Computers, Materials & Continua

Tech Science Press

Doi:10.32604/cmc.2025.064147

ARTICLE

Enhanced Coverage Path Planning Strategies for UAV Swarms Based on SADQN Algorithm

Zhuoyan Xie[1], Qi Wang[1,*], Bin Kong[2,*] and Shang Gao[1]

[1]School of Computer Science and Engineering, Jiangsu University of Science and Technology, Zhenjiang, 212003, China
[2]Experimental Centre of Forestry in North China, Chinese Academy of Forestry, Beijing, 102300, China

*Corresponding Authors: Qi Wang. Email: wangqi@just.edu.cn; Bin Kong. Email: konbin@caf.ac.cn

Received: 06 February 2025; Accepted: 06 May 2025; Published: 03 July 2025

ABSTRACT: In the current era of intelligent technologies, comprehensive and precise regional coverage path planning is critical for tasks such as environmental monitoring, emergency rescue, and agricultural plant protection. Owing to their exceptional flexibility and rapid deployment capabilities, unmanned aerial vehicles (UAVs) have emerged as the ideal platforms for accomplishing these tasks. This study proposes a swarm A*-guided Deep Q-Network (SADQN) algorithm to address the coverage path planning (CPP) problem for UAV swarms in complex environments. Firstly, to overcome the dependency of traditional modeling methods on regular terrain environments, this study proposes an improved cellular decomposition method for map discretization. Simultaneously, a distributed UAV swarm system architecture is adopted, which, through the integration of multi-scale maps, addresses the issues of redundant operations and flight conflicts in multi-UAV cooperative coverage. Secondly, the heuristic mechanism of the A* algorithm is combined with full-coverage path planning, and this approach is incorporated at the initial stage of Deep Q-Network (DQN) algorithm training to provide effective guidance in action selection, thereby accelerating convergence. Additionally, a prioritized experience replay mechanism is introduced to further enhance the coverage performance of the algorithm. To evaluate the efficacy of the proposed algorithm, simulation experiments were conducted in several irregular environments and compared with several popular algorithms. Simulation results show that the SADQN algorithm outperforms other methods, achieving performance comparable to that of the baseline prior algorithm, with an average coverage efficiency exceeding 2.6 and fewer turning maneuvers. In addition, the algorithm demonstrates excellent generalization ability, enabling it to adapt to different environments.

KEYWORDS: Coverage path planning; unmanned aerial vehicles; swarm intelligence; Deep Q-Network; A* algorithm; prioritized experience replay

1 Introduction

In recent years, due to their efficiency, flexibility, safety, and other advantages, unmanned aerial vehicles (UAVs) have been widely applied in various fields such as surveillance, agriculture, disaster management, and power line inspections [1–5]. These applications require UAV to quickly cover the target area while acquiring environmental information and avoiding obstacles [6]. Therefore, efficient coverage path planning (CPP) is critical to determining the operational efficiency and quality of the UAV. However, in practical applications, a single UAV is limited by its endurance and is suitable only for small-scale tasks [7]. For large-scale missions, UAV swarm are typically used. This paper focuses on the coverage task of the UAV swarm in order to further improve efficiency and reduce time and energy consumption.

Researchers have proposed various methods to address the UAV swarm coverage path planning problem, which can be broadly classified into centralized and distributed approaches. In centralized methods, a central planner allocates tasks based on prior environmental knowledge and mathematical optimization to find the optimal path [8]. However, these methods assume that environmental information is available, which underestimates the complexity of the CPP problem. Distributed methods, on the other hand, enable collaborative decision-making through communication between UAVs and a ground station, avoiding collisions [9], but still face challenges in complex and irregular environments. With the development of artificial intelligence, more and more researchers are applying machine learning to path planning problems, with reinforcement learning (RL) being the most representative approach [10]. However, its research in coverage path planning is still in the early stages [11].

This paper proposes a UAV swarm-based Coverage Path Planning methodology, aiming to ensure complete coverage of the target area while minimizing time and energy consumption, taking into account factors such as no-fly zones, boundary constraints, and flight conflicts. The main contributions of this paper include:

(1) We proffer an enhanced cell decomposition method, discretizing the irregular target area into grids whilst concurrently minimizing the grid-based target area to curtail the task execution time to the greatest extent possible.
(2) We put forward a multi-scale map fusion method, which amalgamates and updates local environment maps perceived by individual UAVs through communication among UAVs, augmenting the observation range of UAVs and further augmenting coverage efficiency.
(3) We present a distributed SADQN algorithm, leveraging the A* algorithm to assist in initial action selection for UAV swarm DQN training, expediting the training process, and employing prioritized experience replay to train neural networks, hastening the convergence of the algorithm.

The remainder of this work is structured as follows. Section 2 furnishes an overview of germane work in the field of multi-UAV coverage path planning. Section 3 introduces the model of UAV swarm coverage tasks and delineates objectives and constraints. Section 4 presents the SADQN algorithm for UAV swarm coverage along with its minutiae. Section 5 showcases the simulation results of the algorithm. Finally, Section 6 concludes the paper and deliberates on future research directions.

2 Related Works

At present, a copious amount of research efforts have been dedicated to the coverage path planning problem concerning UAV swarms. This part undertakes a comprehensive survey of pertinent work from both centralized and distributed perspectives.

In [12], a grid decomposition-based coverage method is proposed, which optimizes the allocation of coverage areas for multiple UAVs by constructing a linear programming model. Lu et al. introduced a turning-minimization multi-robot spanning tree coverage algorithm that transforms the problem into one of finding the maximum independent set in a bipartite graph and then employs a greedy strategy to minimize the number of turns in the spanning tree's circumnavigation coverage path [13]. Maza and Ollero presented a cooperative strategy in which the ground control station divides the target area into multiple non-overlapping, obstacle-free subregions and assigns each subregion to a UAV based on its relative capabilities and initial position [14]. However, since this method initiates task assignments from the center of each subregion, it may result in considerable redundant coverage. Building upon this, Chen et al. proposed a novel path planning method based on a success-history-adaptive differential evolution variant combined with linear population reduction. This approach establishes the relationship between all possible starting

points of subregion paths and the overall multi-region path to enhance the efficiency of multi-region coverage tasks [15].

These methods rely on the availability of environmental information, underestimating the complexity of the CPP problem and being applicable only to deterministic scenarios. To augment UAVs' robustness in complex environments, distributed algorithms can be enlisted. In [16], a coordinated search scheme based on model predictive control and communication constraints was designed to effectively enable UAV swarms to search for both static and dynamic targets in uncertain scenarios. Qiu et al. proposed a novel complete coverage path planning method for mobile robot obstacle avoidance based on bio-inspired neural networks, rolling path planning, and heuristic search approaches [17]. Bine et al. introduced an energy-aware ant colony optimization algorithm that leverages multi-UAV Internet technologies to coordinate and organize UAVs in order to avoid airspace collisions and congestion [18]. Li et al. proposed a DQN-based coverage path planning method that approximates the optimal action values via DQN, while combining a sliding window approach with probabilistic statistics to handle unknown environments. This method optimizes coverage decisions and enhances the adaptability and performance of multi-UAV missions in unknown scenarios [19]. Moreover, in [20], a comprehensive coverage path planning framework was constructed using deep reinforcement learning techniques, which integrates convolutional neural networks with long short-term memory networks. The framework simultaneously maximizes cumulative rewards and optimizes an overall cost weight based on kinetic energy.

Although the above research results have obvious application effects in their respective research fields, there is still significant room for improvement in solving the problem of collaborative coverage path planning for UAV swarms in discrete environments, while simultaneously reducing path repetition rate and energy consumption.

3 Framework of the Model

3.1 Enhanced Approximate Cell Decomposition

The coverage task in this study is conducted on a discretized map. However, real-world environments are often irregular, which results in many redundant grids during the discretization process. In irregular terrain environments, to improve task execution efficiency, we have adopted an innovative strategy: rotating the target area to minimize the decomposed grid map, thereby minimizing task execution time.

Fig. 1 illustrates this enhanced approximate cell decomposition method, where black lines represent the boundaries of the irregular target area, red grids represent the area that UAVs need to cover, black blocks represent obstacles in the environment, and gray grids represent no-fly zones. Fig. 1a illustrates the initial situation of cell decomposition in an irregular area. To optimize the terrain redundancy issue, we first identify all vertices (x_i, y_i) of the target area and select the longest edge AB. Point A is chosen as the reference point, and the entire target area is translated using Eq. (1) to obtain coordinates $\left(x_i^{'}, y_i^{'}\right)$, as shown in Fig. 1b. Then, using edge AB as the rotation axis, the entire target area is rotated using Eq. (2) to obtain coordinates $\left(x_i^{''}, y_i^{''}\right)$, and where l_i is the length of edge AB, and θ_i is the rotation angle. Fig. 1c represents the final grid target area.

$$\left(x_i^{'}, y_i^{'}\right) = (x_i - x_a, y_i - y_a) \tag{1}$$

$$\left(x_i^{''}, y_i^{''}\right) = (l_i * \cos\theta_i, l_i * \sin\theta_i) \tag{2}$$

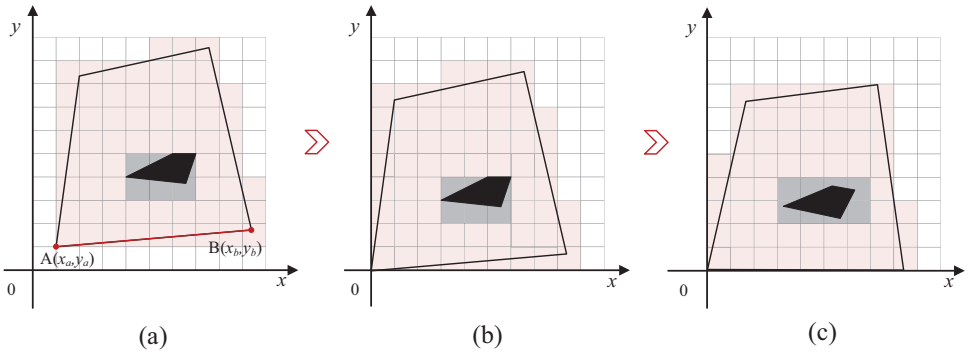

Figure 1: Illustration of enhanced approximate cell decomposition in irregular terrain

3.2 Multi-Scale Maps

UAV swarms often face issues such as trajectory overlap, coverage gaps, and collisions when performing coverage path planning in complex environments with obstacles. To address these challenges, we propose a method for integrating and updating multi-scale maps to achieve close cooperation among UAVs.

Specifically, a UAV swarm system consisting of N UAVs is established, where each UAV is denoted as $UAV_i, (i \in N)$. These UAVs are tasked with coverage missions in a $l \times w$ grid area, with each grid's geometric center denoted as $P = (x, y)$, where $x \in \{1, 2, \ldots, l\}$, $y \in \{1, 2, \ldots, w\}$. Thus, the position information of the i-th UAV can be represented as $P_i = (x, y)$. Let G represent the target grid area in the irregular environment where CPP needs to be performed, requiring $P_i \in G$. When a UAV reaches the position above point P it indicates that the current grid has been fully covered. Let $Covr_{x,y}$ represent the environmental information status value of the current grid: $Covr_{x,y} = 0$ indicates that the grid has not been covered by any UAV, $Covr_{x,y} = 1$ indicates that the grid has been covered, and $Covr_{x,y} = -1$ indicates that the UAV has detected an obstacle in the grid. Therefore, the coverage status matrix map_i of the i-th UAV can be represented as follows:

$$map_i = \begin{pmatrix} Covr^i_{1,1} & \cdots & Covr^i_{1,w} \\ \vdots & \ddots & \vdots \\ Covr^i_{l,1} & \cdots & Covr^i_{l,w} \end{pmatrix} \tag{3}$$

In this equation, $Covr^i_{x,y}$ represents the coverage status of UAV_i at grid (x, y). The process of multi-scale map fusion and update for the UAV swarm can be represented using the coverage status matrix:

$$map_i \xleftarrow{update} map\left(\max_{i \in N} Covr^i_{x,y}\right) \tag{4}$$

Fig. 2 shows the system framework for three UAVs working together on the CPP task. UAVs interact with the environment, choose actions, and receive rewards. Specifically, we use local environmental maps to record the areas covered by each UAV and no-fly zones. At each time step, UAVs exchange local environmental map information with the ground station and update the multi-scale maps, enabling action decisions based on global observations.

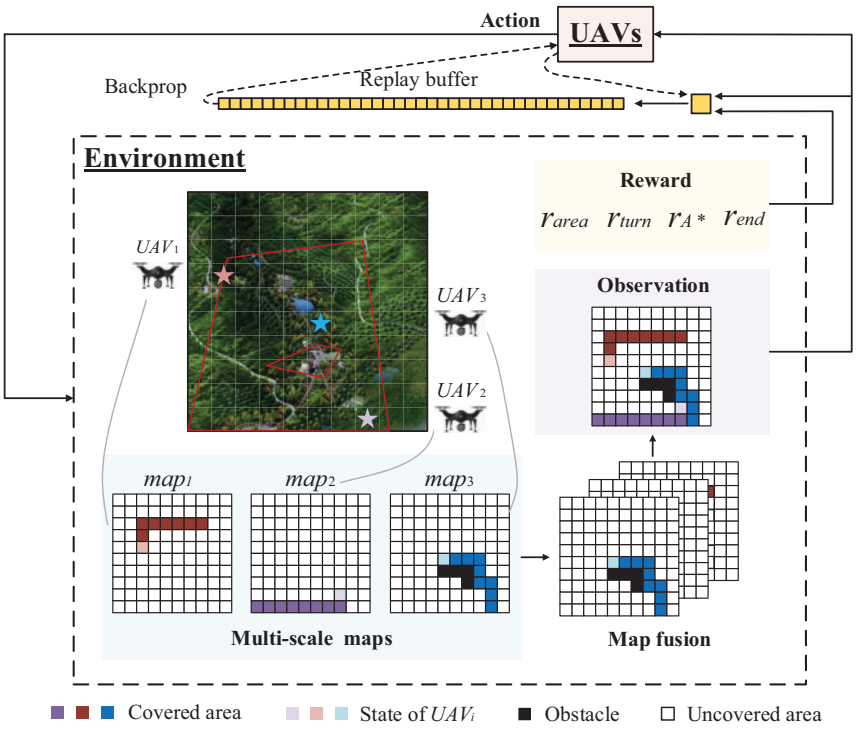

Figure 2: Illustration of the CPP system framework

3.3 Performance Metrics

In summary, this paper adopts a distributed approach, allowing each UAV to take off simultaneously from different initial positions. At each step, UAVs collaborate using multi-scale map integration while independently making action decisions. During this process, each UAV only needs to cover a portion of the target area, completing its sub-task. Therefore, the completion time of the last UAV's sub-task determines the total completion time of the CPP task. This paper assumes that the UAV maintains a constant speed during flight and uses the number of task completion steps $Step$ as an important metric to evaluate the CPP problem, which can be expressed as follows:

$$Step = \max_{i \in N} step_i \tag{5}$$

where $step_i$ represents the number of steps required for UAV_i to complete its subtask. We need to set multiple metrics for precise measurement to better assess the coverage task performance. First, we define P^{lapped} as the set of all duplicate grids within the target area, and P_i^{cover} as the set of all grids covered by the path of UAV_i. Thus, the difference between the two represents the effective coverage grid set. Based on this, we define the coverage efficiency of the UAV swarm CPP task as follows:

$$\eta = \frac{\sum_{i=1}^{N} f\left(P_i^{cover} - P^{lapped}\right)}{Step} \tag{6}$$

where the function $f(\bullet)$ is used to obtain the number of elements in a set. The CPP problem for the UAV swarm requires maximizing coverage efficiency within the target area, meaning minimizing the task completion steps. Additionally, the number of turns in the path needs to be counted f_{turn}.

4 Distributed SADQN Algorithm

4.1 DQN Algorithm

The Deep Q-Network (DQN) algorithm aims to learn the state-action value (Q-value) function through a deep neural network to find the optimal policy, enabling the agent to take appropriate actions in each state to maximize future cumulative rewards. The update process of its Q-value function is as follows:

$$Q(s,a;\theta) = r + \gamma \max_{a'} Q(s',a';\theta^{-}) \tag{7}$$

The parameters θ in Eq. (7) are the weights of each layer in the neural network, γ is the discount factor, and r is the reward value for the current action. The core idea of the DQN algorithm is to minimize the loss function and continuously adjust the weights θ of the neural network using gradient descent, so that the Q-values output by the network are as close as possible to the actual Q-values. The loss function can be defined as follows:

$$L(\theta) = E_{s,a,r,s'}\left[\left(r + \gamma \max_{a'} Q(s',a';\theta^{-}) - Q(s,a;\theta)\right)^2\right] \tag{8}$$

Specifically, DQN uses two neural networks with the same structure but different parameters: the online network and the target network. In Eq. (8), $Q(s,a;\theta)$ represents the output of the online network, which is used to evaluate the Q-value of the current state-action pair; $Q(s',a';\theta^{-})$ represents the output of the target network, and the target Q-value calculation process is reflected in Eq. (7). The parameters are then updated based on the loss function, with the online network updating its parameters at each iteration, while the target network updates its parameters only periodically.

In a given state, the agent in a reinforcement learning task can only choose one action, either exploitation or exploration. Therefore, a balance between the two needs to be achieved in order to obtain the optimal task outcome. To further improve learning efficiency, this paper improves the ε-greedy strategy. This paper proposes a swarm A*-guided Deep Q-Network (SADQN) algorithm, in which a heuristic action guidance mechanism based on the A* algorithm is introduced during the DQN training process to replace the traditional random action selection strategy. This improved action selection strategy not only preserves the adaptive exploration ability of reinforcement learning but also accelerates the accumulation of effective experiences through heuristic guidance. The ε-greedy strategy designed in this paper is as follows:

$$a_t = \begin{cases} \text{A}^{\star} & ,\rho < \varepsilon \\ \arg\max_{a} Q(s,a;\theta) & ,\rho \geq \varepsilon \end{cases} \tag{9}$$

where ρ is a randomly generated number between 0 and 1, and in this paper, ε is set to 0.9. When $\rho < \varepsilon$, the UAVs select the action with the maximum Q-value based on the Q-network; otherwise, it makes action decisions with the A* algorithm. Subsequently, we introduce how the A* algorithm selects actions for coverage path planning. Each UAV needs to plan its path based on the globally observed map fused from multi-scale maps. Starting from the current position P_i, the objective is to minimize the following:

$$f(x) = g(x) + h(x) \tag{10}$$

where $g(x)$ represents the cost incurred from the starting point P_i to the current position x_i, which is the sum of the cost of each action g_i taken from the starting point to the current position, where $g_i \in G$ and $G = \{0, 0.1, 0.4, 0.2, 0.2\}$. $h(x)$ represents the estimated cost at point x, which is set as the heuristic cost of the given position, i.e., the Manhattan distance between this position and the starting point.

4.2 State Space, Action Space, Reward

The state space, action space, and instantaneous reward for the UAV_i at each time step t can be respectively represented as s_i^t, a_i^t, r_i^t.

a. State space and action space

In a discrete grid map, the position of the UAV_i at time step t can be denoted as $P_i^t = (x, y)$. In the distributed approach, to enhance cooperation, the state space of each UAV is relatively complex, including the current position, action decisions, and local environmental map information. Therefore, the state vector of UAV_i can be represented as $s_i^t = \{P_i^t, a_i^t, map_i^t\}$.

The algorithm discretizes the UAV's action space, where the UAV can move diagonally through horizontal and vertical actions. Thus, the action set of the UAVs is represented as $A = \{a_0, a_1, a_2, a_3, a_4\}$. Where a_0 represents the stationary action, and a_1, a_2, a_3, a_4 represent actions moving upwards, downwards, leftwards, and rightwards, respectively. Therefore, the action of UAV_i at time t can be represented as $a_i^t \in A$.

b. Reward

The core objective of the CPP for the UAV swarm is to maximize coverage efficiency while ensuring safe flight operations. To achieve this goal, we have designed multiple task-oriented reward functions to guide the UAV swarm in achieving optimal collision-free coverage. Specifically, to avoid path repetition and collisions, we have designed a reward based on effective coverage, denoted as r_{area}:

$$r_{area} = \begin{cases} 1, map\,(x, y) = 0 \\ -1, map\,(x, y) = 1 \\ -10, map\,(x, y) = -1 \end{cases} \tag{11}$$

In Eq. (11), $map\,(x, y)$ represents the environmental state information value of the current grid. The UAV_i will be assigned different coverage reward values based on the different environmental state of the current grid. In addition, frequent turns by the UAV_i can lead to additional energy consumption, significantly increasing energy usage. To reduce mission energy consumption, we have designed a turn-based reward r_{turn}:

$$r_{turn} = \begin{cases} 0.5, a = a_1 \\ -0.5, a = a_3, a_4 \\ -1, a = a_0, a_2 \end{cases} \tag{12}$$

In Eq. (12), the UAV_i is assigned different turning reward values according to its selected flight action. To incentivize the UAVs to use the A* planned path in the early stages of training and accelerate the training process, a positive reward $r_{A*} = 0.1$ is provided when each UAV employs the A* algorithm for action decision-making. And when the UAV swarm completes the CPP task, each UAV receives a reward $r_{end} = 1$ for reaching its final position state.

The total reward r_i^t obtained by the UAVs at each time step is composed of the four components mentioned above, namely:

$$r_i^t = r_{area} + r_{turn} + r_{A*} + r_{end} \tag{13}$$

4.3 Prioritized Experience Replay

The SADQN algorithm of this paper deployed on UAV_i has its Q-network taking the current state s_i as input and outputting the estimated Q-value $Q\,(s_i, a_i, \theta_i)$; the target Q-network takes the next state $s_i^{'}$ as

input and outputs the maximum Q-value $\max_{a_i'} Q\left(s_i', a_i'; \theta_i^-\right)$. Therefore, the target Q-value can be represented as Eq. (14).

$$y_i = \begin{cases} r_i & , is\ end \\ r_i + \gamma \max_{a_i'} Q\left(s_i', a_i'; \theta_i^-\right), & otherwise \end{cases} \tag{14}$$

To effectively utilize learning experiences, we adopt a prioritized experience replay strategy to improve learning efficiency and stability during training, accelerating the algorithm's convergence. In the SADQN algorithm proposed in this paper, the UAV_i takes action a_i in the current state s_i, transitions to the next state s_i', and receives the corresponding reward r_i. The experience sample $\left(s_i, a_i, r_i, s_i'\right)$ is then stored in the experience replay buffer D_i of UAV_i. UAV_i selects experience samples from its replay buffer based on the TD-error of the j-th sample to determine the sampling probability. The TD-error $\delta_i^{(j)}$ represents the difference between the current Q-value and the target Q-value, and it is defined as follows in Eq. (15):

$$\delta_i^{(j)} = \left(y_i - Q\left(s_i, a_i; \theta_i\right)\right)^{(j)} \tag{15}$$

$$P_i(j) = \frac{\left(p_i^{(j)}\right)^{\alpha}}{\sum_k \left(p_i^{(k)}\right)^{\alpha}} \tag{16}$$

$$p_i^{(j)} = \left|\delta_i^{(j)}\right| + \varepsilon \tag{17}$$

$$\omega_i^{(j)} = \alpha \cdot \left(n \cdot p_i^{(j)}\right)^{-\beta} \tag{18}$$

Eq. (16) defines the sampling probability for each sample, where $p_i^{(j)}$ represents the priority of the i-th sample in the experience pool, as defined in Eq. (17). Additionally, k represents the number of samples in the experience pool, and the parameter α controls the strength of prioritized replay. Eq. (18) defines the importance weight $\omega_i^{(j)}$, which is used to eliminate the bias introduced by prioritized experience replay. The parameter β determines the intensity of bias correction. Therefore, the loss function considering the priority of experience samples can be defined as:

$$L_i(\theta_i) = E\left[\omega_i^{(j)} \cdot \left(\delta_i^{(j)}\right)^2\right] \tag{19}$$

$$\theta_i \leftarrow \theta_i + \alpha \delta_i^{(j)} \cdot \nabla_{\theta_i} \left(Q\left(s_i, a_i; \theta_i\right)\right)^{(j)} \tag{20}$$

Eq. (20) is used to update the parameters θ_i of the Q-network. Every C steps, the parameters of the Q-network are copied to the target network (i.e., $\theta_i^- = \theta_i$), and the target Q-values y_i are generated using the target Q-network for the subsequent C steps. The detailed process of the SADQN algorithm is as follows (Algorithm 1):

Algorithm 1: Distributed SADQN algorithm

Input: number of UAVs N, number of episodes E, number of steps T, minibatch k, frequency of target-update C, experience pool capacity M

Initialize: initialize the target area observe and the initial position of UAVs, the replay memory $D_i \leftarrow \varnothing$, initialize action-value function Q with random weights θ_i, initialize target action-value function Q with weights $\theta_i^- = \theta_i$

(Continued)

Algorithm 1 (continued)

For $episode = 1, E$ **do**
  Initialize sequence s_i^0
For $step = 1, T$ **do**
  For $UAV_i = 1, N$ **do**
    With probability ε select $a_i^t = argmax\left[Q\left(s_i^t, a_t; \theta_i\right)\right]$ otherwise, select the action calculated by A*
    Execute action a_i^t in emulator and observe reward r_i^t and next state s_i^{t+1}
    If collide or complete the coverage task **then**
      break
    Store transition $\left(s_i^t, a_i^t, r_i^t, s_i^{t+1}\right)$ in $D_i(|D_i| > M,$ remove old samples)
    Sample k transitions from D_i, calculate the transition priority by Eq. (17) and the weight by Eq. (18)
    Perform a gradient descent step to the network parameters θ_i by Eq. (20)
    Compute TD-error by Eq. (15) and update transition priority
    Every C steps reset $\theta_i^- = \theta_i$
  End For
  update the state ${s_i}^{t+1} \leftarrow {s_i}^t, i = 1, 2, \ldots, N$
End For
End For

5 Simulation Result

This section validates the correctness and effectiveness of the proposed SADQN method by comparing its results with those of six other solutions, including GBNN, A*, TMSTC*, SADQN-nA, SADQN-nP, and DDPG. All algorithms were run on Python 3.9 on a Windows 11 system. The specific parameter settings of the algorithm are as follows: the learning rate $lr = 0.0001$, the discount factor $\gamma = 0.85$, the batch size for stochastic gradient $B = 64$, the experience replay buffer capacity $M = 1000000$, the neural network parameters θ_i for each UAV are randomly initialized, the target network update frequency $C = 4$, the parameter α for prioritized experience replay is set to 0.6, and the parameter β is set to 0.7.

5.1 Analysis Performance of SADQN

We map the UAV flight trajectories onto a two-dimensional plane. Fig. 3a shows the irregular environment, where the initial discretized map is shown in Fig. 3b, and the final optimized map is presented in Fig. 3c. And in the irregular environment depicted in Fig. 3c, three UAVs are deployed to perform the coverage path planning task. The green dashed lines represent the starting positions of the UAVs.

As shown in Fig. 4, the final coverage results of different algorithms are presented. We can observe that all the algorithms have completed the coverage task. Moreover, the paths generated by the proposed SADQN method and the A* algorithm show no overlapping segments, while significant repeated coverage is observed in the path maps of the other algorithms.

As shown in Table 1, the average results of the CPP task under different algorithms are presented. From this, we can observe that in the irregular environment, the average coverage efficiency of the proposed SADQN method can reach 2.798, which indicates that it effectively balances the workload distribution among the UAVs while minimizing repeated coverage. Notably, the task completion efficiency of our method is superior to that of the GBNN, A*, and TMSTC* algorithms. Where the GBNN algorithm exhibits a significant amount of redundant path planning. While the A* and TMSTC algorithms demonstrate high coverage

efficiency, they fall short of the SADQN algorithm in terms of the number of task completion steps, primarily due to their lack of a collaborative mechanism. Our SADQN algorithm achieves cooperative coverage among UAVs by sharing environmental information through multi-scale maps, making it better suited for complex environments and resulting in superior path planning. Additionally, the SADQN algorithm performs better in terms of step count and number of turns, benefiting from our turn reward mechanism, which effectively reduces energy consumption caused by frequent turns. DDPG adopts a deterministic policy, making it prone to getting stuck in local optima during exploration. Although it enhances exploration by adding noise, this mechanism is insufficient in large state spaces and irregular environments, resulting in slow learning and suboptimal performance.

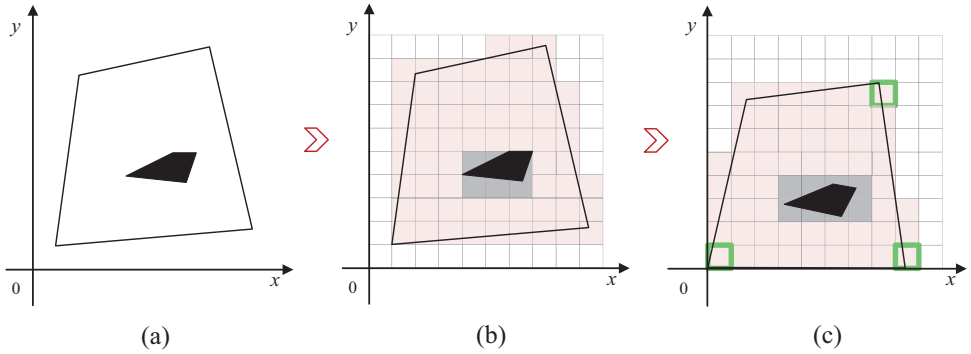

Figure 3: Schematic diagram of the discretized target area

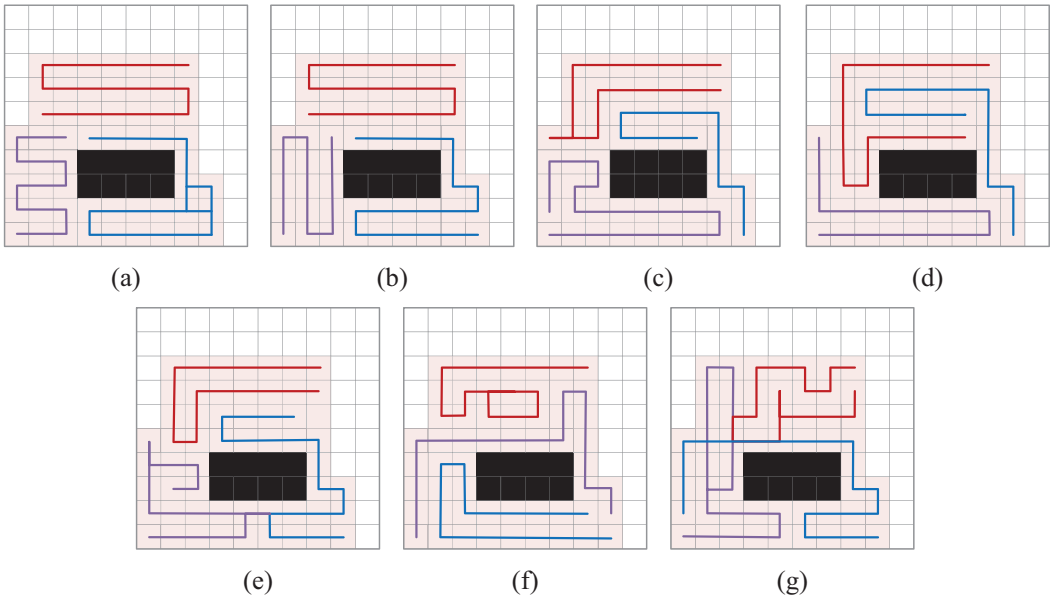

Figure 4: Comparison of coverage results for different algorithms. (a) GBNN; (b) A*, (c) TMSTC*; (d) SADQN; (e) SADQN_nP; (f) SADQN_nA; (g) DDPG

Table 1: Results of completion steps, coverage efficiency, and turning counts

Algorithm	Completion steps	Coverage efficiency η	Turning counts
GBNN	29	1.517	21
A*	21	2.667	17
TMSTC*	22	2.409	17
SADQN	19.5	2.798	15.25
SADQN_nP	20.75	2.490	20.25
SADQN_nA	21.25	2.408	17.50
DDPG	23.5	2.062	26

Fig. 5 illustrates the reward variation curves of four different algorithms during training, with the shaded areas representing the standard deviation of the reward values, reflecting the range of reward fluctuations. From Fig. 5, we can observe that the SADQN algorithm converges the fastest, stabilizing after 200 episodes. The reward values of the SADQN_nA algorithm are relatively low, indicating that the A*-guided algorithm provides effective prior knowledge, accelerating the training process. The reward increase of the SADQN_nP algorithm is slower, suggesting that prioritized experience replay effectively utilizes important experiences, enhancing training efficiency. The combination of the A* algorithm and prioritized experience replay makes the SADQN algorithm perform the best. The DDPG algorithm converges slowly with lower reward values, primarily because its exploration mechanism and action selection methods are unsuitable for complex environments, resulting in significantly inferior performance.

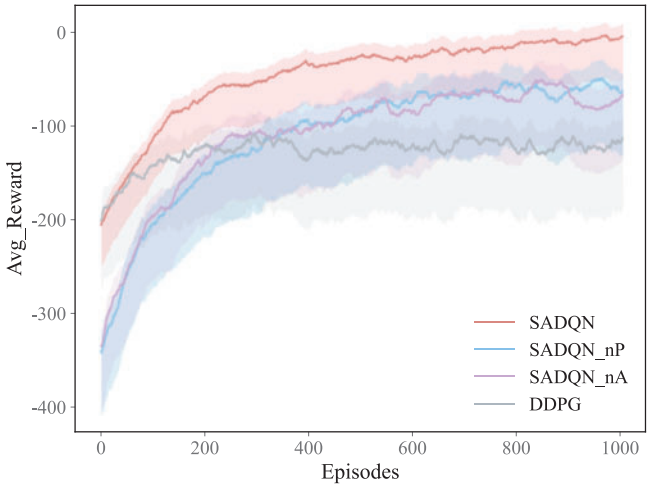

Figure 5: Reward curve comparison graph

The UAV swarm may encounter unexpected situations such as malfunctions, energy depletion, or communication interruptions while performing complex tasks, causing some UAVs to stop working. In such cases, ensuring full coverage becomes crucial. As shown in Fig. 6, in a coverage task coordinated by three UAVs, one UAV stops working due to an unexpected issue. The remaining two UAVs can quickly adjust their strategies and take over the coverage task for the entire area to ensure the successful completion of full coverage.

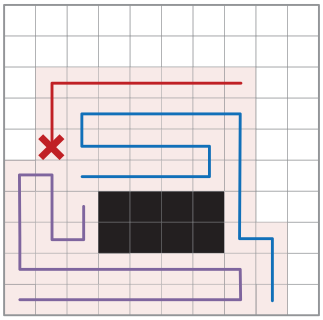

Figure 6: Response of the UAV swarm to the failure situation

5.2 Generalization Ability

To validate the algorithm's generalization ability, this section compares the results of the CPP problem in two irregular environments, Env1 and Env2, as shown in Fig. 7a,b. It can be observed that the number of coverage grids in the target area was reduced by 10.1% and 9.1%, respectively, after processing with the improved cell decomposition method. In this section, three UAVs were deployed to perform coverage path planning tasks in two experimental scenarios shown in Fig. 7a,b. The green lines outline the starting positions of each UAV, and they will collaborate to achieve full coverage of the target area.

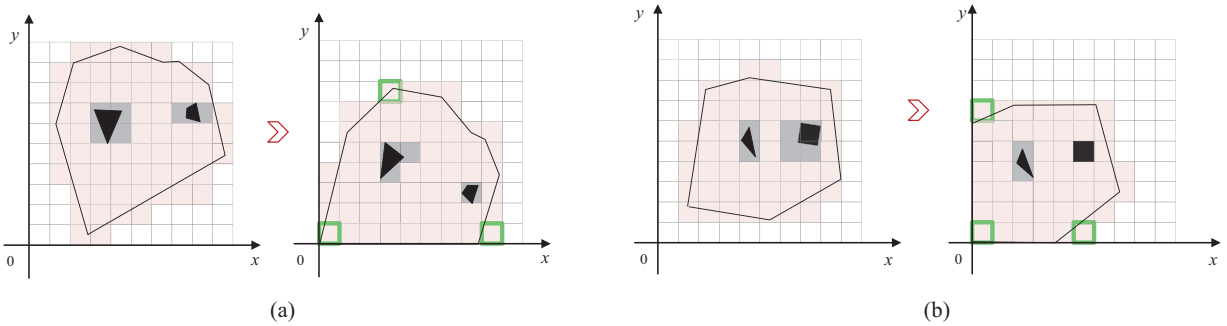

Figure 7: Discretization of irregular environments

Table 2 presents the completion steps, coverage efficiency, and turning counts. From Table 2, it can be observed that all algorithms achieved complete coverage. Our proposed SADQN method performed well in both environments, with minimal occurrence of duplicate coverage. Compared to other algorithms, SADQN demonstrated lower average steps to completion steps, further enhancing its coverage efficiency, with an average coverage efficiency exceeding 2.6. Moreover, while our method ensures a high coverage efficiency through reasonable path planning, it also involves fewer turns, enabling more efficient coverage path planning.

Table 2: Experimental results of coverage path planning in different environments

Irregular environments	Algorithm	Completion steps	Coverage efficiency η	Turning counts
	GBNN	26	2.077	27
	A*	27	2.074	18

(Continued)

Table 2 (continued)

Irregular environments	Algorithm	Completion steps	Coverage efficiency η	Turning counts
Env1	TMSTC*	22	2.409	19
	SADQN	21	2.661	18.5
	SADQN_nP	23.5	2.261	24.75
	SADQN_nA	24.25	1.717	27.75
	DDPG	24.75	1.887	27.75
Env2	GBNN	22	1.636	23
	A*	18	2.389	17
	TMSTC*	19	2.053	24
	SADQN	16.75	2.685	17
	SADQN_nP	18.25	2.340	18.5
	SADQN_nA	18.25	2.388	19
	DDPG	19	2.168	22.85

6 Conclusion

This research initiates with the discretization of irregular environments through an enhanced cell decomposition technique, thereby effectively partitioning the complex terrains into manageable grids. Simultaneously, the observation scope of UAVs is expanded by leveraging multi-scale maps, endowing them with a broader field of view to better perceive and adapt to the surroundings. An inventive SADQN algorithm is put forth, which ingeniously incorporates the A* algorithm to bolster the DQN process, specifically devised to tackle the intricate coverage path planning challenges that UAV swarms encounter in complex and irregular settings. The proposed algorithm has been rigorously validated within obstacle-ridden irregular environments. The experimental outcomes unequivocally demonstrate its superiority over existing benchmark algorithms. In terms of convergence speed, it exhibits a remarkable acceleration, swiftly arriving at optimal solutions and reducing the computational burden. Regarding completion steps, it streamlines the overall process, minimizing unnecessary maneuvers and enhancing operational efficiency. The coverage efficiency soars to an impressive 2.6, signifying a significant leap in the thoroughness of area coverage. Moreover, the algorithm also considers the turning counts, optimizing flight paths to curtail energy consumption associated with frequent directional changes. Notably, in the face of sudden failures, the resilience of the system shines through. The remaining UAVs are still capable of tenaciously adhering to the original CPP task objectives, ensuring the continuity and integrity of the mission. This robustness adds an extra layer of reliability to UAV swarm operations, especially in critical or unpredictable scenarios.

Acknowledgement: The authors are thankful to researchers in Jiangsu University of Science and Technology for the helpful discussion.

Funding Statement: The authors received no specific funding for this study.

Author Contributions: The authors confirm contribution to the paper as follows: study conception and design: Zhuoyan Xie, Qi Wang; data collection: Zhuoyan Xie; analysis and interpretation of results: Zhuoyan Xie, Bin Kong; draft manuscript preparation: Zhuoyan Xie, Shang Gao. All authors reviewed the results and approved the final version of the manuscript.

Availability of Data and Materials: The data that support the findings of this study are available from the corresponding authors, upon reasonable request.

Ethics Approval: Not applicable.

Conflicts of Interest: The authors declare no conflicts of interest to report regarding the present study.

Nomenclature

l, w	Length and width of the target area
x, y	Target area coordinate
G	Target grid area
N	Total number of Unmanned Aerial
UAV_i	The i-th Unmanned Aerial Vehicle
P_i	The i-th UAV's grid center coordinates
$Covr_{x,y}$	Environmental information state value
map_i	The i-th UAV's coverage state matrix
$step_i$	The i-th UAV's number of steps
η	Coverage efficiency
A	Set of the UAV actions
s, a, r	State, action, and reward of the DQN algorithm
s_i^t, a_i^t, r_i^t	State, action, and reward of the UAV_i at step t
θ	DQN neural network parameters
$Q(s, a; \theta)$	Value function of the DQN algorithm
y_i	Target Q-value
$L(\theta)$	Loss function
ε	Greedy algorithm parameters

References

1. Cabreira TM, Brisolara LB, Paulo RFJ. Survey on coverage path planning with unmanned aerial vehicles. Drones. 2019;3(1):4. doi:10.3390/drones3010004.
2. Mukhamediev RI, Yakunin K, Aubakirov M, Assanov I, Kuchin Y, Symagulov A, et al. Coverage path planning optimization of heterogeneous UAVs group for precision agriculture. IEEE Access. 2023;11:5789–803. doi:10.1109/access.2023.3235207.
3. Pérez-González A, Benítez-Montoya N, Jaramillo-Duque Á, Cano-Quintero JB. Coverage path planning with semantic segmentation for UAV in PV plants. Appl Sci. 2021;11(24):12093. doi:10.3390/app112412093.
4. Muñoz J, López B, Quevedo F, Monje CA, Garrido S, Moreno LE. Multi UAV coverage path planning in urban environments. Sensors. 2021;21(21):7365. doi:10.3390/s21217365.
5. Cabreira TM, Di Franco C, Ferreira PR, Buttazzo GC. Energy-aware spiral coverage path planning for UAV photogrammetric applications. IEEE Robot Autom Lett. 2018;3(4):3662–8. doi:10.1109/lra.2018.2854967.
6. Nedjati A, Izbirak G, Vizvari B, Arkat J. Complete coverage path planning for a multi-UAV response system in post-earthquake assessment. Robotics. 2016;5(4):26. doi:10.3390/robotics5040026.
7. Collins L, Ghassemi P, Esfahani ET, Doermann D, Dantu K, Chowdhury S. Scalable coverage path planning of multi-robot teams for monitoring non-convex areas. In: Proceedings of the 2021 IEEE International Conference on Robotics and Automation (ICRA); 2021 May 30–Jun 5; Xi'an, China. Piscataway, NJ, USA: IEEE; 2021. p. 7393–9.
8. Bolu A, Korçak Ö. Path planning for multiple mobile robots in smart warehouse. In: Proceedings of the 2019 7th International Conference on Control, Mechatronics and Automation (ICCMA); 2019 Nov 6–8; Piscataway, NJ, USA: IEEE; 2019. p. 144–50.

9. Luna MA, Molina M, Da-Silva-Gomez R, Melero-Deza J, Arias-Perez P, Campoy P. A multi-UAV system for coverage path planning applications with in-flight re-planning capabilities. J Field Robot. 2024;41(5):1480–97. doi:10.1002/rob.22342.
10. Jonnarth A, Zhao J, Felsberg M. End-to-end reinforcement learning for online coverage path planning in unknown environments. arXiv:2306.16978. 2023.
11. Heydari J, Saha O, Ganapathy V. Reinforcement learning-based coverage path planning with implicit cellular decomposition. arXiv:2110.09018. 2021.
12. Cho SW, Park JH, Park HJ, Kim S. Multi-uav coverage path planning based on hexagonal grid decomposition in maritime search and rescue. Mathematics. 2021;10(1):83. doi:10.3390/math10010083.
13. Lu J, Zeng B, Tang J, Lam TL, Wen J. TMSTC*: a path planning algorithm for minimizing turns in multi-robot coverage. IEEE Robot Autom Lett. 2023;8(8):5275–82. doi:10.1109/lra.2023.3293319.
14. Maza I, Ollero A. Multiple UAV cooperative searching operation using polygon area decomposition and efficient coverage algorithms. In: Distributed autonomous robotic systems. Vol. 6. Tokyo, Japan: Springer; 2007. p. 221–30 doi: 10.1007/978-4-431-35873-2_22.
15. Chen G, Shen Y, Zhang Y, Zhang W, Wang D, He B. 2D multi-area coverage path planning using L-SHADE in simulated ocean survey. Appl Soft Comput. 2021;112(7):107754. doi:10.1016/j.asoc.2021.107754.
16. Xu S, Zhou Z, Li J, Wang L, Zhang X, Gao H. Communication-constrained UAVs coverage search method in uncertain scenarios. IEEE Sens J. 2024;24(10):17092–101. doi:10.1109/jsen.2024.3384261.
17. Qiu X, Song J, Zhang X, Liu S. A complete coverage path planning method for mobile robot in uncertain environments. In: Proceedings of the 2006 6th World Congress on Intelligent Control and Automation; 2006 Jun 21–23; Dalian, China. Piscataway, NJ, USA: IEEE; 2006. p. 8892–6.
18. Bine LM, Boukerche A, Ruiz LB, Loureiro AA. A novel ant colony-inspired coverage path planning for internet of drones. Comput Netw. 2023;235:109963. doi:10.1016/j.comnet.2023.109963.
19. Li W, Zhao T, Dian S. Multirobot coverage path planning based on deep Q-network in unknown environment. J Robot. 2022;2022(1):6825902. doi:10.1155/2022/6825902.
20. Le AV, Vo DT, Dat NT, Vu MB, Elara MR. Complete coverage planning using Deep Reinforcement Learning for polyiamonds-based reconfigurable robot. Eng Appl Artif Intell. 2024;138(4):109424. doi:10.1016/j.engappai.2024.109424.