AN IMPROVED Q-LEARNING ALGORITHM WITH OPTIMIZED INITIALIZATION AND ANNEALED BOLTZMANN EXPLORATION

Tang Yi, Yu.O.German
Department of Information Technologies in Automated Systems,
Belarussian State University of Informatics and Radioelectronics
Minsk, Republic of Belarus
E-mail: tangyijcb@163.com, jgerman@bsuir.by

This paper proposes a hybrid enhancement method for Q-learning that combines direction-sensitive Q-table initialization with annealing-based Boltzmann exploration. Initialization leverages geometric priors to bias actions toward the target without leaking obstacle information; the annealing-based Boltzmann method achieves a smooth transition from extensive exploration to exploitation. By leveraging the symmetry of isometric states and an adaptive exploration strategy, the improved Q-learning algorithm achieves faster convergence in discrete action environments.

Introduction

In intelligent transportation systems (ITS), navigation and path planning are foundational capabilities that underpin applications such as autonomous driving, unmanned delivery, and warehouse logistics. Typical scenarios include efficient traversal within urban road networks, congestion and obstacle avoidance in warehouse grid environments, and policy evaluation in simulated traffic networks. Given the discrete layout of environments, constrained traversability, and complex dynamics, achieving efficient and robust path planning in unknown or partially known settings carries substantial research and engineering significance.

From an application perspective, an effective navigation strategy should satisfy three criteria: reachability, efficiency, and robustness. Methodologically, traditional rule-based or heuristic planners (e.g., fixed-cost search and hand-crafted policies) often require extensive parameter tuning and complete map information in large-scale, dynamically changing scenarios; they generalize poorly to novel layouts or constraints and incur high maintenance costs[1].

Against this backdrop, reinforcement learning (RL), with its capacity to learn adaptively through interaction with the environment, has emerged as a powerful approach to path planning in grid-based settings. Among RL methods, Q-learning method has been widely adopted for grid navigation with discrete actions due to its simplicity, computational lightness, and strong interpretability.

I. Weaknesses of existing methods

Despite substantial progress, Q-learning and its variants continue to face efficiency and robustness bottlenecks in large-scale, obstacle-dense, and sparse-reward grid environments. First, the lack of reliable navigational priors at initialization leads to directionless early exploration when Q-tables are uniform or random. Although distance- or heuristic-based initializations can accelerate learning to some extent, they tend to over-bias the agent toward blocked directions under high obstacle density, and their benefits are highly map-dependent. Second, exploration mechanisms are largely heuristic and insensitive to learning progress. Specifically, fixed ϵ-greedy induces high variance and repeated probing, while Boltzmann/entropy-regularized policies or adaptive ϵ decay can improve stability but introduce additional hyperparameters and tuning burdens, and still exhibit oscillations, cul-de-sac trapping, and premature greediness in heavily cluttered settings[2]. In terms of scalability, approaches based on local-search decomposition and hybrids with evolutionary/swarm intelligence enhance global exploration but increase computational complexity and parameter sensitivity, limiting real-time deployment.

Overall, existing improvements partially alleviate exploration and reward sparsity issues, yet remain highly sensitive to priors and hyperparameters. As obstacle density and map scale grow, robustness and generalization remain inadequate.

II. Problem statement

We model path planning in a 2D grid environment as a discrete Markov Decision Process (MDP) $\langle \mathcal{S}, \mathcal{A}, \mathcal{P}, \mathcal{R}, \gamma \rangle$. On a grid of size $H \times W$, the state $s = (i, j) \in \{0, \ldots, H-1\} \times \{0, \ldots, W-1\}$ denotes the current position of agent, with a designated start s_{start} and goal s_{goal}. The action set $\mathcal{A} = \{\text{up}, \text{down}, \text{left}, \text{right}\}$ corresponds to four-connected movements.

The reward function is designed to jointly capture task completion, collision penalties, and time costs[3]. with the objective of maximizing the expected discounted return which can be formed below in Equation (1):

$$\max_{\pi} \; \mathbb{E}_{\pi} \left[\sum_{t=0}^{T_{\max}} \gamma^t r_t \;\middle|\; s_0 = s_{\text{start}} \right] \qquad (1)$$

where $\gamma \in [0, 1)$ is the discount factor and r_t is the instantaneous reward induced by the current transition. For small to medium discrete state spaces, we adopt tabular Q-learning as a baseline solver to estimate the action-value function in a model-free manner and iteratively improve the policy. The update rule is shown in Equation (2):

$$Q(s_t, a_t) \leftarrow Q(s_t, a_t) + \alpha\Big(r_t + \gamma \max_{a' \in A} Q(s_{t+1}, a') - Q(s_t, a_t)\Big) \tag{2}$$

where $\alpha \in (0, 1]$ is the learning rate. Exploration–exploitation trade-offs are handled via ϵ-greedy.

III. Method description

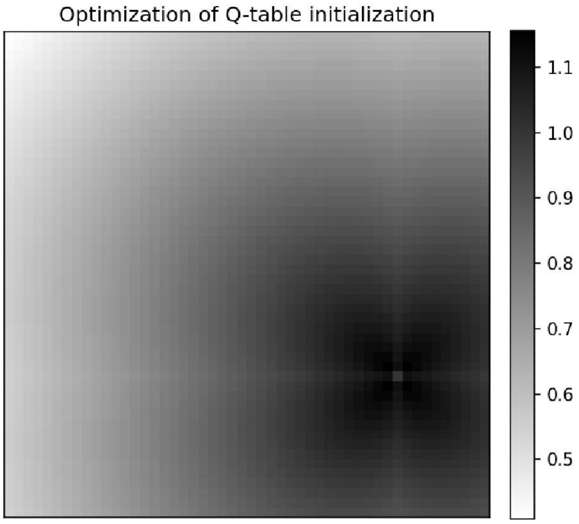

Figure 1 – Optimized initialization of Q table

Optimized Q Table Initialization: To mitigate the drawback of standard initialization that ignores navigational priors, we provide a warm start to reduce early blind exploration. Unlike zero or random initialization, we compute $Q_0(i, j, a)$ using goal direction and distance, where (i, j) denotes the state position, a denotes the action index, and Q_0 is the initial Q-value. The initialization is defined as in Equation(3):

$$Q_0(i, j, a) = \beta_0 + \beta_1 \left(1 - \hat{d}(i, j; g_i, g_j)\right) + \beta_2 \kappa(i, j, a) \tag{3}$$

where $\beta_0 = 0$ is a constant bias term, $\beta_1 = 0.8$ is the distance weight, and $\beta_2 = 0.5$ is the direction weight; $\hat{d}(i, j; g_i, g_j)$ is the normalized Manhattan distance (with (g_i, g_j) denoting the goal position), computed using $\hat{d}(i, j; g_i, g_j) = \frac{|i-g_i|+|j-g_j|}{(H-1)+(W-1)+\epsilon_0}$, ($\epsilon_0 = 1\text{e}{-8}$ is a small constant to avoid division by zero). $\kappa(i, j, a)$ is the action alignment score, computed via the dot product between the unit goal vector field $\vec{u}_g(i, j)$ and the unit action vector $\vec{u}_a$. The calculation is shown below in Equation (4):

$$\kappa(i, j, a) = \max\left(0, \vec{u}_g(i, j) \cdot \vec{u}_a\right) \tag{4}$$

This method biases actions that are closer to the goal and aligned in direction, significantly accelerating convergence. Figure 1 shows the visualization of optimized initialization of the table.

Annealed Boltzmann Exploration To address the limitations of standard ϵ-greedy with high noise and insufficient late-stage exploration. we adopt an annealed Boltzmann scheme whose temperature decays exponentially with episode e[4]. The calculation for temperature T_e is in Equation(5):

$$T_e = \max(T_{\min}, T_0 \cdot \rho^e) \tag{5}$$

where $T_0 = 1.5$ is the initial temperature, $T_{\min} = 0.02$ is the lower bound, and $\rho = 0.998$ is the decay rate. The action probability $P(a \mid s)$ is computed as:

$$P(a \mid s) = \frac{\exp\left(Q'(s, a)/T\right)}{\sum_{a' \in A} \exp\left(Q'(s, a')/T\right)} \tag{6}$$

To improve efficiency, we add a UCB bonus(Upper Confidence Bound bonus): $Q'(s, a) = Q(s, a) + c\sqrt{\ln(t + 1)/(N(s, a) + \epsilon)}$, where $Q'(s, a)$ is the adjusted Q-value, $c = 0.1$ is the exploration constant, t is the total time step count, and $N(s, a)$ is the state–action visit count. This mechanism encourages actions with higher uncertainty and promotes detour discovery in cluttered environments, contrasting with the inefficient exploration reported for standard Q-learning in computationally complex dynamic warehouses. Visit counts are tracked via state and action arrays, and the global step t increments per action.

IV. conclusion

We propose an improved Q-learning method for grid-based navigation, centered on the synergy between direction-sensitive Q-table initialization and annealed Boltzmann exploration. The former provides a mild yet explicit directional prior, while the latter implements a smooth, adaptive softening mechanism to transition from exploration to exploitation. Without revealing obstacle information and while retaining a model-free paradigm, the method is expected to improve sample efficiency and convergence stability.

1. Zhang, Y., Zhao, W., Wang, J., & Yuan, Y. (2024). Recent progress, challenges and future prospects of applied deep reinforcement learning: A practical perspective in path planning. Neurocomputing, 608, 128423.
2. Pei, M., An, H., Liu, B., & Wang, C. (2021). An improved dyna-q algorithm for mobile robot path planning in unknown dynamic environment. IEEE Transactions on Systems, Man, and Cybernetics: Systems, 52(7), 4415-4425.
3. Ben-Akka, M., Tanougast, C., & Diou, C. (2025). Novel design of reward and epsilon-greedy decay strategy tailored for Q-learning in optimizing local mobile robot path planning. Knowledge-Based Systems, 113836.
4. Zhou, Q., Lian, Y., Wu, J., Zhu, M., Wang, H., & Cao, J. (2024). An optimized Q-Learning algorithm for mobile robot local path planning. Knowledge-Based Systems, 286, 111400.