Dean&Francis

ISSN 2959-6157

Research on Obstacle Avoidance and Path of an Intelligent Robot Based on Reinforcement Learning

Sinan Qi

Department of Applied Physics,
Hohai University, Nanjing, Jiangsu,
213200, China
qisinan@lsu.edu.gn

Abstract:

This paper focuses on the application and verification of the Q-learning algorithm and the Sarsa algorithm in a robot obstacle avoidance scenario. With the increasing application of intelligent robots, the complex dynamic environment puts forward higher requirements for their obstacle avoidance ability. Traditional obstacle avoidance algorithms are difficult to adapt to a changing environment. Reinforcement learning shows strong adaptability and obstacle avoidance effects through the interaction between robots and the environment, which has become a current research hotspot. In this paper, based on Q-learning and the Sarsa algorithm, a Python program is used to build the experimental environment, and the test scene is processed graphically to facilitate the observation of the obstacle avoidance path of the robot. Both Q-learning and the State-Action-Reward-State-Action (SARSA) algorithm avoid conventional obstacles and reach the end point by the shortest path in the experiment. In the dangerous obstacle scene, the Q-learning algorithm can still avoid obstacles and find the shortest path, while the Sarsa algorithm selects a longer route. The verification results show that the two algorithms have their advantages and disadvantages, which provides a reference for the selection and optimization of robot obstacle avoidance algorithms and has important practical significance and theoretical value. This study aims to promote the development of robot obstacle avoidance technology and provide a useful reference for research and application in related fields.

Keywords: Reinforcement learning, AI, robot, obstacle avoidance

ISSN 2959-6157

1. Introduction

·In recent years, reinforcement learning has provided a new paradigm for robot obstacle avoidance with its ability to autonomously optimize decisions by interacting with the environment. By combining the environment representation and policy optimization of neural networks, deep reinforcement learning enables the robot to learn complex mapping relationships from raw sensory data, and then realize dynamic obstacle avoidance. However, current methods still face challenges, such as high computing resource consumption leading to poor real-time performance, large demand for training data and limited generalization ability, and insufficient Robustness of policies in complex scenarios. Wu Kun showed in his research on Q-learning that the off-line Q-learning algorithm has its limitations and may lead to estimation failure due to distributed offset [1]. In addition, Dongbin Zhao used the Sarsa algorithm to introduce experience replay training to enhance machine deep learning, which is better than the Q-learning algorithm in some aspects of deep learning [2]. Scholar Li Fan pointed out in his article that some recommendation systems based on DQN inherently deal with the difference between the fixed complete action space inherent in the Q network and the gradually reduced available action space during the recommendation period, and the Q table generation was optimized with the help of this research [3].

This paper will conduct experiments on two classical algorithms for different scenarios, and introduce experience replay and reward and punishment mechanisms to verify the learning performance of the two algorithms in the same scenario.

2. Research Methods

Combined with the reinforcement learning data set in python provided by Sichakar-Valentyn, the python experimental environment was built, different experimental maps and agents were created, obstacles, punishment mechanisms and end points were added to the map, and the obstacle avoidance behavior and final route of the agent in the map were observed and recorded. In this experiment, a total of three maps were created. The models were the robot avoiding obstacles to reach the end point in the city, the mouse avoiding obstacles to find cheese, and the mouse handling cliff obstacles (punishment mechanism). In each experiment, the Q-learning algorithm and the Sarsa algorithm are used to compare the performance of the two algorithms in different scenarios.

3. Process of Experiment

The scene of experiment 1 was built, as shown in Fig. 1, and the experimental map with a size of 9*9 units was constructed. It was necessary to avoid obstacles such as trees and roadblocks and find the optimal route from the starting point to the end point, driven by Q-learning and the Sarsa algorithm.

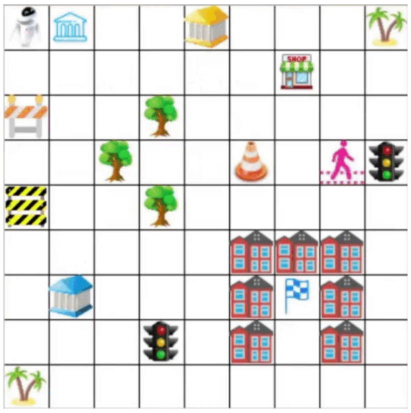

Fig. 1. Experiment I, the robot avoids obstacles in the city to reach the end of the experiment (Photo/Picture credit: Original).

The second scene of the experiment was set up, as shown in Fig. 2, and the experimental map with a size of 25*25 units was constructed. Driven by Q-learning and the Sarsa algorithm, it was necessary to avoid walls and find the optimal route from the starting point to the end point.

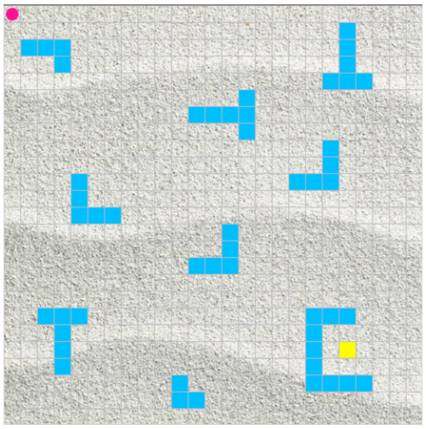

Fig. 2. Experiment II map where mice avoid walls to find cheese (Photo/Picture credit: Original).

The scene of experiment 3 is set up, as shown in Fig. 3, and a 5*9 unit size experimental map is constructed. The obstacle is a cliff, and the penalty mechanism is introduced. When falling into the cliff, it is necessary to start again and calculate the number of steps accumulated, which may lead to a higher number of steps in the final route, and the corresponding score of the optimal route becomes worse. Driven by Q-learning and the Sarsa algorithm, it reaches the end point from the starting point to avoid obstacles and find the optimal route.

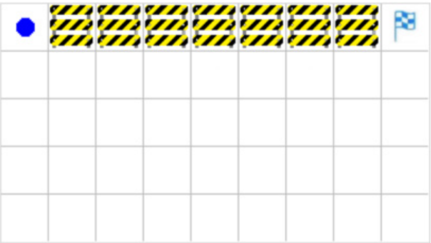

Fig. 3. Map of Experiment 3, where mice avoid the cliff to reach the endpoint (Photo/Picture credit: Original).

4. Experimental Results

The results of experiment I are shown in Fig. 4. In the robot simple obstacle avoidance experiment, the two algorithms show consistent routes, and both avoid obstacles and reach the end point with the shortest route.

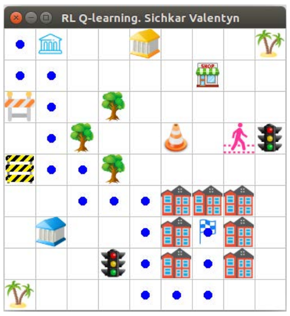

Fig. 4. The robot avoids obstacles to reach the endpoint (Photo/Picture credit: Original).

The results of experiment 2 are shown in Fig. 5. The two algorithms also show consistency, and the mice both take the shortest route to avoid the obstacles and find the cheese.

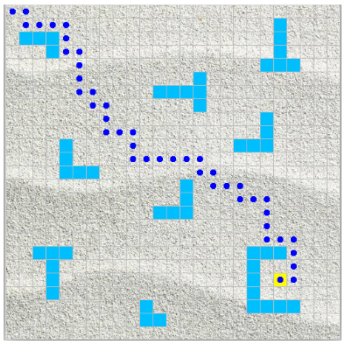

Fig. 5. Mice avoid obstacles to find cheese (Photo/Picture credit: Original).

There were significant differences in the results of experiment 3, among which the results of the Q-learning algorithm were shown in Fig. 6. The mouse approached the cliff to reach the end and successfully found the shortest path. Sarsa algorithm chooses the path away from the cliff, which is farther than the route, but safer, and Sarsa algorithm considers this route to be the optimal route.

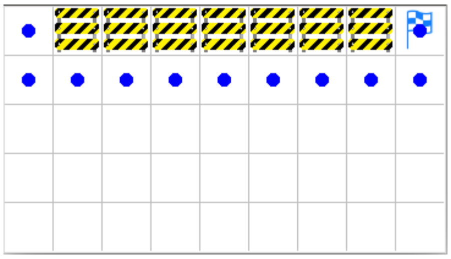

Fig. 6. Final route of the mouse under the Q-learning algorithm (Photo/Picture credit: Original).

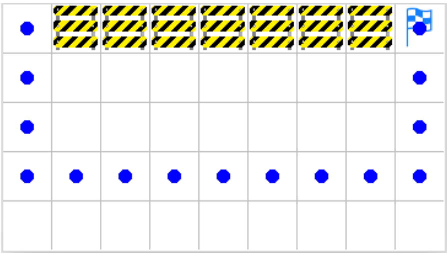

Fig. 7. The final route of the mouse under the Sarsa algorithm (Photo/Picture credit: Original).

5. Analysis of Results

By observing the experimental results in Fig. 4, it can see the next action decision made by the mobile robot: the final expedient action sequence to reach the goal is as follows: down-right-down-down-right-down-down-right-down-down-right-up-up-up-up. In the experiments using the Q-learning algorithm, for Experiment 1, the shortest path found to reach the target consists of 16 steps, and the longest path found to reach the target consists of 185 steps. Similar to Environment 1, the mouse found the shortest path to the goal after using the Q-learning algorithm, consisting of 42 steps.

Experiments 1 and 2 were implemented based on different algorithms. It can be seen that in the face of different obstacles, environments, and reward mechanisms, the experimental models found the shortest route, and the results were the same. However, after introducing the penalty mechanism, the two algorithms have a big difference (Figs. 6 and 7): Q-learning chooses the shortest path, which is close to the cliff and only takes 10 steps to reach the end, while sarsa algorithm chooses a long 14-step route in order to avoid the risk. The reason is that both algorithms

ISSN 2959-6157

are calculated based on a Q-table, but Q-learning is an offline method, and Sarsa is an online algorithm. In essence, the Q-Learning algorithm only considers whether the whole route is optimal when updating the Q value, while the Sarsa algorithm will consider whether there will be negative rewards in each step. As a result, Q-learning will be more aggressive, and Sarsa will take the safer, longer route.

6. Future Prospects

This experiment only aims at the existing data set, combined with the current version of the algorithm for comparative verification experiments, aiming at guiding the selection of two algorithms in the design of intelligent robots, there are still many shortcomings. In the future development of intelligent algorithms, first of all, more reward and punishment mechanisms can be introduced. Refer to Mohammad Reza Bonyadi's research, for some environments, the reward mechanism is applied after a series of actions rather than after each action, which makes the logic of selection of algorithms in different environments unclear. Thus, the desired results cannot be achieved [4]. Or refer to the articles of Sreyas Ramesh and Shamima Najnin to combine the two algorithms for special scenarios, such as cross-context noun learning and microgrid energy management, and develop and upgrade strategies on the original ability to improve the effectiveness of the algorithm in specific scenarios. Improved solutions for different scenarios [5, 6].

7. Conclusion

According to the above experimental results, it can be found that both algorithms have advantages and disadvantages, but the application scenarios of robots in reality are more complex: for example, more complex routes, different degrees of reward mechanisms, for example, when facing the scene of "cliff" and "small pit", how to make a choice: choose to take a long way to avoid risk or move closer to obtain a better route. Therefore, in the route planning of different scenarios, the two algorithms should be organically combined to obtain a better route. If a judgment mechanism is added, which algorithm should be selected for route planning in the face of a special obstacle. For example, when facing dangerous obstacles such as cliffs, Sarsa algorithm can be adopted to prioritize safety, and when facing ordinary walls, the negative feedback mechanism can be reduced, and the Q-learning algorithm can be selected to find the shortest route around.

References

[1] Wu K, Zhao Y, Xu Z, Che Z, Yin C, Harold Liu C, Feng F, Tang J. ACL-QL: Adaptive conservative level in q-learning for offline reinforcement learning. IEEE Trans Neural Netw Learn Syst. 2025, 36(6): 11399-11413.

[2] Dongbin Z, Haitao W, Kun S, Yuanheng Z. Deep reinforcement learning with experience replay based on SARSA. IEEE Symposium Series on Computational Intelligence. 2016.

[3] Li F, Qu H, Zhang L, Fu M, Chen W, Yi Z. Q-ADER: An effective q-learning for recommendation with diminishing action space. IEEE Trans Neural Netw Learn Syst. 2025, 36(5): 8510-8524.

[4] Bonyadi MR, Wang R, Ziaei M. Self-punishment and reward backfill for deep q-learning. IEEE Trans Neural Netw Learn Syst. 2023, 34(10): 8086-8093.

[5] Ramesh S, N SB, Sathyavarapu SJ, Sharma V, A A NK, Khanna M. Comparative analysis of Q-learning, SARSA, and deep Q-network for microgrid energy management. Sci Rep. 2025, 15(1): 694.

[6] Najnin S, Banerjee B. Pragmatically framed cross-situational noun learning using computational reinforcement models. Front Psychol. 2018, 9: 5.