Neurocomputing 468 (2022) 181–197

Contents lists available at ScienceDirect

Neurocomputing

journal homepage: www.elsevier.com/locate/neucom

Path planning and dynamic collision avoidance algorithm under COLREGs via deep reinforcement learning

Xinli Xu [a], Peng Cai [b], Zahoor Ahmed [c], Vidya Sagar Yellapu [c], Weidong Zhang [d,c,*]

[a] *School of Mechanical Engineering, University of Shanghai for Science and Technology, Shanghai 200093, China*
[b] *Department of EE Big Data, Human Horizons Technology Co., Ltd., Shanghai 200082, China*
[c] *Department of Automation, Shanghai Jiao Tong University, Shanghai 200240, China*
[d] *School of Information and Communication Engineering, Hainan University, Haikou 570228, Hainan, China*

ARTICLE INFO

Article history:
Received 11 May 2021
Revised 11 August 2021
Accepted 25 September 2021
Available online 8 October 2021
Communicated by Zidong Wang

Keywords:
Unmanned surface vehicle
COLREGs
Dynamic collision avoidance
DDPG
Cumulative priority sampling mechanism

ABSTRACT

As one of the core technologies of the automatic control system for unmanned surface vehicles (USVs), autonomous collision avoidance algorithm is the key to ensure the safe navigation of USVs. In this paper, path planning and dynamic collision avoidance (PPDC) algorithm which obeys COLREGs is proposed for USVs. In order to avoid unnecessary collision avoidance actions, the risk assessment model is developed, which is used to determine the switching time of path planning and dynamic collision avoidance. In order to train the algorithm which complies with the COLREGs, the encounter situation is divided quantitatively, which is regarded as the input state of the system, so that the high-dimensional input is successfully avoided. The state space of the USV is defined by relative parameters to improve the generalization ability of the algorithm, meanwhile, a network structure based on DDPG is designed to achieve the continuous output of thrust and rudder angle. Combined with path planning, collision avoidance, compliance with COLREGs and smooth arrival task, four kinds of reward functions are designed. In order to solve the problem of low training efficiency of experience replay mechanism in DDPG, cumulative priority sampling mechanism is proposed. Through the simulation and verification in a variety of scenarios, it is proved that PPDC algorithm has the function of path planning and dynamic collision avoidance in compliance with COLREGs, which has good real-time performance and security.

1. Introduction

With the rapid development of shipping economy in the world, the total number of ships has increased rapidly, which brings new challenges to the safe navigation of ships [1]. Although the equipment of modern ships is very advanced, accidents such as ship collisions and hitting rocks still occur frequently. According to statistical data, 89–96% of collision accidents are caused by human factors [2]. On the one hand, the ship failed to strictly comply with Convention on the International Regulations for Preventing Collisions at sea, (COLREGs) [3]. On the other hand, the complexity of ship's types and environment brings challenges to the helmsman. Especially when multiple ships meet, it is very difficult for the helmsman to evaluate the risk of each ship and determine the best strategy for collision avoidance. The research of autonomous collision avoidance algorithm for ships will fundamentally solve the safety problems caused by human factors, which is of great significance to ensure navigation safety and reduce the loss of life and property. Therefore, the automatic pilotage algorithm of ships is a valuable research topic.

During the whole voyage, when the environment is fully known and there is no change, the path planning of the USV can be completed at one time, that is static collision avoidance. However, the actual navigation environment is changing rapidly, which requires the USV to have the ability to deal with the sudden threat of dynamic obstacles, namely dynamic collision avoidance. So far, many achievements have been made in the research of static collision avoidance in known environment, while the dynamic collision avoidance in unknown environment is becoming a hot issue in the field of unmanned surface vehicles [4]. The common dynamic collision avoidance methods include dynamic path planning method, artificial neural network method, fuzzy logic method, velocity obstacle method, artificial potential field method and so on. Dynamic path planning algorithm is a continuous path planning method, which ensures that the obtained path is dynamic optimal solution and global optimal solution. Zhang and Zhang [5], Liu and Wang [6] study the problem of collision avoidance by path

[*] Corresponding author.
E-mail address: wdzhang@sjtu.edu.cn (W. Zhang).

https://doi.org/10.1016/j.neucom.2021.09.071
0925-2312/© 2021 Elsevier B.V. All rights reserved.

planning control. Artificial neural network is applied to dynamic navigation, literature [7–10] study the problem of collision avoidance by neural network method. Cheng et al. [11] introduce fuzzy logic method into control system of USVs, which is used to deal with complex dynamic threats. Velocity obstacle method skillfully transforms the collision avoidance problem in the position space into the velocity space [12], and calculates the optimal avoidance velocity based on the evaluation function. Artificial potential field method is a kind of virtual force method, which can be applied to the scene with dynamic obstacle ships because of its excellent real-time performance. Xu et al. [13] propose a dynamic collision avoidance algorithm via layered artificial potential field with collision cone, which realizes the dynamic collision avoidance of USVs in complex environment.

These methods have the capability of dynamic collision avoidance, but there are many more challenges that need to be addressed. First of all, due to the marine environment is complex and unknown, many possible scenarios are difficult to design [13], so it is difficult to establish an accurate collision avoidance model. Even worse, USVs are occasionally too ill-equipped to meet the complex control law produced by analytic algorithms. In addition, when the USV is sailing in an unknown environment, it is difficult to make a reasonable strategy for collision avoidance, which can balance all the dangerous situations.

In view of the above problems, the advantages of deep reinforcement learning appear gradually. It adopts a model free learning method [14], which has strong fitting ability, based on which useful features can be learned from high-dimensional data such as position, velocity and angle, then control strategies can be obtained to deal with complex navigation problems [15]. Reinforcement learning is gradually applied to the collision avoidance control of USVs. Cheng [16] proposes a concise obstacle avoidance algorithm with the deep Q-networks architecture. Wang [17] proposed a greedy obstacle avoidance algorithm on the basis of actor-critic architecture to achieve the goal with very little training cost. The reinforcement learning method above is used to solve the static collision avoidance problem of USVs.

In addition to the dynamic constraints, the USV is also subject to the influence of other ships, i.e. the actual COLREGs constraints are required for the USV to avoid collision in an unknown environment. To solve these problems, this paper proposes a path planning and dynamic collision avoidance (PPDC) algorithm. The main innovations and contributions of this work are summarized as follows:

Firstly, this paper designs the risk assessment model of the USV based on the dynamic arena [18] and the safe distancc of approach (SDA). They can reflect the security domain and dangerous situation of the USV in real time, which can help to determine the appropriate time to avoid collision, and control the USV to take corresponding actions.

Secondly, the marine environment is complex and changeable. When multi obstacle ships appear at the same time, the input dimension of the algorithm will be large and the calculation time will be long. In this paper, the encounter situation of COLREGs is divided quantitatively, and the encounter situation is regarded as one of the input states of the system. The input dimension is reduced and the convergence speed of the algorithm is improved.

Thirdly, the classical DQN algorithm is only suitable for the control of simple discrete actions, while the complex continuous control will lead to the number of explosive actions. The PPDC algorithm designed in this paper is used to solve the problem of continuous action space in the process of control, and realize the continuous output of thrust and rudder angle.

Finally, DDPG algorithm is lack of guidance strategy, which leads to low training efficiency and slow convergence speed. Cumulative priority sampling mechanism is designed, which significantly improves the learning efficiency of the USV.

This paper is organized as follows. Section 2 provides the mathematical model. Section 3 designs the detailed structure of the algorithm. The simulation results are in Section 4. This paper is wrapped up with some concluding remarks in Section 5.

2. Mathematical model

2.1. Dynamic model of the USV

This paper mainly studies the movement in the horizontal plane [19,20]. The main motion variables are the surge velocity u, the sway velocity υ, and the yaw velocity r in the body-fitted frame {b}. In the inertial coordinate system {n}, the position of x-axis, position of y-axis and heading angle are respectively represented by x, y and ψ. It is assumed that the USV can be regarded as a rigid body and the geodetic coordinate system is a typical inertial coordinate system. The motion model of the ship is shown in Fig. 1.

The nonlinear dynamic equations of motion can be expressed as follows:

$$\dot{\eta} = T(\psi)v$$

$$M\dot{v} = -N(v)v - g(v) + \tau + \tau_w \tag{1}$$

where$T(\psi)$ is the transformation matrix, $T^{-1}(\psi)T(\psi) = I_{3\times3}$. The position vector $\eta \overset{\text{def}}{=} [xy\psi]^T \in \mathbb{R}^3$, and the velocity vector $v \overset{\text{def}}{=} [u\upsilon r]^T \in \mathbb{R}^3$. All parameters are shown in Table 1. M is the inertia matrix, $M = M^T = M_A + M_{RB} > 0$, m is the mass of the USV, I_z is the rotary inertia of z-axis for the USV. $X_{\dot{u}}$,$Y_{\dot{v}}$, $Y_{\dot{r}}$, $N_{\dot{v}}$, $N_{\dot{r}}$ are coefficients of added mass. $N(v)v$ is used to describe hydrodynamic force. The impacts of coriolis centripetal force and fluid damping force are considered,$N(v)v = C(v)v + D(v)$. $C(v) = C_A(v) + C_{RB}(v)$, among them $X_u, X_{|u|u}, X_{uuu}$,Y_r, Y_υ,$Y_{|\upsilon|\upsilon}$,$Y_{|r|\upsilon}$,$Y_{|\upsilon|r}$,$Y_{|r|r}$,N_r, N_υ, $N_{|\upsilon|r}$,$N_{|r|\upsilon}$,$N_{|\upsilon|\upsilon}$, $N_{|r|r}$ are the hydrodynamic coefficient. $g(v) = [g_u, g_\upsilon, g_r]^T \in \mathbb{R}^3$ is the unmodeled dynamics.$\tau = [\,\tau_u \quad \tau_v \quad \tau_r\,]^T \in \mathbb{R}^3$ is the control force or yaw moment provided by the servo system. τ_u is the thrust generated by the propeller to drive the USV forward and backward. τ_r is the moment that drives the USV to turn. $\tau_r = \frac{1}{2}C_L\rho AV^2L$, among them, C_L is the lift coefficient, which is related to the geometry of the rudder blade and changes with rudder angle δ. ρ is the density of water, A is the area of rudder blade. V is the velocity of water at the rudder blade.L is the distance from the pressure center of rudder to the center of the USV.

For the underactuated USV,$\tau_v = 0$. $\tau_w = [\,\tau_{wu} \quad \tau_{w\upsilon} \quad \tau_{wr}\,] \in \mathbb{R}^3$ is the time-varying environment interference vector. Uncertainty term $g(v)$ and time-varying environmental interference term τ_w are defined follow the literature [21].

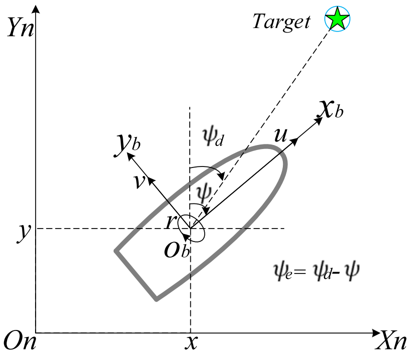


Fig. 1. The motion model of the ship.

Table 1
Parameters of the equations.

$$T(\psi)=\begin{bmatrix}\cos\psi & -\sin\psi & 0\\ \sin\psi & \cos\psi & 0\\ 0 & 0 & 1\end{bmatrix}$$

$$M_{RB}=\begin{bmatrix}m & 0 & 0\\ 0 & m & mx_g\\ 0 & mx_g & I_Z\end{bmatrix}$$

$$C_{RB}(v)=\begin{bmatrix}0 & 0 & -m(x_g r+v)\\ 0 & 0 & mu\\ m(x_g r+v) & -mu & 0\end{bmatrix}$$

$$D(v)=\begin{bmatrix}-X_u-X_{|u|u}|u|-X_{uuu}u^2 & 0 & 0\\ 0 & -Y_v-Y_{|v|v}|v|-Y_{|r|v}|r| & -Y_r-Y_{|v|r}|v|-Y_{|r|r}|r|\\ 0 & -N_v-N_{|v|v}|v|-N_{|r|v}|r| & -N_r-N_{|v|r}|v|-N_{|r|r}|r|\end{bmatrix}$$

$$M_A=\begin{bmatrix}-X_{\dot u} & 0 & 0\\ 0 & -Y_{\dot v} & -Y_{\dot r}\\ 0 & -N_{\dot v} & -N_{\dot r}\end{bmatrix}$$

$$C_A(v)=\begin{bmatrix}0 & 0 & Y_{\dot v}v+Y_{\dot r}r\\ 0 & 0 & -X_{\dot u}u\\ -Y_{\dot v}v-Y_{\dot r}r & X_{\dot u}u & 0\end{bmatrix}$$

$$g(v)=\begin{bmatrix}0.0279uv^2+0.0342v^2r,\\ 0.0912u^2v,0.0156ur^2+0.0278urv^3\end{bmatrix}^T$$

$$\tau_w=[2\cos(0.5t)\cos(t)+0.3\cos(0.5t)\sin(0.5t)$$
$$-3,\ 0.01\sin(0.1t),0.6\sin(1.1t)\cos(0.3t)]^T$$

2.2. Risk assessment model

When the USV is sailing, it is necessary to judge whether there are obstacles in the surrounding environment that may cause collision threat. In order to simplify the expression, our USV is called OU for short in the following paragraphs, target obstacle ship is called TS for short. The schematic diagram of motion parameters between the two ships is shown in Fig. 2. The coordinate of OU is (x_O, y_O), heading angle is φ_O, the velocity is v_O. The coordinate of TS is (x_T, y_T), heading angle is φ_T, the velocity is v_T. The intersection angle of heading is C_T, where $C_T = \varphi_T - \varphi_O$. If $C_T < 0$, then $C_T = C_T + 360^{\circ}$. The relative azimuth of OU is θ_O, the relative azimuth of TS is θ_T, v_R is relative velocity. The angle between the connecting line of the positions for two ships and the direction of v_R is α. DCPA is the distance of closest point of approach (CPA) between OU and TS, $DCPA = R_T * sin\alpha$. TCPA is the time to closest point of approach (CPA),$TCPA = R_T * cos\alpha/v_R$[18]. When $TCPA < 0$, TS has passed the CPA for the two ships, it is no longer a threat to OU. When $TCPA > 0$, TS has not passed the CPA for the two ships, so the collision risk remains.

In the 1980s, the british scholar E.M.Goodwin et al. [22] propose that there is a boundary around the ship, once another ship crosses the boundary of our ship, it means that this ship is a collision threat to our ship. In order to make the boundary inviolable, the british scholar Davis designes a boundary called 'Arena', which is introduced in this paper. It's a circle of radius R_{Arena} centered on our ship. If the target ship is outside the arena, our ship does not need to take any action. In literature [23], The calculation formula of R_{Arena} is as follows:

$$R_{Arena} = v_R\left(T_n + \frac{\varrho \times N \times SDA}{v_R}\right) \tag{2}$$

where, v_R is the relative velocity, T_n is the time required for our USV to turn 90° at full rudder, ϱ is the coefficient, generally ϱ is (1.5, 2). N is the visibility coefficient, when the visibility is good, the value is 1. When visibility is poor, the value is 1.25. SDA is the safe distancc of approach of two ships.

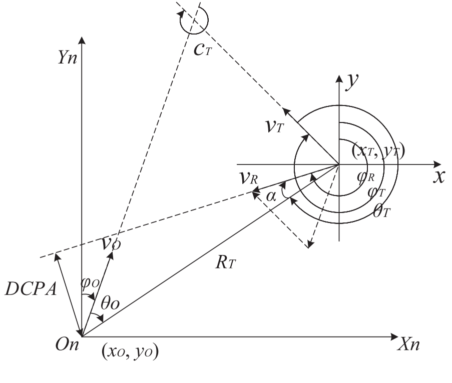


Fig. 2. Schematic diagram of motion parameters between two ships.

According to the COLREGs, "in order to avoid collision with the target ship, it is necessary to take avoidance actions to ensure that the two ships can pass at a safe distance " [3]. Therefore, the safe distancc of approach (SDA) is an important basis to judge whether there is a risk of collision between the two ships. The definition of SDA takes into account the size of the two ships, the relative position error and the yaw motion error [23].

$$SDA = (L_o + L_T) + 2 \times P + (L_o \times \pi/135 + L_T \times \pi/45) \tag{3}$$

Among them, L_o is the length of OU, L_T is the length of TS, P is the variance of kalman filter, $2 \times P$ is the position error of the ship under the interference of wind, waves and currents. $L_o \times \pi/135$ is half of the bandwidth of the track for OU, $L_T \times \pi/45$ is half of the bandwidth of the track for TS.

The above parameters are used to evaluate the collision risk in real time in this paper.

Definition 1. *Determination coefficient of collision risk* $\rho = min(R_{Ti}/R_{Arenai})$.

R_{Ti} is the distance between OU and TS_i (i = 1,2,…,n). When $\rho > 1$, namely$min(R_{Ti}/R_{Arenai}) > 1$, there is no obstacles in the arena of OU, and the task of the USV is 'sailing to the target'. When $\rho \le 1$, there is at least one obstacle in the arena of OU. The risk level is determined by calculating the value of R_{Ti}/R_{Arenai} for each TS. The smaller the ρ is, the higher the risk level is. Therefore, when taking collision avoidance action, the TS_i with the smallest ρ should be avoided first.

2.3. COLREGs constraint model

This paper studies the situation that OU is the give-way ship. Considering that the International Maritime Organization has formulated International Regulations for Preventing Collisions at Sea (COLREGs) [3], it is recommended that the avoidance actions should be determined according to COLREGs. In order to train the algorithm in accordance with COLREGs and reduce the input of state space for the system, the parameter σ_i of encounter situation is defined. When the two ships do not constitute any COLREGs situation, $\sigma_i = 0$, in other cases, σ_i is defined as follows,

(1) Head-on ($\sigma_i = 1$):When the encounter situation of OU and TS is head-on, both ships are directly in front of each other at an angle of ± 5°. The relative azimuth of OU θ_O satisfies the condition $\theta_O \le 5^{\circ}$ or $\theta_O \ge 355^{\circ}$. The relative azimuth of TS θ_T satisfies the condition $\theta_T \le 5^{\circ}$ or $\theta_T \ge 355^{\circ}$. When two ships are involved in an encounter situation, their surrounding regions are painted with the corresponding colors

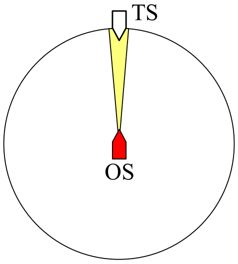


Fig. 3. Head-on.

to visualize the encounter situations better. In Fig. 3, OU's position is at the center of the circle, the velocity is pointing due north, and TS's position falls in the yellow area. At the same time, OU is also in this range of TS. In this case, OU should turn right to avoid TS.

(2) Starboard crossing:

(a) Starboard crossing-small angle($\sigma_i = 2$): When the encounter situation of OU and TS is starboard crossing-small angle, the condition $\theta_T \leqslant 45^\circ,\ 185^\circ \leqslant C_T < 210^\circ$ is satisfied. In Fig. 4, the position of TS falls in the pink sector area, and its velocity direction falls in the gray sector area. In this case, OU should turn right to avoid TS.

(b) Starboard crossing-large angle($\sigma_i = 3$): When the encounter situation of OU and TS is starboard crossing-large angle, the condition $\theta_T \leqslant 112.5^\circ,\ 210^\circ \leqslant C_T \leqslant 360^\circ$ is satisfied. In Fig. 5, the position of TS falls in the wathet sector area, and its velocity direction falls in the gray sector area. In this case, OU should turn left to avoid TS.

(3) Overtaking:

When the encounter situation of OU and TS is overtaking, the condition $112.5^\circ \leqslant \theta_T \leqslant 247^\circ$ is satisfied, and the velocity component of OU in TS's direction is larger than that of TS, namely $v_O > v_T * cosC_T$. In Figs. 6 and 7, TS's position is at the center of the circle and its velocity is pointing due north. OU's position falls in the blue or green areas respectively.

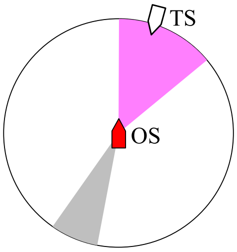


Fig. 4. Starboard crossing-small angle.

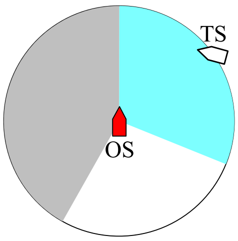


Fig. 5. Starboard crossing-large angle.

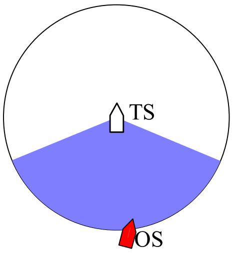


Fig. 6. Overtaking 1.

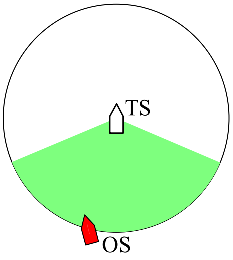


Fig. 7. Overtaking 2.

(a) Overtaking $1(\sigma_i = 4)$:When $\alpha < 90^{\circ}$, $DCPA > 0$, OU should turn right to avoid TS.

(b) Overtaking $2(\sigma_i = 5)$:When $270^{\circ} < \alpha \leq 360^{\circ}$, $DCPA \leq 0$, OU should turn left to avoid TS.

2.4. Strategy of state switching

We define three navigation states of the USV: FREE SAILING state, collision avoidance in compliance with COLREGs state (COLREGs state) and STABLE state. In the FREE SAILING state, the USV does not have the risk of collision with any obstacle ship, so the USV only needs to consider the task of sailing towards the target. In COLREGs state, there is at least one obstacle ship that poses a threat to the USV, and the USV needs to take action. In the COLREGs state, when multiple dynamic ships threaten the USV, these ships should be considered simultaneously. The USV should avoid according to the risk level (Definition 1). After the USV has taken a collision avoidance action and escaped from the collision avoidance state, it is necessary to stabilize the course of the USV in order to prevent it from entering the COLREGs state again. In this case, the USV should be in STABLE state.

By quantifying the navigation state and collision avoidance risk, the state switching strategy of subdivision can be defined. The flow chart of switching for state is shown in Fig. 8. First, the information of target is obtained to judge whether the USV reaches the target or not. If the USV reaches the target, the program ends. Otherwise, the state of the USV needs to be determined. The location, velocity and course of the target point and the obstacle ship are acquired by the sensors, laser radars, AIS navigational instrument and other equipment equipped with the USV. We use a combination of the status of the USV at the last moment and the current obstacle ship (TS_i) with the highest risk to determine the navigation status of the USV. The following are the conditions for determining the navigational status and the actions that should be taken.

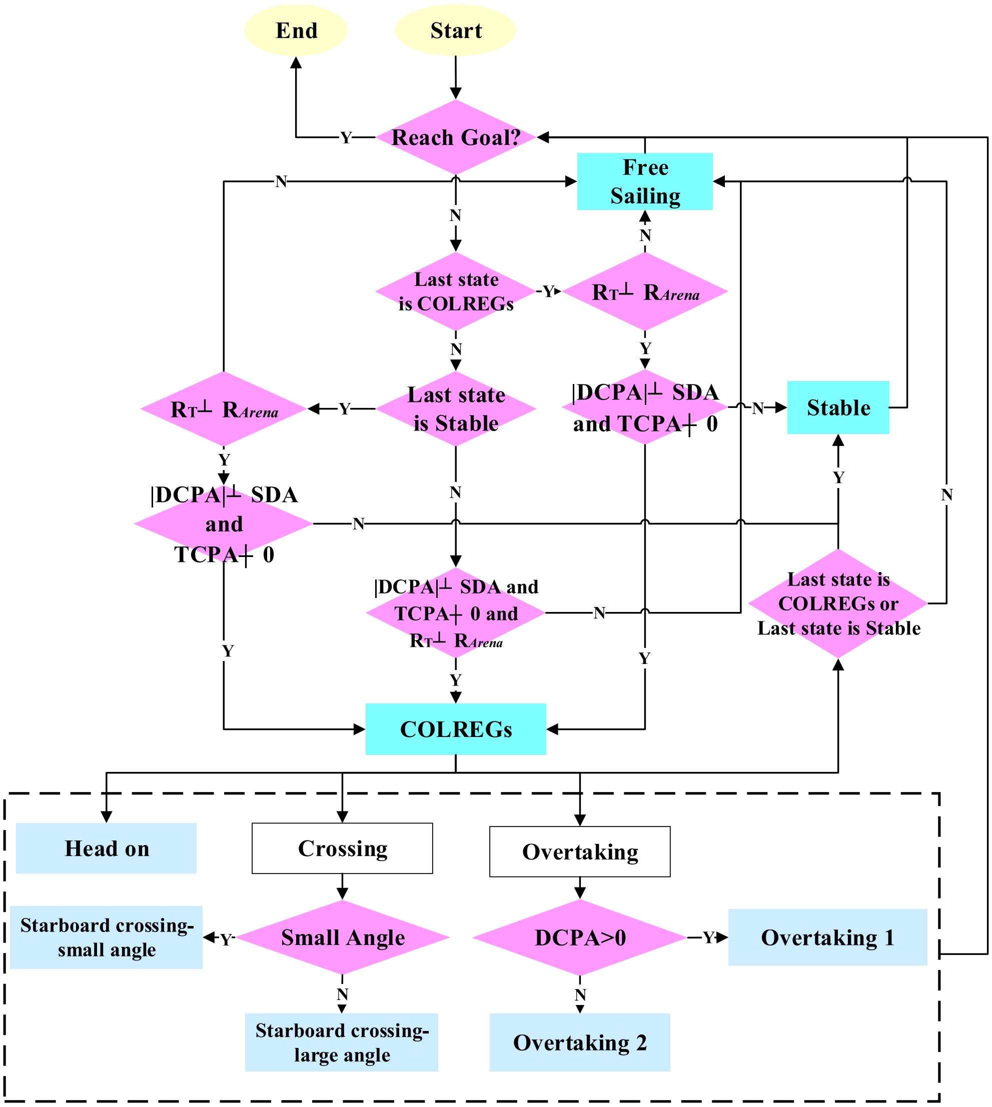


Fig. 8. Flow chart of state switching.

(1) If the following conditions are met. the USV is FREE SAILING state, and its task is to sail towards the target:
  (a) If the last state of USV is COLREGs state or STABLE state, in order to ensure the safety of the USV, we define that the USV is in FREE SAILING sate only when any obstacle ship is not within the arena of OU, that is $min(R_{Ti}/R_{Arenai}) > 1$.
  (b) If the last state of USV is neither COLREGs nor STABLE state, when DCPA > SDA or TCPA < 0 or $min(R_{Ti}/R_{Arenai}) > 1$, that is to say, the shortest encounter distance between the two ships is greater than SDA, or the two ships have passed safely, or all obstacle ships are outside the arena of OU.
  (c) If the last state of USV is neither COLREGs nor STABLE state, while DCPA $\leq$ SDA and TCPA ≥ 0 and $R_{Ti} \leq R_{Arena}$, but the encounter situation does not belong to any sub-state of COLREGs.
(2) If the following conditions are satisfied, the USV is in STABLE state, it needs to keep its heading angle:
  (a) If the last state of USV is COLREGs state, there are two cases:

Case 1: When $R_{Ti} \leqslant R_{Arenai}$ *and* $(DCPA > SDA$ *or* $TCPA < 0)$, that is to say, there is an obstacle ship in the arena of OU, but the minimum encounter distance between the two ships is greater than SDA or the two ships have passed safely.

Case2: When $R_{Ti} \leqslant R_{Arenai}$ *and* $(DCPA \leqslant SDA$ *and* $TCPA \geqslant 0)$, but the encounter situation between the two ships does not belong to any of COLREGs.

  (b) If the last state of USV is STABLE, when $R_{Ti} \leqslant R_{Arenai}$ and (DCPA $>$ SDA or TCPA < 0),or $(R_{Ti} \leqslant R_{Arenai}$ *and* $DCPA \leqslant SDA$ *and* $TCPA \geqslant 0)$,but the current encounter situation does not belong to any of COLREGs.
  (c) If the following condition is satisfied, the USV is in COLREGs state, it needs to take corresponding actions:

No matter what the state of the USV at the last moment is, as long as $(R_{Ti} \leq R_{Arenai} and \text{DCPA} \leq \text{SDA} and \text{TCPA} \geq 0)$ is true at the moment, that is to say, there is at least one ship in the arena of OU, and the minimum encounter distance of the two ships is within the SDA. Then, determine which COLREGs sub-state OU and the other ship belong to (refer section 2).

The flow chart of state switching is shown in Fig. 8.

3. Path planning and dynamic collision avoidance algorithm

The navigation control of ships is very complex, which needs to realize continuous control instructions based on continuous changing marine environment. The path planning and dynamic collision avoidance (PPDC) algorithm proposed in this paper is an algorithm based on DDPG [24,25]. By taking advantage of DDPG's ability to process continuous actions, the state space, action space and reward function for navigation tasks are designed, and an improved experience replay mechanism is designed. The network structure and algorithm framework are designed according to the requirements of navigation task.

3.1. State space

The state space S_t includes the state space of our USV S_O and the environment state space S_{env}. S_O includes the linear velocity v_O, angular velocity ω_O and heading angle ψ, that is

$$S_O = (v_O, \omega_O, \psi) \tag{4}$$

In order to improve the generalization ability of the algorithm, the environment state space S_{env} is composed of the relative parameters including the distanceR_T between TS_i and OU, the azimuth θ_T of TS_i relative to OU, and the azimuth θ_o of OU relative to TS_i, the ratio of OU's velocity to the component of TS's velocity in the direction of the OU's velocity is SF. The encounter situation of the two ships is σ_i. Therefore,

$$S_T = (R_T, \theta_T, \theta_o, SF, \sigma_i) \tag{5}$$

State S_{goal} includes the distance R_{G_t} between OU and target, the error of heading angle ψ_e (where $\psi_e = \psi - \psi_d$. ψ_d, the azimuth of the connecting line between OU and the target, as shown in Fig. 1), OG is the flag of whether our ship has reached the target, when OU reaches the target, the value of OG is 1, otherwise it is 0, that is to say

$$S_{goal} = (R_{G_t}, \psi_e, \mathrm{OG}) \tag{6}$$

So the above parts are integrated, the current state of the system is defined as:

$$S_t = (v_O, \omega_O, \psi, R_T, \theta_T, \theta_o, SF, \sigma_i, R_{G_t}, \psi_e, \mathrm{OG}) \tag{7}$$

All states in the state space are normalized, the normalized range is [−1,1].

3.2. Action space

In order to simplify the complexity of the problem, in the collision avoidance of traditional reinforcement learning, the speed of the USV is usually set to be constant. However, in practice, a constant velocity will result in a greater risk of collision. In order to reduce the risk of collision and perform the task of collision avoidance in the complex marine environment, the actions that USV can take in the process of collision avoidance are defined as acceleration, deceleration and turning.

In order to ensure the continuity of action space and the feasibility of actions, based on the dynamic characteristics of the USV, the velocity and angular velocity can be changed by controlling thrust and rudder angle, therefore, action space is defined as:

$$a = [\tau_u, \delta] \tag{8}$$

τ_u is thrust and δ is rudder angle. The USV is equipped with two electronic thrusters to control the thrust, the thrust range is [−50 N, 50 N], and the rudder angle is [−35°, 35°].

3.3. Reward function

The purpose of reinforcement learning algorithm is to gain the most cumulative rewards. The mainline event of the USV studied in this paper is sailing towards the target. When the mainline event occurs, the mainline rewards should be given in [26]. The USV can get mainline rewards with a certain probability through random attempts. But for the entire voyage, the probability of mainline events is very low. If there is no real-time feedback, it will be difficult for the USV to learn the skill of sailing towards the target. This is the problem of sparse returns in deep reinforcement learning. To solve this problem, continuous reward function is designed. We decompose the various states that the USV may encounter during sailing towards the target, and design the corresponding reward function.

Therefore, this paper designs four kinds of rewards, they are path planning reward, collision reward, COLREGs reward and goal reward.

(1) Path planning reward
   (a) Shape reward

According to the paper about shape reward published by Bratman in 2012 [27], shape reward can improve the speed of the convergence. For the task of sailing towards the target, shape reward is defined as penalty according to the distance between the USV and the target. When the USV is far away from the target, the penalty should be large, otherwise the penalty should be small. Therefore, shape reward is defined as follows,

$$r_{shape} = -\frac{R_{G_t}}{R_{initial}} \tag{9}$$

where R_{G_t} is the distance between the USV and the target at time t, and $R_{initial}$ is the initial distance between the USV and the target.

(b) Heading reward

In order to minimize sailing time and save fuel, it is necessary to keep the USV heading toward the target as far as possible. Therefore, the heading reward is designed to guide the heading towards the target. The heading deviation of USV is defined as ψ_e, where $\psi_e = \psi - \psi_d$. No matter the deviation is positive or negative, it means that the heading of the USV deviates from the ideal heading. Therefore, the heading reward after normalization is defined as,

$$r_{heading} \overset{\text{def}}{=} \begin{cases} \frac{-|\psi-\psi_d|}{180}, & |\psi-\psi_d| \le 180 \\ \frac{-(360-|\psi-\psi_d|)}{180}, & |\psi-\psi_d| > 180 \end{cases} \tag{10}$$

(2) Collision reward

In order to prevent the collision between OU and TS, collision reward should be designed. When a TS appears within the range of the OU's radar, it simply means that the OU has detected the TS, and only when the distance between the OU and TS is less than the threshold does the collision threat need to be considered. Whether TS enters the arena of OU mentioned above is regarded as the standard of threat judgment. When the distance R_{Ti} between OU and TS is less than R_{Arena}, the corresponding reward should be fed back. Collision reward is defined as follows,

$$r_{collision} = \begin{cases} -\left(1 - \frac{R_{Ti}}{R_{Arena}}\right)^2, & R_{Ti} \le R_{Arena} \\ 0, & R_{Ti} > R_{Arena}, \end{cases} \tag{11}$$

(3) COLREGs reward

When the USV encounters other ships in the open sea, in order to ensure the safety of navigation, the collision avoidance actions taken by the USV should comply with COLREGs. COLREGs reward is defined as follows,

$$r_{COLREGs} = \begin{cases} 0, & \textit{Compliance with COLREGs} \\ -\zeta, & \textit{Against COLREGs} \end{cases} \tag{12}$$

When USV obeys the COLREGs, the penalty is 0. Otherwise the penalty is $-\zeta$, ζ is a positive constant.

(4) Goal reward
(a) Goal reach reward

Reaching the target is the main goal of the USV, so the reward function should be designed to promote the USV to obtain this ability as soon as possible. Unlike other rewards, a significant positive reward for reaching the target should be given. We define that when the distance between the USV and target is less than R_{reach}, the USV is regarded as reaching the target. Therefore, goal reach reward is defined as follows,

$$r_{reach} = \begin{cases} 10, & R_{G_t} \leqslant R_{reach} \\ 0, & R_{G_t} > R_{reach} \end{cases} \tag{13}$$

(b) Speed reward

When the USV is near the target, it needs to slow down in advance so that it can reach the target smoothly, so we design speed reward to train the USV to have this skill. At the same time, if the USV is far away from the target, it should sail towards the target as fast as possible. Therefore, speed reward is defined as follows,

$$r_{speed} = \begin{cases} \frac{-|v_o|}{v_{omax}}, & R_{G_t} \le R_{decrease\,v} \\ \frac{v_o - v_{omax}}{v_{omax}}, & R_{G_t} > R_{decrease\,v} \end{cases} \tag{14}$$

where $\boldsymbol{v_o}$ is the real-time velocity of the USV, $\boldsymbol{v_{omax}}$ is the maximum velocity of the USV. We define that the USV should begin to slow down when the distance between the USV and the target point is $R_{decrease\,v}$.

Different reward functions are integrated into a heterogeneous reward function as follows,

$$R = \begin{bmatrix} \lambda_{shape} \\ \lambda_{Heading} \\ \lambda_{Collision} \\ \lambda_{COLREGs} \\ \lambda_{reach} \\ \lambda_{speed} \end{bmatrix}^T \begin{bmatrix} r_{shape} \\ r_{Heading} \\ r_{Collision} \\ r_{COLREGs} \\ r_{reach} \\ r_{speed} \end{bmatrix} \tag{15}$$

$\lambda_{shape}, \lambda_{Heading}, \lambda_{Collision}, \lambda_{COLREGs}, \lambda_{reach}, \lambda_{speed}$ are the weight coefficients of different reward functions.

3.4. Cumulative priority sampling mechanism

In the experience replay mechanism [28,29] of DDPG, the current state, action and other information are stored at each step as the experience of the agent, so as to form the sequence of experience replay. When training the network, mini batch empirical datas are randomly extracted as training samples, this uniform sampling mechanism ignores the importance of experience and the learning efficiency is low.

In order to improve the effect of training, cumulative priority sampling mechanism is proposed in this paper. First, the sample sequences with large cumulative reward are selected from the replay buffer with higher probability. Secondly, samples with large loss value are selected for training from the above sample sequences. This makes full use of the high-quality data in the replay buffer and increases the number of effective action samples. The former ensures the stability of the target value y_i, while the latter accelerates the convergence speed of the iteration of the action-value function.

The series of state transitions from the initial state to the termination state can be described as $\langle S_1, A_1, S_2, R_1 \rangle, \langle S_2, A_2, S_3, R_2 \rangle, \cdots$, such a sequence is called an episode in reinforcement learning [30,31]. The replay buffer is E={g_1,g_2,...,g_m}, where $g_i = \left[\left(S_1^i, A_1^i, S_2^i, R_1^i \right), \left(S_2^i, A_2^i, S_3^i, R_2^i \right), \cdots \right]$ is the i-th sequence in the replay buffer, $S_j = \left(S_j^i, A_j^i, S_{j+1}^i, R_j^i \right)$, S_j is a sample of sequence g_i. The cumulative reward of the agent in an episode is G_i. Let $p_i = G_i + \epsilon$, which is the priority of the i-th sequence. $\boldsymbol{\epsilon}$ is a very small positive number, which is used to ensure that all sequences have a priority greater than 0. The probability$P(i)$ of g_i is:

$$P(i) = \frac{p_i^\alpha}{\sum_{k=1}^m p_k^\alpha} \tag{16}$$

Among them, α is the regulator factor of priority. When α = 0, priority is not considered. When α = 1, sampling is based entirely on priority. Probability$P(i)$ of the sequence is used to sample the sequences in experience buffer E, $i \in \{1, 2, \cdots, m\}$. While sequences with large cumulative rewards are selected with greater probability, sequences with small cumulative rewards also have the chance to be selected, which ensures the diversity of samples.

The sequences selected from experience buffer E is represented by g_i, the total number is S. These sequences form the experience buffer$E$, $E = \{g_1, g_2 \cdots, g_S\}$. Each of the samples of E is represented by S_u, $S_u = \left(S_{ut}, A_{ut}, S_{ut+1}, R_{ut} \right)$,$u \in \{1, 2, \cdots, N\}$. For each sample S_u, the difference between the current Q value $Q\left(S_u, A_u \middle| \theta^Q\right)$ and the target Q value $y_u = R + \gamma Q' \left(S_u', A_u' \middle| \theta^{Q'} \right)$ is calculated, that is

$$\Delta Q_u = y_u - Q\left(S_u, A_u \middle| \theta^Q\right) \tag{17}$$

Let $p_{su} = \Delta Q_u + \epsilon$, which is the priority of the u-th sample, ϵ is a very small positive number, which ensures that all samples have priority greater than 0. The sampling probability $P_s(u)$ of S_u is:

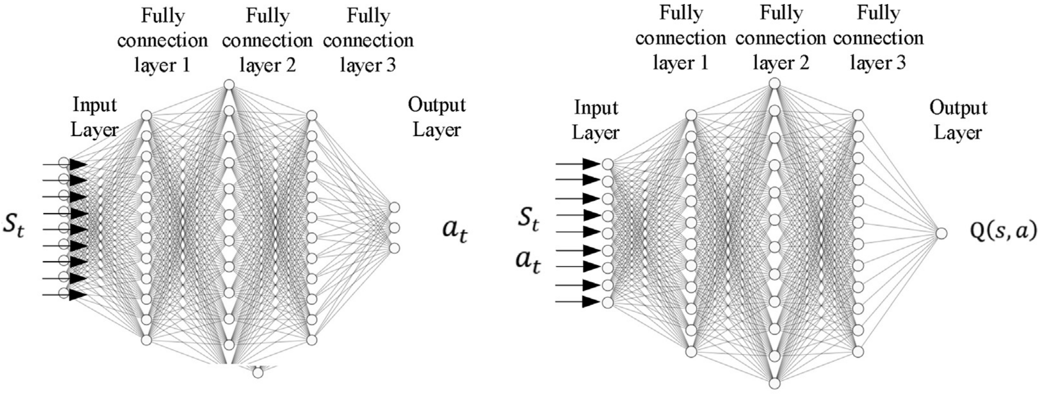
$$P_S(u) = \frac{p_{su}^\beta}{\sum_{k=1}^N p_{sk}^\beta} \tag{18}$$

β is the regulator factor of priority. When β = 0, priority is not considered, when β = 1, sampling is based entirely on priority.

Samples in experience buffer E are sampled according to probability $P_S(u)$. This priority sampling can be seen as a method of adding stochastic factors in selecting experiences since even those with low loss value can still have a probability to be replayed, which guarantees the diversity of sampled experiences. Such diversity can prevent the neural network from being over-fitting.

3.5. Network design

The algorithm framework of PPDC includes neural network module, experience replay module and environment processing module. The neural network module of PPDC includes four parts: on policy network, target policy network, online Q network and target Q network. On policy network and target policy network



(a) Actor neural network.

(b) Critic neural network.

Fig. 9. Structure of network.

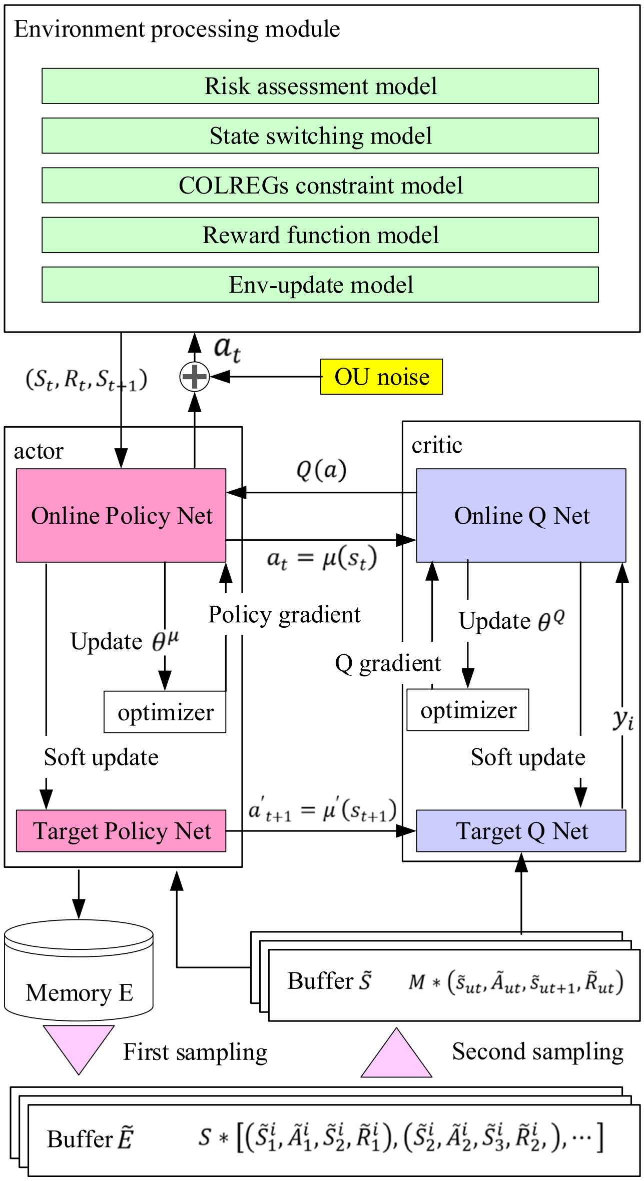


Fig. 10. Training framework of PPDC.

have the same network structure, we call it the actor network. online Q network and target Q network have the same network structure, we call it the critic network. The structure of actor network and critic network is shown in Fig. 9.

Fig. 9(a) is actor network whose input is environmental state S_t. The hidden layer consists of three fully connected layers (300 neurons in each layer), the full connection layer adopts ReLU activation function, and the output layer adopts Tanh activation function. The output is normalized control instruction of angular velocity a_{ω_t} and linear velocity a_{v_t}. Fig. 9(b) is critic network whose inputs include not only the state S_t but also the actions that the actor network outputs. The output of the actor network is the evaluation Q value for (s, a)

Experience replay module consists of three parts, they are respectively experience buffer E, buffer E and buffer S. E is used to store the experience sequence. E is used to store the sequence obtained from the first sampling. S is used to store the sequence obtained from the second sampling.

The environment processing module includes five parts, which are respectively Risk assessment model, State switching model, COLREGs constraint model, Reward function model and Env-update model.

Based on the above definition, we design the training framework of PPDC, which is shown in the Fig. 10. The pseudo code of PPDC is as follows.

Algorithm 1. PPDC algorithm for USVs.

```
Input: The number of episode for training U, frequency of
    training K, the size of experience buffer E N, the number of
    sequence for the first sampling S, the regulator factor of the
    first sampling $\alpha$. The number of samples for the second
    sampling M, the regulator factor of the second sampling $\beta$.
    the discount factor of reward $\gamma$, the end time of training T,
    the update frequency of target Q network L.
Output: Optimal network parameters
1: Initialize the parameters of Online Policy Net and Target
    Policy Net with $\theta^\pi$ and $\theta^{\pi'}$, the parameters of Online Q Net
    and Target Q Net with $\theta^Q$ and $\theta^{Q'}$
2: Initialize experience buffer $E$, the number of sequences in
    the experience buffer E $i$, the storage experience buffer of a
    single sequence $h$, the priority $p_1$
3: for episode = 1,U do
```

a (*continued*)

Algorithm 1. PPDC algorithm for USVs.

```
Initialize a OU process $\varkappa_t$ for action exploration
Receive initial observation state $s_1$
for t = 1, T do
    Select action $a_t = \pi(s_t|\theta^{\pi}) + \varkappa_t$ according to the online policy and exploration noise
    Execute action $a_t$ and observe new state $s_{t+1}$
    Store transition $(s_t, a_t, r_t, s_{t+1})$ in experience buffer E
      if t % K==0 and episode > U
        For $i$ = 1,N do
          $P(i) = \frac{p_i^{\alpha}}{\sum_{k=1}^{m} p_k^{\alpha}}$
            Calculate the probability of the first sampling
          $P(i) = \frac{p_i^{\alpha}}{\sum_{k=1}^{m} p_k^{\alpha}}$
        End for
      Based on sampling probability $P(i)$, S sequences are sampled from E and stored in experience buffer $E$
      For $i$ = 1,S do
          $P_S(u) = \frac{p_{su}^{\beta}}{\sum_{k=1}^{N} p_{sk}^{\beta}}$
          Calculate the probability of each sample
          $P_S(u) = \frac{p_{su}^{\beta}}{\sum_{k=1}^{N} p_{sk}^{\beta}}$
       End for
      Based on sampling probability $P_S(u)$, M sequences are sampled from $E$ and stored in experience buffer $S$
      Minimize the loss function to update critic network:
          $L_i = \frac{1}{N}\sum_i \left(y_i - Q\left(S_i, A_i \middle| \theta^Q\right)\right)^2$
          Using samples in $S$, Minimize the loss function to update critic network:
          $L_i = \frac{1}{N}\sum_i \left(y_i - Q\left(S_i, A_i \middle| \theta^Q\right)\right)^2$
      Update the online policy using the sampled policy gradient:
          $\nabla_{\theta^{\pi}} J(\theta) \approx \frac{1}{N}\sum_i \nabla_a Q\left(s, a \middle| \theta^Q\right)\Big|_{s=s_i, a=\pi(s_i)} \times \nabla_{\theta^{\pi}} \pi(s|\theta^{\pi})\big|_{s=s_i}$
      Update the target networks:
      $\theta^{Q'} \leftarrow \tau\theta^{Q} + (1-\tau)\theta^{Q'}$
      $\theta^{\pi'} \leftarrow \tau\theta^{\pi} + (1-\tau)\theta^{\pi'}$
End for
```

4. Simulation

In this part, the effectiveness of the algorithm is verified by simulation. The computer configuration is as follows: Intel Core i7-9750H six core processor, NVIDIA GTX1660Ti 6 GB graphics card, 16 GB DDR3L memory. The dynamic performance data of the USV is measured by the USV platform of IPAC (Information Processing and Advanced Control Group) in Shanghai Jiaotong University. In the simulation, the USV does not know the follow-up motion of the dynamic obstacle ships, which can only take corresponding actions according to the existing motion of the dynamic obstacle ships. The dynamic performance of the USV is within its limit power range, and its motion ability is controllable. The parameters of the ship are shown in Table 2 below.

The simulation environment is a rectangular area of 600 m*500 m. L_o, L_T is the length of our USV and the target ship respectively. $v_{Omax}, a_{max}, \omega_{max}, \alpha_{max}$ are the maximum linear velocity, linear acceleration, angular velocity and angular acceleration of our USV respectively. v_{Tmax} is the maximum velocity of the target ship. The number of samples in each batch is equal. The size of experience buffer E is 1000 sequences, and each sequence can store up to 500 samples. The number of sequences in the first sampling is 100, and the number of samples in the second sampling is 200. The maximum number of steps for each episode is set to 500. Learning rate of actor network is $\alpha^{\mu} = 1 \times 10^{-4}$, learning rate of critic network is $\alpha^{Q} = 1 \times 10^{-3}$. The discount factor $\gamma = 0.99$, the soft replacement coefficient $\tau = 0.01$.

Table 2
Parameters and performance of ships.

v_{Omax}	3.5 m/s	L_o	1.8 m
a_{max}	0.4 m/s^2	L_T	1.8 m
ω_{max}	0.2 rad/s	v_{Tmax}	3.5 m/s
α_{max}	0.05 rad/s^2		

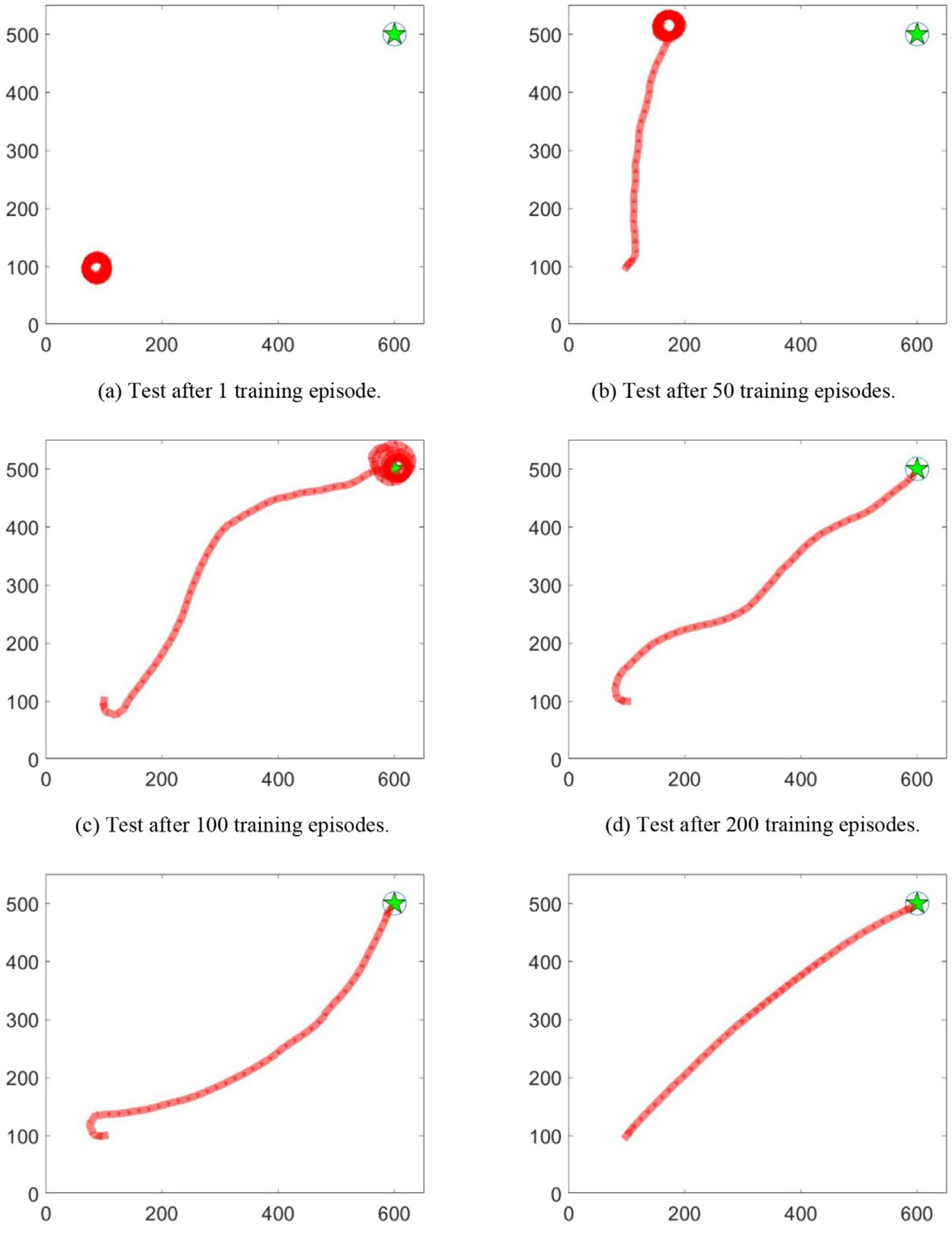
4.1. The task of path planning

In this section, the PPDC algorithm designed above is used for simulation. In the training process, in order to improve the generalization ability of the model, the initial direction of USV's velocity is random. The initial position is (100, 100), the initial velocity is 0, and the initial angular velocity is 0. The target is set at the point (600, 500). The task of path planning is trained with 1000 episodes, the maximum number of steps in each episode is 500, and the network is saved every 10 episodes during the training. When the USV reaches the target, the episodes ends. Fig. 11 shows a few typical moments selected from the training process.

As can be seen from Fig. 11, at the beginning of the training, the USV does not know how to sail, it turns around the starting point. After training 50 episodes, the USV learns to sail forward but the USV does not sail towards the target. The USV sails forward on the longitudinal axis and turns in circles again. After training 100 episodes, the USV can sail to the target, but it makes violent circles around the target without accurately reaching the target area (within ±5 m from the target). After training 200 episodes, the USV can reach the target area accurately, but there are some fluctuations in USV's track. After training 500 episodes, the USV has learned to sail to the target, the fluctuation of track is obviously reduced. After training 1000 episodes, USV has learned to sail to the target, and the path is smooth and almost optimal.

In the path planning task, in order to compare DDPG algorithm with PPDC algorithm, the two algorithms are both trained for 10 times. The results of the first 1000 episodes are extracted, and the average cumulative reward of the two methods are calculated. As shown in Figs. 12 and 13, the abscissa is the number of episodes, and the ordinate is the value of the average cumulative reward. In the early stage of training, the USV is in the stage of exploration and learning experience, and the average cumulative reward is relatively low. With the increase of training times, the USV enters the stage of using experience, the number of reaching the target increases, and the average cumulative reward also increases. After 100 episodes, the average cumulative reward of the cumulative priority sampling mechanism is larger than that of the original DDPG algorithm. In the 750 episodes, PPDC algorithm converges before DDPG algorithm.

According to the above results, PPDC algorithm using cumulative priority sampling mechanism is better than DDPG algorithm, which converges faster and is more stable after convergence, and has higher training efficiency.



(a) Test after 1 training episode.

(b) Test after 50 training episodes.

(c) Test after 100 training episodes.

(d) Test after 200 training episodes.

(e) Test after 500 training episodes.

(f) Test after 1000 training episodes.

Fig. 11. Training effect of path planning task.

4.2. Collision avoidance tasks in various encounter situations

In this section, possible dangerous situations during the encounter between OU and TS are simulated and verified. As shown in Fig. 14(a), the encounter situation of the two ships is head-on. The red ship is OU and the other one is TS. As the two ships get closer, TS enters the arena of OU at the 49th step, and OU enters the COLREGs state. The location detection region is shown in yellow, and OU takes a right turn to avoid TS. Then OU attempts to escape from COLREGs state, at the 51th step, it enters the STABLE state, and keeps its heading angle. At the 78th step, OU escapes from danger, and it returns to the FREE SAILING state. OU finally manages to avoid TS and reaches the target. Fig. 14(b) shows the curves of velocity and force of OU during the sailing.

The OU continuously monitors the environment as it sails. As shown in Fig. 15(a), at the 47th step, OU is in the state of COLREGs, and the encounter situation of two ships is starboard crossing-small angle. In this state, the location detection region is shown in pink, and the velocity detection region is shown in gray. TS is in the position detection region, and its velocity points to the gray region. USV takes a right turn to avoid TS. Then the OU attempts to escapes from COLREGs state, at the 51th step, it enters the STABLE state, and keeps the heading angle. At step 69, OU successfully escapes the danger and returns to FREE SAILING state. The collision avoidance actions complies with COLREGs, and OU finally reaches the target smoothly. Fig. 15(b) shows the curves of velocity and force of OU during the sailing.

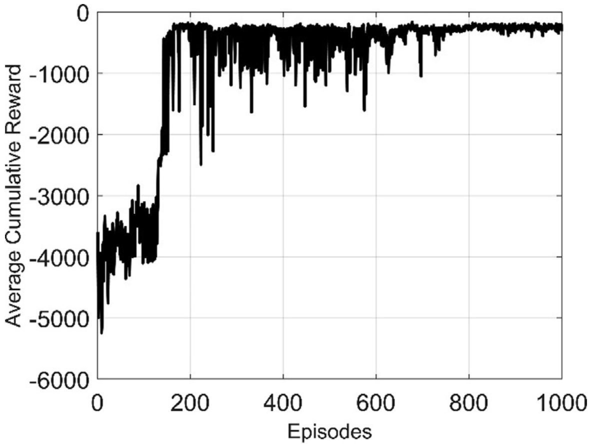


Fig. 12. Average cumulative reward of unimproved experience replay mechanism (DDPG algorithm).

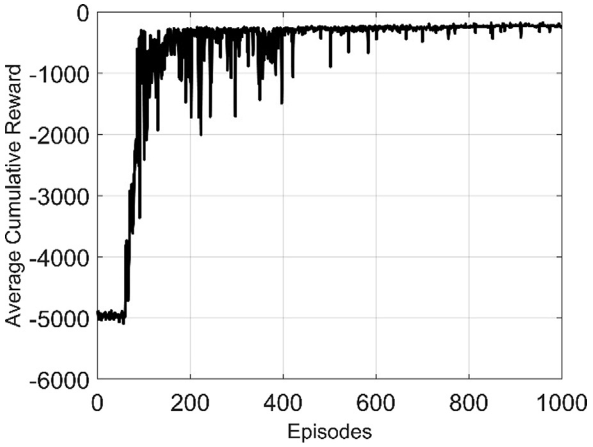


Fig. 13. Average cumulative reward of cumulative priority sampling mechanism (PPDC).

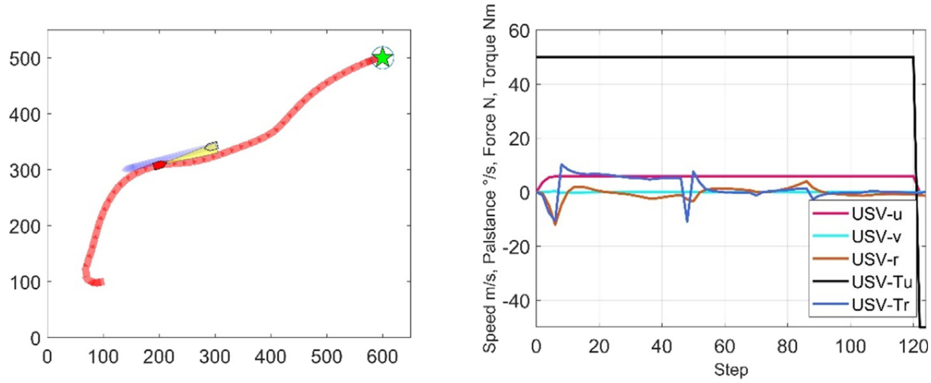
As shown in Fig. 16(a), as two ships approach, TS enters the arena of OU. At the 40th step, OU is in the state of COLREGs. Simultaneously, TS is detected in the starboard crossing-large angle region, which color is wathet, its velocity points to the gray region. The USV takes a left turn to avoid TS. Then the OU attempts to escapes from COLREGs state, at the 43th step, it enters the STABLE state, and keeps the heading angle. At the 48th step, OU escapes from danger, the STABLE state is cancelled, and it returns to the FREE SAILING state. The collision avoidance actions complies with COLREGs, and OU finally reaches the target smoothly. Fig. 16(b) shows the curves of velocity and force of OU during the sailing.

As shown in Fig. 17(a), OU is in the state of COLREGs, and the encounter situation of two ships is overtaking 1. The overtaking 1 region is shown in purple. At the 22th step, OU takes a right turn to avoid TS. Then the OU attempts to escapes from COLREGs state, at the 25th step, it enters the STABLE state, and keeps the heading angle. At the 65th step, OU escapes from danger, the STABLE state is cancelled, and it returns to the FREE SAILING state. The collision avoidance actions complies with COLREGs, and OU finally reaches the target smoothly. Fig. 17(b) shows the curves of velocity and thrust of OU during the sailing.

In Fig. 18(a), the encounter situation of two ships is overtaking 2. The overtaking 2 region is shown in green. At the 41th step, OU takes a left turn to avoid TS. At the 43th step, OU escapes from danger and returns to FREE SAILING state. The collision avoidance actions complies with COLREGs, and OU finally reaches the target smoothly. Fig. 18(b) shows the curves of velocity and thrust of OU during the sailing.

In the above five encounters, the collision avoidance actions all comply with COLREGs, and the OU finally reaches the target smoothly everytime, which indicates that the algorithm is effective.

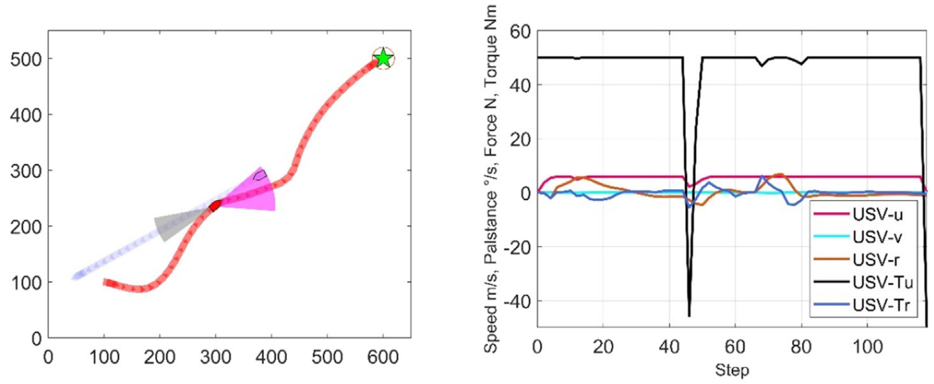
We define the voyage as a success if the USV complies with the COLREGS and finally reaches the destination, otherwise it is a failure. As shown in Fig. 19, we extract the success rate of the first 2000 episodes and obtain the curve of the success rate corresponding to COLREGS. The abscissa is the number of episodes, and the ordinate is the success rate. In the early stages of training, the USV continues to explore the environment, and when a collision occurs, the next episode begins. As the number of episode increases, the accumulation of experience gradually increases. After a period of trial and error, the success rate of the algorithm increases rapidly. Around the 200th episode, the success rates reach about 60%. Around the 1500th episode, success rates level off and reach 97%. It can be seen that the proposed PPDC algorithm can control the USV to make reasonable path planning and comply with the requirements of COLREGs when avoiding dynamic obstacle ships.



(a) Trajectory of the OU and TS.

(b) Velocity and thrust in head-on.

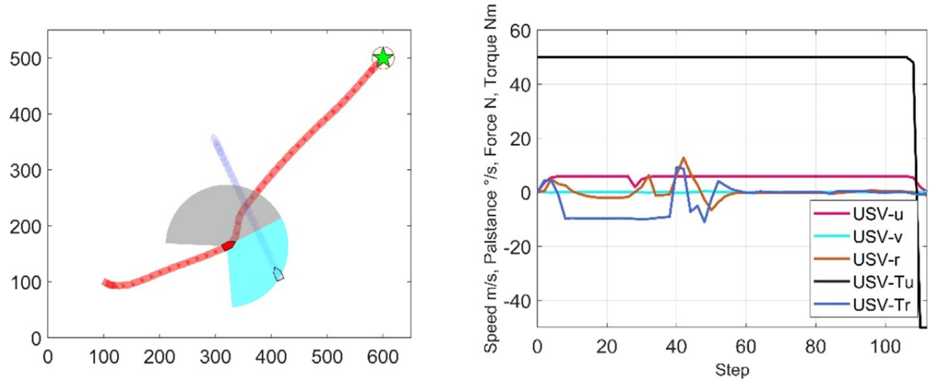
Fig. 14. Head-on.



(a) Trajectory of the OU and TS.

(b) Velocity and thrust in starboard crossing-small angle.

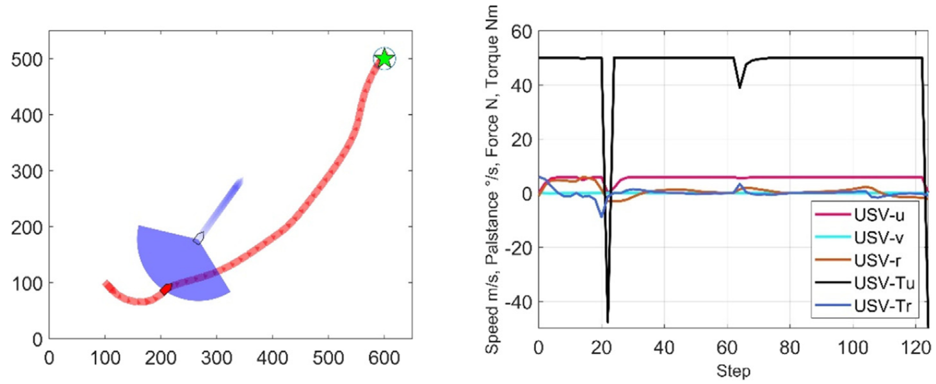
Fig. 15. Starboard crossing-small angle.



(a) Trajectory of the OU and TS.

(b) Velocity and thrust in starboard crossing-large angle.

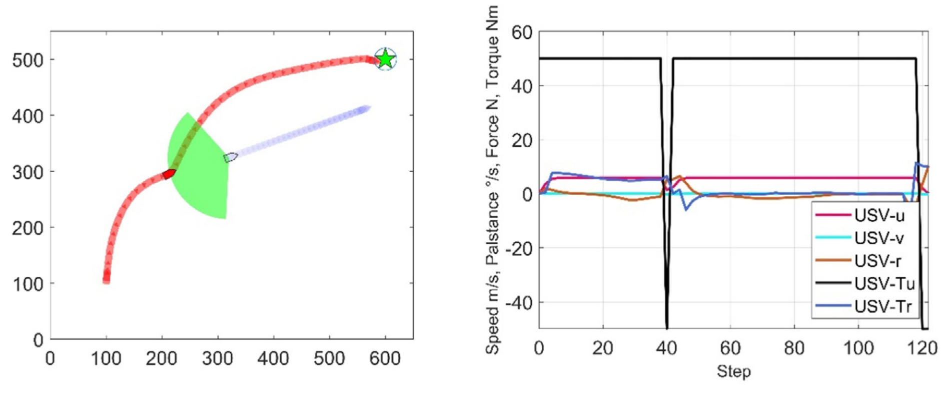
Fig. 16. Starboard crossing-large angle.



(a) Trajectory of the OU and TS.

(b) Velocity and thrust in overtaking 1.

Fig. 17. Overtaking 1.



(a) Trajectory of the OU and TS.

(b) Velocity and thrust in overtaking 2.

Fig. 18. Overtaking 2.

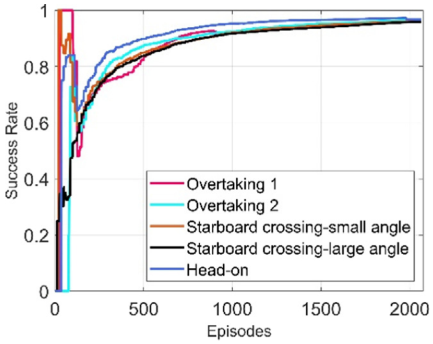
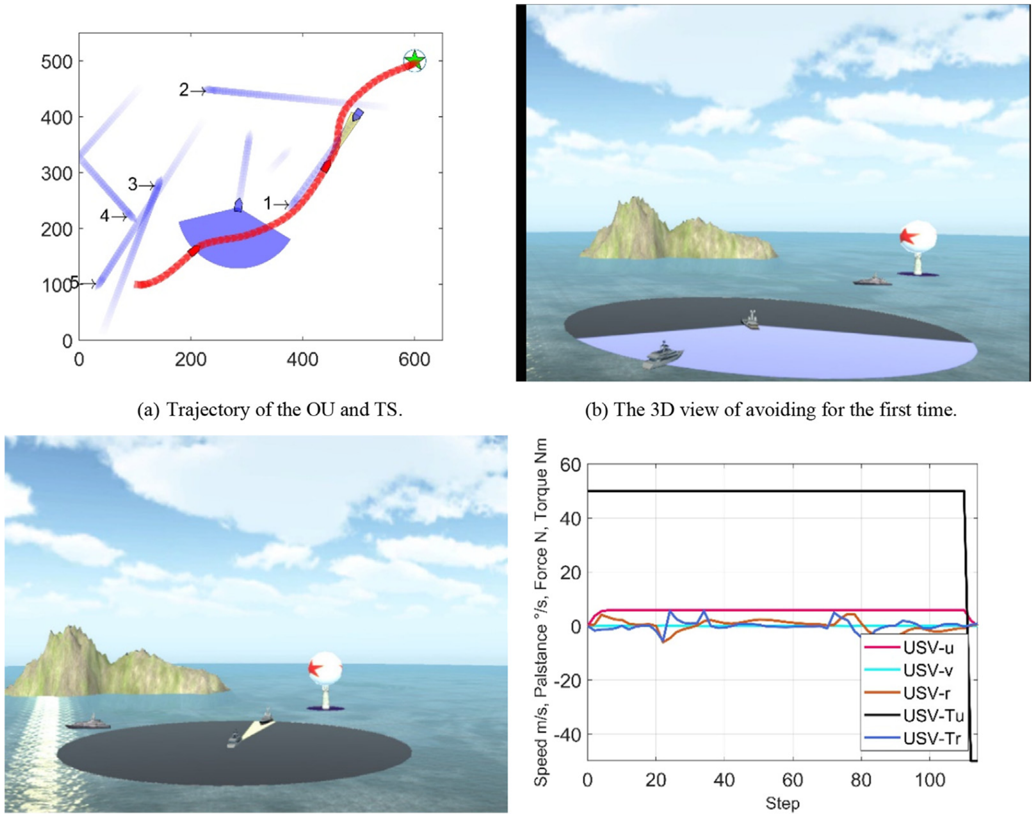


Fig. 19. The curve of success rate.

4.3. Collision avoidance of multiple USVs

In order to verify the path planning and dynamic collision avoidance ability of the algorithm in complex environment, five TS are set with random initial velocities and positions. In order to increase the complexity of collision avoidance mission for OU, the mirror navigation mode is designed. In mirror navigation mode, when an obstacle ship reaches the boundary of a specified area, its path will reflect mirrored, which will ensure that the obstacle ship navigates within the specified area, as shown in Fig. 20(a). However, the USV and the obstacle ship are still very difficult to encounter. To further increase the difficulty of collision avoidance, we set TS 1 as a challenge ship. The reason why it is



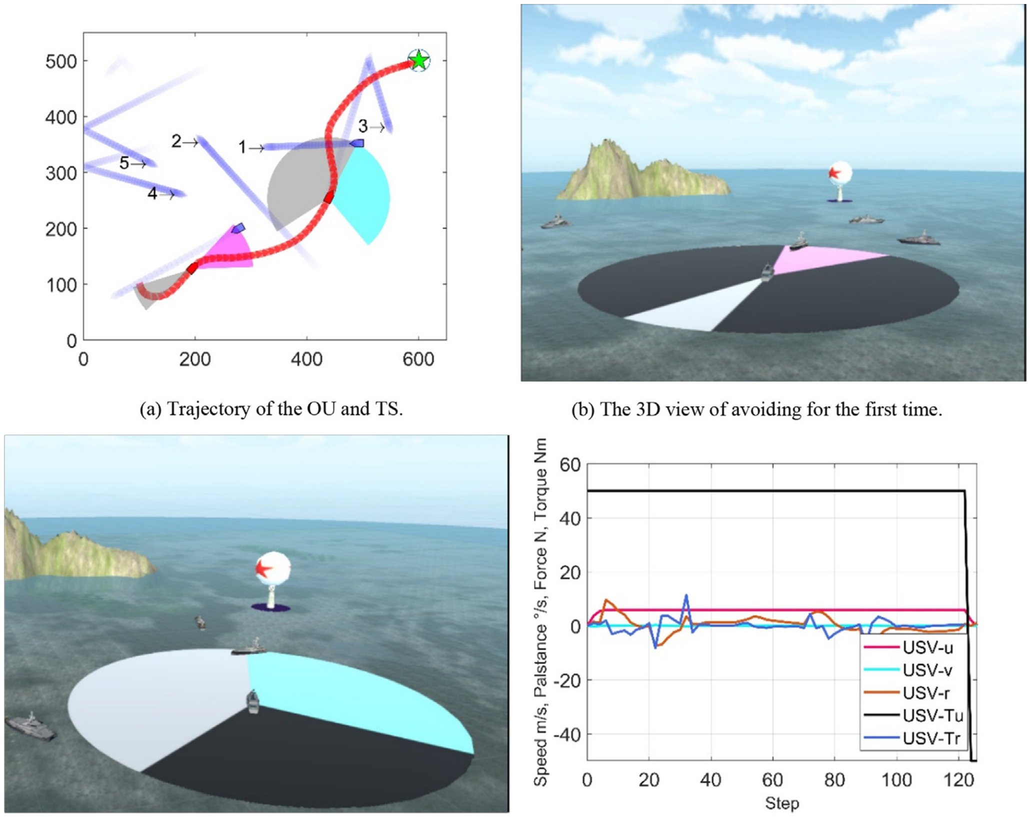
(a) Trajectory of the OU and TS.

(b) The 3D view of avoiding for the first time.

(a) The 3D view of avoiding for the second time.

(d) The curve of velocity and thrust.

Fig. 20. Encounter scenario 1.



(a) Trajectory of the OU and TS.

(b) The 3D view of avoiding for the first time.

(b) The 3D view of avoiding for the second time.

(d) The curve of velocity and thrust.

Fig. 21. Encounter scenario 2.

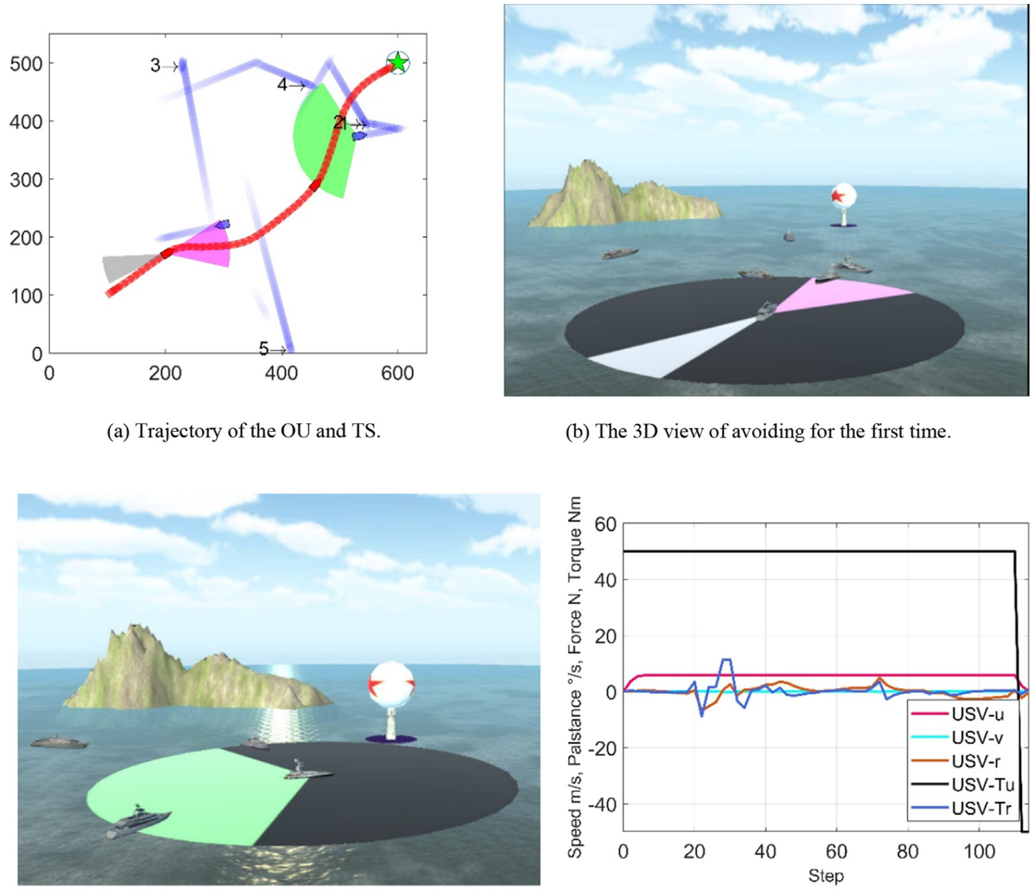
called a challenge ship is that every time the TS 1 is created, it poses a threat to the safety of OU, which ensures that the USV gains experience in collision avoidance. In order to distinguish features easily, we use python and unity3D to simulate jointly, establish 3D navigation environment in unity3D, and complete the training of PPDC algorithm in python.

Fig. 20(a) depicts the trajectory of OU and the other ships. TS 1,2,3,4,5 are generated at the same time. The first ship the OU encounters is TS 3, but it does not pose a threat to OU. Then, OU encounters TS 1, and the situation is overtaking 1. As shown in Fig. 20(b), at step 22, the USV enters the COLREGs state. TS 1 is at the center of the sector, which is located at (284.77, 240.54), while OU is located at (208.15, 162.72). The USV makes a right turn to avoid the danger and escapes at step 34. The USV then detects the newly created TS 1 and the encounter situation is head-on, as shown in Fig. 20(c). The positions of the OU and TS 1 are (496.51, 401.80), (443.55, 313.91) respectively, and OU takes a left turn to avoid TS 1. At step 83, OU dodges TS 1 and begins to sail towards the target. Throughout the voyage, OU avoids TS 1 twice, and the collision avoidance actions comply with COLREGs. OU finally reach the target safely with 116 steps in total. The curve of velocity and thrust for OU during the whole collision avoidance process is shown in Fig. 20(d).

As shown in Fig. 21(a), at step 22, OU detected TS 1, and the encounter situation of the two ships is starboard crossing-small angle. Which is shown in Fig. 21(b), OU locates at (197.45, 131.21), TS 1 locates at (274.78, 197.87). After OU takes a right turn and completes the collision avoidance action it returns to Free Sailing state. As shown in Fig. 21(c), at step 72, OU enters the COLREGs state again. OU takes a left turn to avoid TS 1. Finally, OU arrives at the target safely with 128 steps in total. The curve of velocity and thrust for OU during the whole collision avoidance process is shown in Fig. 21(d).

Fig. 22(a) depicts the trajectory of the OU and other ships. OU detects the TS 1 at the beginning of the voyage, and the encounter situation of the two ships is starboard crossing-large angle, as shown in Fig. 22(b). At the 22th step, OU takes a right turn to avoid the collision and returns to Free Sailing at step 37. Then, in the process of sailing to the target, OU and TS 1 form an overtaking 2 situation, as shown in Fig. 22(c). At this point, OU takes a left turn to avoid TS 1. It can be seen that the encounter situation OU faced is very complicate when approaching to the target. Both TS 2 and TS 4 are in the vicinity of OU, but they do not pose any threat to OU, so the navigation is not affected and OU arrives at the target smoothly with 116 steps in total. The curve of velocity and thrust for OU during the whole collision avoidance process is shown in Fig. 22(d).

In conclusion, we can see that with this algorithm, OU can avoid multiple burst dynamic obstacle ships which has the ability of path planning and dynamic collision avoidance in compliance with COLREGs for the USV.



(a) Trajectory of the OU and TS.

(b) The 3D view of avoiding for the first time.

(c) The 3D view of avoiding for the second time.

(d) The curve of velocity and thrust.

Fig. 22. Encounter scenario 3.

5. Conclusion

This paper studies the PPDC algorithm of USVs for path planning and dynamic collision avoidance in unknown marine environment. In order to avoid unnecessary collision avoidance actions, dynamic arena and the safe distancc of approach (SDA) are used to build real-time risk assessment model for USVs, the switching time of “FREE SAILING state ”, “STABLE state” and “COLREGs state” is designed. Dynamic model of ship is established, which makes the output action more reasonable and effective. The encounter situation of complying with COLREGs is divided quantitatively, and the high-dimensional input is successfully avoided. In this paper, four kinds of reward functions are designed, which are path planning reward, collision reward, COLREGs reward and goal reward. When we define the state space, we use relative parameters to improve the generalization ability of the algorithm, at the same time, distance, heading angle, velocity, collision avoidance, COLREGs and other factors are considered. Cumulative priority sampling mechanism is proposed, which makes full use of the high-quality data in the experience buffer, the number of effective actions of samples is increased. Compared with DDPG algorithm, PPDC algorithm improves the efficiency of exploration and achieves better results on the average cumulative reward.

The performance of the algorithm is verified on the simulation platform of Shanghai Jiaotong University. Through the task of path planning, task of collision avoidance in five encounter situations and task of collision avoidance for multi ships in three scenarios, it is verified that the algorithm has the ability of path planning and dynamic collision avoidance in compliance with COLREGs for USVs. Experimental results show that the algorithm has the characteristics of real-time and safety in avoiding multi dynamic obstacle ships.

How to establish a more accurate dynamic model and achieve smoother thrust and torque control will be the next research direction.

CRediT authorship contribution statement

Xinli Xu: Conceptualization, Methodology, Software, Visualization, Formal analysis, Writing – original draft, Writing - review & editing. **Peng Cai:** Visualization, Formal analysis, Data curation, Software, Writing – original draft. **Zahoor Ahmed:** Visualization, Formal analysis, Investigation, Writing - review & editing. **Vidya Sagar Yellapu:** Validation, Formal analysis, Investigation, Writing - review & editing. **Weidong Zhang:** Supervision, Resources, Funding acquisition, Project administration, Writing - review & editing.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This paper is partly supported by the Key R&D Program of Hainan (ZDYF2021GXJS041) and the Key R&D Program of Guangdong (2020B1111010002).

References

[1] D. Kim, K. Hirayama, M. Okimoto, Ship collision avoidance by distributed tabu search, TransNav: Int. J. Mar. Navig. Saf. Sea Transp. 9 (1) (2015) 23–29, https://doi.org/10.12716/1001.09.01.03.

[2] T. Statheros, G. Howells, K.M. Maier, Autonomous ship collision avoidance navigation concepts, technologies and techniques, J. Navig. 61 (1) (2008) 129–142, https://doi.org/10.1017/S037346330700447X.

[3] S. Mankabady, The law of collision at sea, North-Holland, 1972

[4] C. YongBo, M. YueSong, Y. JianQiao, S. XiaoLong, et al., Three-dimensional unmanned aerial vehicle path planning using modified wolf pack search algorithm, Neurocomputing 266 (3) (2017) 445–457, https://doi.org/10.1016/j.neucom.2017.05.059.

[5] G. Zhang, X. Zhang, Concise robust adaptive path-following control of underactuated ships using DSC and MLP, IEEE J. Ocean. Eng. 39 (4) (2014) 685–694, https://doi.org/10.1109/joe.2013.2280822.

[6] L. Liu, D. Wang, Z. Peng, Path following of marine surface vehicles with dynamical uncertainty and time-varying ocean disturbances, Neurocomputing 173 (2) (2015) 799–808, https://doi.org/10.1016/j.neucom.2015.08.033.

[7] Y. Lu, C. Wen, T. Shen, W. Zhang, Bearing-based adaptive neural formation scaling control for autonomous surface vehicles with uncertainties and input saturation, IEEE Trans. Neural Networks Learn. Syst. (2020), https://doi.org/10.1109/TNNLS.2020.3025807.

[8] G. Zhang, S. Chu, X. Jin, W. Zhang, Composite neural learning fault-tolerant control for underactuated vehicles with event-triggered input, IEEE Trans. Cybern. 51 (5) (2020) 2327–2338, https://doi.org/10.1109/TCYB.2020.3005800.

[9] G. Zhang, M. Yao, J. Xu, W. Zhang, Robust neural event-triggered control for dynamic positioning ships with actuator faults, Ocean Eng. 207 (2020), https://doi.org/10.1016/j.oceaneng.2020.107292 107292.

[10] J. Li, G. Zhang, C. Liu, W. Zhang, COLREGs-constrained adaptive fuzzy event-triggered control for underactuated surface vessels with the actuator failures, IEEE Trans. Fuzzy Syst. (2020), https://doi.org/10.1109/TFUZZ.2020.3028907.

[11] Y. Cheng, Z. Sun, Y. Huang, W. Zhang, Fuzzy categorical deep reinforcement learning of a defensive game for an unmanned surface vessel, Int. J. Fuzzy Syst. 21 (2) (2019) 592–606, https://doi.org/10.1007/s40815-018-0586-0.

[12] P. Fiorini, Z. Shiller, Motion planning in dynamic environments using velocity obstacles, Int. J. Robot. Res. 17 (7) (1998) 760–772, https://doi.org/10.1177/027836499801700706.

[13] X. Xu, W. Pan, Y. Huang, W. Zhang, Dynamic collision avoidance algorithm for USVs via layered APF with collision cone, J. Navig. 73 (6) (2020) 1306–1325, https://doi.org/10.1017/S0373463320000284.

[14] R.S. Sutton, Introduction: the challenge of reinforcement learning, Mach. Learn. (1992).

[15] A. Krizhevsky, I. Sutskever, G.E. Hinton, Imagenet classification with deep convolutional neural networks, Adv. Neural Inf. Process. Syst. 42 (11) (2012) 1097–1105, https://doi.org/10.1145/3065386.

[16] Y. Cheng, W.D. Zhang, Concise deep reinforcement learning obstacle avoidance for underactuated unmanned marine vessels, Neurocomputing 272 (5) (2018) 63–73, https://doi.org/10.1016/j.neucom.2017.06.066.

[17] X. Wang, X. Liu, T. Shen, W. Zhang, A greedy navigation and subtle obstacle avoidance algorithm for USV using reinforcement learning, in: IEEE Chinese Automation Congress (CAC). 2019, 2019, pp. 770–775, https://doi.org/10.1109/cac48633.2019.8996917.

[18] X. Bi, C. Jia, L. Wu, G. Jiang, Research on the domain model of ship collision avoidance action, J. Dalian Mar. Univ.: Nat. Sci. Ed. 29 (1) (2003) 9–12, https://doi.org/10.16411/j.cnki.issn1006-7736.2003.01.003.

[19] T.I. Fossen, Handbook of Marine Craft Hydrodynamics and Motion Control, New York, 2011.

[20] B. Yi, R. Ortega, D. Wu, W. Zhang, Orbital stabilization of nonlinear systems via Mexican sombrero energy shaping and pumping-and-damping injection, Automatica 112 (2020), https://doi.org/10.1016/j.automatica.2019.108661 108661.

[21] Z. Peng, D. Wang, T. Li, Z. Wu, Leaderless and leader-follower cooperative control of multiple marine surface vehicles with unknown dynamics, Nonlinear Dyn. 74 (2) (2013) 95–106, https://doi.org/10.1007/s11071-013-0951-3.

[22] E.M. Goodwin, A statistical study of ship domains, J. Navig. 28 (3) (1975) 328–344, https://doi.org/10.1017/s0373463300041230.

[23] W. Zhao, Collision Avoidance and Maritime Safety of Ships, Dalian Maritime University, 2006.

[24] W. Liu, Z. Wang, X. Liu, et al., A survey of deep neural network architectures and their applications, Neurocomputing 234 (2) (2017) 11–26, https://doi.org/10.1016/j.neucom.2016.12.038.

[25] R.S. Sutton, A.G. Barto, Reinforcement Learning: An Introduction, Cambridge, 1998.

[26] J. Schmidhuber, Deep learning in neural networks: an overview, Neural Networks. 61 (3) (2015) 85–117, https://doi.org/10.1016/j.neunet.2014.09.003.

[27] J. Bratman, S.P. Singh, J. Sorg, R.L. Lewis, Strong mitigation: nesting search for good policies within search for good reward, Adapt. Agents Multi Agents Syst. 6 (2) (2012) 407–414, 10.1.1.484.2952&rep=rep1&type=pdf.

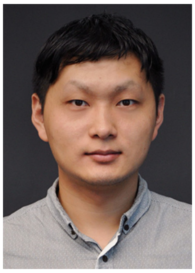
[28] D. Silver, A. Huang, C.J. Maddison, et al., Mastering the game of go with deep neural networks and tree search, Nature 529 (7587) (2016) 484–489, https://doi.org/10.1038/nature16961.

[29] K.V. Mnih, D Silver Kavukcuoglu, et al., Human-level control through deep reinforcement learning, Nature 518 (7540) (2015) 529–538, https://doi.org/10.1038/nature14236.

[30] Zhu Mei, Wang Xue, Wang Yin, Human-like autonomous car-following model with deep reinforcement learning, Transp. Res. Part Cemerg. Technol. 78 (5) (2018) 348–368, https://doi.org/10.1016/j.trc.2018.10.024.

[31] H. Modares, F.L. Lewis, M.B. Naghibi Sistani, Integral reinforcement learning and experience replay for adaptive optimal control of partially-unknown constrained-input continuous-time systems, Automatica 50 (1) (2014) 193–202, https://doi.org/10.1016/j.automatica.2013.09.043.

Xinli Xu received the M.S. and Ph.D. degrees in College of Mechanical and Electric Engineering from Changchun University of Science and Technology, China, in 2013, and 2017, respectively. She then worked as a Postdoctoral Fellow at Shanghai Jiao Tong University. She joined University of Shanghai for Science and Technology in 2020. Her research interests include deep reinforcement learning, automatic driving and unmanned system.



Peng Cai received the B.Eng. degree from the College of Automotive Engineering, Jilin University, China, in 2012. He joined FAW R&D Center as the head of Hongqi vehicle networking ecology in 2012, then he joined Human Horizons Technology Co., Ltd as the head of EE big data in 2018. His research interests include automatic driving, deep learning and big data.

Zahoor Ahmed received the B.Sc. degree in Electronics Engineering from the University College of Engineering and Technology (UCET), The Islamia University of Bahawalpur, Pakistan, in 2008 and the M.S. degree in Electrical Engineering from Government College University Lahore, Pakistan, in 2013. He is currently working towards his PhD degree in Control Science and Engineering at Automation Department, School of Electronics Information and Electrical Engineering, Shanghai Jiao Tong University, Shanghai, China. His research interests include multiagent system, robust control and networked control system.

Vidya Sagar Yellapu received his B.Tech. in Electrical and Electronics Engineering from JNTU, India in 2007, M.Tech. in Electrical Power Systems from JNTU Anantapur, India in 2009, and Ph.D. in Engineering from HBNI, India in 2016. He is currently a Postdoctoral Fellow at Department of Automation, SJTU, China. His main research interests are statistical data analyses, statistical process control, Kalman filters, estimation theory, multi-sensor fusion, multiscale modelling, vibration monitoring, Fault detection and diagnosis and control systems.

Weidong Zhang received his BS, MS, and PhD degrees from Zhejiang University, China, in 1990, 1993, and 1996, respectively. He joined Shanghai Jiao Tong University in 1998 as an Associate Professor and has been a Full Professor since 1999. From 2003 to 2004 he worked at the University of Stuttgart, Germany, as an Alexander von Humboldt Fellow. He is a recipient of National Science Fund for Distinguished Young Scholars of China and Cheung Kong Scholars Program. Presently he is Director of the Engineering Research Center of Marine Automation, Shanghai Municipal Education Commission. His research interests include control theory and pattern recognition theory and their applications in several fields, including power/chemical processes and USV/UAV/AUV. He is the author of 1 book and more than 300 refereed papers, and holds 61 patents. The Quantitative Control Theory he presented has been widely applied in 35 different backgrounds by more than 30 groups.