Path Planning for Ferry Crossing Inland Waterways Based on Reinforcement Deep Learning

Xiaoli Yuan [1,2], Chengji Yuan [1,2], Wuliu Tian [3,4,*], Gan Liu [1,2] and Jinfen Zhang [1,2,5,*]

[1] School of Transportation and Logistics Engineering, Wuhan University of Technology, Wuhan 430063, China

[2] National Engineering Research Center for Water Transport Safety (WTS Center), Wuhan University of Technology, Wuhan 430063, China

[3] Maritime College, Beibu Gulf University, Qinzhou 535000, China

[4] Hubei Key Laboratory of Inland Shipping Technology, Wuhan University of Technology, Wuhan 430063, China

[5] Inland Port and Shipping Industry Research Co. Ltd. of Guangdong Province, Guangzhou 512100, China

[*] Correspondence: tianwuliu@whut.edu.cn (W.T.); jinfen.zhang@whut.edu.cn (J.Z.);
Tel.: +86-27-86589905 (J.Z.)

Abstract: Path planning is a key issue for safe navigation of inland ferries. With the development of ship intelligence, how to enhance the decision–support system of a ferry in a complex navigation environment is one of the key issues. The inland ferries need to cross the channel frequently and, thus, risky encounters with target ships in the waterway are more frequent, so they need an intelligent decision–support system that can deal with complex situations. In this study, a reinforced deep learning method is proposed for path planning of inland ferries during crossing of the waterways. In the study, the state space, action space and reward function of the Deep Q-network (DQN) model are designed and improved to establish an autonomous navigation method for ferries considering both economy and safety. The DQN model also takes into account the crossing behavior, navigation economy and safety. Finally, the model is applied to case studies to verify its effectiveness.

Keywords: deep reinforcement learning; DQN; autonomous navigation; ferry crossing behavior

1. Introduction

Nowadays, shipping has undertaken a huge amount of transportation due to its low cost and economy. With the increase in waterway transportation volume, more ships will be navigating in busy and congested waterways. For ferries, the waterways' environment and encounter situations are complex, which will threaten navigation safety if decisions are made improperly. Although some studies have been carried out to predict environmental and operational scenarios, ferries are also faced with unknown and uncertain factors derived from encountered ships and the environment. The ferry should reach the endpoint to deliver goods and/or customers; meanwhile, they need to avoid collision with other ships. Hence, path planning and collision avoidance is one of the key issues for inland ferries.

For ships navigating in open waterways, long term path planning aims to plan a route in advance. However, ships navigating in narrow waterways need short-term path planning focusing on collision avoidance. In fact, path planning for narrow waterways should consider local collision avoidance together with a path from the start point to end point. Moreover, in congested waterways, the probability that one ship reacts to avoid collisions with more than one ship is likely to increase, thereby making collision avoidance decision-making more complex. Thus, an intelligent and autonomous path planning method considering all trajectories' decisions throughout the whole journey is needed to support human decisions.

In this study, an autonomous ferry-crossing navigation decision–support model is proposed, considering economy and safety based on deep reinforcement learning. The action space and state space are modified in order to adjust to the inland ferries.

2. Literature Review and Motivation

Many path planning methods for ships have been proposed extensively. In general, path planning can be divided into global path planning and local path planning. Global path planning refers to finding a long-term route that can avoid collisions with static obstacles and dynamic obstacles. Local path planning mainly refers to a path derived from collision avoidance decisions. Global path planning is utilized to find an acceptable path before navigation [1], and local path planning focused on short time and real time paths formed by anti-collision actions [2]. In general, path planning algorithms can be classified into two main categories: the classical algorithms and intelligent algorithms [3,4]. The former is usually utilized in global path planning. For classical algorithms in global path planning, the grid based algorithm and sampling-based algorithm are the most common ones, such as A*, D* and RRT (Rapid-exploration Random Tree). The grid based algorithms aim to search for grids by defining cost values for every neighbor point[5]. The iterations are updated until the path is finished. In most cases, the grid based method are utilized in known navigation environments [6-9]. The sampling-based algorithms are conducted by determining adaptive steps to achieve efficient path planning. These algorithms are primitively utilized in static path planning. In order to extend their applications, some studies made improvements from the view of a time slice [9]. However, these improvements would increase the calculation to a large amount. Then, the jump point of grid based algorithms together with a dynamic window are raised to reduce the calculations.

For local path planning and collision avoidance, some indexes such as DCPA (distance at closest point of approach), TCPA (time to the closest point of approach) and relative distance are utilized to assess the collision risk between two encountered ships, and subsequently, reciprocal or distributed collision avoidance decision algorithms are formulated to make anti-collision strategies [10-12]. However, these studies are based on the identification of all encountered scenarios, and their strategies are determined subsequently [13]. If there is a scenario that is not identified in early times, these models may result in failure. They are not robust and resilient when facing environment changes. Moreover, they are more suitable for open waterways and do not perform well enough to manage a large number of moving obstacles in narrow or complex waterways [4,14]. In order to assist autonomous navigation, some researchers make a combination of global and local path planning. Song, A.L et al. [15] established a two-level dynamic obstacle avoidance algorithm to achieve global path planning and local collision avoidance. The two levels are divided by defined thresholds which separate the entire path into nonemergency and emergency situations. However, this division sometimes may be trapped into local optimization, especially in a multi-ship encounter situation.

In summary, traditional algorithms lay a solid foundation for the development of path planning. However, they have some disadvantages in dealing with complex environment and autonomous navigation. Firstly, they usually fall into local optimal solutions. Secondly, the path planning strategies stored in the decision system are not sufficient to face dynamic obstacles, and any slight changes will result in planning failure. Thirdly, their abilities to interact with the environment are quite limited even when some traditional algorithms are improved in the time dimension. Fourthly, their ability in the integration of global and local path planning should be further improved.

Therefore, it is particularly important to formulate intelligent algorithms to fill the gaps. Intelligent algorithms are also called model-free algorithms. They can adapt to complex situations by interaction and communication with the obstacles. Among the intelligent algorithms, DRL (Deep reinforcement learning) makes a combination of DL (deep learning) and RL (reinforcement learning). DL is capable of perception capability, while

RL has the decision-making capability. Thus, DRL has human-level decision-making capabilities to deal with high-dimensional sensory input and actions. Deep Q-network (DQN), combining DL and Q-learning, was first raised in [16]. It improves the stability of the combination [17] and allows the agent to modify its strategy according to the received reward or punishment. Due to the ability of DQN, it has been utilized in robot and vehicle path planning. Yuan, J et al.[18] established a double DQN algorithm for an autonomous underwater vehicle to avoid collision with moving obstacles considering running time, total path and planning time. Bhopale, P et al. [19] modified a traditional Q-learning algorithm to deal with multiple obstacle avoidance problems for autonomous underwater vehicle. Meanwhile, DQN has been conducted to model ship path planning and navigation in some recent studies. Shen, H et al.[20] developed a DQN based approach for automatic collision avoidance of multiple ships incorporating ship maneuverability, human experience and navigation rules. Wang, W et al. [21] integrated COLREGs into the DRL algorithm and trained over multiple ships in rich encountering situations. These studies show that the application of DQN in path planning is promising, especially for the autonomous navigation of ships. In contrast with common ships, path planning for ferries in congested waterways has some inherent travelling patterns, which are dependent on seamanship and navigation rules for ferries. Thus, path planning for ferries should be further studied to promote autonomous navigation, especially in complex and congested waterways. During path planning, some mandatory rules should be followed, such as COLREGs and other related ferry rules. For ferry encounter situations, rules of crossing are different from other encounter situations, especially in the inland waterways of China.

Motivated by these studies and problems, we extend our previous studies on ferries to learn an autonomous path planning policy based on DQN [22,23]. In this study, an autonomous path planning model based on DQN is raised by recognition of ferries' crossing patterns defined by navigational experience and rules. This model can achieve autonomous path planning as defined by crossing patterns.

The remainder of this paper is organized as follows. Section 3 provides a brief overview of the formulation of autonomous path planning for ferries. Section 4 introduces the construction of a DQN-based path planning algorithm. The case studies are presented in Section 5. Conclusions and future work are presented in Section 6.

3. Construction of Autonomous Path Planning for Ferries

For the autonomous path planning of ferries, the main point is how to travel in a safe and efficient way. Safety and economy are the two conflict indexes. Therefore, path planning for ferries attempts to find a compromise between them. Economy is mainly referred to as the shorter path to reduce fuel and time consumption. Safety is determined by travelling relative distance and ship domain. Since the paths are short in ferry crossing situations, environment factors such as bathymetry and current only have limited influence on a short route (Tan et al. 2018), thus, such factors are not considered in this research.

3.1. Rules for Ferry Navigation

Before DQN-based path planning model construction, crossing patterns of ferries should be specified in the first place. Usually, a ferry should abide by COLREGs in open waterways. In congested and coastal waterways such as Dover Strait [24], Alesund area of Norway and the Gulf of Finland [25], a ferry should also follow COLREGs. However, in inland waterways of Yangtze river, rules have been adjusted based on COLREGs, especially when a ferry cross the channel. In order to clarify the encountered situations and responsibilities, a first-person perspective is implemented. The similarities and differences discussed in [26] are listed as follows:

(1)   COLREGs for normal ferry navigation

When navigating in the channel and encountering other ships, a ferry may be involved in crossing, overtaking and head-on situations, similar with normal ships. For a

ferry's normal navigation, they should follow COLREGs to avoid collisions [27,28]. The relevant rules with respect to collision avoidance for ferries are Rule 10, Rule 13, Rule 14, Rule 15, Rule 16 and Rule 17. Rule 10 applies to traffic separation schemes. Rule 13, Rule 14 and Rule 15 apply to overtaking, head on and crossing situations. Rule 16 and Rule 17 determine actions taken by given-way ships and stand-on ships. For ferries' normal navigation, four different encounter scenarios and their anti-collision responsibility according to COLREGs are demonstrated in Figure 1.
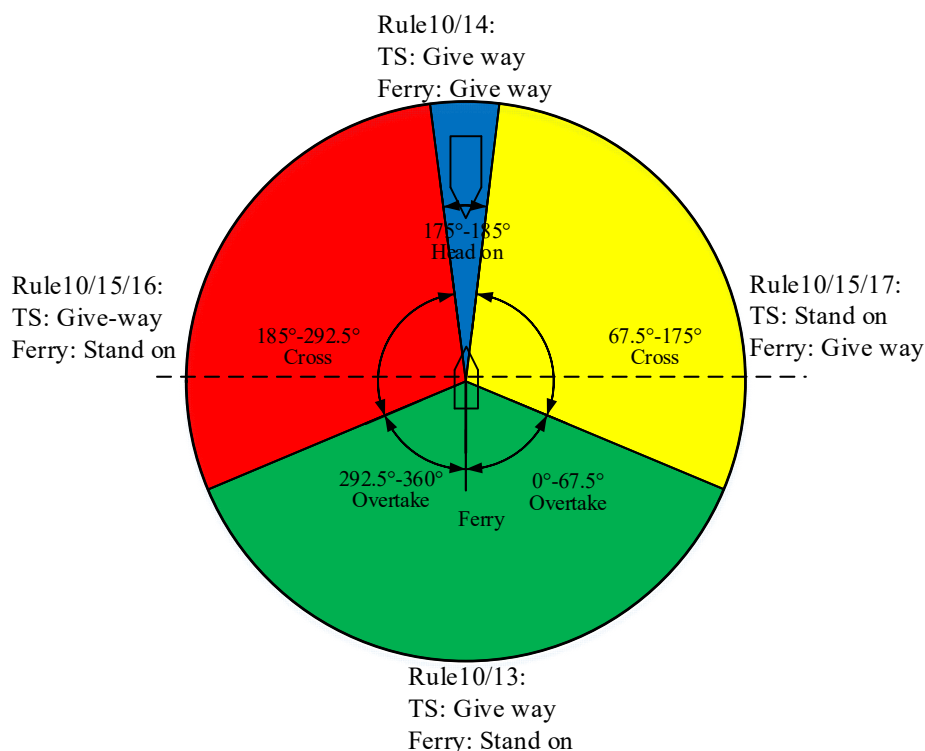

Figure 1. Ferry's collision avoidance responsibility according to COLREGs.

(2) Rules for ferry crossing the channel

Compared with the normal navigation of a ferry, crossing the channel is a typical navigation pattern, which creates crossing encounter situations with target ships. Therefore, the Rules of People's Republic of China for Avoiding Collisions were created to further describe the responsibilities of a ferry. The relevant rules are Rule 9 and Rule 14.

Rule 9: During the process of avoidance, the giving way ships should take actions to avoid the given way ships, and the given way ships should also pay attention to the actions of the giving way ships.

Rule 14: When ferries encounter other ships traveling along the channel or river, ferries must take avoiding actions. Travelling from the bow of other ships arbitrarily is forbidden for ferries.

Rule 9 determines anti-collision responsibilities in accordance with COLREGs. Rule 14 explains actions that a ferry should take to avoid collision when crossing the channel. What is more, crossing patterns when a ferry crosses the channel have been clarified as travelling from the bow and stern of target ships in the Yangtze River. When a ferry crosses the channel, four encountering situations are also created, similar with normal navigation. The responsibility for the four scenarios when a ferry is crossing the channel, according to COLREGs and Rules of People's Republic of China for Avoiding Collisions, is shown in Figure 2.
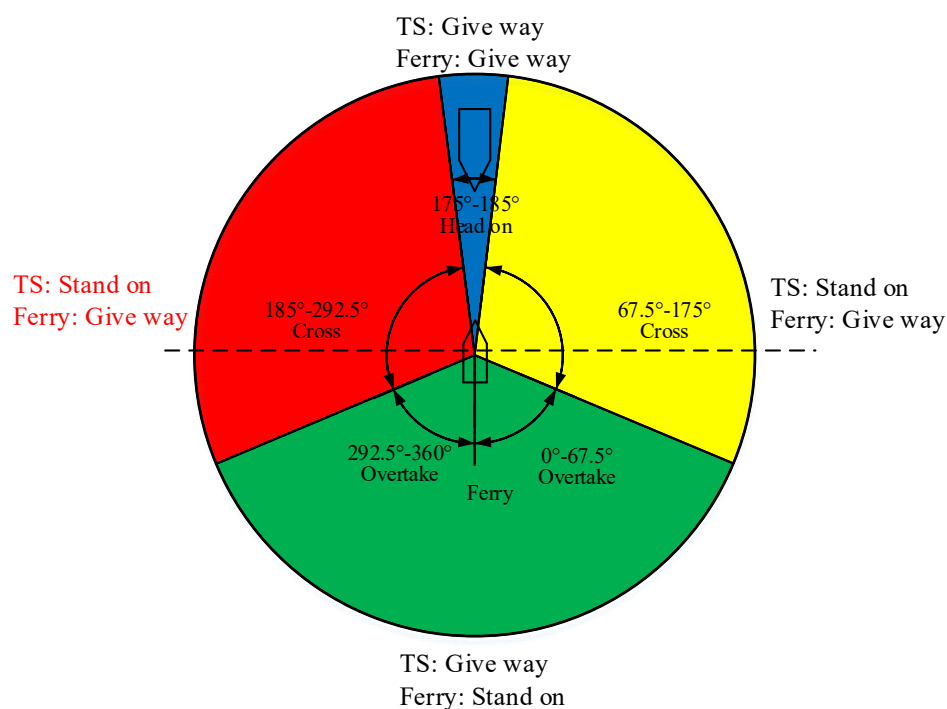
Figure 2. Responsibility of four encounter scenarios according to COLREGs and inland rules.

As shown in Figure 2, ferries should take actions to avoid collision in cross situations. To follow the inland rules, there are two problems that ferries have to deal with. Firstly, inland waterways are narrow and congested, and multiple ships are involved in encounter situations. A ferry needs to decide on travel from the bow or stern of target ships. Travelling from the stern of one target ship will result in travelling from the bow of others. Rules for a ferry crossing the channel are ambiguous when target ships travel from the left side of ferries. The crossing risk, such as relative crossing distance, is not specific, which will result in confusion for navigators. Secondly, travelling from the bow of target ships will be safer and more economical than from the stern of target ships in some encounter situations. Crossing patterns should be further quantified. Navigation experience is a good reference [29], and crossing patterns can be inferred from navigation experience by the FCPD (Ferry Crossing Patterns Determination) model based on machine learnings algorithms. Moreover, Xgboost shows better performance than other machine learning algorithms as discussed in [26]. Then, the predicted crossing patterns will be served as inputs for state space.

3.2. Collision Risk

The collision risk of a ferry can be simplified as the relative distance between a ferry and a target ship. Considering navigation rules of inland waterways and crossing patterns, a ferry's relative distance and collision risk can be divided into three levels, namely, Grade 1, Grade 2 and Grade 3, as shown in Figure 3. The collision risk of a path can be expressed as collision risk at every time slice.
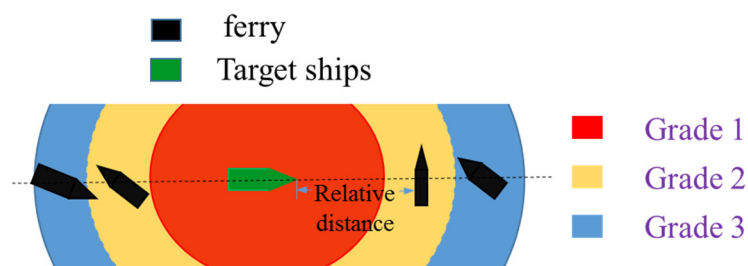
Figure 3. Crossing safety grades classification.

As described in [26], crossing grades are described as follows. The boundary of every grades is fuzzy and can be determined by historical travelling data and expert experience. Grade 1: This area is called the critical area. It is formed by a ship domain where the crossing actions are forbidden. Crossing actions that lead to violating this area should not be allowed. However, inland waterways are narrow and congested, and some crossing actions are still taken in this area in historical encounter situations. Thus, rewards will be utilized to find a balance between safety and economy which will be solved by DQN. Grade 2: The yellow area is constrained by Grade 1 and Grade 3. Most of the navigators take crossing actions in this area. When complex encounter situations occur, the ferry will choose a series of satisfactory actions to meet the economy and safety requirements based on rewards. The crossing's relative distance within this area will result in a Grade 2 collision risk of path. Grade 3: At long range, crossing patterns rarely occurred in this area. Crossing operations in this area are time consuming and are usually not preferable. Some crossing actions are still taken in this area in historical encounter situations because navigators are risk averse. Thus, collision risk of DQN based path planning can be updated considering a navigator's risk preference. We use τR^{Policy} to describe collision risk grades of ferry.

$$\tau R^{Policy} \in \{Grade1, Grade2, Grade3\} \tag{1}$$

3.3. DQN Configuration

Intelligent or autonomous path planning are comprised by manned or unmanned path planning. There are three categories, namely, fully unmanned path planning, partly unmanned path planning and fully manned path planning. Although the path planning method raised in this paper can be utilized in fully unmanned path planning, human operation should also be considered. Then, a key point should be solved when the system is authorized to command and when a human operator must take over, since the ferry crossing route is short and there are crossing situations near the start-point and endpoint of path. Hence, the autonomous path planning system is authorized to command when a ferry starts to travel. The navigators can take over the command when the crossing actions are finished and the crossing encounter situations are solved. The alternation of autonomous system and human operation can be determined by navigators. For autonomous path planning, it can be defined based on the sequential decision-making problem. The Markov Decision Process (MDP) is typically utilized to model sequential decision-making problems. In MDP, the decision-maker, who is called an agent, executes an action in the environment, and the environment, in turn, yields a new state and reward. The main idea of this model is that agents take corresponding actions based on the current state and obtain rewards and the value of the next state by interacting with the environment. The aim is to find strategies to maximize cumulative rewards by interacting with the environment. More formally, suppose agents execute an action $a_t \in A(s_t)$ following a policy π_θ with current state s_t, then the agent receives rewards from the environment R_t, and updates

itself in a new state S_{t+1}. The reward R is the feedback that quantifies the action quality of agents. The cumulative rewards at a given moment can be described as follows:

$$U_t = R_t + \gamma R_{t+1} + \gamma^2 R_{t+2} + \gamma^3 R_{t+3} + \ldots + \gamma^{n-t} R_n \tag{2}$$

γ is a discount factor. This parameter can take any value between 0 and 1. This factor is used to ensure that the future rewards are not worth. The state-action function $Q_\pi(s,a)$ following policy π_θ can be derived from U_t. The expected value of state value function is $V_{\pi(s)}$.

4. Construction of DQN Based Path Planning

Construction of DQN based path planning is to establish state space and action space for a ferry. State space refers to a series of acquired data which can be utilized for path planning. Action space refers to action related parameters serving as key elements of a path.

4.1. State Space

State space often includes position-related information of ferry and target ships. Due to the decision process of DQN, it is assumed that target ships can be identified in real time, and information on the environment is also available. Unlike other DQN-based collision avoidance algorithms, the environment should consider both endpoints and target ships. Therefore, the state space is divided into two parts, namely, relative state space between a ferry and endpoints s^{End} and relative state space between s^{Target}. s^{End} can be expressed as follows:

$$\Delta x^{End} = x^{End} - x^F$$
$$\Delta y^{End} = y^{End} - y^F \tag{3}$$
$$s^{End} = [\Delta x^{End}, \ \Delta y^{End}, \ v^F, \ c^F]$$

x^F、y^F、v^F、c^F means position, velocity and course of ferry. x^{End}、y^{End} means transferred position of the end point. Then, s^{Target} can be expressed as follows:

$$\Delta x^{TF} = x^T - x^F$$
$$\Delta y^{TF} = y^T - y^F$$
$$\Delta c^{TF} = c^T - c^F \tag{4}$$
$$\Delta v^{TF} = v^T - v^F$$
$$s^{Target} = [\Delta x^{TF}, \ \Delta y^{TF}, \ \Delta v^{TF}, \ \Delta c^{TF}]$$

x^T、y^T、v^T、c^T means position, velocity and course of target ship. In order to consider collision risk, R^{Policy} is added. Then, state space is updated as:

$$S = \begin{cases} [\Delta x^{End}, \Delta y^{End}, v^F, c^F, \Delta x^{TF}, \Delta y^{TF}, \Delta v^{TF}, \Delta c^{TF}], D^{TF} \leq R^{Policy} \\ [\Delta x^{End}, \Delta y^{End}, v^F, c^F, \tau R^{Policy}, \tau R^{Policy}, 0,0], D^{TF} > R^{Policy} \end{cases} \tag{5}$$

When encounter situations occur, the crossing patterns predicted by Xgboost are integrated into state space. As described in [26], "1" means crossing from the bow of target ships, and "0" means crossing from the stern of target ships. If the encounter situations are determined, the crossing pattern is adjusted to −1 in order to keep the structure of the state space unchanged. The state space is updated as follows:

$$S = \begin{cases} [\Delta x^{End}, \Delta y^{End}, v^F, c^F, \Delta x^{TF}, \Delta y^{TF}, \Delta v^{TF}, \Delta c^{TF}, CP], D^{TF} \leq R^{Policy}, Com = UN \\ [\Delta x^{End}, \Delta y^{End}, v^F, c^F, \tau R^{Policy}, \tau R^{Policy}, 0, 0, -1], D^{TF} > R^{Policy} \text{ or } Com = FN \end{cases} \quad (6)$$

$p = [lon^F, lat^F, v^F, c^F, lon^T, lat^T, v^T, c^T]$. CP describes crossing patterns of ferry. Com is a Boolean value, and UN means crossing actions are disrupted. FN means crossing actions are finished. D^{TF} describes the relative distance between ferry and target ships.

4.2. Action Space

Usually, there are three kinds of action spaces for ships: changing velocity while maintaining the course, changing course while maintaining velocity and changing both velocity and course. For path planning and collision avoidance, ferries prefer changing course to changing speed because frequent speed changes will cause damage to the main engine. Thus, in this study, the action space is established by course. A ferry can choose to turn to starboard, port or maintain its course during collision avoidance, so the action space is discretized into three discrete values:

$$a = [-\Delta c, 0, \Delta c] \quad (7)$$

4.3. Reward Function Design

The objective of reinforcement learning is to maximize the reward. A ferry needs to make decisions considering economy and safety.

4.3.1. Economy Reward

The economy reward aims at finding a shorter path and shorter navigation time. If a ferry is closer to the endpoint at time slice t than at time slice t − 1, then a positive reward will be generated; otherwise, a negative reward will be obtained. When a ferry reaches the endpoint, the reward is given by 500.

$$r_{end}^t = 500 \quad (8)$$

When a ferry navigates beyond the scope, the reward is set as:

$$r_{beyond}^t = -200 \quad (9)$$

Suppose d' represents the distance between the ferry and the end point at time slice t, and d represents the distance between the ferry and the end point at time slice $t − 1$, then the reward at time slice t can be expressed as:

$$r_{distnace}^t = d - d' \quad (10)$$

Meanwhile, the shorter navigation time is achieved by using a decay factor to ensure that the reward decreases when the navigation time increases. Therefore, the economy reward can be expressed as:

$$r_{economy}^t = (d - d')\beta \quad (11)$$

In addition, when the ferry is close to the endpoint, the decay factor is applied. In order to consider navigation safety, a weighted factor *w* is applied. Then, the overall economy reward is listed as follows:

$$R^t_{economy} = \begin{cases} 500, reach\ the\ endpoint \\ -200, beyond\ the\ scope \\ (d-d')\beta w, d-d' > 0 \\ (d-d')w, d-d' < 0 \end{cases}, t \in T \qquad (12)$$

The reward meets the economy requirement and can successfully guide the ferry to reach the endpoint in a shorter time when there are no encountered ships.

4.3.2. Safety Reward

Suppose d^t_s represents the distance between the ferry and target ship at time slice *t*, and d^{t-1}_s represents the distance between the ferry and target ship at time slice $t-1$. If $R^{Policy} \leq d^t_s$, then the reward at time slice *t* can be expressed as:

$$r^t_s = (d^t_s - d^{t-1}_s)(1-w), \qquad R^{Policy} \leq d^t_s \qquad (13)$$

The violation of the ship domain is forbidden, so when a ferry navigates near τR^{Policy}, the economy weight is set to be 0, and the negative reward is increased. If $R^{Policy} \geq d^t_s$, the reward is described as follows:

$$r^t_s = \begin{cases} (d^t_s - d^{t-1}_s)\rho, d^t_s - d^{t-1}_s < 0 \\ (d^t_s - d^{t-1}_s)\sigma, d^t_s - d^{t-1}_s \geq 0 \end{cases} \qquad (14)$$

When $d^t_s - d^{t-1}_s < 0$, a ferry is crossing towards target ships; this situation is dangerous, so a weighted factor ρ is added to increase negative rewards. When $d^t_s - d^{t-1}_s > 0$, a ferry tries to keep away from target ships, the relative distance is unsatisfactory, so a weighted factor σ is proposed. If the relative distance is smaller than the ship domain, then the reward is −500. Thus, the safety reward is listed as follows:

$$R^t_s = \begin{cases} (d^t_s - d^{t-1}_s)(1-w), R^{Policy} \leq d^t_s, R^{domain} \leq d^t_s \\ -500, R^{domain} > d^t_s, R^{Policy} > d^t_s \\ \begin{cases} (d^t_s - d^{t-1}_s)\rho, d^t_s - d^{t-1}_s < 0 \\ (d^t_s - d^{t-1}_s)\sigma, d^t_s - d^{t-1}_s \geq 0 \end{cases}, R^{Policy} > d^t_s, R^{domain} \leq d^t_s \end{cases} \qquad (15)$$

5. Simulation Results and Experimental Comparison

In order to validate the DQN based path planning model, simulation with undefined crossing patterns and actual case studies with defined crossing patterns is raised. The ship domain is set to be 1.5 L, and the relative distance has been removed by ship domain. The R^{Policy} is set to be 3 L. What is more, σ and ρ are set to be 1 and 2; w is variable to show comparison results. Crossing patterns are derived from historical data, so crossing patterns of simulation cannot be obtained; otherwise, crossing patterns of actual case studies can be predicted by FCPD model. Firstly, a simulation with undefined crossing patterns with w changes from 0.1 to 0.9 is raised to find a satisfactory value of w. Then, two actual case studies with defined crossing patterns are proposed to validate the model compared with an actual path.

5.1. Simulation with Undefined Crossing Patterns

A simulation of an encounter situation is formulated to validate the proposed model with undefined crossing patterns. A ferry encounters three target ships during the voyage. Then, the static and dynamic parameters of the target ships as well as the ferry are present in Table 1.

Table 1. Simulated parameters.

Type	Length (m)	Width (m)	Start Point (m, m)	End Point (m, m)	Course (°)	Velocity (m/s)
Ferry	79	14	(100, 700)	(900, 1200)	90	3
TS1	79	14	(400, 1000)	-	210	3
TS2	79	14	(500, 100)	-	0	5
TS3	79	14	(1000, 300)	-	50	2

As mentioned before, w can be utilized to show the relative importance of safety and economy. In this case study, an optimal value can be obtained by offering a series of values from 0.1 to 0.9 at 0.1 intervals. The time slice is set to be 5s. The ship domain is twice the ship length. Each environment is trained 2000 times. The reward is normalized as shown in Figure 4.
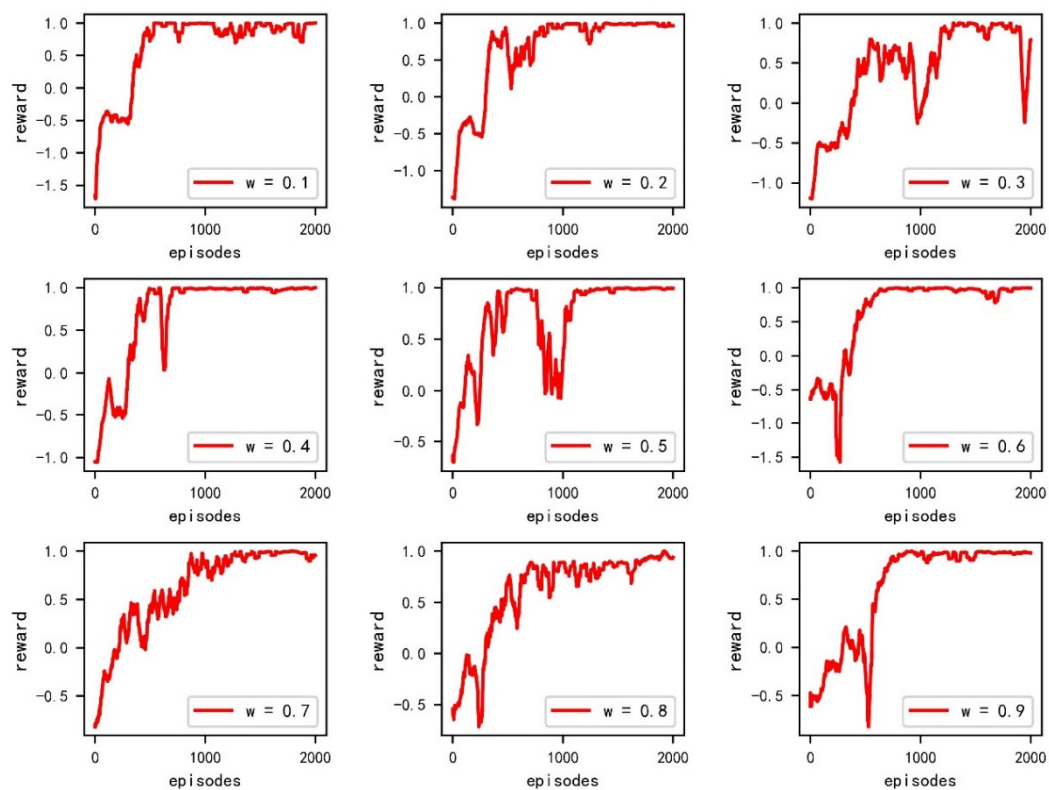
Figure 4. Normalization reward with different values of w.

From the convergence curves shown in the figure, all curves have a fast convergence speed and good stability, except when w = 0.3 and is having an oscillation. The length of the total voyage is shown in Figure 5.
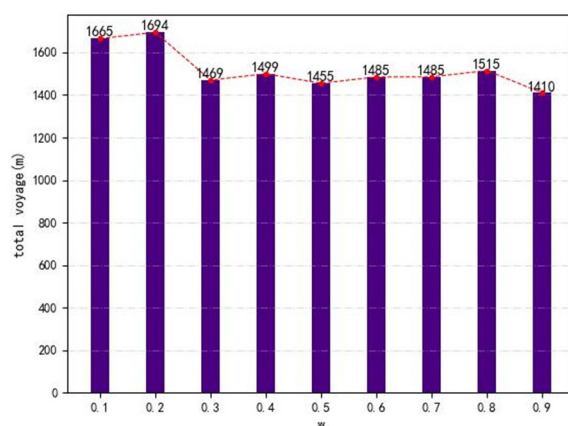
Figure 5. Total voyage with different w.

Overall, w has a negative correlation with total voyage. The total voyage will be longer with w increases, as shown in Figure 5. However, there is a small impact on total voyage when w ranges from 0.3 to 0.9. The reason is that w exists within encounter situations. When encounter situations are clear, w is useless. Then, closest relative distances are calculated to show the difference in Figure 6.
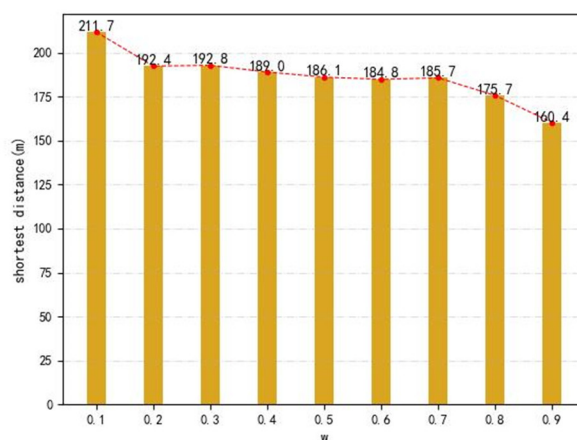

Figure 6. Closest relative distances with different values of w.

It can be seen that the closest relative distances between the ferry and the target ships decrease with the increase in w, but all the closest relative distances are larger than the ship domain (158 m). Therefore, it can be inferred that the navigation safety is decreasing with the increase in w. The ship domains are not violated. Then, the trajectories, their DCPA, TCPA and relative distance are visualized when w is set to be 0.1 and 0.9 in Figures 7–10.
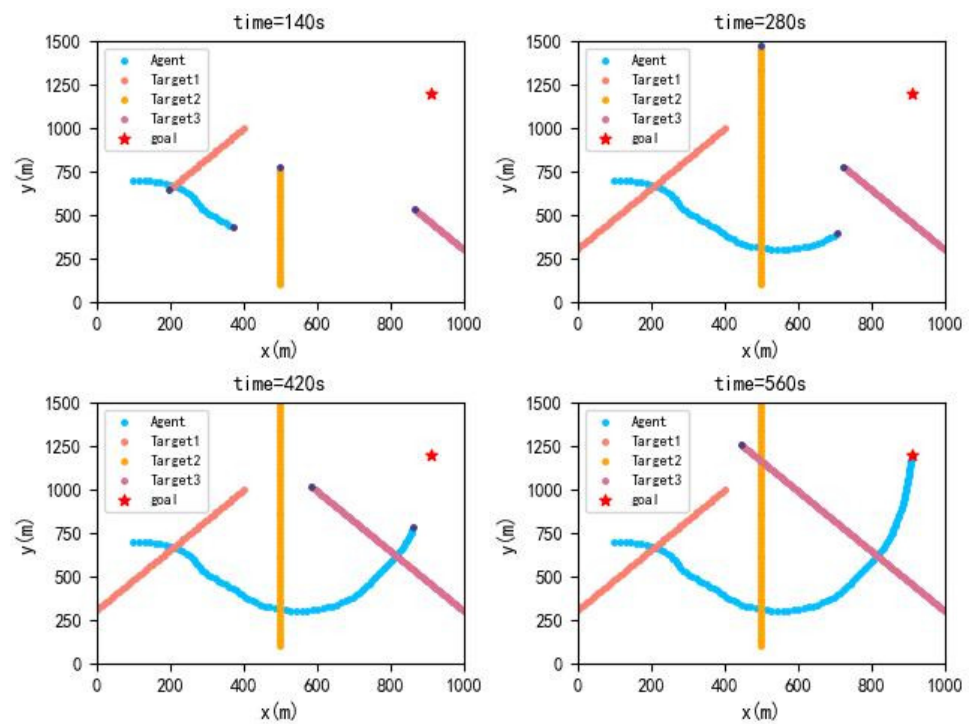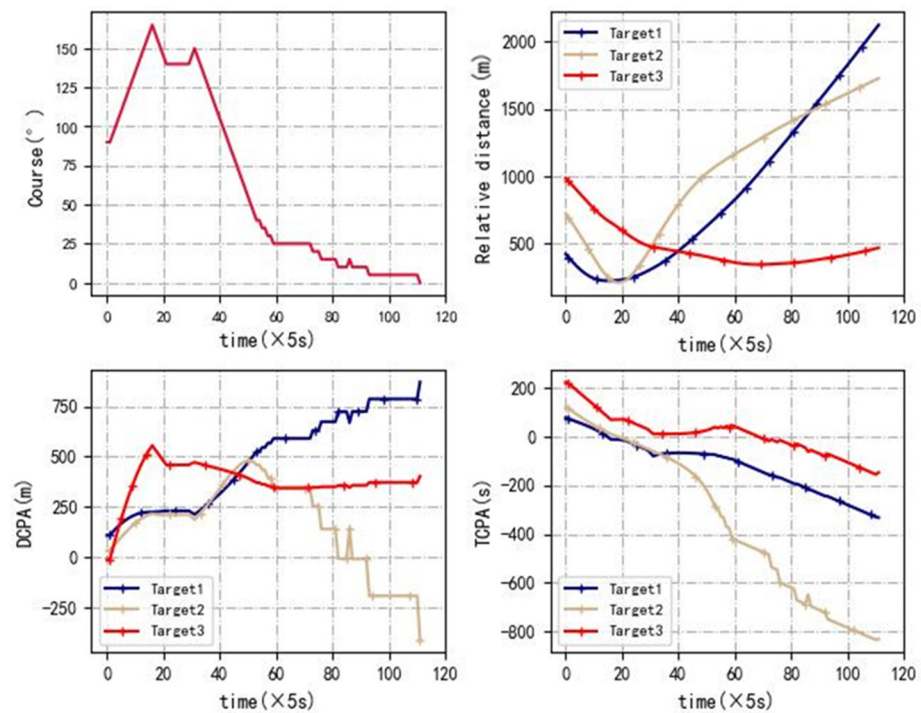
Figure 7. Trajectories when w = 0.1.

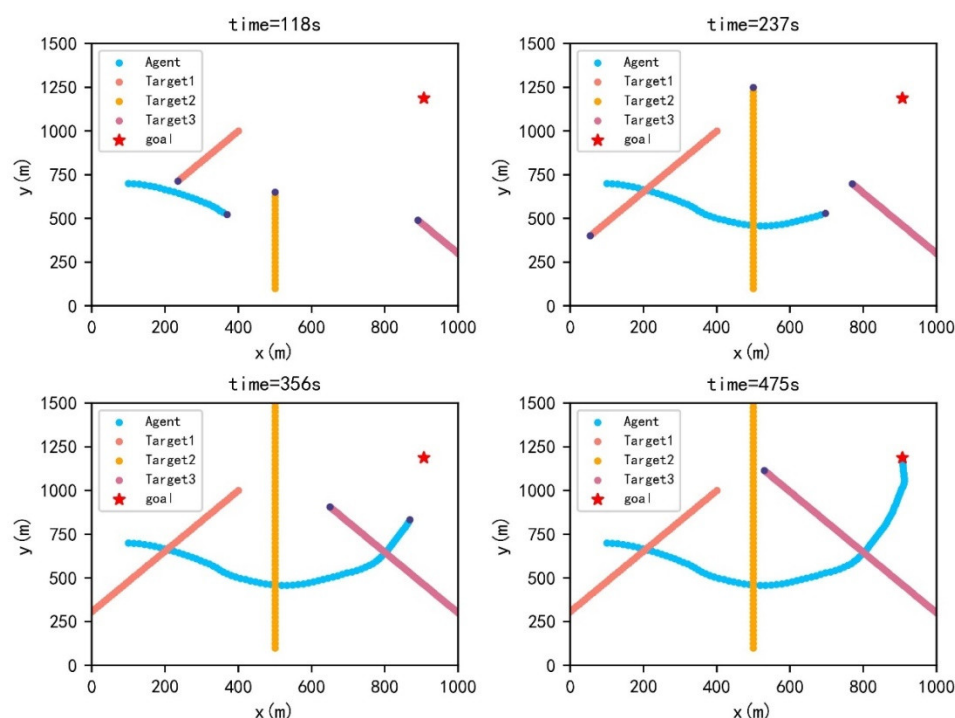**Figure 8.** Course of ferry, relative distance, DCPA and TCPA when w = 0.1.

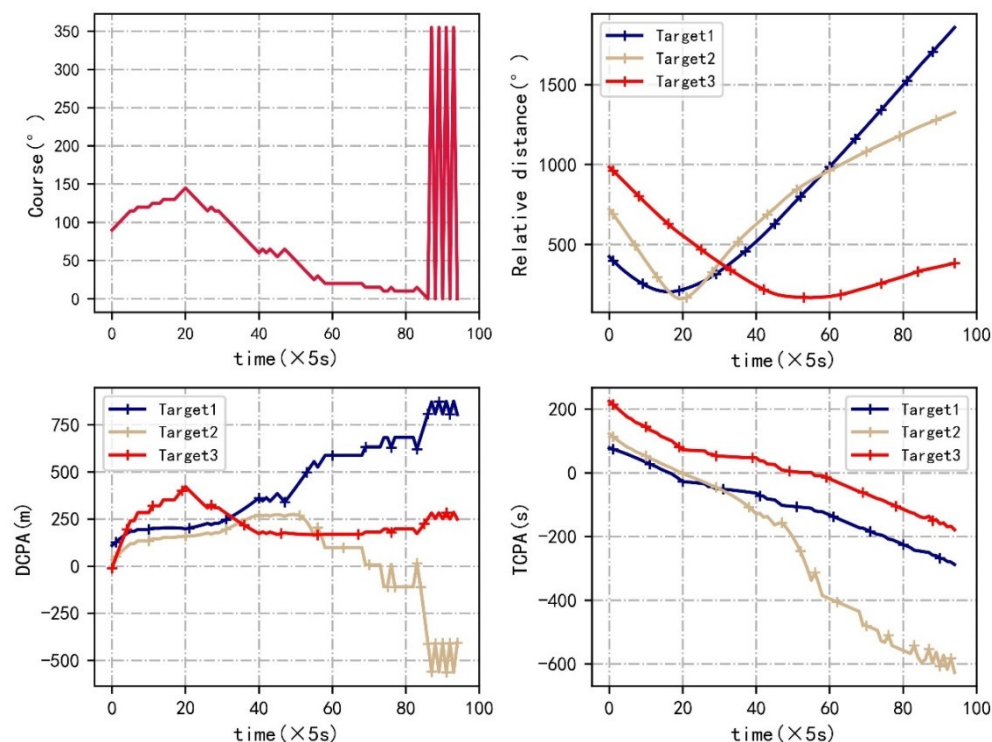**Figure 9.** Trajectories when $w = 0.9$.

Figure 10. Course of ferry, relative distance, DCPA and TCPA when $w = 0.9$.

The course of a ferry is changeable. The reason for ferry course oscillation is that the clockwise course is positive. Therefore, a relative course of 0° and 350° is 10°. Compared with $w = 0.1$, the trajectories, DCPA, TCPA and relative distance when $w = 0.9$ have some differences. The maximum course change reaches 75° in the first 100 s, while the maximum course change is 55° when $w = 0.1$. For crossing actions, a ferry prefers to avoid collision with target ships at early times when $w = 0.1$. However, a ferry tends to reach the endpoint when $w = 0.9$. The relative distance is larger when $w = 0.1$. As shown in

DCPA and TCPA, there is no collision risk for both. Moreover, the navigation time is 475 s when w = 0.9, while the navigation time is 560 s when w = 0.1.

5.2. Case Studies with Defined Crossing Patterns

Two encounter situations are raised to validate the DQN based path planning when w is set to be 0.7. The comparison results are conducted between the actual and optimized agent path. Animations can be found in Appendix A.

5.2.1. Case Study 1

A historical ferry voyage is selected in which three ships encountered a ferry in the Jiangsu Section of the Yangtze River. The latitude and longitude of the destination are (31.9499° N, 118.6090° E), and the start point is (31.94659° N, 118.60532° E). Then, the dynamic parameters of the target ships as well as the ferry are present in Table 2.

Table 2. Dynamic and static parameters.

Type	Length (m)	Width (m)	Position (E, N)	Course (°)	Velocity (m/s)
Ferry	79	14	(118.60532, 31.94659)	138.4	0.514
TS1	59	11	(118.60925, 31.948575)	226.8	2.161
TS2	67	12	(118.611207, 31.949801)	224.0	2.109
TS3	106	18	(118.605601, 31.94153)	62.4	3.647

A different weight can be selected to show the difference. The w is set to be 0.7. The training results are shown in Figure 11.
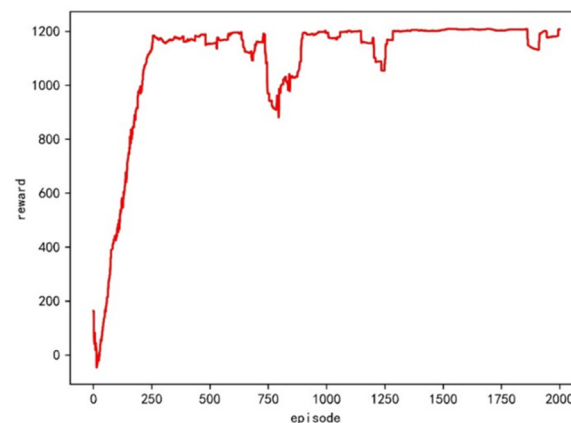

Figure 11. Reward when w = 0.7.

The training starts to converge around 250 times, some fluctuations appear from 750 to 1000 times, and then training is stable. The actual and predicted crossing patterns are listed in Table 3.

Table 3. Crossing patterns comparison.

Type	TS1	TS2	TS3
Ferry	1	1	0
Agent	1	1	0

As described in the table, the actual and predicted crossing patterns are the same. A ferry and a trained agent travel from the bow of TS1 and TS2 and stern of TS3. The actual and trained trajectories are shown in Figure 12.
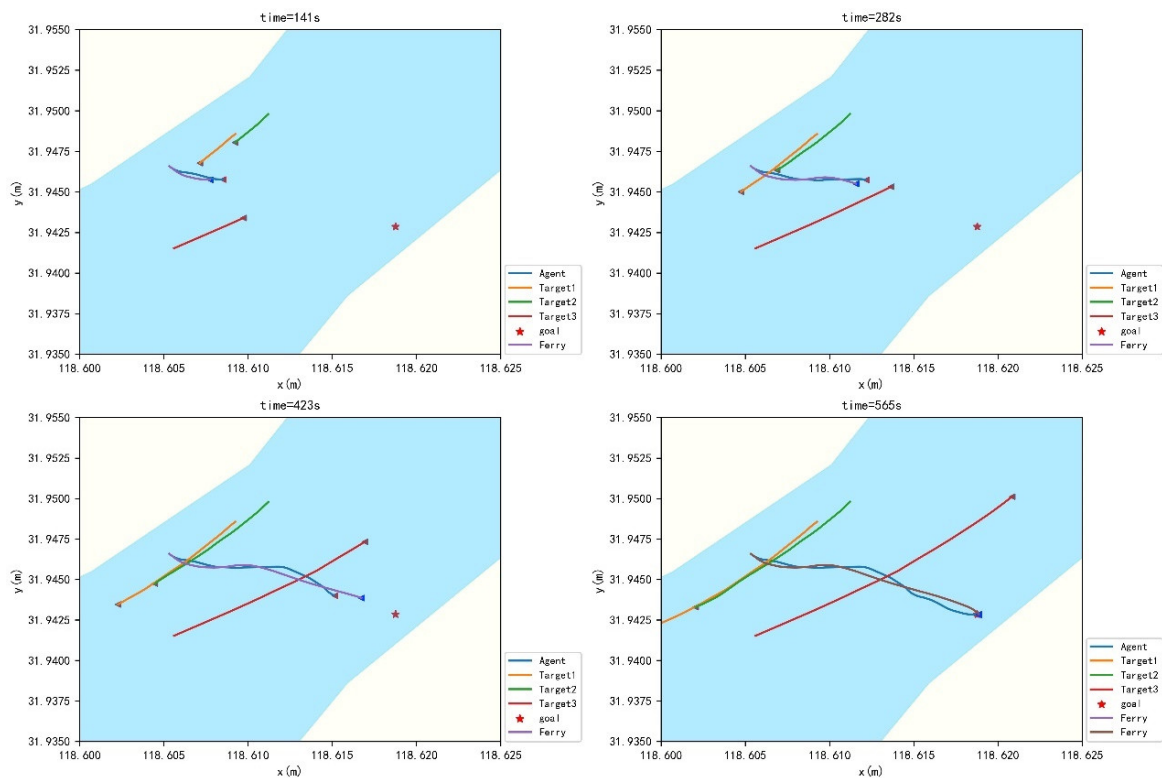
Figure 12. Trained and actual trajectories.

The introduction of the XgBoost algorithm to predict the crossing patterns, combined with the adjusted reward function for path planning, is feasible. The trained trajectories match the real trajectories satisfactorily. Then, DCPA, TCPA and relative distance are shown in Figure 13.
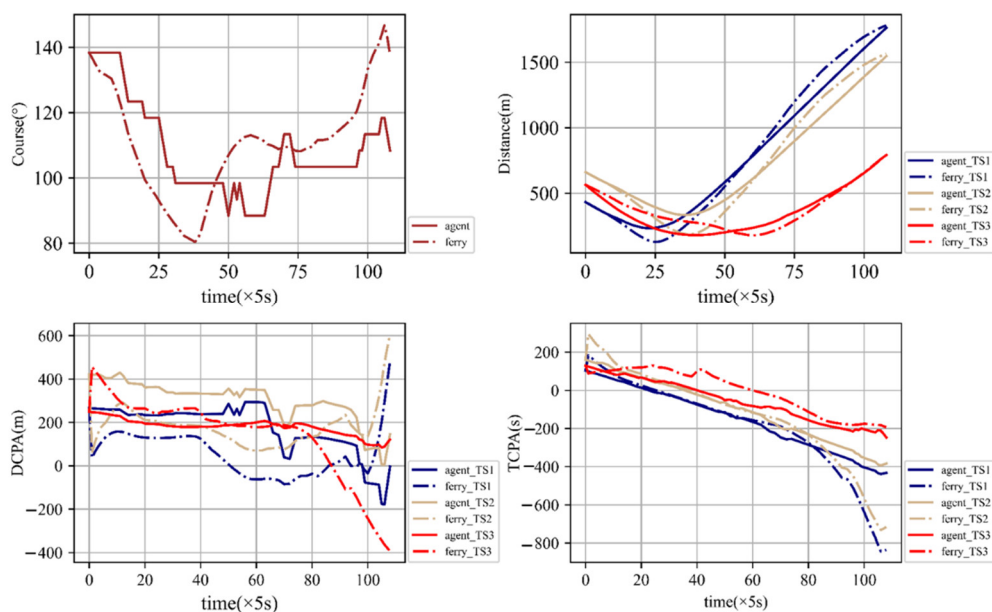

Figure 13. Course, relative distance, DCPA and TCPA.

As shown in Figure 13, the course of the trained agent has more fluctuations than the ferry's real trajectory. The relative distance between agent and target ships is longer than the relative distance between ferry and target ships, which means trained trajectories are

safer than actual ones. For DCPA and TCPA, agent and ferry almost share the same expected TS3. It can be inferred that the trained trajectories can avoid collisions with target ships and perform better in safety.

5.2.2. Case Study 2

A historical ferry voyage is selected in which four ships are encountered with a ferry in the Jiangsu Section of the Yangtze River. The latitude and longitude of the destination are (31.9499° N, 118.6090° E), and the start point is (31.9429° N, 118.619067° E). Then, the dynamic parameters of the target ships as well as the ferry are present in Table 4.

Table 4. Dynamic and static parameters.

Type	Length (m)	Width (m)	Position (E, N)	Course (°)	Velocity (m/s)
Ferry	79	14	(118.619067, 31.9429)	322.2	0.412
TS1	87	15	(118.622383, 31.955626)	231.2	3.141
TS2	55	10	(118.609067, 31.942509)	55.8	2.932
TS3	49	9	(118.604057, 31.939152)	58.2	2.564
TS4	67	13	(118.609341, 31.942036)	55.5	2.264

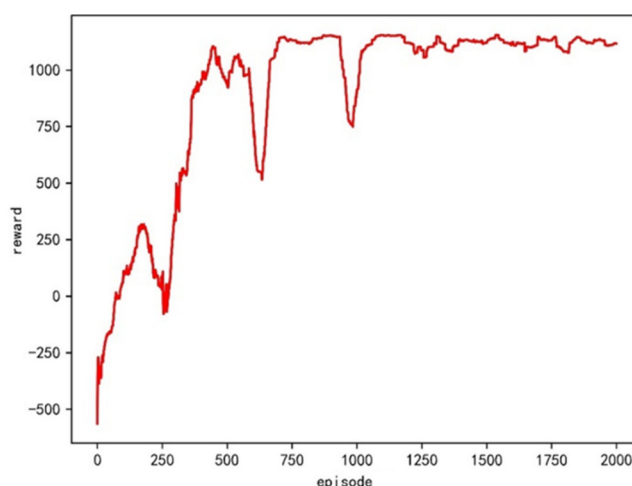The w is set to be 0.7. The training results are shown in Figure 14.

Figure 14. Reward when w = 0.7.

The training starts to converge around 1000 times, and then training is stable. The actual and predicted crossing patterns are listed in Table 5.

Table 5. Crossing patterns comparison.

Type	TS1	TS2	TS3	TS4
Ferry	1	0	1	0
Agent	0	0	0	0

As described in the table, the actual and predicted crossing patterns are different. A ferry travels from the bow of TS1 and TS3 and from the stern of TS2 and TS4. The trained agent travels from the bow of TS1, TS2, TS3 and TS4. The actual and trained trajectories are shown in Figure 15.
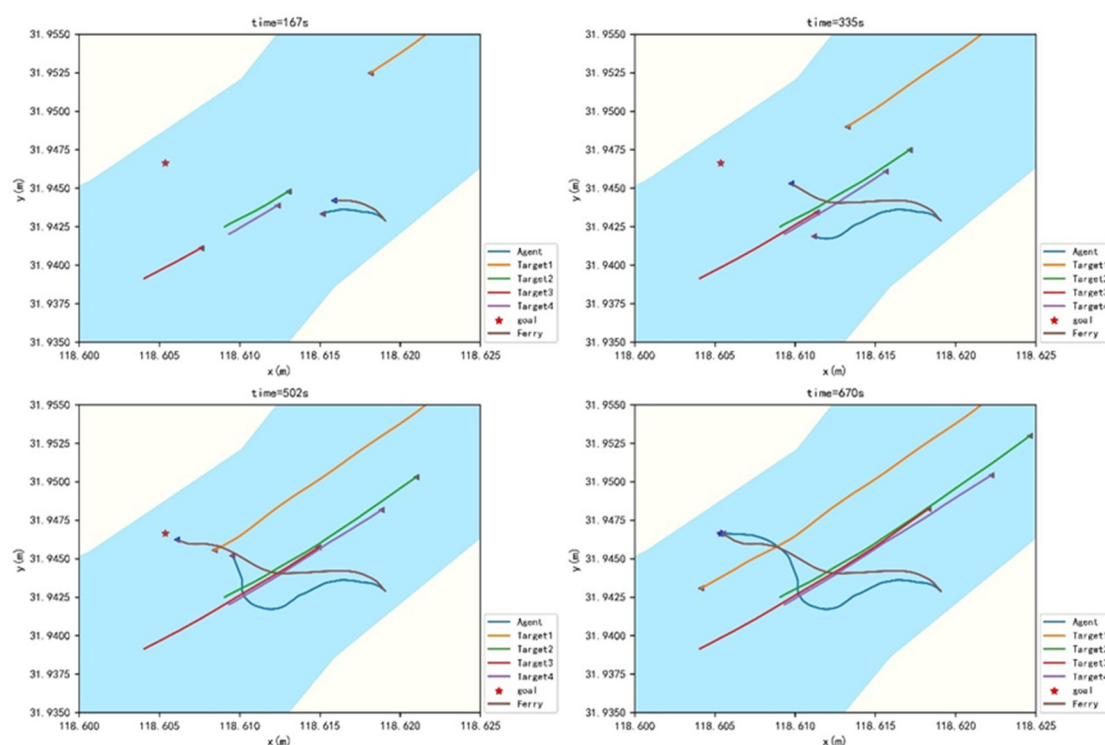
Figure 15. Trained and actual trajectories.

As shown in Figure 15, the agent spends more of the voyage in avoiding collision with target ships. The crossing patterns of the agent are safer than a ferry. However, the economy of a ferry is better than an agent. The DCPA, TCPA and relative distance are calculated as shown in Figure 16.
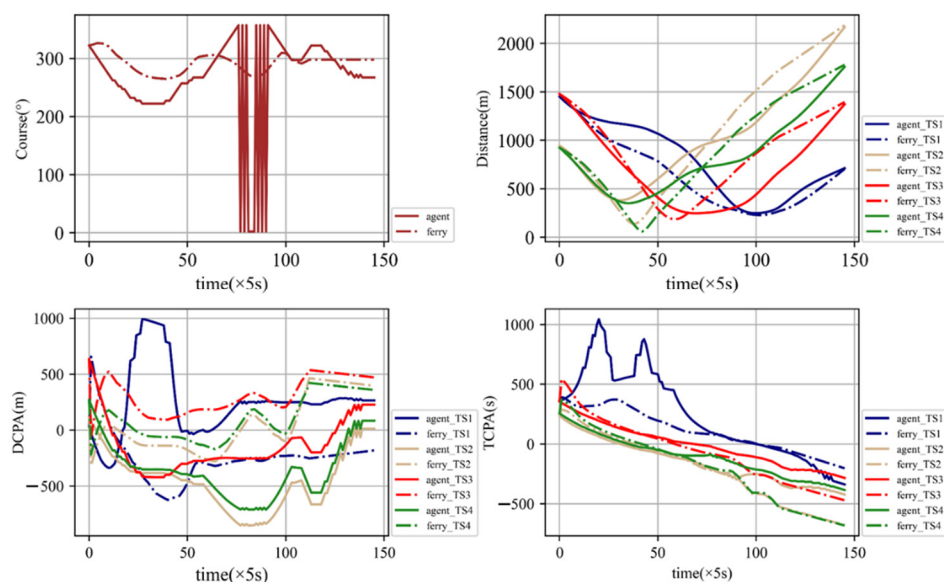

Figure 16. Course, relative distance, DCPA and TCPA.

As shown in the figures, the ferry avoids TS2 and TS4 from 100 s to 250 s, while the agent avoids TS2, TS3 and TS4 from 100 s to 400 s. Since the agent avoids more ships, the collision avoidance operation time is longer than for the ferry. The minimum relative distance between the agent and TS2 is 404 m and between the agent and TS4 it is 401 m,

respectively. The minimum relative distance between a ferry and TS2 is 116 m and between a ferry and TS4 is 71 m, respectively. For ferries, the TCPA of TS2 and TS4 is close to 0 at 200 s. At this moment, the DCPA of TS2 and TS4 is −139 m and −62 m. For an agent, the DCPA of TS2 and TS4 is −350 m and −339 m. For TS2 and TS4, the trajectory of the agent is safer than the trajectory of the ferry. For TS3, the minimum relative distance of the ferry is 247 m, and the minimum relative distance of the agent is 258 m. The trajectory of the agent is safer than the trajectory of the ferry. As for TS1, the crossing pattern is 0 for the agent, but the crossing pattern is 1 for the ferry. Although the minimum relative distance between the agent and TS1 is shorter than between the ferry and TS1, the crossing pattern of the agent is safer than the ferry. Thus, the overall trajectory of the agent is safer than the ferry, but the economy of the ferry performs better than the agent.

6. Conclusions

This study proposes an intelligent path planning algorithm for ferries crossing the busy waterways. Rules for inland waterways indicate that when a ferry encounters a target ship, it should take actions to avoid the target ship and try to pass through the stern of the target ship. In order to incorporate these rules into the model, this study uses historical data for crossing behavior prediction and inputs them into the state space. Moreover, different thresholds are selected for the reward function to balance the economy and safety. Then, the model is utilized in case studies. The results of Case 1 show that the autonomous navigation trajectory based on the model is, in general, similar with the actual trajectory, indicating that the model can be applied to the autonomous navigation of the ferries. The results of Case 2 show that the trajectory of the ferry based on the model is better than the actual trajectory in terms of safety. This study tries to achieve a satisfactory path while considering safety and economy. If navigators prefer to economize, then the path will be more risky; otherwise, a longer path will be generated. The autonomous path planning system is constructed by training for safety rewards and economy rewards. Safety and economy cannot be achieved simultaneously. Case studies show that the generated paths are safer than the historical path, but are longer than the historical path. In practice, the safety and economy of a path planning system are determined by navigator preference. In future research, this study will be further optimized in speed for better application in practice.

Author Contributions: Conceptualization, J.Z.; methodology, X.Y. and C.Y.; software, X.Y. and J.Z.; validation, X.Y. and J.Z.; formal analysis, J.Z.; investigation, G.L. and W.T.; resources, J.Z.; data curation, X.Y. and C.Y.; writing—original draft preparation, X.Y. and C.Y.; writing—review and editing, J.Z.; visualization, X.Y. and C.Y.; supervision, J.Z.; project administration, J.Z.; funding acquisition, J.Z. All authors have read and agreed to the published version of the manuscript.

Data Availability Statement: Not applicable.

Appendix A. Result Visualization

The visualization of case studies is available at https://www.researchgate.net/publication/367250875_autonomous_path_planning_navigation accessed on 23 January 2023.

References

1. Zaccone, R.; Martelli, M. A collision avoidance algorithm for ship guidance applications. *Journal of Marine Engineering & Technology* **2019**, *19*, 62-75, doi:10.1080/20464177.2019.1685836.
2. Yuan, X.; Zhang, D.; Zhang, J.; Zhang, M.; Guedes Soares, C. A novel real-time collision risk awareness method based on velocity obstacle considering uncertainties in ship dynamics. *Ocean Engineering* **2021**, *220*, 108436, doi:10.1016/j.oceaneng.2020.108436.
3. Baziyad, M.; Saad, M.; Fareh, R.; Rabie, T.; Kamel, I. Addressing Real-Time Demands for Robotic Path Planning Systems: A Routing Protocol Approach. *IEEE Access* **2021**, *9*, 38132-38143, doi:10.1109/access.2021.3058121.
4. Luman, Z.; Myung-Il, R. COLREGs-compliant multiship collision avoidance based on deep reinforcement learning. *Ocean Engineering* **2019**, *191*, 106436, doi:10.1016/j.oceaneng.2019.106436.
5. Guo, S.; Zhang, X.; Zheng, Y.; Du, A.Y. An Autonomous Path Planning Model for Unmanned Ships Based on Deep Reinforcement Learning. *Sensors* **2020**, *20*, 426-460, doi:10.3390/s20020426.
6. Duchoň, F.; Babinec, A.; Kajan, M.; Beňo, P.; Florek, M.; Fico, T.; Jurišica, L. Path Planning with Modified a Star Algorithm for a Mobile Robot. *Procedia Engineering* **2014**, *96*, 59-69, doi:10.1016/j.proeng.2014.12.098.
7. Plaza, P.M.; Hussein, A.; Martin, D.; Escalera, A. Global and local path planning study in a ROS-based research platform for autonomous vehicles. *Advanced Transportation* **2018**, *5*, 1-11, doi:10.1155/2018/6392697.
8. Fu, B.; Chen, L.; Zhou, Y.; Zheng, D.; Wei, Z.; Dai, J.; Pan, H. An improved A* algorithm for the industrial robot path planning with high success rate and short length. *Robotics and Autonomous Systems* **2018**, *106*, 26-37, doi:10.1016/j.robot.2018.04.007.
9. Liu, L.; Yao, J.; He, D.; Chen, J.; Huang, J.; Xu, H.; Wang, B.; Guo, J. Global Dynamic Path Planning Fusion Algorithm Combining Jump-A* Algorithm and Dynamic Window Approach. *IEEE Access* **2021**, *9*, 19632-19638, doi:10.1109/access.2021.3052865.
10. Li, J.; Wang, H.; Guan, Z.; Pan, C. Distributed Multi-Objective Algorithm for Preventing Multi-Ship Collisions at Sea. *Journal of Navigation* **2020**, *73*, 971-990, doi:10.1017/s0373463320000053.
11. Zhang, J.; Zhang, D.; Yan, X.; Haugen, S.; Guedes Soares, C. A distributed anti-collision decision support formulation in multi-ship encounter situations under COLREGs. *Ocean Engineering* **2015**, *105*, 336-348, doi:10.1016/j.oceaneng.2015.06.054.
12. D'Amato, E.; Mattei, M.; Notaro, I. Distributed Reactive Model Predictive Control for Collision Avoidance of Unmanned Aerial Vehicles in Civil Airspace. *Journal of Intelligent & Robotic Systems* **2019**, *97*, 185-203, doi:10.1007/s10846-019-01047-5.
13. Liu, J.; Zhang, J.; Yan, X.; Soares, C.G. Multi-ship collision avoidance decision-making and coordination mechanism in Mixed Navigation Scenarios. *Ocean Engineering* **2022**, *257*, 111666, doi:10.1016/j.oceaneng.2022.111666.
14. Li, L.; Wu, D.; Huang, Y.; Yuan, Z.-M. A path planning strategy unified with a COLREGS collision avoidance function based on deep reinforcement learning and artificial potential field. *Applied Ocean Research* **2021**, *113*, 102759, doi:10.1016/j.apor.2021.102759.
15. Song, A.L.; Su, B.Y.; Dong, C.Z.; Shen, D.W.; Xiang, E.Z.; Mao, F.P. A two-level dynamic obstacle avoidance algorithm for unmanned surface vehicles. *Ocean Engineering* **2018**, *170*, 351-360, doi:10.1016/j.oceaneng.2018.10.008.
16. Mnih, V.; Kavukcuoglu, K.; Silver, D.; Rusu, A.A.; Veness, J.; Bellemare, M.G.; Graves, A.; Riedmiller, M.; Fidjeland, A.K.; Ostrovski, G.; et al. Human-level control through deep reinforcement learning. *Nature* **2015**, *518*, 529-533, doi:10.1038/nature14236.
17. Lv, L.; Zhang, S.; Ding, D.; Wang, Y. Path Planning via an Improved DQN-Based Learning Policy. *IEEE Access* **2019**, *7*, 67319-67330, doi:10.1109/access.2019.2918703.
18. Yuan, J.; Wang, H.; Zhang, H.; Lin, C.; Yu, D.; Li, C. AUV Obstacle Avoidance Planning Based on Deep Reinforcement Learning. *Journal of Marine Science and Engineering* **2021**, *9*, 1166, doi:10.3390/jmse9111166.
19. Bhopale, P.; Kazi, F.; Singh, N. Reinforcement Learning Based Obstacle Avoidance for Autonomous Underwater Vehicle. *Journal of Marine Science and Application* **2019**, *18*, 228-238, doi:10.1007/s11804-019-00089-3.
20. Shen, H.; Hashimoto, H.; Matsuda, A.; Taniguchi, Y.; Terada, D.; Guo, C. Automatic collision avoidance of multiple ships based on deep Q-learning. *Applied Ocean Research* **2019**, *86*, 268-288, doi:10.1016/j.apor.2019.02.020.
21. Wang, W.; Wu, Z.; Luo, H.; Zhang, B. Path Planning Method of Mobile Robot Using Improved Deep Reinforcement Learning. *Journal of Electrical and Computer Engineering* **2022**, *2022*, 1-7, doi:10.1155/2022/5433988.
22. Cai, M.; Zhang, J.; Zhang, D.; Yuan, X.; Soares, C.G. Collision risk analysis on ferry ships in Jiangsu Section of the Yangtze River based on AIS data. *Reliability Engineering & System Safety* **2021**, *215*, 107901, doi:10.1016/j.ress.2021.107901.
23. Zhang, M.; Zhang, D.; Fu, S.; Kujala, P.; Hirdaris, S. A predictive analytics method for maritime traffic flow complexity estimation in inland waterways. *Reliability Engineering and System Safety* **2022**, *220*, 108317, doi:10.1016/j.ress.2021.108317.
24. Wu, B.; Li, G.; Zhao, L.; Aandahl, H.-I.J.; Hildre, H.P.; Zhang, H. Navigating Patterns Analysis for Onboard Guidance Support in Crossing Collision-Avoidance Operations. *IEEE Intelligent Transportation Systems Magazine* **2022**, *14*, 62-77, doi:10.1109/mits.2021.3108473.
25. Zhang, M.; Conti, F.; Sourne, H.e.L.; Vassalos, D.; Kujala, P.; Lindroth, D.; Hirdaris, S. A method for the direct assessment of ship collision damage and flooding risk in real conditions *Ocean Engineering* **2021**, *237*, 109605, doi:10.1016/j.oceaneng.2021.109605.
26. Yuan, X.; Zhang, D.; Zhang, J.; Cai, M.; Zhang, M. Crossing behavior decision-making for inland ferry ships based on Machine Learning. In Proceedings of the 6th International Conference on Transportation Information and Safety (ICTIS) Wuhan, China, 22–24 October 2021; pp. 1509–1517.

27.   Zhang, M.; Montewka, J.; Manderbacka, T.; Kujala, P.; Hirdaris, S. A Big Data Analytics Method for the Evaluation of Ship - Ship Collision Risk reflecting Hydrometeorological Conditions. *Reliability Engineering & System Safety* **2021**, *213*, 107674, doi:10.1016/j.ress.2021.107674.

28.   Zhang, M.; Kujala, P.; Hirdaris, S. A machine learning method for the evaluation of ship grounding risk in real operational conditions. *Reliability Engineering and System Safety* **2022**, *226*, 108697, doi:10.1016/j.ress.2022.108697.

29.   Zhang, J.; Liu, J.; Hirdaris, S.; Zhang, M.; Tian, W. An interpretable knowledge-based decision support method for ship collision avoidance using AIS data. *Reliability Engineering and System Safety* **2022**, *230*, 108919, doi:10.1016/j.ress.2022.108919.