Bi-Criteria Optimization for Balancing Communication Connectivity and Distance Based on β-DQN in UAV Trajectory

Xuan Liu , Zhexin Xu , Yi Wu

College of Photonic and Electronic Engineering, Fujian Normal University, China

INTRODUCTION

Unmanned Aerial Vehicles (UAVs) play a crucial role in search and rescue missions due to their flexibility and mobility. However, maintaining communication with Ground Control Stations (GCS) during maritime missions poses challenges. Optimizing UAV flight paths helps reduce communication interruptions and flight distance, saving time and energy. Traditional optimization algorithms improve path efficiency but often overlook communication issues. Deep Reinforcement Learning (DRL) performs well in this regard, but its reward settings often require multiple attempts, potentially affecting the strategy's optimization and effectiveness. This study proposes the β-DQN algorithm, which optimizes path distance and communication through an innovative reward and decay mechanism.

Problem Formulation

Grid-Based Map Construction for UAV Path Planning

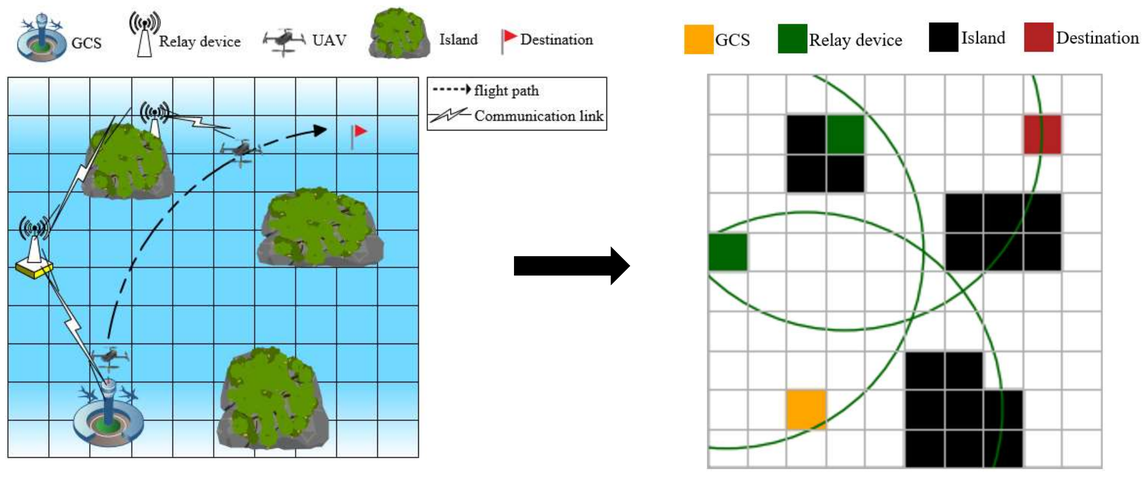


Fig. 1 The UAV starts from the GCS (yellow), aiming to reach the mission site (red) while avoiding island obstacles (black), which are above the UAV's flight altitude, and maintaining communication with the GCS. Path planning is grid-based: white cells are flyable areas, and green cells are relay nodes ensuring connectivity. The goal is to minimize both flight distance and communication loss, with distances measured by grid cells.

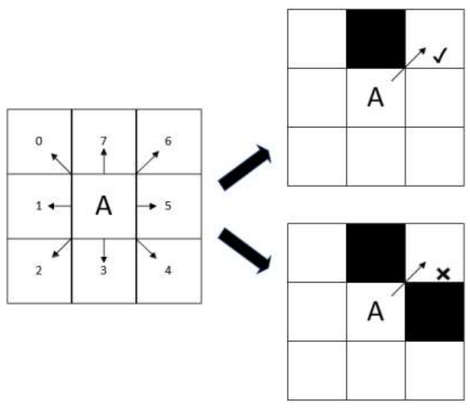


Fig. 2 In the grid-based scenario, the UAV can move to 8 adjacent grid cells, with an action space of {0, 1, 2, 3, 4, 5, 6, 7}. Black blocks represent obstacles, and if the UAV attempts to pass through one, it will remain in its current position. Diagonal movements may encounter dead-ends, meaning the UAV is blocked by obstacles on two sides and cannot proceed.

Total Path Cost Definition

Equation 1 The total path cost θ is a weighted sum of the flight distance ratio θ_{dis} and the communication loss ratio θ_{nc}. Here, θ_{dis} the ratio of the UAV's flight distance to that of the baseline algorithm, and θ_{nc} is the ratio of the communication loss distance to the baseline. The A* algorithm serves as the baseline for both metrics.

$$\theta = (1-n)\theta_{\text{dis}} + n*\theta_{\text{nc}} \qquad n \in [0,1]$$

$$\theta_{\text{dis}} = \frac{d(\tau)}{d(\tau_{A^*})}$$

$$\theta_{\text{nc}} = \frac{nc(\tau)}{nc(\tau_{A^*})}$$

Methodology

In traditional DQN, when the reward *R* for reaching the destination gradually decays over time, penalties for communication interruptions can outweigh these rewards, causing the action-value function $Q(s_t, a_t)$. to turn negative. Moreover, since neural networks typically initialize the action-value function to values close to zero, this makes it difficult to find a decision path significantly better than others in the early stages of training, thus affecting both the stability and convergence of learning. To address these issues, β-DQN introduces the following improvements.

This paper redefines the reward model, incorporating the communication interruption penalty into the reward decay, allowing the algorithm to converge more quickly and simplifying parameter tuning. The reward is divided into two parts: *r*, used to guide the path in the early stages and gradually decaying to zero with ε; and *R*, representing the final reward for reaching the destination, which is set much higher than *r* .The decay factor Γ applies to both flight distance and communication loss, with a communication interruption penalty β included in the decay, unifying it with the penalty for increased flight distance. This ensures faster convergence and simpler parameter tuning, as shown below.

$$r = (r_1 + r_2 + r_3)\cdot\varepsilon$$

$$r_1 = \begin{cases} -3 & a \in \{1,3,5,7\} \\ -5 & a \in \{0,2,4,6\} \end{cases}$$

$$r_2 = \begin{cases} 0 & |x_c' - x_d| + |y_c' - y_d| \le |x_c - x_d| + |y_c - y_d| \\ 8 & |x_c' - x_d| + |y_c' - y_d| - 1 = |x_c - x_d| + |y_c - y_d| \\ 12 & |x_c' - x_d| + |y_c' - y_d| - 2 = |x_c - x_d| + |y_c - y_d| \end{cases}$$

$$r_3 = \begin{cases} 0 & x_c' = x_c \wedge y_c' = y_c \\ -20 & else \end{cases}$$

$$\Gamma = \begin{cases} \gamma, & C_{\text{p}} = 1 \text{ and } a \in \{1,3,5,7\} \\ \gamma^{\sqrt{2}}, & C_{\text{p}} = 1 \text{ and } a \in \{0,2,4,6\} \\ \gamma^{1+\beta}, & C_{\text{p}} = 0 \text{ and } a \in \{1,3,5,7\} \\ \gamma^{\sqrt{2}(1+\beta)}, & C_{\text{p}} = 0 \text{ and } a \in \{0,2,4,6\} \end{cases}$$

Results

Performance Comparison of Different Algorithms

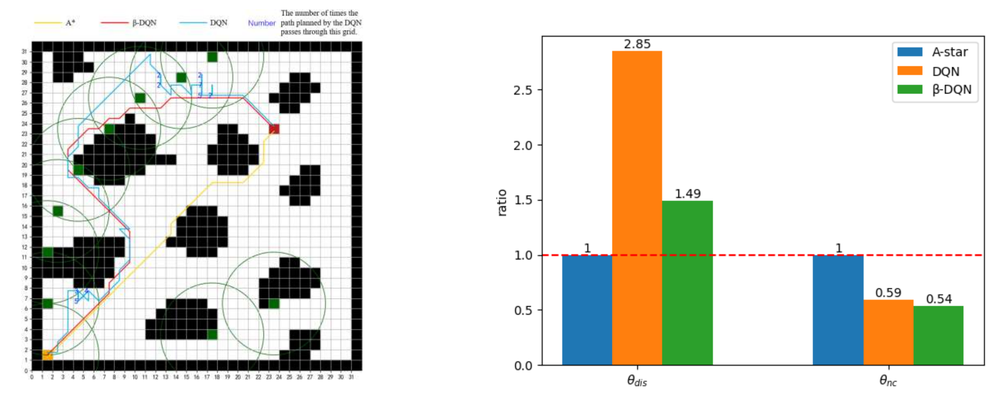


Fig. 3 The comparison shows that the β-DQN algorithm reduces grid revisits and performs better in planning compared to traditional methods like A* and DQN. It also achieves a lower total path cost by effectively minimizing no-communication areas, using θ_{dis} and θ_{nc} as metrics.

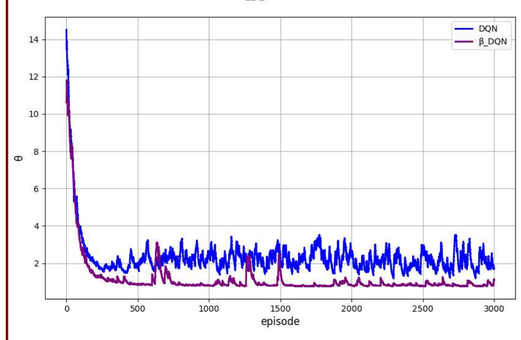
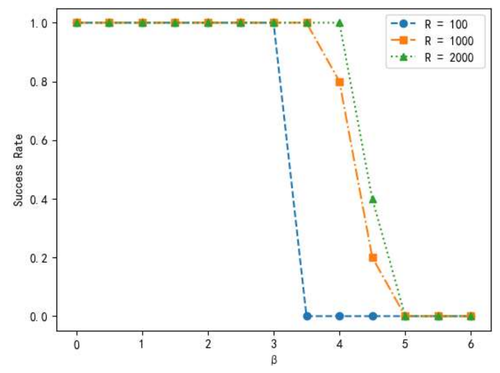


Fig. 4 The comparison shows that the improved β-DQN algorithm converges faster and more stably to the path with the lowest total cost, compared to traditional DQN. The traditional DQN tends to converge to suboptimal paths due to early penalties for communication interruptions. The improved algorithm incorporates these penalties into reward decay, enhancing both convergence speed and stability.

Effect analysis of β-DQN with different parameters

Fig. 5 The success criterion is reaching convergence within 3000 training episodes. Different *R* values affect the β range for successful path planning. For *R* = 100, 1000, and 2000, the ranges of β for 100% success are [0,3], [0,3.5], and [0,4]. As *R* increases, the effective β range expands, enabling broader successful planning.



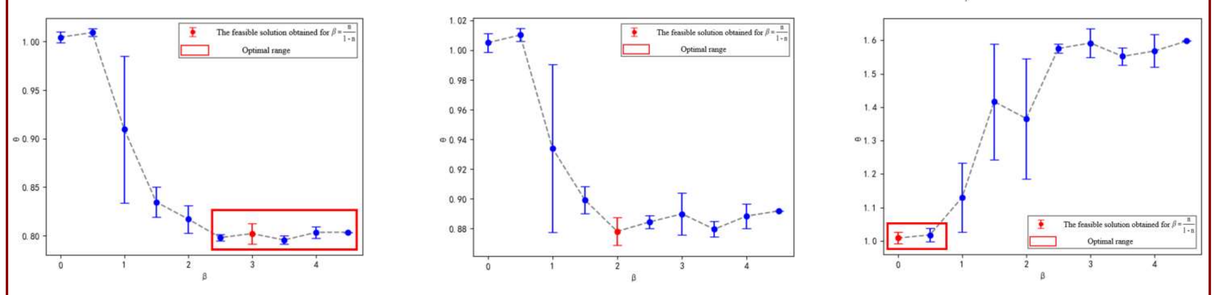


Fig. 6 This shows the relationship between *n and* β in the reward setting, where β is set as $n/(1\text{-}n)$. Optimal solutions are achieved when β matches this formula, with examples at *n* = 3/4 and $\beta = 3$, *n* = 1/3 and $\beta = 2$, and *n* = 0 with $\beta = 0$.

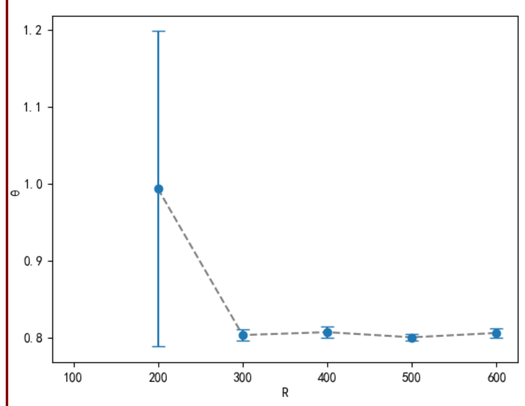


Fig. 7 The total path cost θ is affected by different *R* values with fixed β and *n*. When *R* is set to 100, path planning fails, while at $R{=}200$, some inconsistencies arise. As *R* increases further, the optimal path cost is achieved, but increasing *R* beyond a certain point does not improve results.

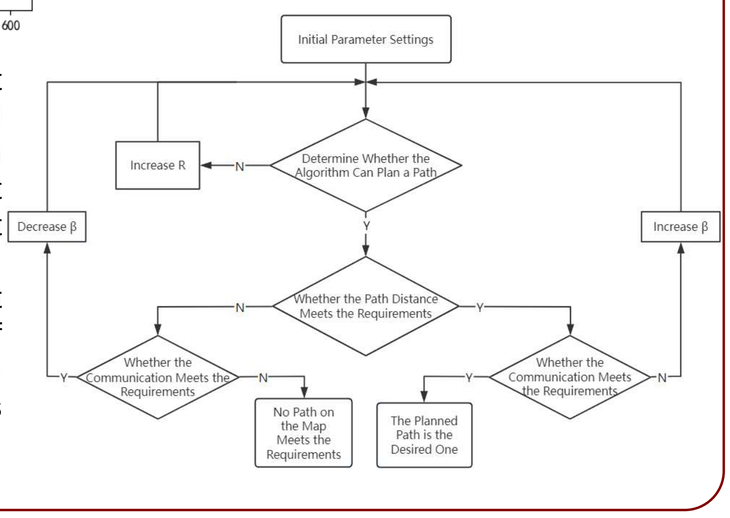


Fig. 8 In practical applications, it is recommended to set the initial *R* to 2000 and β to 0. If path planning fails, increase *R*. If flight distance is met but communication fails, increase β; if communication is met but flight distance fails, decrease β. If neither is met, no valid path exists. If both are met, the path is valid.

Conclusion

In maritime UAV path planning, minimizing flight distance and communication interruption is key. This paper introduces β-DQN, which incorporates communication interruption penalties. This adjustment prevents the optimal path's action-value function from becoming negative, improving early exploration. The value of β balances communication and path efficiency, leading to faster and more stable results. The study also verifies the relationship between β and *n,* showing that beyond a certain *R*, increasing *R* does not enhance optimization, and provides a guideline for practical tuning in real-world applications.