Physics and Electronic Engineering, Shanxi University, Taiyuan, China. His current research interests include distributed machine learning, UAVs, and resource allocation.

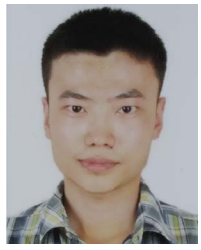
Zhicai Zhang (Member, IEEE) received the Ph.D. degree from the School of Information and Communication Engineering, Beijing University of Posts and Telecommunications, Beijing, China, in 2014. He is currently an Assistant Professor with the School of Computer Science and Technology, Hainan University, Haikou, China. From 2017 to 2018, he was with Carleton University, Ottawa, ON, Canada, as a Visiting Scholar. His research interests include edge intelligence, distributed machine learning, and blockchain. He was the recipient of the Best Paper Award at IEEE Globecom'20.

Laurence T. Yang (Fellow, IEEE) received the B.E. degree in computer science and technology and the B.Sc. degree in applied physics from Tsinghua University, Beijing, China, in 1992, and the Ph.D. degree in computer science from the University of Victoria, Victoria, BC, Canada, in 2006. He is currently a Professor with the School of Computer Science and Technology, Hainan University, Haikou, China, the School of Computer Science and Technology, Huazhong University of Science and Technology, Wuhan, China, and the Department of Computer Science, St. Francis Xavier University, Antigonish, NS, Canada. His research interests include parallel and distributed computing, embedded and ubiquitous computing, and Big Data. His research has been supported by the National Sciences and Engineering Research Council and the Foundation for Innovation.

Lin Cai (Fellow, IEEE) received the M.A.Sc. and Ph. D. degrees in electrical and computer engineering from the University of Waterloo, Waterloo, ON, Canada, in 2002 and 2005, respectively. Since 2005, she has been with the Department of Electrical and Computer Engineering, University of Victoria, Victoria, BC, Canada, and she is currently a Professor. She is a NSERC E.W.R. Steacie Memorial Fellow, and an Engineering Institute of Canada (EIC) Fellow. In 2020, she was elected as a Member of the Royal Society of Canada's College of New Scholars, Artists and Scientists, and a 2020 Star in Computer Networking and Communications by N2Women. Her research interests include in communications and networking, with a focus on network protocol and architecture design supporting emerging multimedia traffic and the Internet of Things. She was the recipient of the Outstanding Achievement in Graduate Studies.

Jia Luo received the Ph.D. degree from the Chongqing University of Posts and Telecommunications, Chongqing, China, in 2020. He is currently a Lecturer with the Chongqing University of Posts and Telecommunications. From April 2018 to April 2019, he is a Visiting Scholar with Carleton University, Ottawa, ON, Canada. His current research interests include UAV communications, mobile edge computing, and artificial intelligence algorithms.

Zhe Zhang (Member, IEEE) received the Ph.D. degree in electrical and computer engineering from Carleton University, Ottawa, ON, Canada in 2019. He is currently an Assistant Professor with the Nanjing University of Posts and Telecommunications (NJUPT), Nanjing, China. Before he joined NJUPT, he was a Research Engineer with Huawei Ottawa Research Center, Ottawa. His research interests include multimedia communications, in-network caching, software-defined networking, and Internet of things.



Chenmeng Wang received the Ph.D. degree in information and telecommunication engineering from the Chongqing University of Posts and Telecommunications, Chongqing, China, in 2018. From 2019 to 2021, he was a Postdoctoral Fellow with the University of Alberta, Edmonton, AB, Canada. From 2015 to 2017, he was a Visiting Ph.D. Student with Carleton University, Ottawa, ON, Canada. He is currently an Associate Professor with Hainan University, Haikou, China. His research interests include small-cell HetNets, multi-access edge computing, massive MIMO systems and resource allocation in mobile networks.