Path Planning of Maritime Autonomous Surface Ships in Unknown Environment with Reinforcement Learning

Chengbo Wang[1], Xinyu Zhang[1](✉), Ruijie Li[1], and Peifang Dong[2]

[1] Key Laboratory of Marine Simulation and Control for Ministry of Communications, Dalian Maritime University, Dalian, China
wangcb@dlmu.edu.cn, zhang.xinyu@sohu.com
[2] School of Mechanical Engineering,
Nanjing University of Science and Technology, Nanjing, China

Abstract. Recently, artificial intelligence algorithms represented by reinforcement learning and deep learning have promoted the development of autonomous driving technology. For the shipping industry, research and development of maritime autonomous surface ships (MASS) has academic value and practical significance. In an unknown environment, MASS interacts with the environment to conduct behavioral decisions-making, intelligent collision avoidance, and path planning. Reinforcement learning balances exploration and exploitation to improve its own behavior by interacting with the environment to obtain rewarded data. Thus, to achieve intelligent collision avoidance and path planning for MASS in unknown environments, a path planning algorithm of MASS based on reinforcement learning is established. Firstly, the research status of unmanned ships and reinforcement learning is reviewed. The four basic elements of reinforcement learning are analyzed: environment model, incentive function, value function and strategy. Secondly, the port environment model, sensor model, MASS behavioral space, reward function, and action selection strategy were designed separately. Besides, the reward function consists of avoiding obstacles and approaching the target point. Finally, based on the python and pygame platform, a simulation experiment was carried out with Rizhao Harbor District as a case study to verify that this method has better self-adaptability. The model successfully avoids obstacles through online trial and error self-learning and plans adaptive paths in unknown environments.

Keywords: Reinforcement learning · Collision avoidance · Path planning · Maritime autonomous surface ships

1 Introduction

With the rapid development of artificial intelligence, unmanned technologies can better serve various fields, including environmental monitoring, object tracking, industrial manufacturing, and hazardous environment exploration. From mobile robots to drones and unmanned boats, the path planning technology has been continuously improved and matured. However, for marine autonomous systems, path planning and obstacle avoidance are new research subjects and hot topics for scholars and experts.

F. Sun et al. (Eds.): ICCSIP 2018, CCIS 1006, pp. 127–137, 2019.
https://doi.org/10.1007/978-981-13-7986-4_12

Path planning is one of the key technologies of the Marine Autonomous System (MAS). Its essence is to avoid obstacles and reach the target point with the optimal path. According to the degree of knowledge of the environmental information, the path planning is divided into a global path planning based on prior knowledge of the environment or an expert system, and a local path planning based on perception information. The disadvantage of the global path planning based on expert system is that it is difficult to obtain information on the marine environment. It is difficult to form a complete and accurate knowledge base and expert system, and it cannot meet online learning. Local path planning requires sensors to acquire environmental information in real time, real-time positioning of unmanned ships, determining the distribution of local obstacles, and calculating an optimal path online. At present, in the field of mobile robots, drones and unmanned boats, effective path planning methods include artificial potential field method, neural network method, fuzzy logic and genetic algorithm. Among them, Chen, Qi, Zhang [1] and others proposed that the improved artificial potential field method avoids the problem of the local minimum point of the unmanned boat and realizes the path planning of the unmanned boat. The effectiveness of the method is verified by simulation. Guo and Wang [2] and others proposed an improved quantum particle swarm optimization algorithm based on modified particle update location, which improved the global optimization ability and convergence performance of UAV path planning algorithms. Yan, Huang, Zhu et al. [3] proposed an underwater robot path planning method based on biological neural dynamics, and Yan simulated and implemented tasks such as autonomous planning of observation paths, return routes, and obstacle avoidance in unknown dynamic underwater environments. The multi-objective genetic algorithm for the local path planning of surface water unmanned boats was designed by Chen [4], taking the shortest local route and the minimum heading variation as the optimization objectives, and the ship's close encounter distance model and the "International Convention on the rules of collision avoidance by sea in 1972" are the constraints.

However, these methods usually need to assume complete environmental information. However, MAS in unknown environments rarely have prior knowledge of the environment. In large number of practical applications, the system needs a strong ability to adapt to uncertain environments. Reinforcement Learning has the advantage of interacting with the environment. Reinforcement learning completes online learning through trial and error algorithms, meeting adaptive learning without prior knowledge of the environment. Duan, Zhang, and Zhang [5] proposed a fuzzy control rule that can realize real-time motion planning of underwater robots based on reinforcement learning self-learning and self-adjusting planning algorithms. Liang [6] proposed an autonomous learning method based on Q-learning algorithm to solve the problem of adaptive path planning for mobile robots in unknown environments, and they completed adaptive path planning through simulation experiments. Zhao, Zheng, Zhang, Liu [7] and others used adaptive and random detection methods (ARE) to complete the mission of UAV navigation and obstacle avoidance, and the UAV path planning is realized with reinforcement learning. Konar, Chakraborty, Singh [8] and others used Q-learning four derived features to propose a new deterministic Q-learning algorithm for path planning of mobile robots, and simulations validated the modified algorithm applicability in path planning. A model-based reinforcement learning algorithm - Dyna-Q algorithm was proposed by Hwang, Jiang, Chen [9] et al. KS Hwang extracts status information between virtual neighbors for indirect learning and combines depth-first search methods to solve the problem of path planning and labyrinth of mountain bikes. Cruz and Yu [10] and others proposed a modified multi-agent reinforcement learning algorithm to perform greedy search by estimating unknown environmental information combined with neural networks and kernel smoothing techniques to implement multi-agent path planning.

The above domestic and foreign route planning methods have achieved good results in the fields of mobile robots, unmanned aerial vehicles and unmanned boats, but they are still at a preliminary stage of research in the intelligent navigation and route planning of MAS. In response to these challenges, this paper will innovate the application of reinforcement learning to the path planning of MAS of unmanned cargo ships. This article uses reinforcement learning selection strategies to complete obstacle avoidance for unmanned cargo ships, realizing the adaptive path planning of Rizhao Port area under the unknown prior knowledge.

2 Reinforcement Learning and Basic Elements

2.1 Reinforcement Learning Principle

Reinforcement learning is also known as enhanced learning and further learning [11]. Unmanned ships learn online by interacting with the environment in a reinforcement learning system. Figure 1 shows the reinforcement learning schematic. In reinforcement learning, MASS selects an action A_t based on the current state S_t and has an impact on the environment. MASS receive environmental feedback R_t (award or punishment). The MASS selects the next action based on the current feedback signal and the environment. The principle of choice is to maximize the probability of positive feedback from the environment. In simple terms, the purpose of reinforcement learning is to choose an optimal strategy that maximizes the total return of unmanned ships in interaction with the environment.

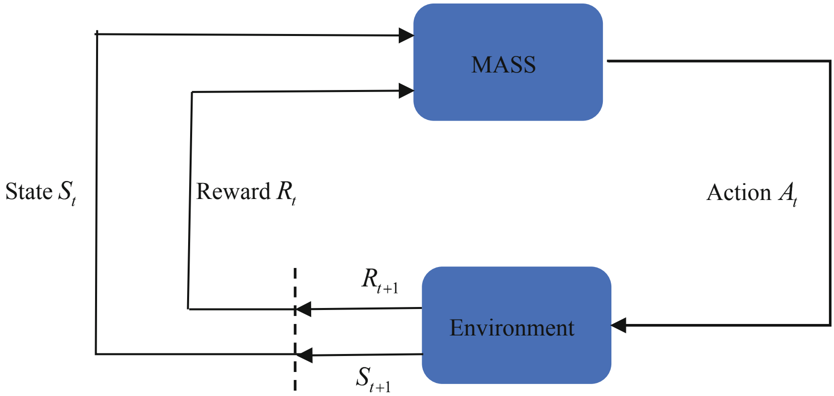


Fig. 1. The reinforcement learning schematic.

Reinforcement learning in discrete time can essentially be regarded as Markov Decision Process (MDP). A Markov decision process for MAS is defined by the following five-tuple: (S, A, P_a, R_a, γ). S represents the limited state space of the unmanned ship; A represents the action space of the unmanned ship, that is, the collection of all the behavior spaces of the unmanned ship in any state, such as left rudder, right rudder, acceleration, deceleration, follow-up, and ship stopping. $P_a(s, s') = P(s'|s, a)$ is a conditional probability that represents the probability that the unmanned ship will reach the next state s' under state s and action a. $R_a(s, s')$ is reward function that represents the incentive obtained from the state s to the state s' of an unmanned ship under action a. $\gamma \in (0, 1)$ is the attenuation factor of the reward, and the reward at the next moment is attenuated according to this factor [12, 13].

Currently used reinforcement learning algorithms include Q-Learning, SARSA-Learning, TD-Learning, and adaptive dynamic programming [14]. This article uses Q-Learning online learning algorithm. Q-Learning is considered as an incremental dynamic programming. By optimizing the action function to find the optimal strategy, the expectation of cumulative returns is maximized. The Q-Learning algorithm not only satisfies the adaptability of the system in the environment, but also ensures that the algorithm converges in the learning process [15].

2.2 Basic Elements of Reinforcement Learning Model

In the reinforcement learning of MAS, in addition to the unmanned ship and the environment, there are four major elements [15]: environment model, reward function R, value function $Q(s)$ and strategy π.

Environment Model. MASS environment models include unmanned ships, obstacles, starting points, target points, and sensors. In the environment model, the MASS acquires information on obstacles and target points through the sensors. After executing the search strategy, the model will make feedback to judge the rewards and punishments of the action strategies.

Reward Function. The reward function is the enhanced signal fed back by the environment after the MASS's action strategy interacts with the environment. It is used to evaluate the quality of the action strategy. If it helps to achieve the goal, it will reward. Instead, punish. The purpose of reinforcement learning is to choose the optimal strategy to maximize the ultimate return of the MASS.

$$r(s_t, a_t) = \mathbb{E}_{P(s_{t+1}|s_t, a_t)}[r(s_t, a_t, s_{t+1})] \tag{1}$$

Value Function. The value function refers to the mathematical expectation of the cumulative returns in the process of moving from the current state to the target state of the MASS under the action search strategy. The value function $Q^{\pi}(s, a) \in \mathbb{R}$ will determine the action search strategy of the driverless system.

$$Q^{\pi}(s, a) = \mathbb{E}_{P^{\pi}(t)}[G(t)|s_1 = s, a_1 = a] \tag{2}$$

Strategy. The problem to be solved in the path planning of MASS is to find an optimal "strategy", to make the greatest return. Essentially, the strategy is a mapping of the state S of MASS to the action A, denoted as $\pi : S \rightarrow A$.

$$\pi^{*} = \arg\max_{\pi} \mathbb{E}_{P^{\pi}(t)}[G(t)] \tag{3}$$

3 Design of MASS Track Decision Model

3.1 Environmental Model Establishment

This research builds a two-dimensional simulation environment based on python and pygame. In the two-dimensional coordinate system, each coordinate point corresponds to a state of MASS. Each state can be mapped to each element of the set of environmental states S. In the simulation environment model, there are two state values for each coordinate point, which are 1 and 0, respectively, where 1 represents the navigable area, which is displayed as a white area in the environmental model; 0 represents the area of the obstacle and is shown as a black area in the environmental model. Figure 2 shows the simulation environment model, which simulates the two-dimensional map of the environment state size 656 × 808. In the environmental model, obstacles such as static ships, breakwaters, and basin foundations were simulated, which positional information is unknown to MASS.

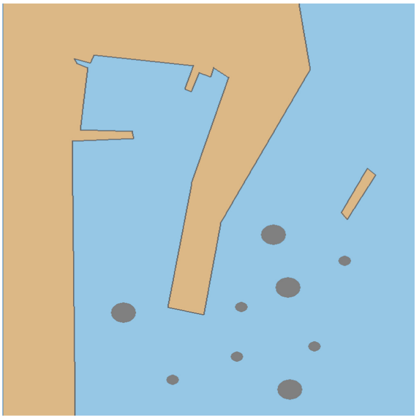


Fig. 2. The simulation environment.

3.2 Expression of Action Space

After setting the initial point and target point of the MASS, it is considered as a mass point during the simulation. The autonomous navigation of MASS is a continuous state during the actual navigation. Thus, research needs to generalize the observational behavior of MASS O into discrete actions $\hat{A} = Generalization(A', O)$. Generally, the search movement of MASS is four discrete actions: up, down, left, and right. When the environment appears corners, the searching behavior in the diagonal direction is increased.

Focusing on the unmanned ship's particle quality, the study defines the actual operational space model of MASS as eight discrete actions: up, down, left, right, up_{-45°, up_{+45°, $down_{+45^\circ}$, $down_{-45^\circ}$. That is the matrix of formula (4):

$$A = [-1, 1 \quad 0, 1 \quad 1, 1 \quad -1, 0 \quad 1, 0 \quad -1, -1 \quad 0, -1 \quad 1, -1] \tag{4}$$

3.3 Design of Reward Function

The reward function plays an important role in the reinforcement learning system of MASS. Function can be used to evaluate the effectiveness of decision-making of MASS and the safety of obstacle avoidance. It is search-oriented. For MASS, the reward function consists of safety, comfort, and arrival targets. In designing the reward function, the following elements should be considered as much as possible [16]:

Approaching Target Point. Searching behavior of MASS with reinforcement learning in unknown environmental conditions should bring MASS closer to the target point. Closer to the incentive function will choose rewards, otherwise it will punish:

$$\mathrm{R_{distance}} = -\lambda_{\mathrm{distance}} \sqrt{(\mathrm{x} - \mathrm{x_{goal}})^2 + (\mathrm{y} - \mathrm{y_{goal}})^2} \tag{5}$$

Safety. In the Q-Learning algorithm model, the unknown environment where the MASS is located is divided into state spaces, which are divided into safety state areas and obstacle areas. The MASS should select the action search strategy that satisfies the safety of the ship in the local area of obstacles, and avoid the obstacles "early, clear, and large". Thus, in the reward function, the penalty value will be added to the behavior near the obstacle, and vice versa. This paper generalizes the excitation function as a nonlinear piecewise function:

$$R = \begin{cases} 10, & \mathrm{d_g(t)} = 0 \\ 2, & s = 1 \text{ and } (\mathrm{d_g}(t) - \mathrm{d_g}(t-1)) < 0 \\ -1, & s = 0 \\ -1, & s = 1 \text{ and } (d_o(t) - \mathrm{d}_o(t-1)) < 0 \\ 0, & \text{else} \end{cases} \tag{6}$$

Where: $s = 0$ represents collision between unmanned ship and obstacle; $s = 1$ represents MASS sailing in a safe area. $d_g(t)$ represents the distance between the target point and the MASS at time t; $d_g(t-1)$ represents the distance between the target point and the MASS at time $t-1$; $d_o(t)$ represents the distance between the obstacle and the MASS at time t; $d_o(t-1)$ represents the distance between the obstacle and the MASS at time $t-1$.

3.4 Action Selection Strategy

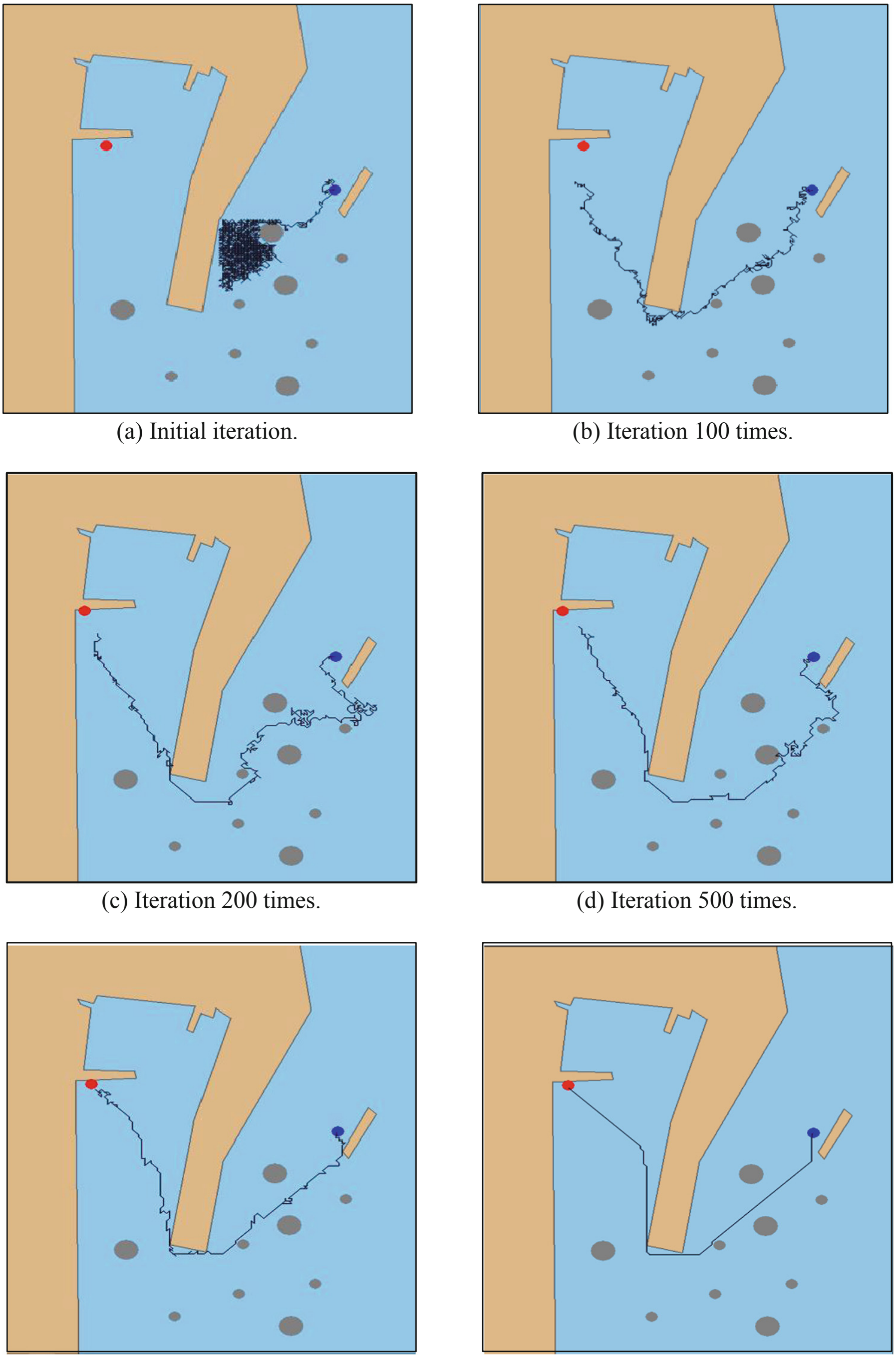
In the reinforcement learning system, on the one hand, the MASS needs online trial and error to find the optimal search strategy, namely exploration; On the other hand, it is necessary to consider the entire route planning. The expectation of the entire MASS to obtain rewards is maximum, namely, utilization. This study uses $\varepsilon - \mathit{greedy}$ strategy, which balances exploration and utilization. Its meaning is that when the search behavior maximizes the action value function, the probability of selecting the action is $1 - \varepsilon + \frac{\varepsilon}{|\mathrm{A(s)}|}$, and the probability of selecting other actions is $\frac{\varepsilon}{|\mathrm{A(s)}|}$:

$$\pi(\mathrm{a}|\mathrm{s}) \leftarrow \begin{cases} 1 - \varepsilon + \frac{\varepsilon}{|\mathrm{A(s)}|} & \text{if} \quad \mathrm{a} = \mathrm{argmax_a Q(s,a)} \\ \frac{\varepsilon}{|\mathrm{A(s)}|} & \quad \text{else} \end{cases} \tag{7}$$

4 Experiments

Based on the model in Sect. 3, a design simulation experiment was established to verify the effectiveness of Q-Learning-based obstacle avoidance and path planning for MASS. The model validation platform uses python and pygame. The path planning framework of the unmanned ship consists of two parts: approaching the target point and avoiding obstacles. When there are no obstacles in the environment or the obstacles are not within the safe encounter distance, the unmanned ship will randomly select the action near the target point with probability $\varepsilon/|\mathrm{A(s)}|$; When an obstacle appears within a safe encounter distance, an unmanned ship navigates obstacles by interacting with the environment through the reward function. Some of the model parameters in the experiment were set to: gamma $\mathit{gamma} = 0.005$, $\gamma = 0.9$, $\omega = 0.02$, $v_0 = 8kn$.

This study sets the initial position (529, 364) and target point (164, 279) of MASS. In the early stage of experimental iteration shown in Fig. 3(a), the MASS collides with obstacles at different time steps. After the collision, the MASS will return to the previous step and reselect the action strategy. In the initial iteration, the MASS cannot judge the temptation area in the simulation environment and it is trapped in the “trap” sea area in the simulation port pool; After 100 iterations, the system gradually plans effective paths, but collision obstacles occur many times in the process and the planning path fluctuates greatly. From 200 iterations to 500 iterations in Fig. 3(b), the collision phenomenon gradually decreases and the planned path fluctuates. As is shown in Fig. 3(c) in the 1000 iterations, all obstacles are effectively avoided and the planned path is weak and gradually stable; Until the 1500th iteration, shown in Fig. 3(d), the probability of random search is the smallest, and the reinforcement learning system plans the final fixed path to reach the target point. The simulation experiment results are shown in Fig. 3.



(a) Initial iteration.

(b) Iteration 100 times.

(c) Iteration 200 times.

(d) Iteration 500 times.

(e) Iteration 1000 times.

(f) Iteration 1500 times.

Fig. 3. The simulation experiment results.

The number of epochs of the model is taken as the abscissa, and the number of steps required to move from the starting to the end of each iteration is plotted as the ordinate to visually observe the training speed and training effect of this method. The iterative convergence trend is shown in Fig. 4. The MASS has too little information about the state of the environment in the early stage of interacting with the environment, and collisions and path planning are fluctuating. As the number of iterations increases, the MASS accumulates learning experience and completes the adaptation to the environment, ultimately successfully planning the path and reaching the target point.

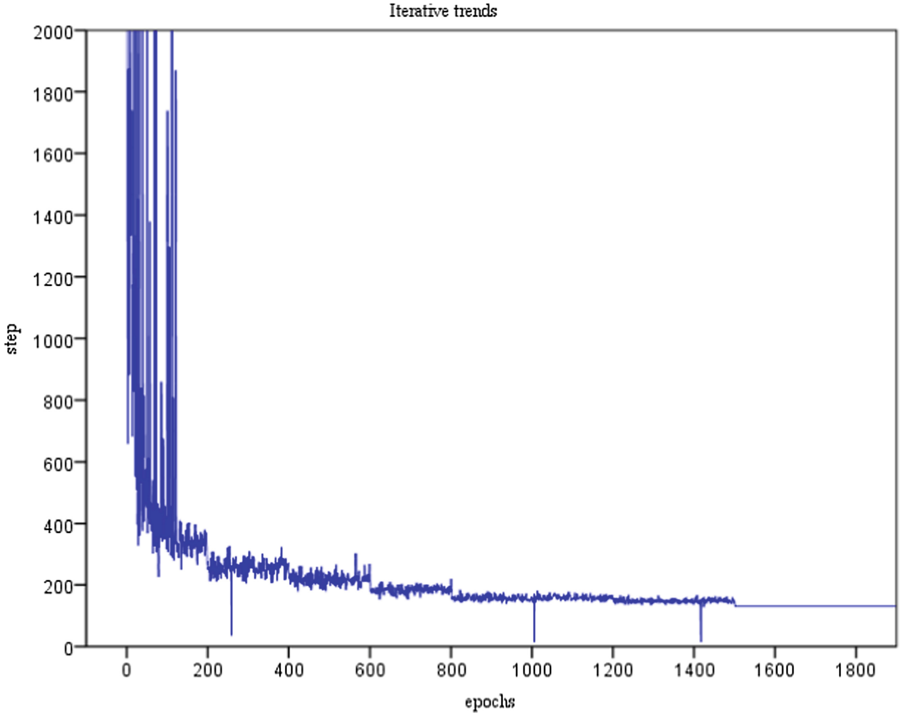


Fig. 4. Iterative trends.

5 Conclusion

This research proposes a path planning method for MASS based on the Q-Learning algorithm. Based on the four basic elements of Q-Learning, an unmanned ship path planning model was established. Simulation experiments verify that this method has better self-adaptive ability. Through the online trial-and-error self-learning, MASS can avoid the obstacles and plan the adaptive path in the unknown environment, which verified the effectiveness of the method.

In the early stage of interacting with the environment, the MASS has too little information about the state of the environment, and there are collisions and large fluctuations in route planning. As the number of iterations increases, the unmanned ship system accumulates learning experience and completes adaptation to the environment, ultimately successfully planning the path and reaching the target point. However, reinforcement learning algorithms still require many improvements, if reinforcement learning is required to be applied in an unmanned scenario.

(1) The Q-Learning algorithm based on Markov decision process can obtain the optimal path through trial and error algorithm, but its convergence speed is slower and the number of iterations is more. The first improvement is to improve the adaptive ability of reinforcement learning so that a small number of iterations can be used to learn the correct behavior with only a small number of samples.
(2) In the actual navigation process, the behavior of the unmanned cargo ship has complex continuity. In this simulation experiment, only simple generalization is done to divide the driving behavior of MASS into up, down, left, right and etc., the second direction of improvement is to enrich the behavioral decision space, making it closer to real ship driving behavior.

In the future research, we also can continuously increase the complexity of the unknown environment, increase the ability to predict and improve the self-adaptive of reinforcement learning so that it can be better applied to the actual situation.

Acknowledgements. This work was supported by the National Natural Science Foundation of China (Grant 51779028), and the Fundamental Research Funds for the Central Universities (Grant 3132018146, 3132016315 and 3132016321).

References

1. Chen, C., Geng, P., Zhang, X.: Path planning research on unmanned surface vessel based on improved potential field. Ship Eng. **37**(9), 72–75 (2015)
2. Guo, Y., Wang, X.: UAV path planning based on improved quantum-behaved particle swarm optimization algorithm. Ship Eng. **45**(1), 99–112 (2016)
3. Yan, M., Huang, B., Zhu, D.: A novel path planning algorithm based on neurodynamics for observation of underwater structures. Ship Ocean Eng. **46**(2), 103–107, 112 (2017)
4. Chen, H.: Preliminary research on local path planning for unmanned surface vehicle. Doctoral dissertation of Dalian Maritime University (2016)
5. Duan, Q., Zhang, M., Zhang, J.: Underwater robot local path planning method based on fuzzy neural network. Ship Eng. **1**, 54–58 (2001)
6. Liang, Q.: Reinforcement learning based mobile robot path planning in unknown environment. Mech. Electr. Eng. Mag. **29**(4), 477–481 (2012)
7. Zhao, Y., Zheng, Z., Zhang, X., et al.: Q learning algorithm-based UAV path learning and obstacle avoidance approach. In: 36th Chinese Control Conference (CCC), Dalian, pp. 3397–3402. IEEE CPP (2017)
8. Konar, A., Chakraborty, I.G., Singh, S.J., et al.: A deterministic improved Q-learning for path planning of a mobile robot. IEEE Trans. Syst. Man Cybern. Syst. **43**(5), 1141–1153 (2013)

9. Hwang, K.S., Jiang, W.C., Chen, Y.J.: Pheromone-based planning strategies in Dyna-Q learning. IEEE Trans. Industr. Inf. **13**(2), 424–435 (2017)
10. Cruz, D.L., Yu, W.: Path planning of multi-agent systems in unknown environment with neural kernel smoothing and reinforcement learning. Neurocomputing **233**, 34–42 (2017)
11. Dong, W.: Research on mobile robot piath planning based on Q-learning. Doctoral dissertation of Shandong University of Science and Technology (2013)
12. Liu, Q., Zhai, J., Zhang, Z., et al.: A survey on deep reinforcement learning. Chin. J. Comput. **41**(1), 1–27 (2018)
13. Zhao, D., Shao, K., Zhu, Y., et al.: Review of deep reinforcement learning and discussions on the development of computer Go. Control Theory Appl. **33**(6), 701–717 (2016)
14. Huang, B., Cao, G., Wang, Z.: Reinforcement learning theory, algorithms and application. J. Hebei Univ. Technol. **35**(6), 34–38 (2006)
15. Szepesvari, C.: Algorithms for reinforcement learning. In: International Conference on Computing, pp. 103–127. Morgan & Claypool, San Rafael (2010)
16. Cheng, Y., Zhang, W.: Concise deep reinforcement learning obstacle avoidance for underactuated unmanned marine vessels. Neurocomputing **272**, 63–73 (2017)