Navigation behavioural decision-making of MASS based on deep reinforcement learning and artificial potential field

WANG Cheng-bo, ZHANG Xin-yu*, ZHANG Jia-wei, DING Zhi-guo, and AN Lan-xuan

Key Laboratory of Marine Simulation and Control for Ministry of Communications, Dalian Maritime University, China

wangcb@dlmu.edu.cn, *zhang.xinyu@sohu.com

Abstract. To realize intelligent obstacle avoidance and local path decisions for maritime autonomous surface ships (MASS) in uncertain environments, a navigation behavioural decision-making model based on deep reinforcement learning (DRL) algorithm improved by artificial potential field (APF) is proposed. Based on the analysis of navigation decision system and perception principle, the action space, reward function, motion search strategy and action value function are designed respectively for the purpose of steering to collision avoidance. The navigation behavioural decision-making model for MASS is improved by adding the prior information, the gravitational potential field and the obstacle repulsion potential field to update the initial action state value function and search path. Python and Pygame modules are used to build a simulation chart. Effectiveness of the algorithm is verified, with Tianjin Xingang port as a study case. The simulation results show that the APF-DRL algorithm is better than the DRL algorithm in training iteration time and piloting decision path, which improves the self-learning ability of MASS, and can meet the requirements of MASS path decision and adaptive obstacle avoidance.

1. Introduction

In July 2017, the State Council in China issued the “New Generation Artificial Intelligence Development Plan” [1]. It is proposed to focus on the breakthrough of the independent unmanned system computing architecture and the autonomous control of the drone and the intelligent technology of auto driving in cars, ships and rail transit. From the results of the Google AlphaGo being able to beat the top Go players [2], in some respects the level of machine intelligence has been able to reach or exceed humans. With the rapid development of artificial intelligence, big data technology and new sensors, the auto-driving bus “Apollo” was officially mass-produced [3]. The same is true for the shipping industry. The future shipping system will only rely less and less on personnel, and the efficiency of ship traffic management will be higher and higher. MASS is an inevitable trend in the future development.

MASS are the integration of multiple intelligent systems [4]. The navigation behavior decision system plays the role of “Navigation Brain”. The problem that needs to be solved is to determine the best navigation strategy and collision avoidance path based on the environmental information of the MASS. At present, most academics only conduct research on USV [5] or ship collision avoidance decision-making, such as genetic algorithm [6], particle swarm algorithm [7] and simulated annealing algorithm [8]. However, in an unknown environment, the prior knowledge of the environment is difficult to obtain, and it is difficult to form a complete and accurate knowledge base. The rule-based algorithm is difficult to cope with various situations. Therefore, in a large number of practical situations, the MASS
needs to have strong adaptive ability. Recently, deep reinforcement learning (DRL) combined with deep neural network models and reinforcement learning has made significant progress in the field of autonomous navigation such as autonomous cars and USV. Among them, Pinxin Long et al. [9] proposed a novel end-to-end framework based on deep neural network to generate effective reactive multi-agent navigation reactive collision avoidance strategy. Wang P et al. [10] used reinforcement learning to train smart cars to learn the automatic lane change behavior so that it can intelligently perform lane change in a variety of even unpredictable situations. Zhelo O et al. [11] studied the behavioral exploration strategy of mobile machines based on deep reinforcement learning, and realized adaptive navigation of mobile robots by setting reward signals. Amir Ramezani Dooraki and Deok Jin Lee [12] combined a multi-layer perceptron neural network and reinforcement learning to propose a deep reinforcement learning path planning algorithm and verified the algorithm's self-exploration and obstacle avoidance ability in unknown environment on a simulation framework, Gazebo. Michael Everett et al. [13] introduced a long-short-time memory network (LSTM) based on no specific behavior rules, and built a deep reinforcement learning algorithm to realize the safe navigation and obstacle avoidance of robots in pedestrians.

In summary, DRL completes self-training by interacting with the environment. High-dimensional input such as original image or environmental state is used in the training process to achieve adaptive ability to the unknown environment. However, the deep Q learning algorithm based on Markov process has slow convergence speed and many iterations, and it is easy to fall into local iteration. Therefore, this paper establishes MASS piloting behavioral decision-making model that introduces APF to improve DRL. The increase of gravitational potential field is the initial Q value of DRL, which avoids the huge calculation amount in complex environment, and effectively prevents the MASS from falling into the concave trap in the environment, which accelerates the iterative speed of the algorithm. Finally, the simulation experiments in Python and Pygame environment verify the feasibility and efficiency of the algorithm.

2. System description

The navigation behavioral decision-making system is the core system of the MASS. Its effectiveness directly determines the navigation safety, and acts like a brain of the MASS. During the voyage, the “brain” imagination, reasoning and decision-making process are very complicated. After the destination state information is clarified and the global route is obtained, it is necessary to generate reasonable and safe piloting actions according to the dynamic environmental situation around the ship (e.g. Acceleration, deceleration, steering). In this process, the "brain" needs to quickly and accurately reason based on multi-source heterogeneous information such as traffic rule knowledge, piloting experience knowledge, and chart information stored in its memory unit. In the environmental model, the MASS senses the relative position and position between the ship and the obstacle in real time through the perception system. Figure 1 shows the perception principle of the MASS, where $S_p(x_p,y_p)$ is the MASS's position; $S_f(x_f,y_f)$ is the target point; $v_0$ is the MASS's speed; $\varphi_o$ is the MASS's course. $S_o(x_o,y_o)$ is the obstacle's position; $\varphi_o$ is the relative orientation of the obstacle and the MASS; $\text{dis}_{u-o}$ is the ship's arrival Target point distance; $\text{dis}_{u-o}$ indicates the distance between the MASS and the obstacle. The current system-aware environmental status information can be expressed as $\text{obs}_t = [v_0, \varphi_o, \varphi_o, \text{dis}_{u-o}, \text{dis}_{u-o}]^T$. In the decision-making process, the system not only obtains the current state $\text{obs}_t$, but also obtains the historical state of the historical observation in the memory pool ( $\text{obs}_{1:T_p}$, $i=1,\ldots,T_p$ ), $T_p$ represents the total duration of the observed memory, and the data input of the final training of the navigation decision-making system at the time $t$ can be expressed as:
\[ X_{\text{perception}}(t) = \begin{bmatrix} \text{obs}_t & \text{obs}_{t-1} & \cdots & \text{obs}_{t-T} \end{bmatrix} = \begin{bmatrix} v_t & \phi_t & \delta_t & \text{dis}_{t-P} & \text{dis}_{t-O} \\ v_{t-1} & \phi_{t-1} & \delta_{t-1} & \text{dis}_{t-1-P} & \text{dis}_{t-1-O} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ v_{t-T} & \phi_{t-T} & \delta_{t-T} & \text{dis}_{t-T-P} & \text{dis}_{t-T-O} \end{bmatrix} \]  

(1)

![Diagram](image)

**Figure 1.** Perception principle.

3. **Navigation behavioural decision-making model for steering collision avoidance**

In this paper, Q-Learning is combined with neural network to establish a navigation behavioral decision model based on Deep Q-Learning. The Deep Q-Learning algorithm uses an empirical playback algorithm. The basic idea is to remember the historical information that the algorithm performs in this environment. In reality, the number of environmental states and action states is extremely large, and it needs to be generalized by neural network. Therefore, long short-term memory (LSTM) is selected as the deep neural network. As shown in Figure 2, it is a framework diagram of a navigation behavioral decision-making model for the MASS based on DRL. In the memory pool, the current state of the MASS is observed as the input of the LSTM, and the Q value table of the action that can be performed in the current state is output, and the behavior strategy corresponding to the maximum Q value is learned through training.
3.1. Collision avoidance behavioural space
After setting the initial and target points, the MASS is regarded as a mass point. In the real navigation process, the autonomous navigation of the MASS is a continuous state, and it is necessary to generalize the observation behavior $O$ of the MASS into a discrete action $\hat{A} = \text{Generalization}(A', O)$. Under normal circumstances, the search action is four discrete actions of east, south, west and north. When the environment appears at the corner, the search behavior in the diagonal direction is increased. Centering on the mass of the MASS, the actual motion space model $A$ of the MASS is defined as eight discrete actions of east, south, west, north, southeast, northeast, southwest and northwest, i.e. matrix $A$:

$$A = \begin{bmatrix}
(-1,1) & (0,1) & (1,1) & (-1,0) & (-1,-1) & (0,-1) & (1,0) & (1,-1)
\end{bmatrix}$$

(2)

3.2. Reward function
In the DRL system, the excitation function plays an important role in evaluating the effectiveness of MASS behavioural decision-making and the safety of obstacle avoidance. It has a search-oriented role. For MASS, the reward function consists of safety, comfort and arrival target points. The goal of DRL is to find the search strategy when the return value is the largest in the driverless process. When designing the reward function, factors such as approaching the target point and safe collision avoidance should be considered as much as possible.

Since the environmental state set and the MASS motion state in the Deep Q-Learning algorithm are both limited, and the actual MASS transportation process is a continuous systematic event, the excitation function must be generalized to be nonlinear piecewise function:
In the formula (3), $s = 0$ indicates that the MASS collides with the obstacle; $s = 1$ indicates that the MASS is sailing in the safe area; $d_s(t)$ indicates the distance of the MASS from the target point at time $t$; $d_s(t-1)$ indicates the distance from the MASS to the target point at time $t-1$; $d_o(t)$ indicates the distance from the obstacle to the MASS at time $t$; $d_o(t-1)$ indicates the distance from the obstacle to the MASS at time $t-1$.

### 3.3. Collision avoidance action selection strategy

In the navigation behavioural decision-making system, the MASS needs online trial and error on the one hand to find the optimal search strategy, that is, exploration; on the other hand, it must consider the whole path planning, the maximum expectation of the MASS to receive the reward, that is, the exploit. This article uses the $\varepsilon$-greedy strategy, which balances exploration and utilization. The meaning is that when the search behavioural maximizes the action value function, the probability of selecting the action is $1 - \varepsilon + \frac{\varepsilon}{|A(s)|}$, and the probability of selecting other actions is $\frac{\varepsilon}{|A(s)|}$. For states with $|A(s)|$ action-state spaces, the action execution strategy is:

$$
\pi(a|s) \leftarrow \begin{cases} 
1 - \varepsilon + \frac{\varepsilon}{|A(s)|} & \text{if } a = \arg \max_a Q(s,a) \\
\frac{\varepsilon}{|A(s)|} & \text{else}
\end{cases}
$$

(4)

Where, $a$ is navigation behavior of MASS. $s$ is navigation state of MASS. $|A(s)|$ is number of state.

### 3.4. Action value function

The state value function refers to the mathematical expectation of the cumulative return of the MASS in the process of moving from the current state to the target state under the action search strategy. The value function $Q(s,a) \in \mathbb{R}$ will determine the motion search strategy of the driverless system [15].

$$Q(s,a) = \mathbb{E}_{P(t)}\{R(t)|s_t = s, a_t = a\}$$

(5)

Where, $R(t)$ represents the cumulative return of the MASS at time $t$, and $P(t)$ represents the search probability of the action at time $t$.

### 3.5. State value function

The state value function describes the value of a state when it follows the policy. This is the expected return when the behaviour from the state begins with our strategy $\pi$.

$$V^\pi(s,a) = \mathbb{E}_{P(t)}\{R(t)|s_t = s\}$$

(6)

### 4. Improved DRL algorithm for navigation behavioral decision of MASS based on APF

In this section, the APF is used to improve the DRL algorithm, to form the APF-DRL algorithm. The basic principle of the APF is to create a virtual gravitational potential field around the obstacle and the target point. The MASS will be attracted by any mass gravitational force and repulsion with obstacles, shore bases and other ships.
The gravitational field function is [16]:

\[ U_{gr}(q) = \frac{1}{2} k_{gr} \| q - q_s \|^2 = \frac{1}{2} k_{gr} \rho^2(q) \]  

(7)

Where, \( k_{gr} \) represents the gravitational constant and \( \rho(q) = \| q - q_s \| \) represents the Euclidean distance of the MASS from the target point.

The repulsion field function is [16]:

\[ U_{rep}(q) = \begin{cases} 
\frac{1}{2} \eta \left( \frac{1}{\rho(q,q_{obs})} - \frac{1}{\rho_0} \right) & \text{if } \rho(q,q_{obs}) \leq \rho_0 \\
0 & \text{if } \rho(q,q_{obs}) > \rho_0 
\end{cases} \]  

(8)

Where, \( \eta \) is the factor of the magnitude of the repulsive force, \( q \) is the coordinates of MASS, and \( q_{obs} \) is the coordinates of each obstacle. \( \rho(q,q_{obs}) \) is the distance between the MASS and the obstacle, \( \rho_0 \) is the radius of influence of each obstacle.

There is no prior knowledge of the environment in DRL algorithm. All state value functions \( V(s) \) in the initial state are equal or completely random. Each action step is generated in a random state. That is to say, the state transition probability is equal in the Markov decision process model. For navigation behavioural decision-making problems, the return value will only be changed when the destination is reached, or an obstacle is encountered. The sparsity of the reward function \( R \) leads to low initial decision efficiency and multiple iterations. Especially for large-scale unknown environments, there are a large number of invalid iterative search spaces.

Combining APF to improve the DRL navigation behavioural decision-making:

- Determining the initial point position \((x_{initial}, y_{initial})\) and the target point position \((x_{goal}, y_{goal})\). In the initialization state value function, the target point is used as the gravitational potential field at the centre of the potential field [17]. It is initialized to the environment prior information according to the position of the target point, and the initialization value is set to be greater than or equal to 0.

- The algorithm searches the environment layer by layer. If an obstacle is found, the speed of MASS in the state \( s(t+1) = -A \) at \( t+1 \).

- Under the action of the potential field function, the state-action value function table is updated using the environment state value function:

\[ Q(s_i,a_i) = r + \gamma V(s_{i+1}) \]  

(9)

- The MASS randomly searches for state behavioral from the starting point, considering only \( v(S(t+1)) \geq 0 \) as the available state. The \( \varepsilon - greedy \) strategy is adopted, and the state-action value is updated every time it moves. When the target point is reached, the iteration ends and the next iteration will start from the starting point.

For the navigation behavioral decision-making of MASS in complex environment, the action state space is huge and the iteration speed is slow. When the target point input update initialization value function is added to improve the gravitational potential field, the MASS has directionality and can move quickly. The destination, while the random strategy ensures that it does not fall into local iterations.

5. Simulation verification and result analysis

5.1. Environmental model

This section selects the part of Tianjin Port as the environment modelling area to verify the effectiveness of the navigation behavioural decision-making model and algorithm of the MASS. The simulation experiment 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 the MASS, and each state can be mapped to each element in the environmental state set. In the simulation environment model,
there are two state values for each coordinate point, which are 1 and 0. One of them represents the navigable area, which is displayed as the sea blue area in the environmental model, and 0 represents the obstacle area in the environment. The model appears as a white and light black area. Figure 3 shows the simulation environment model, including obstacles such as ships, breakwaters, and harbour shores. For MASS, the location information of these obstacles is unknown.

![Figure 3. The simulation environment.](image)

5.2. Navigation behavioural decision-making based on DRL

This section demonstrates the traditional navigation behavioural decision-making model combined with reinforcement learning and deep learning. The decision-making goal consists of two parts: trending target point and obstacle avoidance. When there are no obstacles or obstacles in the environment which will not be within the safe distance, the MASS will randomly select the action to approach the target point with probability $\frac{e}{A(s)}$. When the obstacle appears in the safe encounter distance, the MASS passes the excitation function. Interact with the environment to avoid obstacles. Some of the model parameters in the experiment are set as: learning rate $\alpha=0.5$, attenuation factor $\gamma=0.8$, parameters $w=0.02$, ship speed $v_r=8kn$.

The experiment sets the initial position (582,324) and target point (190,655) of MASS in pixels. As shown in Figure 4(a), in the initial iteration, MASS cannot determine the temptation area in the simulation environment and falls into the “trap” sea area in the simulation port pool. As shown in Figure 4(b), after 500 iterations, the system gradually plans the effective path, but the collision obstacle phenomenon occurs many times in the process, and the planning path fluctuates significantly. From 500 to 1000 iterations, the collision phenomenon is gradually reduced, and the planning path fluctuation is slowed down, as shown in Figure 4(c). All the obstacles are effectively avoided in the iteration 1000 times and the planned path fluctuations are weak and gradually stabilized. As shown in Figure 4(d), until the 2000th iteration, the probability of random search is the smallest, and the system plans the final fixed path through the piloting behavioral decision-making to reach the target point.
5.3. Navigation Behavioural Decision-making Based on APF-DRL

The experiment sets the initial position (582,324) and target point (190,655) of MASS in pixels. In the early stage of the experimental iteration, MASS will also collide with the obstacle at different time steps, and there is no collision after the collision in the experiment. The MASS will return to the previous step and re-select the action strategy. Compared with the experiment in Section 5.2, as shown in Figure 5(a), in the initial iteration, MASS does not fall into a local iteration, the system first plans an effective path, and the collision obstacle phenomenon occurs multiple times in the process. As shown in Figure 5(b), after 500 iterations, the system plans a better path, the collision obstacle phenomenon decreases, and the path of the experiment with the same iteration step is shorter than that in Section 5.2. As shown in Figure 5(c), compared with the Section 5.1 iteration steps, the collision phenomenon is reduced and path fluctuation is significantly slowed down. As displayed in Figure 5(d), at the 1500th iteration, the system has completed the navigation decision-making and reached the optimal piloting strategy until the final 2000 iteration.

5.4. Result analysis

By comparing the two sets of experiments, it is found that APF-DRL algorithm has better adaptive effect. The algorithm of APF-DRL combined with artificial potential field not only avoids local iteration, but also improves the convergence speed. Figure 6 shows the algorithm iteration convergence graph.
The red line in the figure indicates the improved DRL iteration trend, and the black line indicates the DRL iteration trend. It can be intuitively seen that the APF-DRL algorithm performs better.

![Iterative convergence trend comparison result.](image)

Figure 6. Iterative convergence trend comparison result.

Two sets of experimental data were extracted, and the performance of two navigation behavioral decision-making algorithms were compared and analyzed from the aspects of the number of local iterations, large fluctuation iterations (waves more than 500 times), collision rate, optimal decision iteration number, and optimal decision iteration time, and the sample results are shown in table 1.

| Verification experiment | Trapped into local iterations frequency | Fluctuations > 500 iterations frequency | Collision ratio | Iteration times | The iteration time of optimal decision (s) |
|-------------------------|----------------------------------------|----------------------------------------|----------------|----------------|----------------------------------------|
| DRL                     | 47                                     | 638                                    | 1.88%          | 2147           | 859                                    |
| APF-DRL                 | 2                                      | 103                                    | 0.08%          | 1501           | 442                                    |

From Table 1, it can be seen that the collision ratio of APF-DRL is almost 0, successfully avoided local iteration. Besides, the number of fluctuations and trial and error rate are reduced, whereas the iteration speed is faster. In summary, by comparing the convergence and decision-making effects of APF-DRL and DRL algorithms from multiple aspects, this study finds that the APF-DRL is more adaptive to navigation behavioral decision-making and path planning in an unknown environment.

6. Conclusion

In this paper, the target point of the MASS is used as the centre of the gravity potential field, and the initialized value table is updated. The navigation behavioural decision-making model of steering collision avoidance is established by designing the behavioural space, the reward function, the motion search strategy and the dynamic value function. Compared with DRL, the APF-DRL algorithm can improve the reliability and convergence effect of MASS in exploring the optimal piloting behavioural decision-making in unknown environment. The training time is obviously shortened and can get better piloting behavioural decision-making and obstacle avoidance effects. Taking Tianjin Port of China as
an example, the APF-DRL has a better navigation decision-making strategy in the uncertain environment. In the future research, the uncertain dynamic obstacle environment will be increased, and the adaptive ability of the navigation behavioural decision-making model and the iterative speed of behavioural decision-making will be improved, so that it can be better applied to the actual scene.

Acknowledgments
This work was supported by the National Key Research and Development Program of China (Grant no. 2018YFB1601502), and the National Natural Science Foundation of China (Grant no. 51779028).

References
[1] China State Council, “A new generation of artificial intelligence development planning ”, 2017, http://www.gov.cn/zhengce/content/2017-07/20/content_5211996.htm
[2] Phoenix Technology, “Artificial Intelligence Big Event: AlphaGo Robot vs. Go Master Li Shishi”, 2016, http://www.techweb.com.cn/irouter/2016-03-07/2290885.shtml
[3] Iyiou, “Baidu ‘Apolon’ mass production off the assembly line, this L4 level automatic driving bus will be sent to Beijing, Japan and other places”, 2018, http://baijiahao.baidu.com/s?id=1605027032898360744&wfr=spider&for=pc
[4] GAO Zong-jiang,ZHANG Ying-jun,SUN Pei-hua, “Research summary of unmanned ship”, Journal of Dalian Maritime University. 2017, Volume 43,Issue 2, pp. 1-7.
[5] FAN Yun-sheng,LIU Jian,WANG Guo-feng,SUN Yu-tong, “Dynamic path planning for unmanned surface vehicle based on heterologous information fusion”, Journal of Dalian Maritime University. 2018, Volume 44,Issue 1, pp. 9-16.
[6] Chen Hua, “Preliminary Research on Local Path Planning for Unmanned Surface Vehicle”, Dalian of China: Dalian Maritime University, 2016. (in Chinese)
[7] Zhou, Hao, Dongming Zhao, and Xuan Guo. "Global Path Planning of Unmanned Surface Vessel Based on Multi-objective Hybrid Particle Swarm Algorithm." International Conference on Bio-Inspired Computing: Theories and Applications. Springer, Singapore, 2017.
[8] YANG Bai-cheng, ZHAO Zhi-lei, “Multi-ship encounter collision avoidance decisions based on improved simulated annealing algorithm”, Journal of Dalian Maritime University. 2018, Volume 44, Issue 2, pp. 22-26.
[9] Long, Pinxin, Wenxi Liu, and Jia Pan. "Deep-learned collision avoidance policy for distributed multiagent navigation." IEEE Robotics and Automation Letters 2.2 (2017): 656-663.
[10] Wang, Pin, Ching-Yao Chan, and Arnaud de La Fortelle. "A reinforcement learning based approach for automated lane change maneuvers." 2018 IEEE Intelligent Vehicles Symposium (IV). IEEE, 2018.
[11] Zhelo, Oleksii, et al. "Curiosity-driven exploration for mapless navigation with deep reinforcement learning." arXiv preprint arXiv:1804.00456 (2018).
[12] Dooraki, Amir Ramezani, and Deok Jin Lee. "Memory-based reinforcement learning algorithm for autonomous exploration in unknown environment." International Journal of Advanced Robotic Systems 15.3 (2018): 1729881418775849.
[13] Everett, Michael, Yu Fan Chen, and Jonathan P. How. "Motion planning among dynamic, decision-making agents with deep reinforcement learning." 2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). IEEE, 2018.
[14] Wang, Chengbo, et al. "Path Planning of Maritime Autonomous Surface Ships in Unknown Environment with Reinforcement Learning." International Conference on Cognitive Systems and Signal Processing. Springer, Singapore, 2018.
[15] Wang, Chengbo, et al. "Research on intelligent collision avoidance decision-making of unmanned ship in unknown environments." Evolving Systems (2018): 1-10.
[16] Melingui, Achille, et al. "A novel approach to integrate artificial potential field and fuzzy logic into a common framework for robots autonomous navigation." Proceedings of the Institution of Mechanical Engineers, Part I: Journal of Systems and Control Engineering 228.10 (2014):
[17] Dong Peifang, Zhang Zhi'an, Mei Xinhu et al., "Intensive Learning Path Planning Algorithm for Introducing Potential Field and Trap Search", Computer Engineering and Applications. 2018, Volume 2018, Issue 1, pp. 129-134.