electronics

MDPI

Article

Trajectory Planning for UAV-Assisted Data Collection in IoT Network: A Double Deep Q Network Approach

Shuqi Wang [1,2], Nan Qi [1,2,*], Hua Jiang [1,2], Ming Xiao [3], Haoxuan Liu [1,2], Luliang Jia [4] and Dan Zhao [1]

[1] College of Electronic and Information Engineering, Nanjing University of Aeronautics and Astronautics, Nanjing 210000, China; nuaalhx@nuaa.edu.cn (H.L.); zzdd11@nuaa.edu.cn (D.Z.)
[2] National Mobile Communications Research Laboratory, Southeast University, Nanjing 210096, China
[3] Department of Information Science and Engineering, School of Electrical Engineering and Computer Science, Royal Institute of Technology, 114 28 Stockholm, Sweden
[4] School of Space Information, Space Engineering University, Beijing 101416, China
* Correspondence: qinan@nuaa.edu.cn

Abstract: Unmanned aerial vehicles (UAVs) are becoming increasingly valuable as a new type of mobile communication device and autonomous decision-making device in many application areas, including the Internet of Things (IoT). UAVs have advantages over other stationary devices in terms of high flexibility. However, a UAV, as a mobile device, still faces some challenges in optimizing its trajectory for data collection. Firstly, the high complexity of the movement action and state space of the UAV's 3D trajectory is not negligible. Secondly, in unknown urban environments, a UAV must avoid obstacles accurately in order to ensure a safe flight. Furthermore, without a priori wireless channel characterization and ground device locations, a UAV must reliably and safely complete the data collection from the ground devices under the threat of unknown interference. All of these require the proposing of intelligent and automatic onboard trajectory optimization techniques. This paper transforms the trajectory optimization problem into a Markov decision process (MDP), and deep reinforcement learning (DRL) is applied to the data collection scenario. Specifically, the double deep Q-network (DDQN) algorithm is designed to address intelligent UAV trajectory planning that enables energy-efficient and safe data collection. Compared with the traditional algorithm, the DDQN algorithm is much better than the traditional Q-Learning algorithm, and the training time of the network is shorter than that of the deep Q-network (DQN) algorithm.

Keywords: UAV; trajectory planning; deep reinforcement learning; double deep Q-network (DDQN)

Citation: Wang, S.; Qi, N.; Jiang, H.; Xiao, M.; Liu, H.; Jia, L.; Zhao, D. Trajectory Planning for UAV-Assisted Data Collection in IoT Network: A Double Deep Q Network Approach. *Electronics* **2024**, *13*, 1592. https://doi.org/10.3390/electronics13081592

Academic Editors: Felipe Jiménez and Fabio Grandi

Received: 28 February 2024
Revised: 15 April 2024
Accepted: 17 April 2024
Published: 22 April 2024

1. Introduction

1.1. Background

In recent years, the use of unmanned aerial vehicles (UAVs) as airborne base stations to assist in offloading hotspots in existing ground communication infrastructures and cellular networks has been recognized as a promising candidate technology. UAV-assisted communication, when combined with technologies such as the fifth-generation (5G) networks [1–3] and airborne self-organizing networks [4–6], has the potential to provide Internet of Things (IoT) services from high altitudes, creating an airborne domain for the IoT. In certain geographic regions where operators may not be able to afford to build a cellular infrastructure (e.g., a base station), as an alternative, UAVs can lower the communication costs while performing tasks such as collecting or transmitting data to ground-based IoT devices [7,8]. For instance, UAV-assisted cellular communication technology can efficiently restore wireless services after unexpected damage to facilities, such as from natural disasters (e.g., earthquakes, volcanic eruptions, and floods) or in hotspot areas (e.g., sports stadiums and outdoor events), where ground-based cellular stations are insufficient [9,10]. Compared to traditional satellite relays, UAVs fly at lower altitudes when acting as wireless communication providers, resulting in a higher optical resolution

than that of traditional satellites. UAVs have lower maintenance costs and are typically tens or even hundreds of times less expensive than satellites [11]. Furthermore, UAVs are more flexible than traditional satellites, which can only fly according to preset orbits. UAVs can be deployed in a specific area according to demand to fulfill wireless communication tasks [12].

However, as the IoT networks expand in scalability and system design complexity, collecting data from IoT devices while maintaining stable and superior network performance becomes increasingly challenging. In response to this urgent need, UAVs may be an effective solution, due to their high mobility and flexibility. According to the literature [13], UAVs are increasingly being used to collect data from remote sensors. In cases where UAVs assist in information dissemination or data collection, they can collect data sustainably and cost-effectively, as they are equipped with ground-based wireless sensor networks [14]. However, UAVs encounter several challenges, such as a restricted battery capacity, a limited flight time [15,16] and flight altitude, and the impact of malevolent external interference on the communication link between the UAV and the user [17]. Therefore, it is important to revisit how to ensure that UAVs efficiently perform data collection tasks in complex and realistic environments while avoiding obstacles and interferences.

1.2. Related Work and Contributions

With the rapid advancement of UAV-assisted communication technology, traditional mathematical planning algorithms have demonstrated notable efficacy [18]. For instance, authors [19] have leveraged cellular UAVs to ensure stable connections during missions while minimizing the time taken to reach the destination, which was achieved through the implementation of convex optimization and graph theory techniques. Their results indicated reduced mission completion times and an enhanced signal-to-noise ratio (SNR) throughout the missions. Additionally, researchers [20] have identified the following three distinct phases in the trajectory control process: trajectory generation, trajectory correction, and trajectory smoothing. They proposed an ant-colony-based algorithm for initial trajectory generation and an effective collision avoidance scheme for the flight trajectory of UAVs. Another study [21] proposed a method to address the non-convex problem of task assignment, power allocation, and UAV trajectory in wireless communication services. By employing the block coordinate descent method, the original problem was decomposed into two sub-problems, which were subsequently solved iteratively using Lagrange bifurcation and successive convex approximation techniques. However, it is important to note that the computation time of these algorithms may grow exponentially with the scenario size, and they may not be fully adaptable to the increasingly complex scalable wireless network environment.

The application of machine learning techniques to UAV communications has recently gained attention. Reinforcement learning, a model-free algorithmic framework, has been proposed as an alternative to traditional algorithms. This approach does not require the modeling of specific environment feature parameters and can train strategies from trial and error. It has practical significance for UAV trajectory planning and wireless communication system optimization. For instance, in [22], the authors optimized UAV trajectories and power allocation to maximize the fairness of throughput between sensor nodes. In [23], the authors proposed a deep reinforcement learning (DRL)-based framework that uses convolutional neural networks for feature extraction and deep Q-network (DQN) algorithms for decision making to design energy-efficient remote sensing routes for UAVs. The experimental results from [24] demonstrate that the algorithm is significantly more efficient. This study utilizes a network model of the central layer and environmental information and processes the environmental layer through convolution. However, the works only examine the flight trajectory of UAVs at a certain altitude, ignoring the complexity of the 3D environment and the external unknown interference.

In [25], the authors proposed a DRL approach to minimize the task completion time of cellular-connected UAVs while maintaining good cellular network connectivity. In [26],

the authors proposed a dual Q-learning approach to solve optimization problems involving UAV trajectories under continuous time constraints. The authors of [27] performed coordination between UAVs to avoid collisions. This coordination was achieved through a sense-transmit protocol. The main objective is to determine the optimal motion trajectory through a decentralized Q-learning algorithm. This algorithm reduces the convergence time and ensures an efficient transmission of sensor data. However, in practical scenarios, variations in building height can obstruct the flight path of a UAV, necessitating adjustments in altitude. Conversely, heightened interference is encountered when UAVs approach jamming devices, prompting maneuvers to different altitudes in order to mitigate jamming effects and improve the communication environment. It is worth noting that few published articles have considered the impact of obstacles and jammers on channel quality when planning 3D UAV trajectories.

In contrast to the preceding research endeavors, our study encompasses a heightened level of realism by incorporating a sophisticated environmental model, wherein the collective presence of multiple obstacles and jammers contributes to the degradation of the communication link between the UAV and the ground-based IoT devices. Our objective is the maximization of throughput through the strategic optimization of the UAV's three-dimensional flight trajectory, at a low UAV power consumption, while effectively accomplishing data acquisition from the devices and ensuring the safety of UAV flight operations. The proposed algorithm's performance is verified through extensive simulations.

The main contributions of this paper are as follows:

- Our work considers a complex and realistic urban environment in order to study the effects of obstacles and jammers on 3D UAV trajectory planning. Particularly, during the simulation phase, we randomly generate the positions of jammers and ground devices for each iteration, which makes the scenario more uncertain and complicates the design of the Markov decision process (MDP).
- In this paper, the environmental information is not predetermined, and the UAV dynamically senses and navigates around the obstacles in real time using onboard sensors such as cameras. It also learns from historical environmental information obtained from a memory bank to speed up its decision making.
- To address the problem of the limited computing power of UAVs, we developed a DDQN-based UAV trajectory optimization algorithm. The algorithm sets the reward value according to the scene and converges faster. We also provide flight results under different scene parameters and comparison experiments of various reinforcement learning algorithms to support our view. The article fully demonstrates the simulation experiments and algorithm comparisons that validate the effectiveness and superiority of our approach.

The remainder of this paper is organized as follows: In Section 2, the system model and problem description are outlined. Section 3 details the DDQN-based algorithm for trajectory optimization in data collection scenarios. The experimental and simulation results for trajectory optimization are presented in Section 4. The conclusions and future work are discussed in Sections 5 and 6, respectivley.

2. System Model and Problem Formulation

We consider a smart urban setting in Figure 1, where a UAV operates within an unlicensed spectrum band to collect data from a collection of $\mathcal{U} = \{1, ..., U\}$ stationary ground-based IoT devices dispersed across a designated area. In this area, there may be a set of $\mathcal{J} = \{1, ..., J\}$ static ground directional jammers (for example, Wi-Fi that shares an unlicensed spectrum band with the UAV).

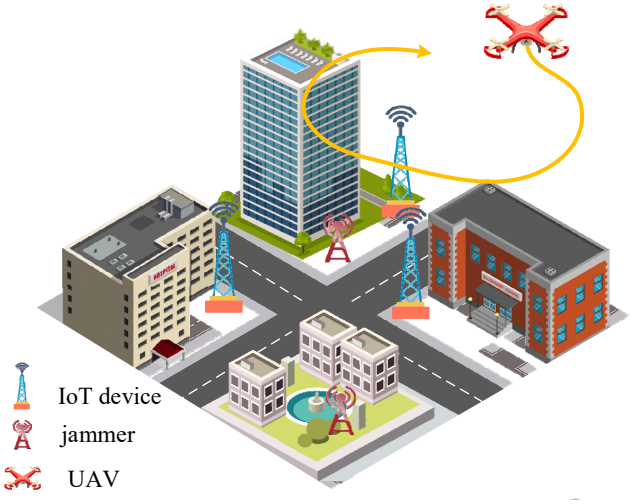


Figure 1. Scenario of UAV data collection.

The maximum duration of a UAV mission is denoted by T, during which the UAV optimal trajectory is designed to maximize data collection from ground IoT devices. For the sake of easy illustration, we assume that T is discretized into equal N time intervals. The UAV position at the time step n is denoted by $\mathbf{q}_n = [x_n, y_n, h_n] \in \mathbb{R}^3$, $\forall n \in N$, while the u-th device is located at $\mathbf{q}_n^u = [x_n^u, y_n^u, 0] \in \mathbb{R}^3, \forall n \in N$, and the position of the j-th jammer is $\mathbf{q}_n^j = \left[x_n^j, y_n^j, 0\right] \in \mathbb{R}^3, \forall n \in N$. Moreover, various obstacle heights are incorporated to simulate a realistic environment.

The action space of each UAV in time step n is defined as follows:

$$\mathbf{a}_n = \left[a_x, a_y, a_z\right] \in \mathcal{A}, \forall n \in [1, N] \tag{1}$$

where $a_x, a_y, a_z \in \{-1, 0, 1\}$ and $\mathcal{A}$ define the set of feasible actions on the UAV's position. Given the executed action, the position of the UAV evolves as follows:

$$\mathbf{q}_{n+1} = \mathbf{q}_n + \mathbf{a}_n \tag{2}$$

2.1. Channel Model

This paper employs a simplified path loss model in Decibel (dB). The channel gain between the UAV and the device u at the time step n is modeled as follows [28]:

$$g_n^u = \begin{cases} \beta_{\text{LoS}} + \alpha_{\text{LoS}} \log_{10}(d_n^u) + \eta_{\text{LoS}}(\text{dB}), & \text{if LoS} \\ \beta_{\text{NLoS}} + \alpha_{\text{NLoS}} \log_{10}(d_n^u) + \eta_{\text{NLoS}}(\text{dB}), & \text{otherwise} \end{cases}, \forall n \in [1, N] \tag{3}$$

where $d_n^u = \|\mathbf{q}_n - \mathbf{q}_n^u\|_2$ is the distance between the UAV and the device u, α is a path loss constant, β is the average channel gain at reference distance $d_0 = 1$m, and η represents the shadowing component following a Gaussian distribution of $\mathcal{N}(0, \sigma^2)$.

The communication link between UAVs and ground devices is affected by the altitude of the UAV, the characteristics of the urban environment, and the interference from other wireless devices. We can assume that the communication between the UAV and ground devices follows a time-division, multiple access mode. The rule is that, at each communication time step, the UAV can collect data from only one ground device, and only the device with remaining data and the highest signal-to-interference plus noise ratio (SINR) can establish a communication link with the UAV at the current time step n. The SINR of the signal received by the UAV from the ground device u at the time step n is as follows:

$$SINR_n^u = \frac{P_u 10^{\frac{g_n^u}{10}}}{I_n + \sigma^2}, \forall n \in [1, N] \tag{4}$$

where P_u is the transmission power at the ground device u, σ^2 is the white Gaussian noise power at the receiver, and I_n is the received interference power that is calculated as $I_n = \sum_{j=1}^{J} P_j 10^{\frac{g_n^j}{10}}$, where P_j represents the transmission power of the jammer j.

Furthermore, the channel throughput can be calculated by Shannon's formula as $R_n^u = B \log_2(1 + SINR_n^u)$, where B is the bandwidth of the channel in bits per second.

2.2. Throughput Maximization Problem Formulation

Here, we denote the link status by $l_n^u \in \{0, 1\}$, where $l_n^u = 1$ indicates the collection of data by the UAV from the u-th device at the time step n, and otherwise $l_n^u = 0$, and the channel access constraint is given as follows:

$$\sum_{u=1}^{U} l_n^u \leq 1, \forall u \in \mathcal{U}, n \in [1, N] \tag{5}$$

Our algorithm aims to optimize the trajectory of the UAV in order to maximize the amount of data collected from the ground equipment during the mission time T. This data collection problem can be formulated as the following optimization problem:

$$\max_{\mathbf{a}_n} \sum_{n=1}^{T} \sum_{u=1}^{U} l_n^u R_n^u \delta_n \tag{6}$$

$$s.t. \mathbf{q}_n \notin \mathcal{B}, \ \forall n \in [1, N] \tag{6a}$$

$$b_n > 0, \ \forall n \in [1, N] \tag{6b}$$

$$SINR_n^u \geq \gamma_{th}, \forall u \in \mathcal{U}, n \in [1, N] \tag{6c}$$

where $\mathbf{a}_n$ is the action of the UAV at step n. Equation (6a) ensures that the UAV avoids collisions with obstacles $\mathcal{B}$. Equation (6b) limits the operation time of the drones, forcing the UAV to end its mission before its battery b_n has run out. Equation (6c) indicates that the communication is interrupted when the SINR produced by the UAV is lower than that of the SINR threshold. This optimization problem is challenging, due to its non-convexity and unknown environment at the decision-making moment. Consequently, conventional model-based approaches are rendered inapplicable.

3. DDQN-Based UAV Trajectory Optimization Algorithm

3.1. Markov Decision Process

In reinforcement learning problems, the MDP is regarded as an idealized form that provides a theoretical framework for achieving goals through interactive learning. In the UAV-assisted communication model, we can use the MDP to simulate the interaction between entities. The complete MDP can be represented by a quaternion $\langle \mathcal{S}, \mathcal{A}, \mathcal{R}, \mathcal{P} \rangle$ [29]. The MDP for trajectory optimization in the data collection scenario is shown in Figure 2, which details the interaction process between the UAV and the environment. Additionally, it provides a detailed description of the process of generating an arbitrary time-slotted state space, where process ⑤ is interchangeable with process ⑥.

$\mathcal{A}$: The action space is defined in (1).

$\mathcal{S}$: The state of the UAV at the time step n is denoted by $\mathbf{s}_n = (\mathbf{s}_{n,1}, \mathbf{s}_{n,2}, \mathbf{s}_{n,3})$. To be specific, $\mathbf{s}_{n,1} = \{\mathbf{q}_n, b_n, L_n\}$ includes the characteristics of the UAV at the time step n, including the UAV's current momentary position $\mathbf{q}_n$, remaining power b_n, and the amount of data that has been collected L_n. $\mathbf{s}_{n,2} = \{\mathbf{q}_n^u, l_n^u, SNR_n^u, d_n^u, D_n^u\}$ includes a characterization of the UAV concerning each ground device, in which d_n^u represents the distance of the UAV from the device and D_n^u represents the amount of data remaining for each device. $\mathbf{s}_{n,3} = \{o_n, o_{n+1}\}$ represents the observation space o_n of the UAV at the time step n and the predicted observation space o_{n+1} at the next time step $n+1$ in the case of a wide observable range of the UAV camera.

$\mathcal{P}$: The transfer probability matrix $\mathcal{P}$ represents a transfer process, i.e., the probability of taking action $\mathbf{a}_n$ to move to the next state $\mathbf{s}_{n+1}$ when the intelligence is in state $\mathbf{s}_n$.

$\mathcal{R}$: The reward space represents the information about the gains made by the UAV in the process of choosing an action and reaching the next state, which can be denoted as $\mathbf{r}_n = \{r_{n,1}, r_{n,2}, r_{n,3}\}$, and the expression for $\mathbf{r}_n$ is defined as follows:

$$\mathbf{r}_n = r_{n,1} - r_{n,2} - r_{n,3} \tag{7}$$

where $r_{n,1} = \sum_{n=1}^{T} \sum_{u=1}^{U} l_n^u R_n^u \delta_n$ is defined as the total amount of data collected by the UAV at each time n, $r_{n,2}$ is the power penalty consumed by the UAV's movement; in addition, a penalty of $r_{n,3}$ is imposed if there is an obstacle in the current observation space o_n, or if the UAV's current position $\mathbf{q}_n$ is outside of the given region.

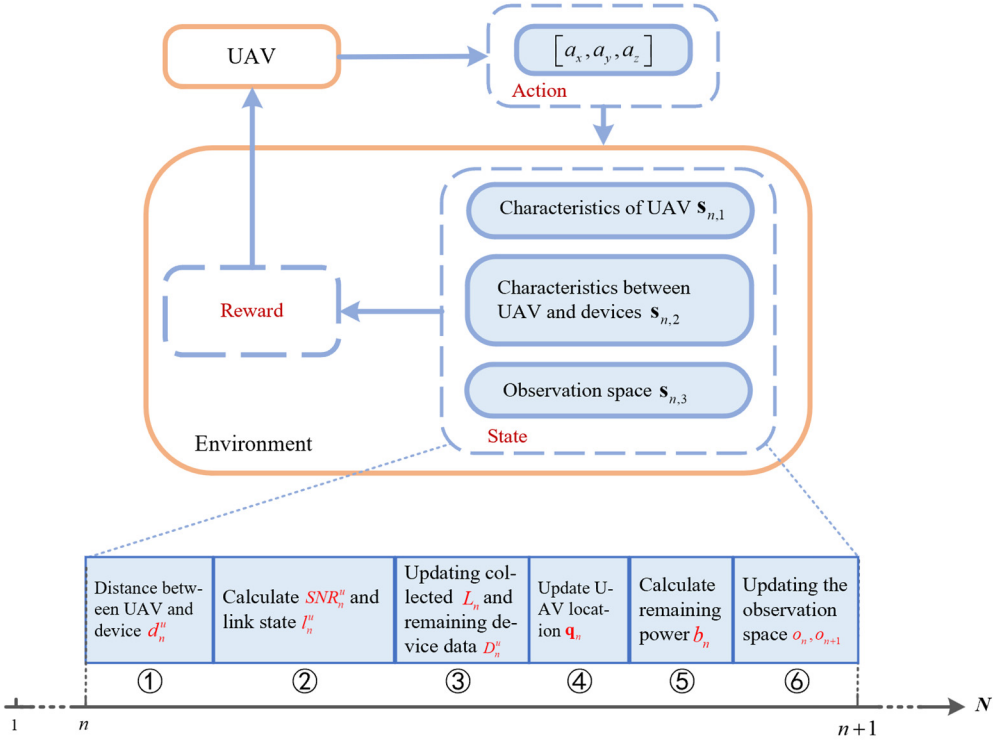


Figure 2. The MDP for trajectory optimization in the data collection scenario.

3.2. DDQN-Based UAV Trajectory Optimization Algorithm

The DQN algorithm is built on top of the standard Q-Learning algorithm framework [29], which utilizes deep learning algorithms to train action-value functions by updating the target network parameters. However, Q-learning and DQN may result in overestimating the Q-values, due to the use of max operations. To avoid the bias caused by this situation, we utilize the double deep Q-network (DDQN) algorithm [30]. Similar to the DQN algorithm, the online network parameters of both of these algorithms are used to generate strategies for the trajectories of intelligence, while the action strategies are used to evaluate the current goals. The difference here is that the DDQN algorithm uses a different set of network parameters for the goodness of the action strategies, i.e., the selection of actions and the evaluation of actions are realized using two different sets of network parameters. Thus, the objective function can be expressed as follows:

$$Q_t^{DDQN} = \mathbf{r}_{t+1} + \gamma Q\left(\mathbf{s}_{t+1}, \underset{a}{\operatorname{argmax}} Q(\mathbf{s}_{t+1}, \mathbf{a}; \theta_t), \theta_t'\right) \tag{8}$$

We use ε-*greedy* to choose the actions $\mathbf{a}_n$, i.e., randomly selecting actions with a probability of ε, and selecting actions based on the Q-values with a probability of $1-\varepsilon$, to ensure that the UAV is somewhat exploratory. Figure 3 shows the architecture of the DDQN.

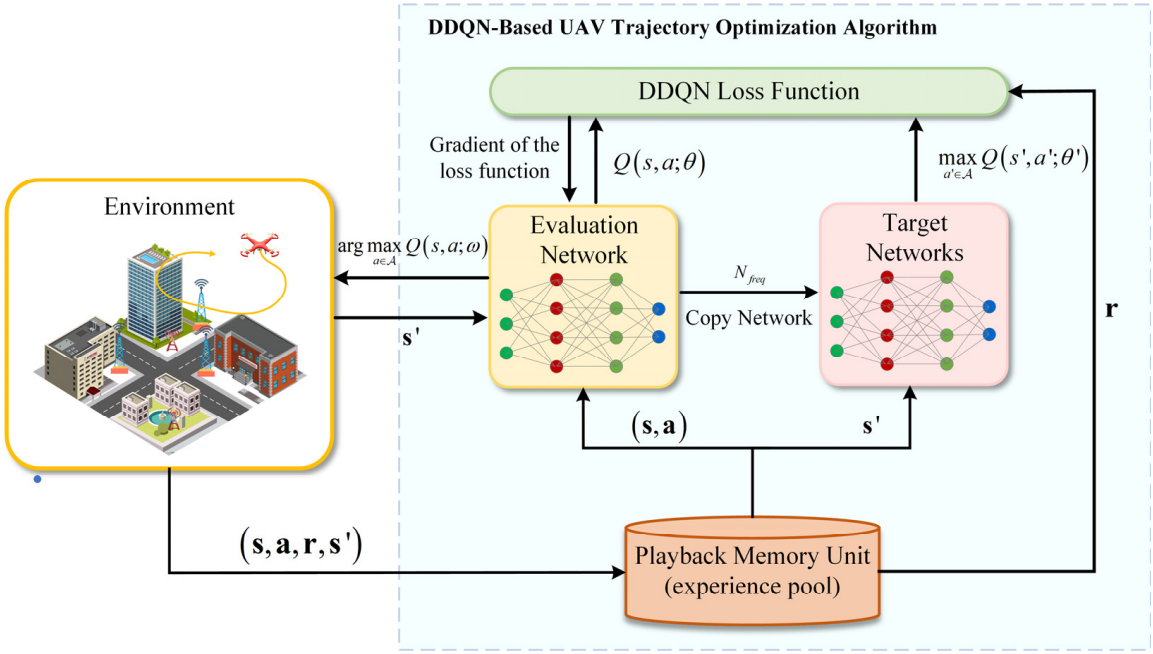


Figure 3. The architecture of DDQN.

Compared with the DQN, the DDQN avoids overestimation to some extent and improves the stability and speed of training. The implementation of the DDQN algorithm in the data collection scenario is presented in Algorithm 1.

Algorithm 1: DDQN-Based UAV Trajectory Optimization Algorithm

Initialize replay memory $\mathcal{D}$, the online network parameters θ, the target network parameters $\theta' = \theta$, and the target network update period N_{freq}.

for episode = 0, 1, . . ., M − 1 **do**
Randomly generating the location of the IoT devices $\mathbf{q}_0^u$, the location of the UAV $\mathbf{q}_0$, the location of the jammers $\mathbf{q}_0^j$, transmission power of UAV P, and interference power P_j.
Time step n = 0, initialization of the environment and state $\mathbf{s}_n$ of the UAV
while $b_n \geq 0$ **do**
choose action $\mathbf{a}_n$ with ε-*greedy* policy, i.e.,

$$\mathbf{a}_n = \begin{cases} \text{randomly select from } \mathcal{A} & w.p.\ \varepsilon \\ \underset{\mathbf{a} \in \mathcal{A}}{\text{argmax}} Q(\mathbf{s}_n, \mathbf{a}_n) & w.p.\ 1\text{-}\varepsilon \end{cases}$$

take action $\mathbf{a}_n$, state $\mathbf{s}_n$, next state s_{n+1}, and reward $\mathbf{r}_n$
store $(\mathbf{s}_n, \mathbf{a}_n, \mathbf{s}_{n+1}, \mathbf{r}_n)$ in replay memory $\mathcal{D}$
$\mathbf{s}_n \leftarrow \mathbf{s}_{n+1}$
$n = n + 1$
end while
randomly sample a minibatch from $\mathcal{D}$

$$y_n = \begin{cases} \mathbf{r}_{n+1}, & \text{if } \mathbf{s}_{n+1} \text{is terminal} \\ \mathbf{r}_{n+1} + \gamma Q\left(\mathbf{s}_{n+1}, \underset{a}{\text{argmax}} Q(\mathbf{s}_{n+1}, \mathbf{a}; \theta_t), \theta_t'\right), & \text{otherwise} \end{cases}$$

do a gradient descent step with loss $\|y_n - Q(\mathbf{s}_n, \mathbf{a}_n; \theta)\|^2$
Replace target network parameters $\theta' \leftarrow \theta$ when $n = N_{freq}$
end for

4. Simulation Results and Discussion

In this section, simulation experiments are conducted, and the performance of the proposed algorithms in different scenarios is explored. To evaluate the efficacy of the proposed method in UAV data collection, the following algorithms are compared: (1) Q-learning; (2) proximal policy optimization (PPO); (3) DQN; (4) dueling DQN; and (5) DDQN. The simulation experiments are conducted with different numbers of devices and obstacles. The convergence performance of the algorithms is compared. In this study, a computer equipped with a 3.60GHz NVIDIA GPU RTX 2080 was used as the experimental platform, and the Adam optimizer was used to update the neural network. The specific simulation experiments are shown in Section 4.1.

4.1. Parameter Initialization

To reduce the training time, the simulation scene in this paper is set at 50 m × 50 m × 10 m, due to the complexity of the algorithm's action and state space [31–34]. The algorithm presented in this paper applies to scenes of arbitrary size, and the simulation parameters are shown in Table 1. Throughout the testing phase, the two-dimensional coordinates of the UAV are positioned at the center of the scene, while its altitude is randomly generated within the range of 3 m to 10 m. The positions of the jammers and devices are randomly distributed within the designated area. Moreover, the data transmission rates of each device are randomly assigned from 15,000 bps to 20,000 bps. When comparing the performance of the different algorithms under identical scenarios, the total fixed data volume remains consistent. Additionally, the parameters for the DDQN algorithm are set as shown in Table 2. The hyperparameters are set according to the simulation experience.

Table 1. Initialization simulation parameters.

Parameters	Value
$\mathcal{N}$	3
$\mathcal{J}$	3
B	100 MHz
P	43 W
P_j	[10, 50]
σ^2	−60 dBm/Hz
β_{LoS}	−30 dB
β_{NLoS}	−35 dB
η_{LoS}	1.41 [28]
η_{NLoS}	2.23 [28]
α_{LoS}	−2.5 [28]
α_{NLoS}	−3.04 [28]

Table 2. Parameter settings.

Parameters	Value
Learning rate α	0.0004
Discount factor γ	0.99
ε	0.01
N_{freq}	100
Minibatch	32
Iteration	80,000

We compare the performance of the DDQN algorithm with other traditional algorithms across various scenarios. During the training phase, each algorithm saves the model that achieved the highest cumulative reward. Subsequently, during the testing phase, 80,000 episodes are run using the best-saved models, and the average reward for each algorithm is computed.

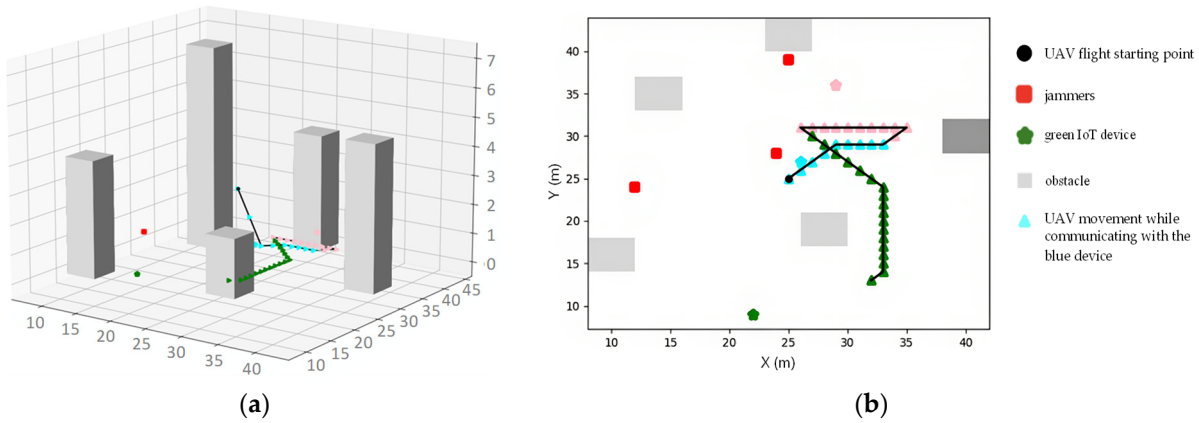
Complexity analysis: The training process has a computational complexity of $\mathcal{O}(|\mathcal{A}| \times |\mathcal{S}| \times \mathcal{N}^{\mathcal{L}})$, where $|\bullet|$ refers to the cardinality of a set, $\mathcal{N}$ is the maximum number of neurons in the hidden layers in θ, and $\mathcal{L}$ is the number of layers of θ.

4.2. Result Analysis with Different Numbers of IoT Devices

In this section, the results of the trajectory optimization problem for the UAV data collection task using the DDQN algorithm are analyzed for various numbers of devices. The UAV is probed in a bounded three-dimensional space, while ground IoT devices and jammers are randomly distributed across a two-dimensional plane. The objective is to optimize the UAV's movement trajectory in order to obtain the highest cumulative reward.

1. The result analysis for the scenario with three devices and five obstacles (3D+5O, in short) is as follows:

An example of a UAV trajectory involving three devices and three jammers is shown in the illustration attached to Figure 4. It is shown that the UAV demonstrates a strategy of bypassing the jammers and obstacles to mitigate interference during the data collection mission and to reduce the likelihood of collisions. Furthermore, the UAV tends to approach devices closely, thereby reducing the distance between them in order to enhance the amount of data collected. The simulation results indicate that the overall trajectory of the UAV aligns with the optimization goal.



(**a**) (**b**)

Figure 4. Example of UAV data collection under the 3G+5O scenario. (**a**) three-dimensional (3D) UAV trajectory; (**b**) Top view of the scenario. (A pentagram represents the ground devices, while a triangular shape symbolizes the trajectory of the UAV.)

Figure 5 shows the convergence performance of the different algorithms in this scenario. The experimental results show that the DDQN algorithm outperforms the other algorithms in terms of the final cumulative average rewards.

2. The result analysis for the scenario with five devices and five obstacles (5D+5O, in short), as follows:

The scenario with five devices and three jammers is depicted in Figure 6. From observing the UAV's trajectory, it is evident that the UAV successfully executes its mission while navigating around the jammers and obstacles. Figure 7 shows the convergence curves of the UAV. The curve shows that the mean reward of each agent shows an upward trend until convergence is reached. Notably, the reward curve of the DDQN algorithm surpasses that of the traditional algorithm, revealing the good performance of the DDQN algorithm.

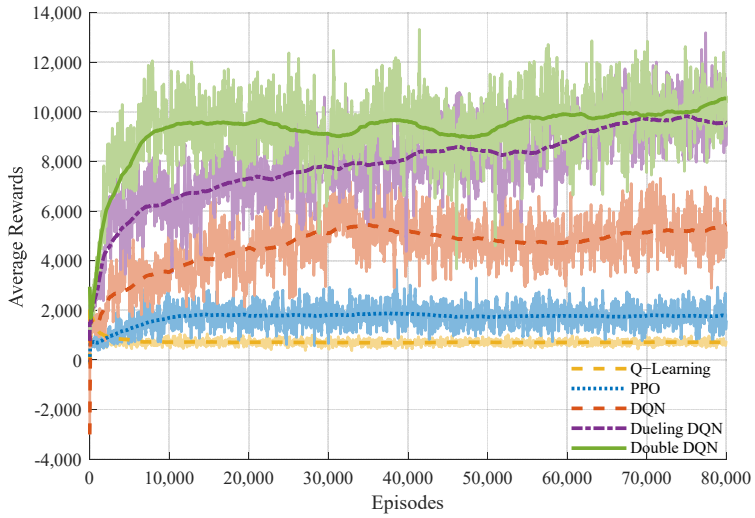


Figure 5. The convergence performance of the different algorithms in 3D+5O.

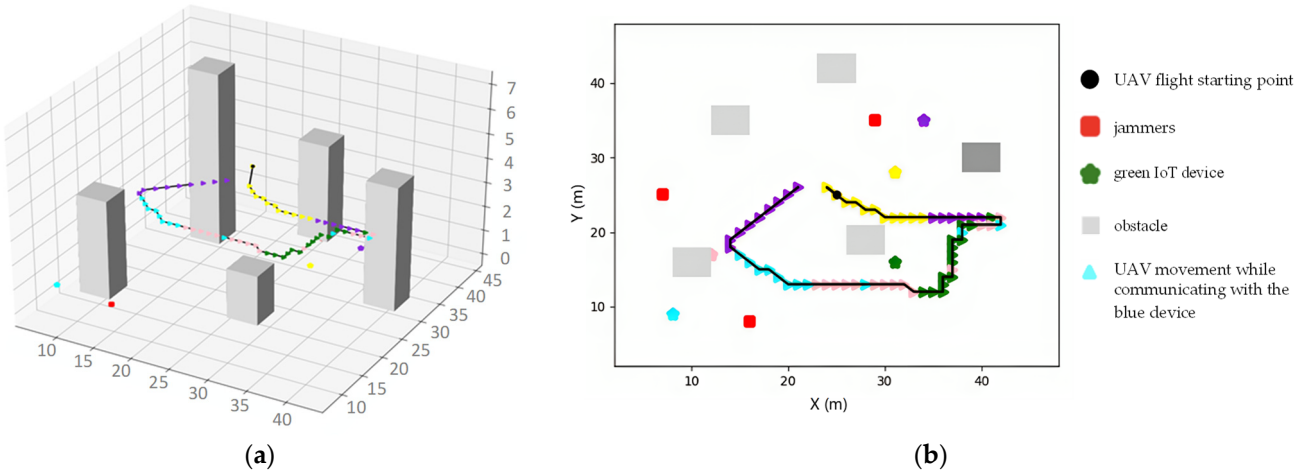


Figure 6. Example of UAV data collection under the 5G+5O scenario. (**a**) 3D UAV trajectory; (**b**) Top view of the scenario.

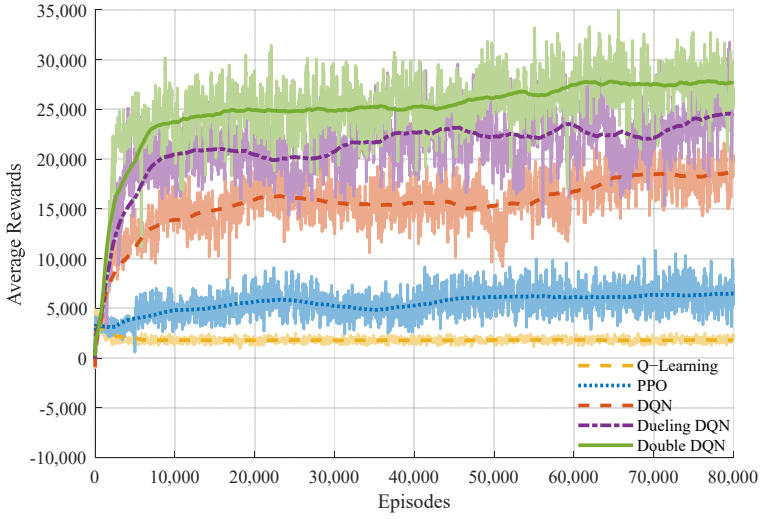


Figure 7. The convergence performance of the different algorithms in 5D+5O.

4.3. Result Analysis with Different Numbers of Obstacles

In this section, we increase the number of obstacles from five in 3D+5O to eight to assess the generalizability of the DDQN algorithm (3D+8O). As can be seen from Figure 8, as the density of the obstacles increases and the environment becomes more intricate, the UAV strategically selects an altitude characterized by lower obstacle density to navigate and complete the data collection task. Figure 9 provides a comparative analysis of the average reward attained by each algorithm, reaffirming the superior performance of DDQN in comparison to the other algorithms.

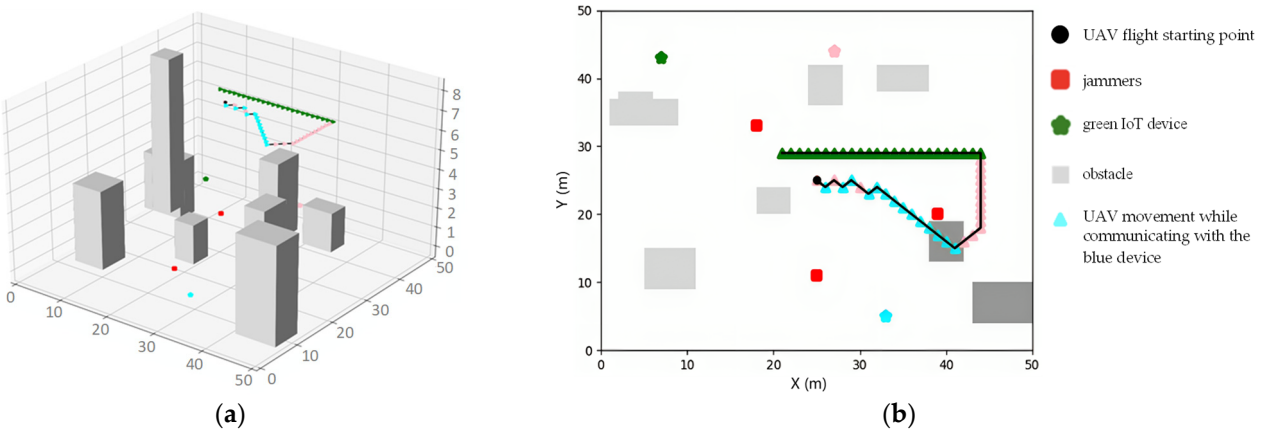


Figure 8. Example of UAV data collection under the 3G+8O scenario. (**a**) 3D UAV trajectory; (**b**) Top view of the scenario.

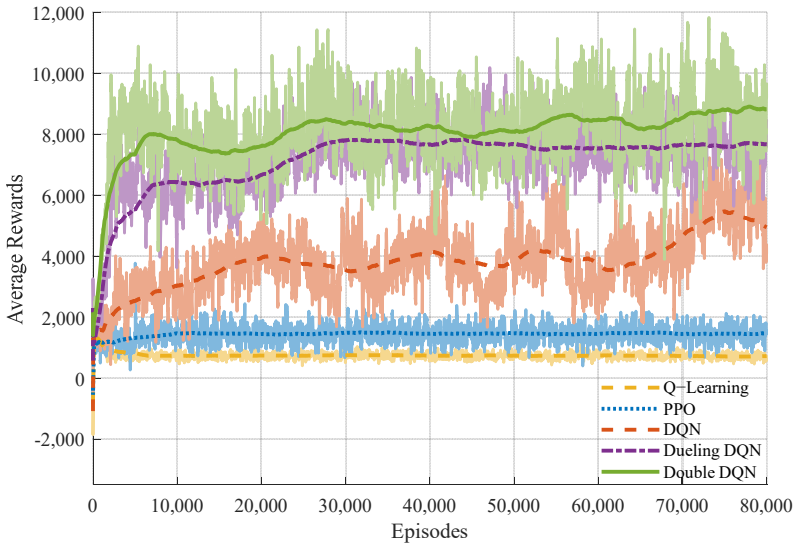


Figure 9. The convergence performance of the different algorithms in 3D+8O.

Figure 10 presents a comparison of the average rewards across the different scenarios under the DDQN algorithm. We obtain the following conclusions: (1) with throughput designated as the reward metric, scenario 5D+5O, characterized by an increased number of ground devices, achieves the maximum reward value compared to 3D+5O; and (2) likewise, in scenario 3D+8O, the higher obstacle density relative to 3D+5O imposes more penalties on the UAVs, resulting in a lower average reward value.

We also compared the convergence rates of the algorithms. The 90% confidence interval of the average reward value of each algorithm is used as the convergence interval to calculate the convergence rate of each algorithm in each scenario, and then the convergence rates of the algorithms in the three scenarios are used to obtain that shown in Figure 11. The

experimental results indicate that, while ensuring the optimal value of the average reward, both the DDQN and the dueling DQN exhibit a better convergence rate compared to the PPO and DQN algorithms. However, the Q-learning algorithm has the fastest convergence rate but the lowest average reward value, resulting in a poor convergence performance.

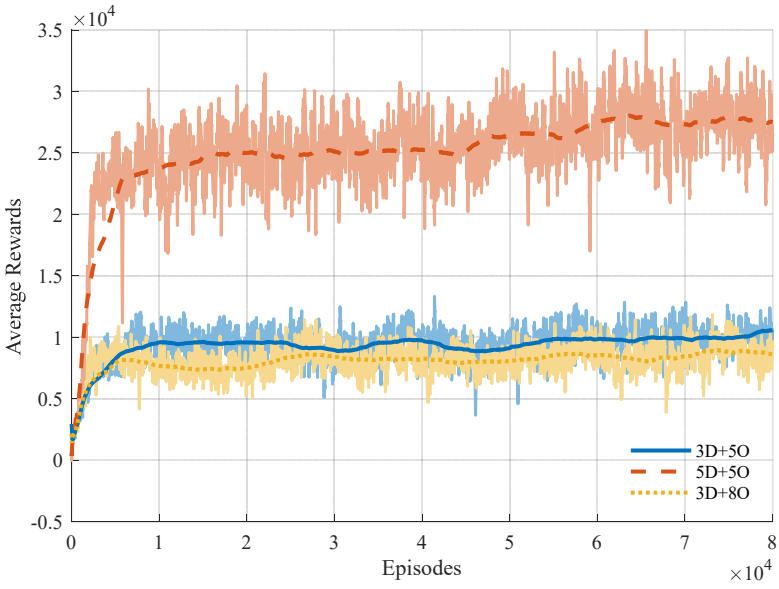


Figure 10. Comparison of average rewards for the scenarios proposed under the DDQN algorithm.

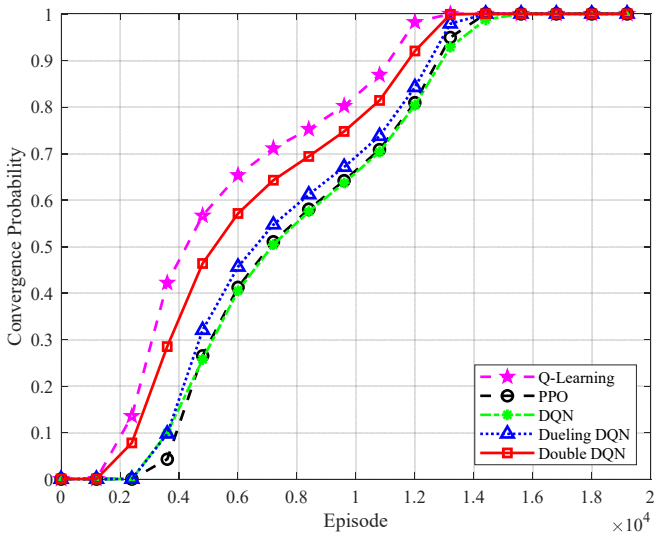


Figure 11. Comparison of convergence probability of the different algorithms.

To further highlight the advantages of the DDQN algorithm, we took 3D+5O as an example; fixed the initial position of the UAV, the ground equipment, and the jammer position after training the network; and observed the test flights under the different algorithms to observe the data collection process of the UAV. The experimental results are shown in Figure 12, which shows that, although the Q-learning algorithm finally reached convergence in the previous experiments, it is still inferior to the dueling DQN and DDQN in terms of the data collection speed. The test results show that the DDQN algorithm can learn the unknown environment better and can collect the data in the shortest number of steps.

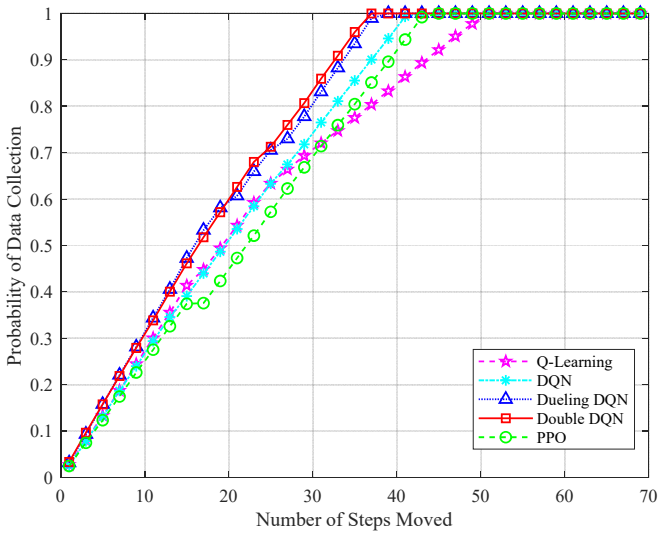


Figure 12. Comparison of probability of data collection of the different algorithms.

Table 3 compares the average training time of each algorithm in different cases. The DDQN algorithm used in this paper has a shorter training time, while ensuring convergence performance. The dueling DQN algorithm is the second-best in terms of convergence performance, surpassed only by the DDQN algorithm. However, it is not as fast as the DDQN algorithm in terms of training time. Furthermore, the proposed algorithm can meet real-time requirements, as its execution time on the system is significantly less than that of its training time.

Table 3. Average training time (seconds per episode) of the UAV.

Method	3D+5O	5D+5O	3D+8O
Q-learning	1.34460	1.28929	1.39889
PPO	1.03520	1.63133	1.27002
DQN	1.75778	4.26592	1.80001
Dueling DQN	2.05745	4.35702	2.13843
DDQN	1.63612	3.95510	1.71541

The simulation results indicate that Q-learning has the worst performance, with an average reward value that is only 10% of that of DDQN. This is because Q-learning requires the storage of the value function of each state–action pair, which can be very difficult, or even infeasible, in a high-dimensional state space. The PPO algorithm is only better than Q-learning because it does not use empirical replay when updating the policy network, resulting in the low utilization of empirical samples. Additionally, the algorithm's performance is sensitive to hyperparameters. On average, DQN achieves only 50% of that of DDQN, due to overestimation, which leads to training instability and performance degradation. It is important to note that this is an objective evaluation and not a subjective one. Despite potential drawbacks such as implementation complexity and longer training times, dueling DQN may still be a better choice in certain problems and scenarios.

5. Conclusions

We studied the problem of 3D trajectory planning for UAVs in data collection network scenarios with jammers and flying obstacles. To achieve this, we proposed a DDQN-based UAV trajectory optimization algorithm that utilizes appropriate reward values. This algorithm enabled the UAVs to efficiently perform data collection tasks in complex and changing environments without prior knowledge of the channel information. We conducted simulations to analyze the impact of different numbers of IoT devices and

obstacle densities on the algorithm's performance. The experimental results demonstrate that the algorithm optimized for throughput outperforms the traditional algorithm in terms of both latency and cumulative reward. Additionally, the results show that the DDQN algorithm is effective in addressing challenges related to trajectory planning.

6. Future Work

While the DDQN algorithm performs well in many situations, it has some limitations. These include hyperparameter sensitivity, high computational resource requirements, and a limited adaptation to non-smooth environments.

In future work, we will conduct out-of-field experiments to observe the UAVs' anti-jamming, obstacle avoidance, and trajectory planning capabilities in a realistic environment. We will also introduce limitations on the UAV batteries, particularly by charging the UAVs in a designated area to ensure that sufficient power is available between tasks. In addition, future considerations will include addressing multi-UAV dynamic scenarios, considering distributed training and hyperparameter tuning, in order to accelerate the training speed for reinforcement learning. Furthermore, when collecting data in UAV IoT networks, it is essential to consider the security and privacy of the transmitted data, adhere to ethical guidelines, and acknowledge social responsibility.

Author Contributions: Conceptualization, S.W. and N.Q.; methodology, S.W. and N.Q.; validation, S.W., H.L. and H.J.; formal analysis, S.W.; investigation, H.J.; writing—original draft preparation, S.W., D.Z.; supervision, L.J. and M.X. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the National Natural Science Foundation of China (No. 62271253, 61901523, and 62001381), the Fundamental Research Funds for the Central Universities (No. NS2023018), in part by the National Aerospace Science Foundation of China under Grant 2023Z021052002, in part by the open research fund of the National Mobile Communications Research Laboratory, Southeast University (No. 2023D09), and the Postgraduate Research and Practice Innovation Program of NUAA (No. xcxjh20230407).

Data Availability Statement: All data underlying the results are available as part of the article and no additional source data are required.

Acknowledgments: The authors would like to thank the editors and the reviewers for their valuable time and constructive comments.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Li, B.; Fei, Z.; Zhang, Y. UAV Communications for 5G and Beyond: Recent Advances and Future Trends. *IEEE Internet Things J.* **2019**, *6*, 2241–2263. [CrossRef]
2. Cao, L.; Wang, H. Research on UAV Network Communication Application Based on 5G Technology. In Proceedings of the 2022 3rd International Conference on Electronic Communication and Artificial Intelligence (IWECAI), Zhuhai, China, 14–16 January 2022; pp. 125–129.
3. Geraci, G.; Garcia-Rodriguez, A.; Azari, M.M.; Lozano, A.; Mezzavilla, M.; Chatzinotas, S.; Chen, Y.; Rangan, S.; Di Renzo, M. What Will the Future of UAV Cellular Communications Be? A Flight from 5G to 6G. *IEEE Commun. Surv. Tutorials* **2022**, *24*, 1304–1335. [CrossRef]
4. Li, P.; Liu, Y.; Deng, X.; Wu, B.; Fu, R. Self-organized Cooperative Electronic Jamming by UAV Swarm Based on Contribution Priority and Cost Competition. In Proceedings of the 2021 IEEE 15th International Conference on Electronic Measurement & Instruments (ICEMI), Nanjing, China, 29–31 October 2021; pp. 49–53.
5. Chen, B.W.; Rho, S. Autonomous Tactical Deployment of the UAV Array Using Self-Organizing Swarm Intelligence. *IEEE Consum. Electron. Mag.* **2020**, *9*, 52–56. [CrossRef]
6. Hu, J.; Wu, H.; Zhan, R.; Rafik, M.; Zhou, X. Self-organized search-attack mission planning for UAV swarm based on wolf pack hunting behavior. *J. Syst. Eng. Electron.* **2021**, *32*, 1463–1476.
7. Xu, X.; Zhao, H.; Yao, H.; Wang, S. A Blockchain-Enabled Energy-Efficient Data Collection System for UAV-Assisted IoT. *IEEE Internet Things J.* **2021**, *8*, 2431–2443. [CrossRef]

8. Cheng, N.; Wu, S.; Wang, X.; Yin, Z.; Li, C.; Chen, W.; Chen, F. AI for UAV-Assisted IoT Applications: A Comprehensive Review. *IEEE Internet Things J.* **2023**, *10*, 14438–14461. [CrossRef]
9. Kosmerl, J.; Vilhar, A. Base stations placement optimization in wireless networks for emergency communications. In Proceedings of the 2014 IEEE International Conference on Communications Workshops (ICC), Sydney, Australia, 10–14 June 2014; pp. 200–205.
10. Zeng, Y.; Zhang, R.; Lim, T.J. Wireless communications with unmanned aerial vehicles: Opportunities and challenges. *IEEE Commun. Mag.* **2016**, *54*, 36–42. [CrossRef]
11. Wang, J.; Jiang, C.; Han, Z.; Ren, Y.; Maunder, R.G.; Hanzo, L. Taking drones to the next level: Cooperative distributed unmanned aerial vehicular networks: Small and mini drones. *IEEE Trans. Veh. Technol.* **2016**, *12*, 73–82. [CrossRef]
12. Pikner, I.; Sivasundaram, S. New Approaches to the Development and Employment of the UAV. *AIP Conf. Proc.* **2012**, *1493*, 752–757.
13. Jakaria, A.H.M.; Rahman, M.A.; Asif, M.; Khalil, A.A.; Kholidy, H.A.; Anderson, M.; Drager, S. Trajectory Synthesis for a UAV Swarm Based on Resilient Data Collection Objectives. *IEEE Trans. Netw. Serv. Manag.* **2023**, *20*, 138–151. [CrossRef]
14. Ding, T.; Liu, N.; Yan, Z.M.; Liu, L.; Cui, L.Z. An efficient reinforcement learning game framework for uav-enabled wireless sensor network data collection. *J. Comput. Sci. Technol.* **2022**, *37*, 1356–1368. [CrossRef]
15. Caposciutti, G.; Bandini, G.; Marracci, M.; Buffi, A.; Tellini, B. Capacity Fade and Aging Effect on Lithium Battery Cells: A Real Case Vibration Test with UAV. *IEEE J. Miniaturization Air Space Syst.* **2021**, *2*, 76–83. [CrossRef]
16. Pan, Y.; Chen, Q.; Zhang, N.; Li, Z.; Zhu, T.; Han, Q. Extending Delivery Range and Decelerating Battery Aging of Logistics UAVs Using Public Buses. *IEEE Trans. Mob. Comput.* **2023**, *22*, 5280–5295. [CrossRef]
17. Youn, W.; Choi, H.; Cho, A.; Kim, S.; Rhudy, M.B. Accelerometer Fault-Tolerant Model-Aided State Estimation for High-Altitude Long-Endurance UAV. *IEEE Trans. Instrum. Meas.* **2020**, *69*, 8539–8553. [CrossRef]
18. Pan, H.; Liu, Y.; Sun, G.; Fan, J.; Liang, S.; Yuen, C. Joint power and 3D trajectory optimization for UAV-enabled wireless powered communication networks with obstacles. *IEEE Trans. Commun.* **2023**, *71*, 2364–2380. [CrossRef]
19. Zhang, S.; Zeng, Y.; Zhang, R. Cellular-Enabled UAV Communication: A Connectivity-Constrained Trajectory Optimization Perspective. *IEEE Trans. Commun.* **2019**, *67*, 2580–2604. [CrossRef]
20. Li, B.; Qi, X.; Yu, B.; Liu, L. Trajectory Planning for UAV Based on Improved ACO Algorithm. *IEEE Access* **2020**, *8*, 2995–3006. [CrossRef]
21. Tang, Q.; Liu, L.; Jin, C.; Wang, J.; Liao, Z.; Luo, Y. An UAV-assisted mobile edge computing offloading strategy for minimizing energy consumption. *Comput. Netw.* **2022**, *207*, 108857. [CrossRef]
22. Wang, Y.; Gao, Z.; Zhang, J.; Cao, X.; Zheng, D.; Gao, Y.; Ng, D.W.K.; Di Renzo, M. Trajectory Design for UAV-Based Internet of Things Data Collection: A Deep Reinforcement Learning Approach. *IEEE Internet Things J.* **2022**, *9*, 3899–3912. [CrossRef]
23. Zhang, B.; Liu, C.H.; Tang, J.; Xu, Z.; Ma, J.; Wang, W. Learning-Based Energy-Efficient Data Collection by Unmanned Vehicles in Smart Cities. *IEEE Trans Ind. Inform.* **2018**, *14*, 1666–1676. [CrossRef]
24. Bayerlein, H.; Theile, M.; Caccamo, M.; Gesbert, D. UAV Path Planning for Wireless Data Harvesting: A Deep Reinforcement Learning Approach. In Proceedings of the GLOBECOM 2020—2020 IEEE Global Communications Conference, Taipei, Taiwan, 7–11 December 2020; pp. 1–6.
25. Zeng, Y.; Xu, X. Path design for cellular-connected uav with reinforcement learning. In Proceedings of the 2019 IEEE Global Communications Conference (GLOBECOM), Waikoloa, HI, USA, 9–13 December 2019; pp. 1–6.
26. Khamidehi, B.; Sousa, E.S. A double q-learning approach for navigation of aerial vehicles with connectivity constraint. In Proceedings of the ICC 2020—2020 IEEE International Conference on Communications (ICC), Dublin, Ireland, 7–11 June 2020; pp. 1–6.
27. Yin, S.; Zhao, S.; Zhao, Y.; Yu, F.R. Intelligent Trajectory Design in UAV-Aided Communications with Reinforcement Learning. *IEEE Trans. Veh. Technol.* **2019**, *68*, 8227–8231. [CrossRef]
28. Esrafilian, O.; Bayerlein, H.; Gesbert, D. Model-aided Deep Reinforcement Learning for Sample-efficient UAV Trajectory Design in IoT Networks. In Proceedings of the 2021 IEEE Global Communications Conference (GLOBECOM), Madrid, Spain, 7–11 December 2021; pp. 1–6.
29. Mnih, V.; Kavukcuoglu, K.; Silver, D.; Graves, A.; Antonoglou, I.; Wierstra, D.; Riedmiller, M. Playing atari with deep reinforcement learning. *arXiv* **2013**, arXiv:1312.5602.
30. Van Hasselt, H.; Guez, A.; Silver, D. Deep reinforcement learning with double q-learning. In Proceedings of the AAAI Conference on Artificial Intelligence 2016, Phoenix, AZ, USA, 12–17 February 2016; Volume 30.
31. Wei, X.; Cai, L.; Wei, N.; Zou, P.; Zhang, J.; Subramaniam, S. Joint UAV Trajectory Planning, DAG Task Scheduling, and Service Function Deployment Based on DRL in UAV-Empowered Edge Computing. *IEEE Internet Things J.* **2023**, *10*, 12826–12838. [CrossRef]
32. Theile, M.; Bayerlein, H.; Nai, R.; Gesbert, D.; Caccamo, M. UAV Path Planning using Global and Local Map Information with Deep Reinforcement Learning. In Proceedings of the 2021 20th International Conference on Advanced Robotics (ICAR), Ljubljana, Slovenia, 6–10 December 2021; pp. 539–546.

33. Hou, X.; Liu, F.; Wang, R.; Yu, Y. A UAV Dynamic Path Planning Algorithm. In Proceedings of the 2020 35th Youth Academic Annual Conference of Chinese Association of Automation (YAC), Zhanjiang, China, 16–18 October 2020; pp. 127–131.
34. Pehlivanoğlu, Y.V.; Bekmezci, İ.; Pehlivanoğlu, P. Efficient Strategy for Multi-UAV Path Planning in Target Coverage Problems. In Proceedings of the 2022 International Conference on Theoretical and Applied Computer Science and Engineering (ICTASCE), Ankara, Turkey, 29 September–1 October 2022; pp. 110–115.