Physical Communication 63 (2024) 102306

Contents lists available at ScienceDirect

Physical Communication

journal homepage: www.elsevier.com/locate/phycom

Full length article

Three-dimensional deep reinforcement learning for trajectory and resource optimization in UAV communication systems☆

Chunlong He [a], Jiaming Xu [a,*], Xingquan Li [b], Zhukun Li [a]

[a] Guangdong Key Laboratory of Intelligent Information Processing, Shenzhen University, Shenzhen, China
[b] Shenzhen Institute of Information Technology, Shenzhen, China

ARTICLE INFO

Keywords:
Multi-user wireless system
UAV
Power control
Trajectory design
PPO
Hierarchical Reinforcement Learning

ABSTRACT

In this paper, we focus on trajectory optimization for multiple three-dimensional unmanned aerial vehicles (UAVs) serving as aerial base stations (BS) to provide wireless coverage for Internet of Things (IoT) devices. These IoT devices are randomly distributed in a three-dimensional region with unknown locations and channel parameters. By optimizing the UAV trajectory and power, we aim to achieve the minimum communication time for all IoT devices. However, the unknown flight time of the UAVs and the IoT device locations make it difficult to formulate an optimization problem between the objective and variables by using traditional convex optimization. To deal with this intractable problem, We propose a solution based on Hierarchical Reinforcement Learning (HRL), specifically modeling it as a Markov Decision Process (MDP) and we divide the training process into two parts, each utilizing the Proximal Policy Optimization (PPO) algorithm for training. Our simulation results show that the proposed HRL offers promising performance for both training and testing phases.

1. Introduction

The unmanned aerial vehicles (UAVs) have seen significant growth due to their high mobility [1]. They have attracted great attention in various fields [2], particularly in wireless communications. Their inherent characteristics such as low costs and flexibility give them the capacity to successfully address issues in conventional terrestrial communications. As a result, UAVs can be utilized as airborne base stations (BS) [3], mobile relays [4,5], mobile edge computing [6,7], emergency search and rescue [8,9] and task offloading [10,11]. UAVs allow systems to provide greater coverage and lower system deployment costs than traditional terrestrial communication systems [12]. In addition, UAVs have the flexibility to exchange information between disaster areas [13].

The trajectory design of UAVs is crucial for fully utilizing their advantages. For instance, in order to maximize the energy efficiency, the UAV utilizes its mobility [14] and finds the most appropriate location to help users communicate [15]. Traditional methods, such as gradient descent and sequential convex approximation have demonstrated excellent performance [16,17] are simple to implement and do not require significant computational resources and expertise compared to deep learning. Nonetheless, the conventional approach for convex optimization is an algorithm that operates continuously. Therefore, if there are alterations in the environmental conditions (such as the user's position), the algorithm must be re-executed. To address this problem, many recent works proposed employing deep reinforcement learning (DRL) to address the joint trajectory design and UAV power allocation problems [18–23]. In [18], the authors have considered hovering and moving cases and utilize DRL to maximize the rate of the total network. In [19], they have studied content delivery in highly congested multi-access cellular networks with data traffic offloaded by edge caching and employ reinforcement learning methods to solve the problem. In [20], the authors have proposed a UAV detection method trajectory based on deep learning. In [21], the authors have integrated UAVs and sixth-generation (6G) communication networks to improve the UAV's energy efficiency using reinforcement learning methods. In [22], the authors have used deep reinforcement learning algorithms to minimize the cost of edge computing for both ground users and UAVs. In [23], the authors have used a heuristic reward function to solve the UAV path planning problem.

The previous research has generally considered only single UAV and focused on optimizing UAV trajectory. Accordingly, we develop a three-dimensional multi-UAV-enabled wireless communication system and specifically address the problem of optimizing the trajectory and power of the UAV when the flight time is unknown. Because the flight

☆ This work was supported in part by the Shenzhen Basic Research Program, China under Grant JCYJ20220531103008018, and Grant 20231120142345001.
* Corresponding author.
E-mail addresses: hclong@szu.edu.cn (C. He), 2100432069@email.szu.edu.cn (J. Xu), lixq@sziit.edu.cn (X. Li), lizhukun777@163.com (Z. Li).

https://doi.org/10.1016/j.phycom.2024.102306
Received 28 December 2023; Accepted 24 January 2024
Available online 2 February 2024

trajectory, resource allocation, user allocation, and coverage of the UAV BS are linked, using traditional methods becomes more complex. In contrast, neural networks possess remarkable fitting capabilities, making them an excellent choice for addressing nonlinear non-convex optimization problems. Once properly trained, they also exhibit strong performance and low time complexity. The contributions of this paper are as follows.

(1) We develop a novel multi-UAV-enabled wireless communication system in a three-dimensional environment, in which we consider minimizing the flight time while meeting the minimum communication rate for all Internet of Things(IoT) devices.

(2) To handle the challenging continuous action space and large-dimensional state space, we present an innovative UAV trajectory optimization algorithm, leveraging the power of Hierarchical Reinforcement Learning (HRL) based on Proximal Policy Optimization (PPO).

(3) We show the performance improvement by comparing our solution to other DRL algorithms, indicating that the proposed HRL outperforms other baseline algorithms.

Paper organizes Section 2 of this paper introduces the multi-UAV communication system. In Section 3, we present an overview of the PPO algorithm's principles and elucidate the framework employed for addressing the trajectory optimization problem. Then, we present the simulation results in Section 4 and provide the paper's conclusion in Section 5.

2. System model and problem formulation

2.1. System model

We consider a system containing M UAVs as BSs and N IoT devices, as shown in Fig. 1. The UAVs have the flexibility to adjust their heights arbitrarily, and IoT devices are dispersed randomly over the area. The UAV is represented by $m \in \mathbb{M}, 1 \le m \le M$ and the IoT devices is represented by $n \in \mathbb{N}, 1 \le n \le N$. We assume that the UAVs only contain the received signal strength indicator (RSSI), not the location data of the ground users. The UAVs chooses all IoT devices that have not finished their communication tasks within the coverage area during each time slot, and the UAVs send information to the selected IoT devices until the minimum data capacity requirements of all IoT devices are met. The optimization goal of this model is to minimize the communication time and flight duration between UAVs and IoT devices. The total time duration T into discrete time slots as $t, 0 \le t \le T$. The coordinate of the nth IoT devices is represented by (x_n, y_n, z_n). And $\boldsymbol{q}_m[t] = (x_m[t], y_m[t])$ to represent the UAV trajectory projected onto the horizontal plane. $\boldsymbol{h}_m[t] = z_m[t]$ to represent the height of the UAV. The flight area of the UAV is restricted to a specific range to ensure it remains within a safe flight range. This range can be represented as $0 \le x_m[t] \le x_{\max}$, $0 \le y_m[t] \le y_{\max}$ and $Z_{\min} \le z_m[t] \le Z_{\max}$.

2.2. Problem formulation

At the tth time slot of the UAV mission, the air-to-ground line-of-sight (LoS) probability between the mth UAV and the nth IoT device is given by

$$P_{Los}(t) = \frac{1}{1 + a\exp(-b(\Phi(t) - a))}, \tag{1}$$

where a and b are constants and are related to the choice of urban environment. $\Phi(t)$ is given by

$$\Phi(t) = \frac{180}{\pi}\arctan(\frac{h_{m,n}[t]}{r_{m,n}[t]}), \tag{2}$$

where $r_{m,n}[t]$ and $h_{m,n}[t]$ represent the horizontal and vertical distances between the UAV and the IoT device, which calculate as follows

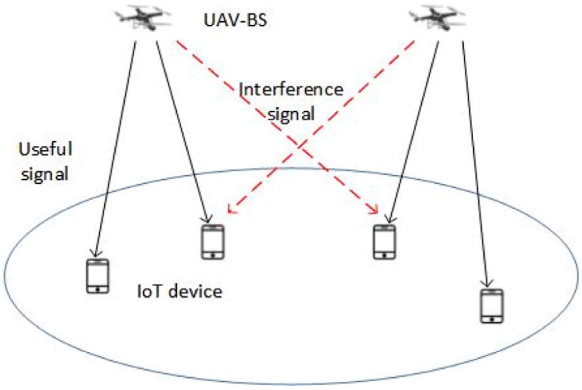

Fig. 1. The system model.

$$r_{m,n}[t] = \sqrt{(x_m[t] - x_n)^2 + (y_m[t] - y_n)^2}, \tag{3}$$

$$h_{m,n}[t] = z_m[t] - z_n. \tag{4}$$

Radio signals lose signal strength due to urban environment masking and scattering in addition to free-space propagation losses. So the path loss model is

$$L_{m,n}(t) = \begin{cases} 20\log(\frac{4\pi f_c d_{m,n}[t]}{c}) + \eta_{LoS}, & \text{LoS link}, \\ 20\log(\frac{4\pi f_c d_{m,n}[t]}{c}) + \eta_{NLoS}, & \text{NLoS link}, \end{cases} \tag{5}$$

where f_c represents the carrier frequency, c represents the speed of light, and $d_{m,n}[t]$ represents the distance between the mth UAV and the nth IoT device, which can be calculated by $d_{m,n}[t] = \sqrt{h^2_{m,n}[t] + r^2_{m,n}[t]}$.

Therefore, the average path loss between the mth UAV and the nth IoT device at the tth time slot can be viewed as

$$\begin{aligned} L(t) =& (20\log(\frac{4\pi f_c d_{m,n}[t]}{c}) + \eta_{LoS})P_{LoS}(t) + \\ & (20\log(\frac{4\pi f_c d_{m,n}[t]}{c}) + \eta_{NLoS})P_{NLoS}(t), \end{aligned} \tag{6}$$

where $P_{LoS}(t)$ and $P_{NLoS}(t)$ are the probability of LoS and non-line-of-sight (NLoS) probabilistic communication, and their conversion relationship is $P_{NLoS}(t) = 1 - P_{LoS}(t)$.

We define the UAV transmission power as $P_m[t]$. In addition, $P^r_{m,n}[t]$ represents the power of the mth UAV received by the nth IoT device at time slot t. It can be calculated by the following equation

$$P^r_{m,n}[t] = P_m[t] - L(t). \tag{7}$$

In order to guarantee the quality of service, it must be ensured that the IoT device receives power exceeding a threshold $P_{\min}$

$$P^r_{m,n}[t] \ge P_{\min}. \tag{8}$$

With the limited battery life and flight time, UAVs need to fly back to origin point in the end. Otherwise, the mission will be deemed as failure. This time slot is represented by t_{end}. so we have,

$$q_m[0] = q_m[t_{end}], \forall m, t_{end} \le T. \tag{9}$$

The UAV can choose to move in each time slot. The equation of motion of the UAV can be expressed as

$$q_m[t+1] = q_m[t] + \begin{bmatrix} \cos(\varphi_m[t]) \\ \sin(\varphi_m[t]) \end{bmatrix} V_m[t], \forall m, t, \tag{10}$$

$$h_m[t+1] = h_m[t] + V^h_m[t], \forall m, t, \tag{11}$$

where $\varphi_m[t]$ represents the heading angle of the UAV and $V_m[t]$ represents the horizontal velocity of the UAV. $V^h_m[t]$ represents the vertical velocity of the UAV. All the parameters have certain range restrictions

and can be mathematically represented by $0 \le \varphi_m[t] \le 2\pi$, $0 \le V_m[t] \le V_{\max}$, $-V^h_{\max} \le V^h_m[t] \le V^h_{\max}$. $V_{\max}$ and $V^h_{\max}[t]$ represents the maximum horizontal and vertical speeds. Moreover the distance between multiple UAVs cannot be too small to prevent collision between UAVs, and the UAV's maximum flying distance within the specified time interval is denoted as $S_{\max}$. The maximum flight altitude of the UAV is $D_{\max}$. Then we have the following equation which can express the above constraint relationship.

$$\left\|q_m[t+1] - q_m[t]\right\| \le S_{\max}, \forall m, t, \tag{12}$$

$$\left\|h_m[t+1] - h_m[t]\right\| \le D_{\max}, \forall m, t, \tag{13}$$

$$\left\|q_j[t] - q_m[t]\right\| > d_{\min}, \forall m, j, t, m \ne j, \tag{14}$$

the signal-to-interference-plus-noise ratio (SINR) at nth IoT device is

$$\gamma_{m,n}[t] = \frac{p^r_{m,n}[t]}{\sum_{j=1, j\ne m}^{M} p^r_{j,n}[t] + \sigma^2}. \tag{15}$$

We establish a set of binary variables that are denoted as $c_{m,n}[t] = 1$. The nth IoT device is served by the mth UAV during time slot t. Otherwise, $c_{m,n}[t] = 0$. We assume that during each time slot t, each IoT device is capable of communicating with only one UAV, as following

$$c_{m,n}[t] \in \{0, 1\}, \forall m, n. \tag{16}$$

If the IoT device is within the coverage range of the UAV and has not completed its data transmission task, then the UAV sends data to the IoT device. The communication rate between the UAV and the IoT device at the tth time slot can be represented as follows

$$r_{m,n}[t] = Blog_2(1 + \gamma_{m,n}[t]), \tag{17}$$

where B is the bandwidth.

The mission of the UAV is divided into two parts, the first part is to complete the communication task of all IoT devices, and the other is back to the origin point autonomously for recharging. The information capacity that each IoT as η_{min}.

For the first part of the task, the collection of sufficient data per IoT device can be expressed as

$$\sum_{t=1}^{t_{end}} \sum_{m=1}^{M} c_{m,n}[t] r_{m,n}[t] \ge \eta_{\min}, \forall n. \tag{18}$$

The formulation of the optimization problem is as follows

$$\min_{\boldsymbol{Q},\boldsymbol{P},\boldsymbol{C},\boldsymbol{H}} t_{end}, \tag{19}$$

$$\text{s.t.} \quad P^r_{m,n}[t] \ge P_{\min}, \tag{19a}$$

$$q_m[0] = q_m[t_{end}], \forall m, \tag{19b}$$

$$\left\|q_j[t] - q_m[t]\right\| > d_{\min}, \forall m, j, t, m \ne j, \tag{19c}$$

$$\left\|q_m[t+1] - q_m[t]\right\| \le S_{\max}, \forall m, t, \tag{19d}$$

$$\left\|h_m[t+1] - h_m[t]\right\| \le D_{\max}, \forall m, t, \tag{19e}$$

$$\sum_{t=1}^{t_{end}} \sum_{m=1}^{M} c_{m,n}[t] r_{m,n}[t] \ge \eta_{\min}, \forall n, \tag{19f}$$

$$c_{m,n}[t] \in \{0, 1\}, \forall m, n. \tag{19g}$$

Due to the unknown locations of IoT devices and channel parameters for UAVs, this optimization problem cannot be solved using conventional algorithms. We use HRL to address the problem and model the system as a Markov decision process (MDP).

3. Hierarchical reinforcement learning

In this section, we attribute problem (19) into a MDP to facilitate the DRL-based solution. We divide the training process of the UAV into two parts,including the horizontal and vertical trajectory that are trained by utilizing PPO algorithm.

3.1. Problem definition

A five-tuple consisting of state, action, transition probability, reward, and discount factor can be used to specify MDP, (S, A, P, Z, γ). The agent firstly observes the current state $s_t \in S$ in time t from the environment, then choose action a_t according to a certain policy $\pi(a|s) = Pr(a_t|s_t)$ and acquire the immediate reward R_t.

We consider the UAV as a MDP the learning decision process of the agent into two subprocesses. Specifically, we ingeniously design the state space and action space corresponding to each subprocess. The state space design, action space design and reward function design are as follows.

(1) State: We assume that the remaining communication data volume of the ith IoT device is denoted as $\eta_i[t]$. N_f represents the number of completed IoT devices tasks. N is the number of the IoT devices within the operating range. Therefore, the state space can be defined as

$$\boldsymbol{S}_{xy}[t] = \left\{ \begin{matrix} \left[\frac{x_1[t]}{X_{\max}}, \frac{y_1[t]}{Y_{\max}}\right], \ldots, \left[\frac{x_M[t]}{X_{\max}}, \frac{y_M[t]}{Y_{\max}}\right] \\ \frac{\eta_1[t]}{\eta_{\max}}, \ldots, \frac{\eta_N[t]}{\eta_{\max}}, \frac{N_f}{N}, \frac{t}{T} \end{matrix} \right\}, \tag{20}$$

$$\boldsymbol{S}_z[t] = \left\{ \begin{matrix} \frac{\eta_1[t]}{\eta_{\max}}, \ldots, \frac{\eta_N[t]}{\eta_{\max}}, \frac{N_f}{N}, \frac{t}{T} \\ z_1[t], \ldots, z_m[t] \end{matrix} \right\}, \tag{21}$$

where define the maximum communication data volume as $\eta_{\max}$. $S_{xy}[t]$ is the state space for training horizontal trajectory optimization, and $S_z[t]$ is the state space for training vertical trajectory. At the same time, $|S_{xy}|_{dim} = 2M + N + 2$, $|S_z|_{dim} = M + N + 2$.

(2) Action: The design of the action vector determines the flight actions of the UAV, so the action space should include all possible actions in the environment. We define the action space to encompass speed, flight direction, and UAV transmission power. Speed control enables it to regulate its movement, while flight direction determines its spatial orientation and path. Additionally, the ability to adjust UAV transmission power plays a pivotal role in optimizing communication and data transfer performance.

$$\boldsymbol{A}_{xy}[t] = \left\{ \begin{matrix} \frac{V_1[t] - V_{\max}/2}{V_{\max}/2}, \ldots, \frac{V_M[t] - V_{\max}/2}{V_{\max}/2} \\ \frac{\varphi_1[t] - \pi}{\pi}, \ldots, \frac{\varphi_M[t] - \pi}{\pi} \\ \frac{p_1[t] - P_{\max}/2}{P_{\max}/2}, \ldots, \frac{p_M[t] - P_{\max}/2}{P_{\max}/2} \end{matrix} \right\}. \tag{22}$$

$$\boldsymbol{A}_z[t] = \left\{ \begin{matrix} \frac{V^h_1[t] - V^h_{\max}/2}{V^h_{\max}/2}, \ldots, \frac{V^h_M[t] - V^h_{\max}/2}{V^h_{\max}/2} \\ \frac{p_1[t] - P_{\max}/2}{P_{\max}/2}, \ldots, \frac{p_M[t] - P_{\max}/2}{P_{\max}/2} \end{matrix} \right\}. \tag{23}$$

$A_{xy}[t]$ is the action space for training horizontal trajectory optimization, and $A_z[t]$ is the action space for training vertical trajectory. Our objective is to minimize the UAV's flight time. Therefore, in terms of speed, we exclusively employ either the maximum UAV velocity or 0 m/s. All features are normalized to fall within the range of [−1,1] before being fed into the neural network. Also, $|A_{xy}|_{dim} = 3M$ and $|A_z|_{dim} = 2M$.

(3) Reward: The design of the reward function directly impacts the neural network's training as it guides the UAV in learning which actions to take in specific states. Both processes are designed with the same reward function since we share a common optimization goal. Five sub-reward parts $\{Z_1(t), Z_2(t), Z_3(t), Z_4(t), Z_5(t)\}$ are proposed.

$Z_1(t)$ is a limit on the distance between UAVs for constraint (19c). $Z_2(t)$ is a reward for finish all tasks and back to the origin. $Z_3(t)$ indicates that the earlier a UAV completes its communication task, the

greater rewards it can receive. $Z_4(t)$ is an intermediate reward that needs to be executed to go back to the take-off point after the UAVs have completed the communication task. $Z_5(t)$ is the penalty for not completing the task at the maximum time for constraint (19f). We also define a time node T_1, which signifies the moment when the UAV starts contemplating its return to the point of origin, and $N_c[t]$ represents the number of the IoT devices who complete communication tasks at the tth time slot.

$$Z_1(t) = \sum_{m=1, m\neq j}^{M} \sum_{j=1}^{M} \xi_{m,j}[t], \tag{24}$$

$$Z_2(t) = T - t_{end}, \tag{25}$$

$$Z_3(t) = N_c[t](T - t), \tag{26}$$

$$Z_4(t) = \begin{cases} \sum_{m=1}^{M} F_m[t](t - T_1) & T_1 \le t \le t_{end}, \\ 0 & \text{otherwise}, \end{cases} \tag{27}$$

$$Z_5(t) = \begin{cases} \sum_{n}^{N} (\eta_{\min} - \eta_n) & \text{if } \eta_n < \eta_{\min} \text{ and } t = t_{end}, \\ 0 & \text{otherwise}, \end{cases} \tag{28}$$

where $F_m[t]$ denotes the distance of mth UAV between origin at time t. η_n is the communication task that the nth IoT device has not completed by the final time slot. One has

$$\xi_{m,j}[t] = \begin{cases} 1, & G_{m,j}[t] \le d_{\min}, \\ 0, & \text{otherwise}, \end{cases} \tag{29}$$

where $G_{m,j}[t]$ means the distance between mth and jth UAV. In conclusion, the whole representation of reward is as

$$Z(t) = \lambda_1 Z_1(t) + \lambda_2 Z_2(t) + \lambda_3 Z_3(t) + \lambda_4 Z_4(t) + \lambda_5 Z_5(t). \tag{30}$$

3.2. PPO

PPO is a DRL algorithm with the actor–critic structure,in which the neural networks are typically divided into two parts: one is the critic function, used to evaluate the value of the current state. The other part is the actor function, which serves as the policy function neural network for selecting actions. It includes two components, the sample actor network and the target actor network, which can be represented as $\pi_\theta(a_t|s_t)$ and $\pi_{\theta'}(a_t|s_t)$. In contrast to the conventional reinforcement learning, PPO requires a critic function to evaluate the worth of the current occurrence of this state, which can generally be fitted using a neural network, also known as a critic neural network. We use $V_\omega(s_t)$ to denote the critic neural network. Since updating neural networks involves calculating gradients, the gradient of the actor neural network can be computed using the following equation,

$$J^{\theta'}(\theta) = \sum_{(s_t,a_t)} \min \left(\begin{array}{c} \frac{\pi_\theta(a_t|s_t)}{\pi_{\theta'}(a_t|s_t)} A^{\theta'}(s_t, a_t), \\ clip(\frac{\pi_\theta(a_t|s_t)}{\pi_{\theta'}(a_t|s_t)}, 1-\varepsilon, 1+\varepsilon) A^{\theta'}(s_t, a_t) \end{array} \right), \tag{31}$$

where $A^{\theta'}(s_t, a_t)$ represents advantage function. It is determined by discount factors β and δ_t and can express as

$$A^{\theta'}(s_t, a_t) = \delta_t + \beta\delta_{t+1} + \cdots + \beta^{T-t}\delta_T, \tag{32}$$

$$\delta_t = R(s_t, a_t|\theta') + \beta v_\omega(s_{t+1}) - v_\omega(s_t). \tag{33}$$

For the critic function, the loss formula can be expressed as

$$loss_c = E[\delta_t^2] \approx \sum_t \delta_t^2. \tag{34}$$

The algorithm for the training process of UAV trajectory planning based on HRL algorithm is summarized in Algorithm 1 (see Fig. 2).

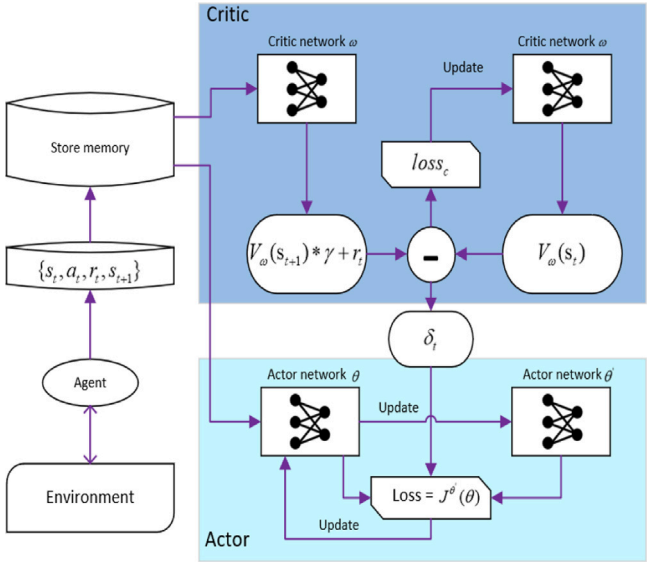

Fig. 2. The framework of Proximal Policy Optimization (PPO) algorithm.

Algorithm 1 Proposed HRL based method

Require: $M, N, T, V_{\max}^h, V_{\max}, P_{\max}$
Initialize policy network parameters θ_{xy}, value function parameters ω_{xy}
Initialize: $\boldsymbol{S}_{xy}$, $\boldsymbol{A}_{xy}$
repeat
  for t=1 to T **do**
    get $\boldsymbol{S}_{xy}[t]$ in (26),
    select $\boldsymbol{A}_{xy}[t]$ according to actor network $\pi_{\theta_{xy}}$
    input $\boldsymbol{A}_{xy}[t]$ to get $\boldsymbol{S}_{xy}[t+1]$ and $Z(t)$
    update $t = t+1$
    if UAVs complete the mission $(t = t_{end})$ **then**
      Break out of the loop
    end if
  end for
  Update the actor: $\theta_{xy} \leftarrow \theta_{xy} + \alpha_{\text{actor}} \nabla_\theta J_{\text{actor}}(\theta_{xy})$
  Update the critic: $\omega_{xy} \leftarrow \omega_{xy} + \alpha_{\text{critic}} \nabla_\omega J_{\text{critic}}(\omega_{xy})$
until
output $\boldsymbol{S}_{xy}$ and $\boldsymbol{A}_{xy}$
Initialize policy network parameters θ_z, value function parameters ω_z
repeat
  for t=1 to T **do**
    get $\boldsymbol{S}_{xy}[t]$ and $\boldsymbol{A}_{xy}[t]$ to change the horizontal trajectory of the UAV
    get $\boldsymbol{S}_z[t]$ in (27)
    select $\boldsymbol{A}_z[t]$ according to actor network $\pi_{\theta_2'}(a_t|s_t)$
    input $\boldsymbol{A}_z[t]$ to get $\boldsymbol{S}_z[t+1]$ and $Z(t)$
    update $t = t+1$
    if UAVs complete the mission $(t = t_{end})$ **then**
      Break out of the loop
    end if
  end for
  Update the actor: $\theta_z \leftarrow \theta_z + \alpha_{\text{actor}} \nabla_\theta J_{\text{actor}}(\theta_z)$
  Update the critic: $\omega_z \leftarrow \omega_z + \alpha_{\text{critic}} \nabla_\omega J_{\text{critic}}(\omega_z)$
until
output $\boldsymbol{S}_z$ and $\boldsymbol{A}_z$
Ensure: Optimal flight trajectory and power

Table 1
Simulation parameters.

Parameters	Value
λ_1	−10
λ_2	0.06
λ_3	0.005
λ_4	−0.00005
λ_5	0.015
f_c	2.5 (Ghz)
$P_{\max}$	26 (dBm)
$V_{\max}$	50 (m/s)
$V^h_{\max}$	20 (m/s)
$Z_{\max}$	400 (m)
$Z_{\min}$	150 (m)
$T_{\max}$	160 (s)
$\eta_{\min}$	36 (bps/Hz)

4. Numerical results

In this part, a detailed description of the simulation parameters and settings is provided. Next, HRL is employed to assess its convergence and stability performance using different reward functions in comparison to various algorithms. The investigation extends to the analysis of the flight trajectory and power consumption of UAVs to substantiate the effectiveness of the HRL algorithm. Following a thorough comparison of the time required by UAVs to complete tasks in diverse scenarios, we delve into considering the algorithm's advantages and disadvantages. Lastly, the upper limit of the algorithm is tested within a high-density scene.

we assume that $M = 2$ UAVs and $N = 10$ IoT devices are randomly coexist within a 2D area of (1.5×1.5) km^2. The altitude of the UAVs is $z_m[0] = 200$ m. Other parameters are shown in Table 1. The complexity of DRL algorithms is intricately tied to the number of multiplicative operations performed during each iteration. So for PPO, the computational time complexity of each time step of a fully connected deep neural network is denoted by $O(\sum_{l_i=1}^{L} l_i * l_{i+1})$, where L denotes the number of network layers, l_i denotes the number of neurons in the ith hidden layer. The HRL algorithm requires two rounds of PPO training. In the PPO algorithm, the critic network takes the sum of the state and action spaces as input and outputs the Q-value, while the actor network takes the state space as input and outputs the action space. Each network has two hidden layers with 256 neurons each.

We set hyper parameters $\varepsilon = 0.2, \lambda = 3 \times 10^{-4}, \gamma = 0.99, \tau = 0.001$. DDPG and A2C have the same reward function and learning rate settings as the PPO. Due to the limitations of algorithms, A2C and DDPG can only play a role in two-dimensional cases, so we only consider these two algorithms in two-dimensional cases.

Fig. 3 shows the training processes of the PPO, PPO without power control (PPONP) ($P_m(t) = P_{\max}$) and other DRL algorithms in terms of horizontal trajectories. We can observe that the PPO algorithm produces greater rewards. Both DDPG and PPO converge at about 10000 episode, however the A2C algorithm fluctuates more. The PPO algorithm has better performance than the other DRL algorithms, their rewards increase rapidly at first, but all of them fall into local optimal solutions as the number of iterations increases. To enhance sample efficiency, the PPO algorithm introduces importance sampling. Additionally, it employs a clipping constraint between the old and new policy to prevent excessive deviations to ensure smooth convergence and reinforcing training process stability. This characteristic enhances the reliability and effectiveness of PPO when addressing reinforcement learning tasks. Compared with the PPONP, the PPO algorithm can adapt the transmission power to minimize interference from other UAVs to bring in additional rewards. Consequently, the overall reward achieved by PPO may surpass that of PPONP.

In Fig. 4, it can be observed that the convergence speed is faster compare with training the horizontal trajectory of the UAV. This is mainly because the state space is relatively smaller when training of

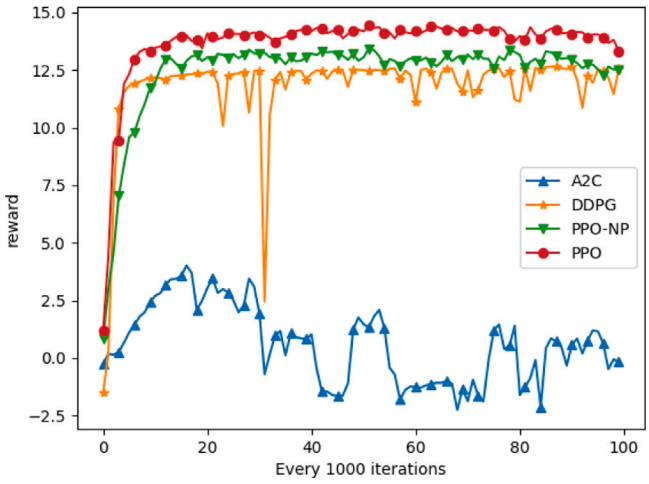

Fig. 3. Average rewards of different DRL algorithms in terms of horizontal trajectories.

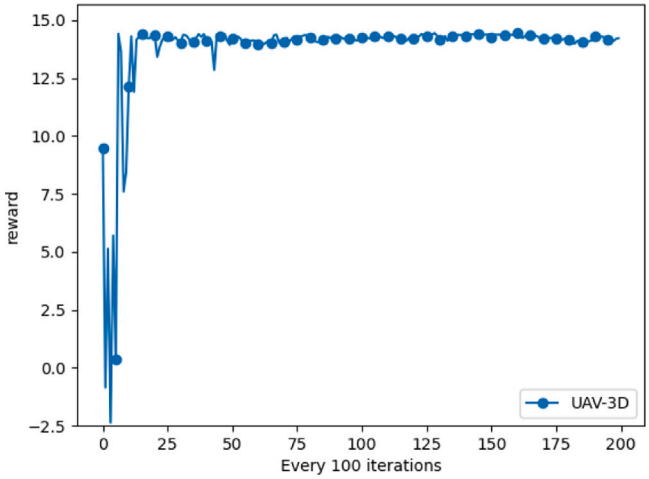

Fig. 4. Rewards of the vertical trajectory of the UAV.

the vertical trajectory. Additionally, with the foundation of horizontal training, the UAV can obtain rewards and reduce a significant amount of training time.

In Figs. 5 and 6, we show the flight trajectory of the UAV. The UAVs are $(0, 0, 150)$ and $(1500, 1500, 150)$ at the initial positions. Since the UAVs do not have the location and channel parameters of the IoT device, the trajectory of the UAVs are not likely to be smooth. In Fig. 5, it can be seen that the two UAVs fly over the IoT devices as much as possible to ensure the communication rate of the IoT devices, and they tend to cover more IoT devices simultaneously during the flight to save service time. When the UAVs finish communicating with all IoT devices, they return to the origin point at the maximum speed. Additionally, from Fig. 6, it can be observed that, to avoid interference, two UAVs try to keep a considerable distance from each other as much as possible.

Fig. 7 shows the transmission power of the UAVs, when the two UAVs communicate with two nearby IoT devices, such as 19s~25s, The power transmission reduces co-channel interference to improve the reception rate. Because of this, interference signals from other UAVs can only be reduced in the absence of transmission power control by modifying the UAV trajectory to move away from other UAVs. We consider that jointly optimizing the transmission power of UAVs provides more flexibility and higher throughput.

In Fig. 8, we can derive several important observations. First and foremost, the A2C algorithm demonstrates limited effectiveness in a

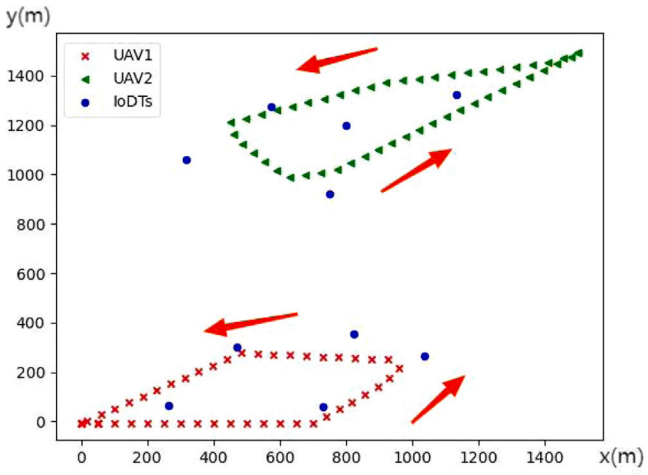

Fig. 5. The horizontal trajectory of UAV.

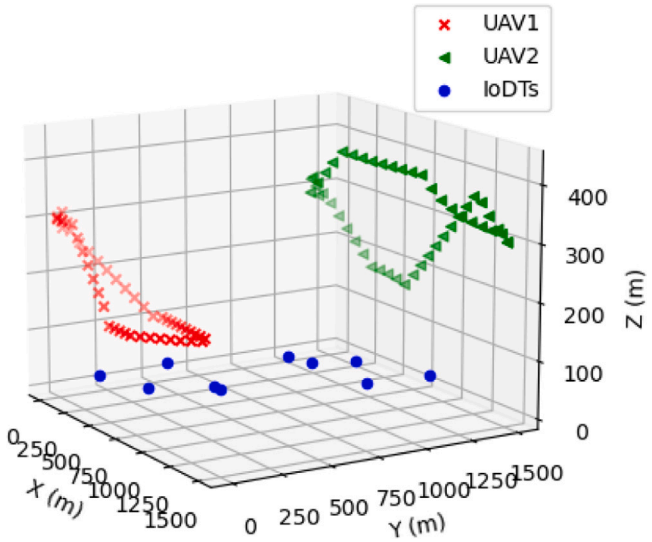

Fig. 6. The vertical trajectory of UAV.

continuous environment. The completion time for a single UAV is notably longer than that for multiple UAVs. The DDPG algorithm, it exhibits poor stability due to the use of a deterministic policy. This means that the algorithm directly outputs a specific action for a given state, leading to sensitivity to minor changes in the environment or initial conditions, especially in the context of continuous action spaces. Finally, comparing the HRL and HRL-NP shows that more flexibility can be obtained, resulting in less time by using power control.

In Fig. 9, with the increase of IoT devices, the proposed HRL algorithm . The escalation of the number of IoT devices implies an exponential rise in complexity. When the number of IoT reaches 14, the A2C algorithm fails to handle such a vast amount of data and causes a breakdown. Due to the use of a deterministic policy, The DDPG algorithm is susceptible to interference from users and leads to local optima. In contrast, our proposed HRL algorithm effectively reduces the complexity introduced by the growing number of IoT devices through a layered training process. It is evident that the HRL algorithm proposed in this paper performs well even in scenarios with a high number of IoT devices, exhibiting high algorithmic efficacy and proving to be more suitable for practical wireless communication service scenarios.

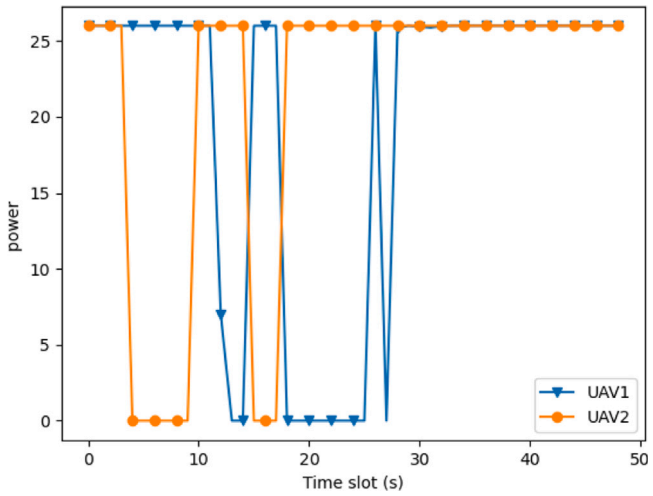

Fig. 7. UAV transmission power versus time for HRL.

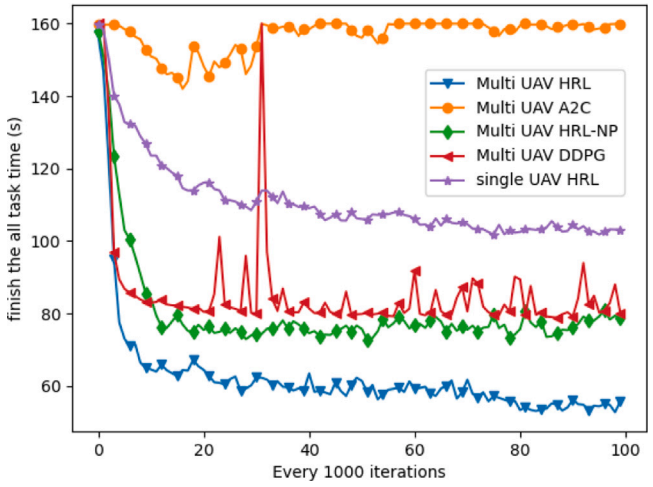

Fig. 8. The time required for the UAV to complete all tasks.

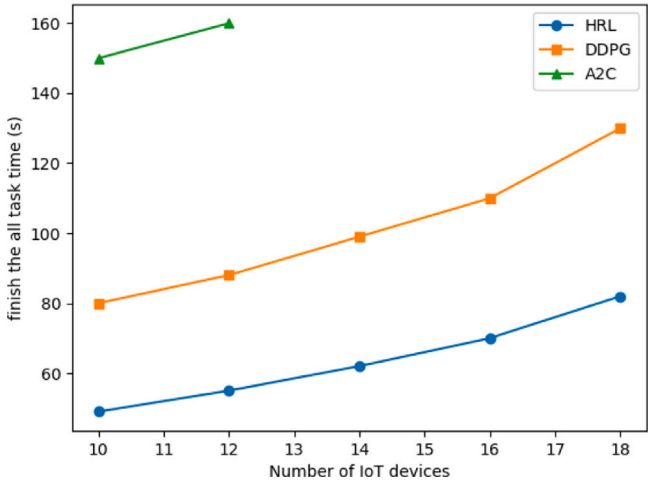

Fig. 9. Completion time variation with changing IoT devices count.

5. Conclusion

In this paper, we aimed to minimize the total time while achieving the minimum communication rate. This was achieved through the optimization of UAV trajectories and transmission power without prior knowledge of the IoT device locations and channel parameters. Regardless of the specific channel models employed, our proposed approach relies solely on the RSSI of the IoT devices.

In order to address the non-convex optimization problem, we modeled it as an MDP and adopt a HRL. Additionally, we conduct a comparative analysis with other DRL algorithms. The numerical results unambiguously illustrated the superior convergence and performance aspects of our proposed HRL algorithm compared to other DRL-based solutions.

CRediT authorship contribution statement

Chunlong He: Writing – review & editing. **Jiaming Xu:** Writing – original draft. **Xingquan Li:** Writing – review & editing. **Zhukun Li:** Conceptualization, Writing – review & editing.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

The authors are unable or have chosen not to specify which data has been used.

References

[1] Y. Zeng, R. Zhang, T.J. Lim, Wireless communications with unmanned aerial vehicles: Opportunities and challenges, IEEE Commun. Mag. 54 (5) (2016) 36–42.

[2] M. Mozaffari, W. Saad, M. Bennis, Y.-H. Nam, M. Debbah, A tutorial on UAVs for wireless networks: Applications, challenges, and open problems, IEEE Commun. surv. Tutor. 21 (3) (2019) 2334–2360.

[3] S. Fang, G. Chen, Y. Li, Joint optimization for secure intelligent reflecting surface assisted UAV networks, IEEE Wirel. Commun. Lett. 10 (2) (2020) 276–280.

[4] Z. Na, C. Ji, B. Lin, N. Zhang, Joint optimization of trajectory and resource allocation in secure UAV relaying communications for Internet of Things, IEEE Internet Things J. 9 (17) (2022) 16284–16296.

[5] L. Wang, B. Li, UAV-enabled reliable mobile relaying under the time-varying Rician fading channel, Alex. Eng. J. 64 (2023) 771–783.

[6] T. Zhang, Y. Xu, J. Loo, D. Yang, L. Xiao, Joint computation and communication design for UAV-assisted mobile edge computing in IoT, IEEE Trans. Ind. Inform. 16 (8) (2019) 5505–5516.

[7] Y. Xu, T. Zhang, J. Loo, D. Yang, L. Xiao, Completion time minimization for UAV-assisted mobile-edge computing systems, IEEE Trans. Veh. Technol. 70 (11) (2021) 12253–12259.

[8] S.H. Alsamhi, A.V. Shvetsov, S. Kumar, S.V. Shvetsova, M.A. Alhartomi, A. Hawbani, N.S. Rajput, S. Srivastava, A. Saif, V.O. Nyangaresi, UAV computing-assisted search and rescue mission framework for disaster and harsh environment mitigation, Drones 6 (7) (2022) 154.

[9] D. Li, X. Yang, F. Zhou, D. Wang, N. Al-Dhahir, Mode adaptive secure UAV relay transmissions, IEEE Trans. Green Commun. Netw. 7 (2) (2023) 787–799.

[10] A.M. Seid, G.O. Boateng, B. Mareri, G. Sun, W. Jiang, Multi-agent DRL for task offloading and resource allocation in multi-UAV enabled IoT edge network, IEEE Trans. Netw. Serv. Manag. 18 (4) (2021) 4531–4547.

[11] A.M. Seid, G.O. Boateng, S. Anokye, T. Kwantwi, G. Sun, G. Liu, Collaborative computation offloading and resource allocation in multi-UAV-assisted IoT networks: A deep reinforcement learning approach, IEEE Internet Things J. 8 (15) (2021) 12203–12218.

[12] S. Ahmed, M.Z. Chowdhury, Y.M. Jang, Energy-efficient UAV-to-user scheduling to maximize throughput in wireless networks, IEEE Access 8 (2020) 21215–21225.

[13] N. Zhao, W. Lu, M. Sheng, Y. Chen, J. Tang, F.R. Yu, K.-K. Wong, UAV-assisted emergency networks in disasters, IEEE Wirel. Commun. 26 (1) (2019) 45–51.

[14] J. Xu, Y. Zeng, R. Zhang, UAV-enabled wireless power transfer: Trajectory design and energy optimization, IEEE Trans. Wirel. Commun. 17 (8) (2018) 5092–5106.

[15] P. Li, J. Xu, Fundamental rate limits of UAV-enabled multiple access channel with trajectory optimization, IEEE Trans. Wireless Commun. 19 (1) (2019) 458–474.

[16] Y. Cai, Z. Wei, R. Li, D.W.K. Ng, J. Yuan, Joint trajectory and resource allocation design for energy-efficient secure UAV communication systems, IEEE Trans. Commun. 68 (7) (2020) 4536–4553.

[17] Q. Wu, Y. Zeng, R. Zhang, Joint trajectory and communication design for multi-UAV enabled wireless networks, IEEE Trans. Wireless Commun. 17 (3) (2018) 2109–2121.

[18] J. Ji, K. Zhu, L. Cai, Trajectory and communication design for cache-enabled UAVs in cellular networks: A deep reinforcement learning approach, IEEE Trans. Mob. Comput. (2022).

[19] P. Karmakar, V.K. Shah, S. Roy, K. Hazra, S. Saha, S. Nandi, Reliable backhauling in aerial communication networks against UAV failures: A deep reinforcement learning approach, IEEE Trans. Netw. Serv. Manag. 19 (3) (2022) 2798–2811.

[20] C. Wang, J. Tian, J. Cao, X. Wang, Deep learning-based UAV detection in pulse-Doppler radar, IEEE Trans. Geosci. Remote Sens. 60 (2022) 1–12.

[21] J. Zhang, W. Ding, Y. Luo, Y. Wang, C. Wang, J. Xiao, Joint trajectory and power control design for UAV anti-jamming communication network, in: 2022 4th International Conference on Advances in Computer Technology, Information Science and Communications, CTISC, Suzhou, China, 2022, pp. 1–6.

[22] Z. Ning, Y. Yang, X. Wang, Q. Song, L. Guo, A. Jamalipour, Multi-agent deep reinforcement learning based UAV trajectory optimization for differentiated services, IEEE Trans. Mob. Comput. (2023) 1–17.

[23] C. Qi, C. Wu, L. Lei, X. Li, P. Cong, UAV path planning based on the improved PPO algorithm, in: 2022 Asia Conference on Advanced Robotics, Automation, and Control Engineering, ARACE, 2022, pp. 193–199.

Chunlong He received the M.S. degree in communication and information science from Southwest Jiao tong University, Chengdu, China, in 2010 and the Ph.D. degree from Southeast University, Nanjing, China, in 2014. From September 2012 to September 2014, he was a Visiting Student with the School of Electrical and Computer Engineering, Georgia Institute of Technology, Atlanta, GA, USA. Since 2015, he has been with the College of Information Engineering, Shenzhen University, where he is currently an Associate Professor. His research interests include communication and signal processing, green communication systems, channel estimation algorithms, and limited feedback techniques. Dr. He is a member of the Institute of Electronics, Information, and Communication Engineering.

Jiaming Xu finished his undergraduate degree in Hunan University of Science and Technology, Hunan, China, in 2021. He is currently pursuing the postgraduation degree with the Shenzhen University Shenzhen China. His research interests include wireless communication networks, resource allocation, and reinforcement learning.

Xingquan Li received his Ph.D. degree from the College of Electronics and Information Engineering, Shenzhen University, Shenzhen, China, in 2019., He was a Postdoctoral Researcher with the College of Electronics and Information Engineering, Shenzhen University. He joined Communication Group, Queen Mary University of London, London, U.K., in 2019, as a Visiting Scholar, where he researched the green communications and resource allocation. He is currently a Lecturer with the School of Microelectronics, Shenzhen Institute of Information Technology, Shenzhen. His research interests include cooperative communications, green communications, and resource allocation.

Zhukun Li received the M.S. degree in the College of Information Engineering, Shenzhen University, Shenzhen, China. His research interests include wireless communication networks, resource allocation, and machine learning algorithms.