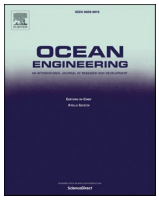
Ocean Engineering 327 (2025) 120956

Contents lists available at ScienceDirect

Ocean Engineering

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



Research paper

Intelligent ship route planning via an A* search model enhanced double-deep Q-network

Xinqiang Chen [a,b,c], Ruiyang Hu [a], Kai Luo [a], Huafeng Wu [d,*], Salvatore Antonio Biancardo [e], Yiwen Zheng [f], Jiangfeng Xian [a,**]

[a] Institute of Logistics Science and Engineering, Shanghai Maritime University, Shanghai, 201306, China
[b] Chongqing Key Laboratory of Green Logistics Intelligent Technology, Chongqing Jiaotong University, Chongqing, 400074, China
[c] Jiangxi Key Laboratory of Intelligent Robot, Nanchang, 330019, China
[d] Merchant Marine College, Shanghai Maritime University, Shanghai, 201306, China
[e] Department of Civil, Construction and Environmental Engineering (DICEA), University of Naples Federico II, Naples, Italy
[f] College of Ocean Science and Engineering, Shanghai Maritime University, Shanghai, 201306, China

ARTICLE INFO

Keywords:
Ship route planning
Deep reinforcement learning
Double deep Q-network
Smart ship

ABSTRACT

Ship route planning provides important on-site information for the purpose of enhancing intelligent ship safety. The study proposes a novel ship route planning method by considering weather conditions, carbon emissions, etc. Initially, a convolutional neural network is employed to build a model that simulates the navigation environment, capturing the maritime conditions. Then, an A* guiding DDQN (A-DDQN) route planning approach is developed by integrating the global pathfinding capability of the A* algorithm with the adaptive learning mechanism of the Double Deep Q-Network (DDQN) algorithm, which enhances navigation efficiency while minimizing randomness and unnecessary course deviations. Based on the designed reward function and the inputted ship state information, the A-DDQN algorithm determines the optimal action strategy for ship navigation. By iteratively executing the optimal action at each step, it generates a route a route from the departure area to the target area. The experimental results indicate that the improved method leads to a 11.93 % reduction in fuel consumption and a 12.16 % decline in carbon emissions, demonstrating superior performance compared to the conventional DDQN algorithm. The research findings can help the maritime community make more reasonable ship routing decisions under varied ship navigation conditions.

1. Introduction

Maritime transport is critical in driving the global economy, with around 80 % of world trade conducted via international shipping. Although shipping is recognized for its Fuel savings and superior economic value, it is facing the dual challenges of a global great recession and climatic change (Cullinane et al., 2022; Raza, 2020; Lee et al., 2023; Shang et al., 2024). The consistent rise in fuel prices has significantly elevated operational expenses, prompting shipping companies to prioritize cost management to maintain their profitability. Additionally, the significant fuel consumption and greenhouse gas emissions exacerbate environmental degradation and climate change, thereby imposing greater demands on the planning of shipping routes (Du et al., 2022a; Armstrong, 2013; Tadros et al., 2023; Chen et al., 2023a; Shang et al., 2023). Under these circumstances, ensuring safe navigation and enhancing energy efficiency has become a core task for enhancing competitiveness and environmental sustainability in the shipping field (Xu et al., 2022; Ventikos et al., 2022). With the advent of intelligent ship technologies, the importance of route planning for optimizing navigation costs and reducing carbon emissions has become increasingly prominent, necessitating effective measures to address environmental challenges and minimize fuel consumption (Zis et al., 2020; Du et al., 2022b; Charalambopoulos et al., 2023; Ma et al., 2024; Zhang et al., 2025).

In the last century, numerous scholars conducted extensive research on route planning issues (Jang et al., 2022). With the ongoing

This article is part of a special issue entitled: Intell Ship Navi published in Ocean Engineering.

* Corresponding author.
** Corresponding author.
E-mail addresses: hfwu@shmtu.edu.cn (H. Wu), jfxian@shmtu.edu.cn (J. Xian).

https://doi.org/10.1016/j.oceaneng.2025.120956
Received 1 November 2024; Received in revised form 10 March 2025; Accepted 10 March 2025
Available online 17 March 2025

advancement of intelligent ship technologies, experts have increasingly integrated smart algorithms into ship route planning, addressing the challenges of automated route generation and yielding significant research outcomes (Li et al., 2023a; Chen et al., 2023b, 2024; Shu et al., 2023). From the perspective of the shipping industry, ship route planning should prioritize navigational safety as the primary objective, and based on this foundation, strive to reduce fuel consumption and lower operational costs (Gkerekos et al., 2020; Zhang et al., 2023). Promoting a green and low-carbon transition has become an important trend in global shipping development. Effective route planning not only enhances navigational safety but also facilitates the efficient use of energy (Zaccone et al., 2018; Liu et al., 2022; Luo et al., 2024). In the issues of route planning, typical research objectives often include minimizing navigation time, reducing fuel consumption, maximizing route safety, and achieving the lowest cost. Shu et al. proposed an optimal control-based ship route planning model for the multi-objective optimization problem of ship route planning in port waterways. In the context of mixed ice-covered and ice-free waters, where there is a trade-off between icebreaking costs and navigation safety for icebreaking fleets in ice zones, an innovative model based on optimal control theory was proposed (Shu et al., 2023, 2024; Gan et al., 2025).

In route planning methods, traditional non-intelligent algorithms can accurately and effectively solve small-scale route planning problems but often struggle to achieve satisfactory results for large-scale optimization issues (Wu et al., 2024; Xing et al., 2023; Hashali et al., 2024; Li et al., 2023b; He et al., 2021). The Dijkstra algorithm employs a method that progressively expands the most efficient route, determining the least expensive route from the initial node to the destination. This process integrates breadth-first search with a greedy technique to reduce costs effectively (Novac et al., 2020; Öztürk et al., 2022). In contrast, the A* algorithm enhances the Dijkstra approach by assessing the route cost to the target node to identify the optimal route (Grifoll et al., 2022). Zheng et al. suggested an improved velocity obstacle method (VO), which integrates ship domain models and the Convention on the International Regulations for Preventing Collisions at Sea (COLREGs) and introduces multiple constraints to establish a collision avoidance model that better aligns with actual navigation requirements. Validation indicates that this method improves the safety and feasibility of ship collision avoidance across a range of emergencies, while also optimizing the precision and cost-efficiency of evasive manoeuvres (Zheng et al., 2024).The artificial potential field algorithm (APF) is utilized within the domain of maritime navigation safety. It exerts an attractive potential field on target points to facilitate the approach of the ship towards its destination while applying a repulsive potential field to obstacles to prevent collisions with them (He et al., 2023; Zhang et al., 2021). Chen et al. introduced a path-planning approach for unmanned surface vehicles (USVs) that combines an adaptive early-warning mechanism with an enhanced ant colony optimization (ACO) algorithm and an artificial potential field (APF) algorithm. The improved method strengthens the ACO algorithm's convergence abilities and boosts the APF algorithm's obstacle avoidance performance, thereby effectively addressing challenges related to local minima and unreachable targets. Simulations and field tests confirmed that this method efficiently plans optimal routes, successfully avoids both static and dynamic obstacles, and demonstrates feasibility and effectiveness in complex marine environments (Chen et al., 2021).

Traditional non-intelligent algorithms typically have the advantages of simplicity and low computational cost, but in some cases, multiple local minima may prevent the acquisition of optimal solutions. In contrast to traditional algorithms, bionic intelligent algorithms exhibit robust global search capabilities, enabling them to circumvent local optimal solutions. Bai et al. developed a plant growth-inspired algorithm to address the path-planning challenge for USVs. This method simulates the phototropism observed in plants, allowing USVs to autonomously navigate around obstacles and reach designated target points in intricate marine environments (Bai et al., 2023). Du et al. introduced an improved fractional-order particle swarm optimization algorithm (FOPSO) that updates the velocity formula with adaptive coefficients, avoiding local optimization and premature problems. Experimental results show that this algorithm produces shorter routes with faster convergence (Du et al., 2022b). These bionic intelligent algorithms often require substantial prior knowledge and extensive computations, but the high computational complexity results in excessive computation time and resource consumption also increases the difficulty of solving problems. Therefore, bionic intelligent algorithms are primarily used for auxiliary decision-making in route planning (Sahoo et al., 2023).

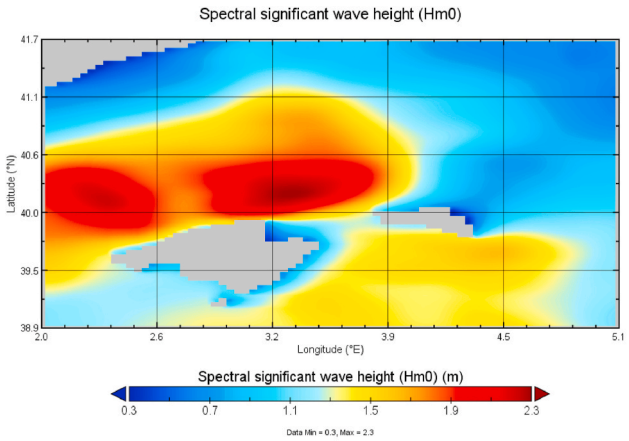
In contrast, machine learning methods exhibit adaptability and possess robust data processing capabilities (Liu et al., 2023; Wu et al., 2022). Deep Reinforcement Learning (DRL) is an algorithm that combines deep learning and reinforcement learning. It leverages deep neural networks to process high-dimensional input data in reinforcement learning, such as images, text or other complex perceptual information, enabling the agent to learn and make decisions in dynamic environments. DRL has made significant progress in various tasks that require the perception of high-dimensional inputs and the formulation of optimal or near-optimal decisions (Wang et al., 2024). Andouglas et al. addressed the challenges of path planning for wind-driven ships by proposing a Q-learning-based algorithm, which divides the planning process into global and local layers, effectively computing safe routes to ensure autonomous sailing under wind influence. However, this method fails to solve the high-dimensional state space problem, resulting in slow convergence (Silva Junior et al., 2020). Lee et al., proposed an approach utilizing the Deep Q-network algorithm, considering both navigable and restricted areas to optimize route planning. Experimental results show that this approach improves navigation efficiency and safety, reducing navigation distance by 1.77 % (Lee et al., 2024). DRL overcomes the drawbacks of traditional algorithms, such as large computational requirements and the need for prior samples, demonstrating strong learning capabilities and adaptability, but it still suffers from slow convergence and unstable route planning, with its adaptability in complex maritime environments requiring further improvement (Chu et al., 2023; Xie et al., 2021; Xiaofei et al., 2022).

To address the issues, a ship route planning approach utilizing the A-DDQN algorithm has been proposed. This method enhances the guidance provided to DDQN by integrating the global route planning features of the A* algorithm, leading to improvements in training efficiency as well as the stability and adaptability of ship route planning. The research plan is primarily divided into two parts, the first involves establishing an environmental model that accurately represents the geographical features and meteorological conditions of the navigable waters, ensuring that route planning thoroughly considers factors such as ship speed loss, geographic obstacles, and external weather impacts. The second component is founded on the Markov Decision Process (MDP) framework, wherein the ship continually modifies its navigation strategy through interactions with the environment to identify the route that yields the highest rewards. The navigation time and distance of the ship are calculated based on the obtained optimal route, and the emissions of exhaust gases are verified based on the STEAM2 model.

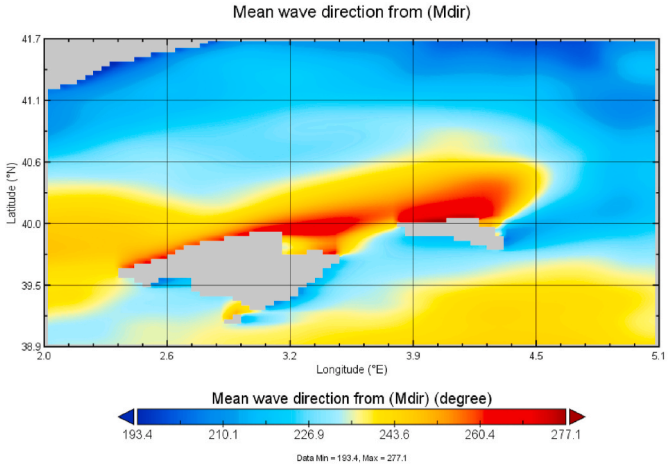
2. Mathematical model

2.1. Marine geographical information

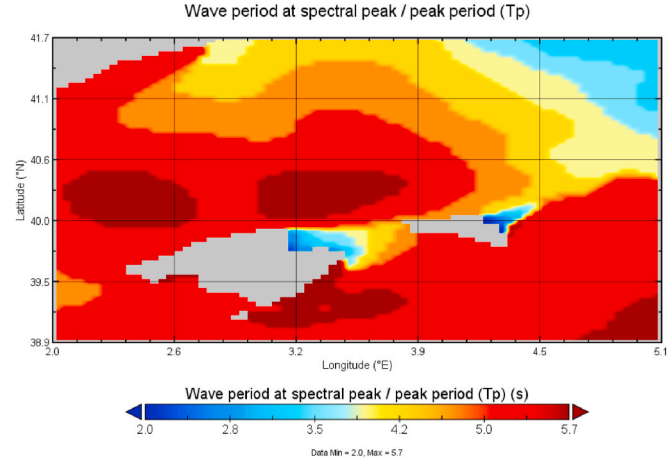
Accurate geo-environmental modelling is fundamental for constructing ship routes, as it precisely represents the static characteristics of the navigation area. In this study, the focus is on ship navigation under the influence of wave dynamics. To effectively evaluate how the marine environment influences ship navigation speeds, data is gathered from the Copernicus Marine Environment Monitoring Service (CMEMS). As shown in Fig. 1, this dataset includes relevant products that pertain to critical factors. Fig. 1(a) illustrates the Spectral Significant Wave Height, while Fig. 1(b) demonstrates mean wave direction from (Mdir). Fig. 1(c)



a) Spectral significant wave height (Hm0) (m)



b) Mean wave direction from (Mdir) (degree)



c) Wave period at spectral peak/peak period (Tp) (s)

Fig. 1. Data visualization.

shows wave period at spectral peak/peak period.

2.2. Speed loss calculation

Evaluating the effect of waves on a ship's navigation speed is essential for assessing its navigational condition. This research utilizes the speed loss calculation method developed by Mannarini et al., which employs empirical formulas to accurately measure the speed reduction caused by wave interactions, taking into account both the relative orientation of the ship to the waves and the significant wave height (Mannarini et al., 2019).The Sea surface wave significant height accurately reflects the intensity of surface waves in the marine environment by measuring the height of the waves and then selecting the average height of the highest third of the waves over a given period. Waves propagating from different directions may exert different dynamic effects on ships, and the Sea surface wave from direction represents the influence of the wave direction in the marine environment on ships or other offshore structures. The Sea surface wave period at variance spectral density maximum is a characteristic parameter of surface waves, describing the specific frequency or period of wave energy distribution. This period is generally considered the dominant wave period in a given maritime region, representing the primary period that best reflects the wave activity in that area.

The Mannarini method posits that waves increase drag on the ship, leading to a reduction in speed. The ship's velocity under wave influence is modelled as its initial speed in calm conditions minus a penalty term determined by wave parameters. The ship's velocity penalty is calculated using a function that considers both the significant wave height and the relative angle θ between the ship's heading and the wave direction. The correlation between wave direction and speed reduction is articulated in Equation (1):

$$v(H_s, \theta) = v_0 - f(\theta) \cdot H_s^2 \tag{1}$$

Where v_0 represents the ship's initial speed in the absence of wave action, H_s is the effective wave height, θ represents the angle measured in a clockwise manner from the ship's heading to the direction of the waves, and $f(\theta)$ is an empirical coefficient that varies with the relative angle.

3. Methodology

3.1. A ship route planning approach utilizing the A-DDQN algorithm

The study proposes an improved DDQN algorithm called A-DDQN that integrates the DDQN algorithm with the A* algorithm. In the construction of the navigation state observation framework, key information from the marine environment model is extracted, and four matrices are designed to represent respectively the ship's position, target point position, land location and navigation speed loss. This ensures that the neural network will ultimately perceive changes in the environment. The A-DDQN algorithm continuously updates the environmental state based on the designed observation framework, enabling intelligent ship navigation.

To avoid the inefficiencies and potential safety risks caused by random exploration in the DDQN method, we propose a strategy that uses the A* algorithm to guide the exploration process. The A* algorithm estimates the shortest route to the target at each step, helping the ship approach the target area as quickly as possible while minimizing unnecessary steering manoeuvres. As training progresses, the exploration rate gradually decreases, and the system ultimately relies entirely on the neural network for decision-making. Fig. 2 presents a depiction of the A-DDQN algorithm.

3.2. DDQN algorithm

DDQN is an advanced algorithm within the DRL that addresses the problem of Q-value overestimation by separating the processes of action selection and the computation of target Q-values. Compared to the traditional DQN algorithm, DDQN effectively reduces decision bias, leading to improved stability and accuracy. The DDQN algorithm determines the best action by leveraging both the current and target Q networks, effectively maximizing the cumulative reward derived from the reward function. The reward function is expressed in Equation (2):

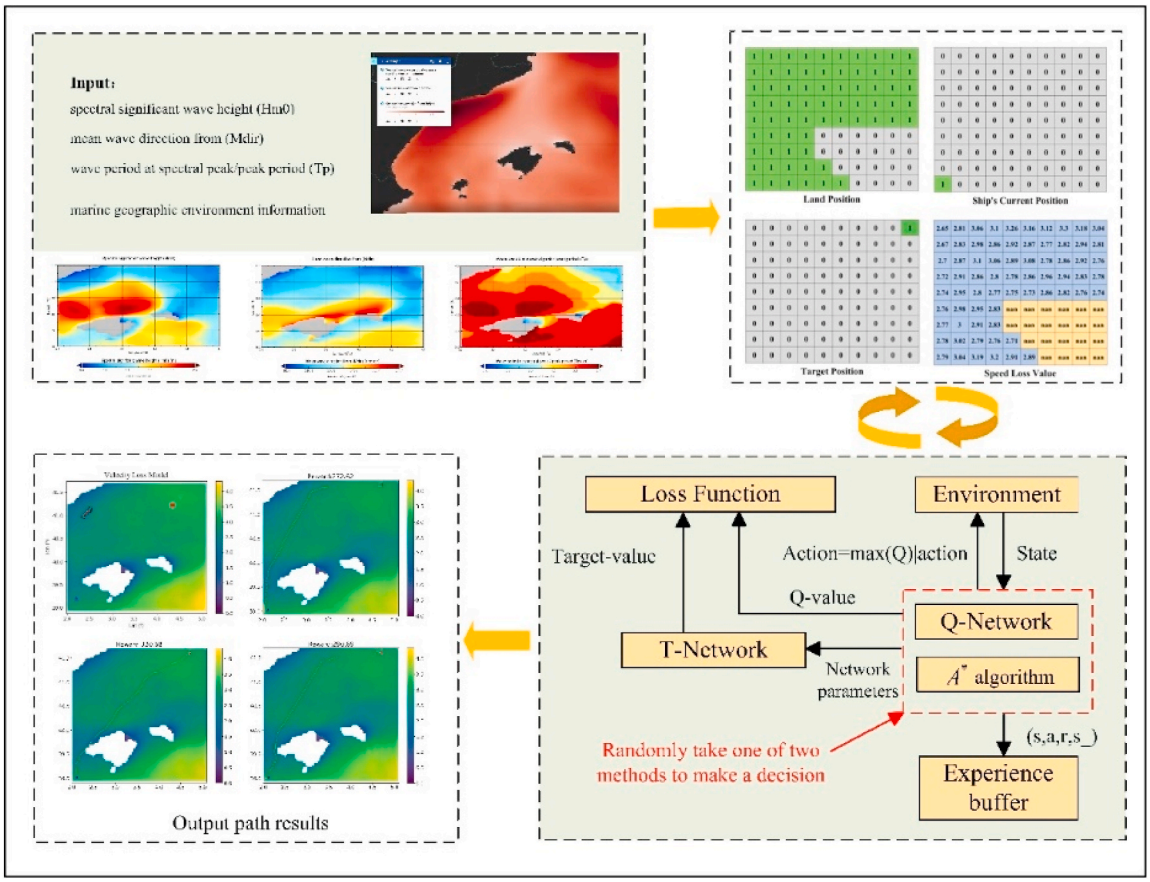


Fig. 2. Illustration of A-DDQN algorithm.

$$R = R_1 + R_2 + \cdots + R_t \tag{2}$$

Where R_t represents the reward obtained at time t.

The loss function $L(\omega)$ is employed to reduce the discrepancy between the current Q-value and the target Q-value, as illustrated in Equation (3):

$$L(\omega) = \left[(Q_c - Q(S_t, a_t; \omega))^2\right] \tag{3}$$

Where $Q(S_t, a_t; \omega)$ represents the output of the current Q network, representing the Q-value of the action a_t under the state S_t; Q_c is the target value to compute the Q-value according to the optimal action under state S_{t+1}, as demonstrated in Equation (4):

$$Q_c = R_t + \gamma Q'(S_{t+1}, \arg\max_a Q(S_{t+1}, a_t, \omega); \omega') \tag{4}$$

Where $\arg\max_{a'}(S_{t+1}, a_t, \omega)$ represents the action associated with the highest Q-value produced by the target Q-network, and $Q'(S_{t+1}, \arg\max_a Q(S_{t+1}, a_t, \omega); \omega')$ is the predicted value generated by the current Q-network.

3.2.1. Ship navigation state observation framework

Essential information is extracted from the marine environment model, representing the ship's state as a set of four matrices with the exact dimensions as the grid-based image of the modelled navigation environment. These matrices, structured as frames with defined height and width, represent the ship's position, the target point location, the land areas and the navigation speed loss values within the corresponding regions, effectively characterizing the ship's navigation state.

As shown in Fig. 3, the combination of four matrices represents the characteristics of the ship's navigation environment. Fig. 3(a) depicts the current position of the ship, where the value 1 denotes the ship's current location and 0 represents the other components of the sea area. Fig. 3(b) indicates the position of the target point, with 1 marking the target point's location and 0 signifying the remaining parts of the sea area. Fig. 3(c) shows the positions of islands and reefs, where 1 corresponds to the open sea and 0 denotes islands or reefs. Fig. 3(d) represents the speed loss values of the ship in each region, with each grid in the matrix indicating the average speed loss value of the ship navigating in the current region. Meanwhile, islands and reefs are represented by missing values, denoted as nan.

Fig. 4 illustrates that the red square indicates the target navigation zone for the ship, the blue triangle signifies the initial position, and the white areas are designated as land. In the oceanic regions, the colour gradient conveys the ship's speed loss, with a shift towards yellow indicating a more significant speed reduction at that location, while a transition to dark blue signifies a lower speed loss. This approach guarantees that each grid includes geographical and environmental features, as well as navigational data about the ship. Additionally, it reflects the degree of wave influence on ship speed, aiding in the avoidance of hazardous routes and improving navigation efficiency and safety.

3.2.2. Action space

In the ship navigation model, the action space is optimized to prevent unnecessary steering decisions caused by limited action modes, as illustrated in Fig. 5. From its current node, the ship can advance to one of 48 neighbouring nodes distributed across three concentric layers. The first layer contains 8 nodes, the second 16, and the third 24, offering 48 distinct action choices. This expanded action space allows for greater navigational flexibility, minimizes unnecessary heading adjustments, and enhances overall navigation efficiency.

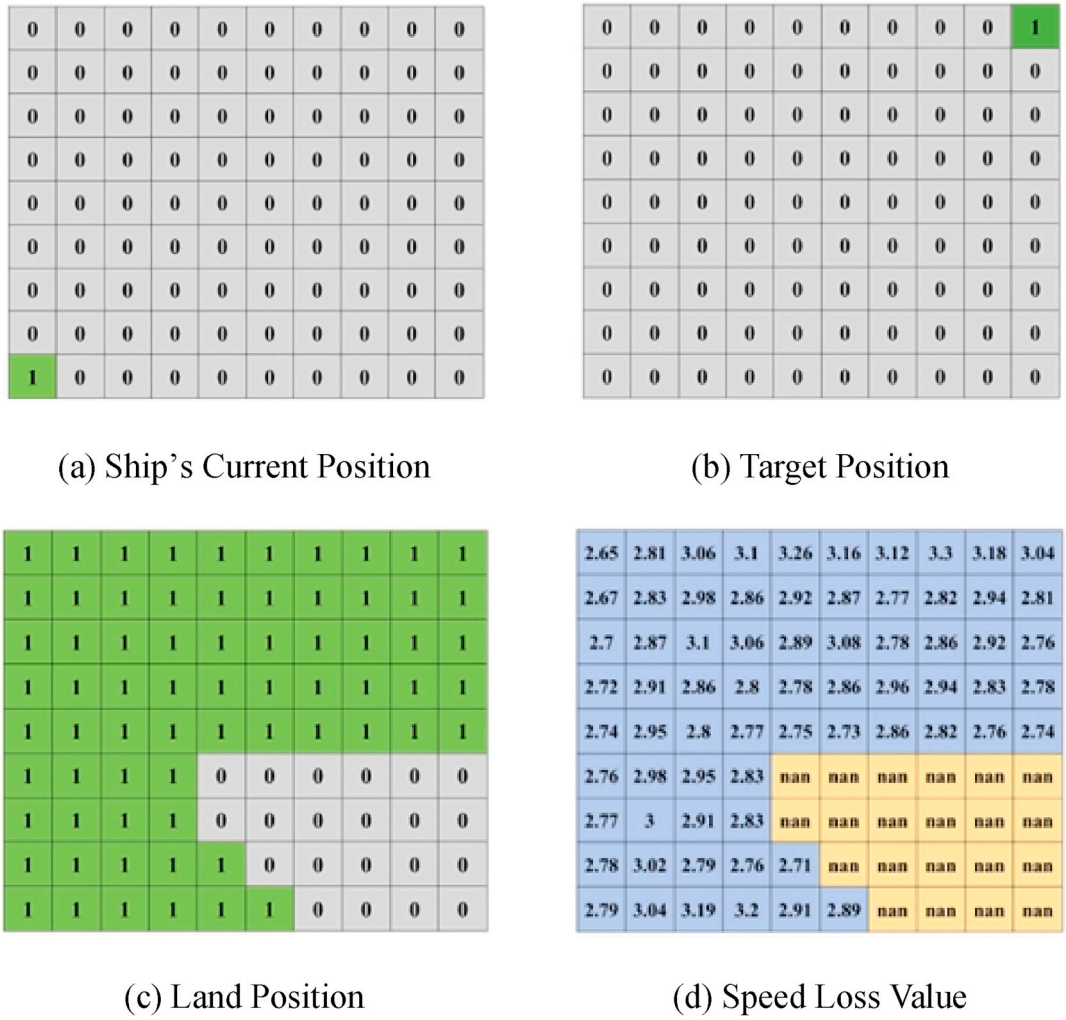


Fig. 3. The navigation state observation space is composed of four matrices.

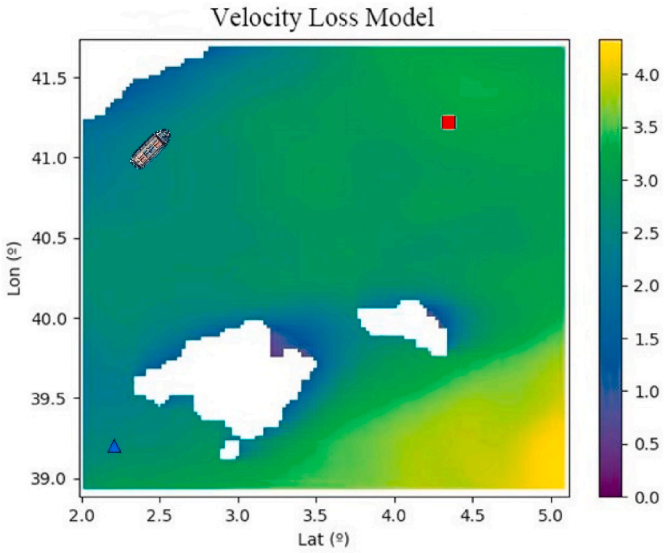


Fig. 4. Schematic diagram of the navigation state observation.

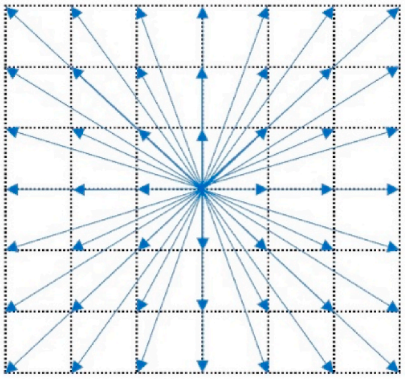


Fig. 5. Schematic diagram of action space.

3.2.3. *Reward function*

In the DDQN algorithm, the reward function plays a vital role in evaluating the outcomes of actions and guiding the ship toward making optimal choices. In this research, the ship is granted a positive reward for actions that move it closer to the target, whereas actions that lead it away from the target or cause a collision result in a negative reward. This study introduces a composite reward function considering geographical, environmental factors and the influence of ocean wind

and waves, which can be mathematically expressed in Equation (5):

$$R_t = R_1 + R_2 + R_3 + R_4 \tag{5}$$

Where R_1 represents the collision penalty at time t, R_2 represents the reward function accounting for the influence of wind and waves, R_3 represents the reward function associated with the distance separating the ship from the target area, and R_4 represents the final reward.

R_1 is a constant value, assigned a significant negative penalty when the ship collides with an island or land, reinforcing the prohibition of collisions, as expressed in Equation (6):

$$R_1 = -100 \tag{6}$$

R_2 quantifies the impact of meteorological factors on ship navigation by applying a penalty proportional to the navigation speed loss, as calculated from Equation (1). Concurrently, a significant negative reward is given when the speed loss value in the navigation area exceeds a certain threshold due to the varying safety requirements for different types of ships prompting the ship to avoid corresponding areas of high winds and waves, which is expressed in Equation (7):

$$R_2 = \begin{cases} k_1 \cdot V_L, V_L < V_{LM} \\ -150, V_L \geq V_{LM} \end{cases} \tag{7}$$

Where V_L represents the ship's current speed loss value, V_{LM} represents the speed loss threshold, and k_1 represents the impact factor that quantifies the effect of speed loss on the ship's navigation.

R_3 demonstrates how the distance between the ship's present location and the target area impacts navigation, motivating the ship to move more efficiently towards its destination, as represented in Equation (8):

$$R_3 = k_2(d_t - d_{t+1}) \tag{8}$$

Where d_t represents the impact of the ship's remaining navigation distance at time t, d_{t+1} represents the impact of the ship's remaining navigation distance at time $t+1$, and k_2 represents the distance influence factor.

R_4 represents the final reward granted when the ship reaches the target area, which is set as a significant positive value. R_4 decreases linearly as the number of actions executed by the ship increases, as expressed in Equation (9):

$$R_4 = 150 \times \left(1 - \frac{M}{M_{\max}}\right) \tag{9}$$

Where M represents the cumulative number of actions taken by the ship, and $M_{\max}$ represents the maximum number of actions, if exceeded, it is considered that the training has entered a local optimum.

3.2.4. Neural network structure

A convolutional neural network (CNN) serves as an approximation model for the Q-value function, facilitating the extraction of essential features from the input environmental states and generating the corresponding action values $Q(s, a)$. The convolutional operation preserves the spatial information of the high-dimensional data while effectively reducing its dimensionality, making it highly applicable to the ship route planning problem.

In the proposed CNN architecture, the input data undergoes a convolutional operation in the first layer to generate 16 feature maps, followed by subsampling through a pooling layer to reduce data volume. A second convolutional layer further refines the features, increasing the number of feature maps to 32, with subsampling again applied via the pooling layer. This convolution-pooling combination allows the network to retain the spatial information of the input while reducing dimensionality and extracting relevant environmental features.

The feature maps produced by the convolutional layers are then flattened and passed through a fully connected layer, resulting in a mapping to a 128-dimensional hidden layer. The output of the hidden layer is subsequently mapped to a 48-dimensional action space, representing the Q-value estimates for the 48 possible actions. At each layer, the rectified linear unit is utilized for nonlinear processing to improve the model's capacity for representation. The definition of the rectified linear unit $f(x)$ can be found in Equation (10):

$$f(x) = \begin{cases} x, x > 0 \\ 0, x \leq 0 \end{cases} \tag{10}$$

Ultimately, the network outputs a 48-dimensional vector corresponding to the 48 action spaces, representing the Q-value for different actions, which is used to select the optimal next action. By integrating convolution and pooling, the network preserves the spatial positional details of the input data while simultaneously decreasing its dimensionality, thus facilitating the extraction of meaningful environmental features. Furthermore, enhancements have been made to the input layer's design to accommodate multidimensional environmental state data. This information is integrated with a fully connected layer to generate the Q-values associated with the action set. The structure of the neural network is depicted in Fig. 6.

3.3. *A** algorithmic guiding DDQN

In the DDQN algorithm, the ε-greedy strategy is frequently utilized for action selection, promoting the agent's exploration of novel state-action pairs. This method helps prevent early convergence on suboptimal solutions caused by an excessive dependence on Q-value assessments. By selecting an action at random with a probability of ϵ, a balance between exploration and exploitation is achieved. Conversely, with a probability of $1-\epsilon$, the action corresponding to the highest current Q-value is chosen. The ε-greedy action selection strategy can be expressed in Equation (11):

$$\pi(s) = \begin{cases} \arg\max_a Q(s_t, a_t), 1-\varepsilon \\ random(A), \varepsilon \end{cases} \tag{11}$$

In the standard DDQN algorithm, random exploration does not guarantee effective exploration, which may lead to irrational turning decisions and reduce computational efficiency. We replace the random exploration with the A* algorithm to enhance search efficiency and improve the adaptability of the neural network. Fig. 2 illustrates the structure of the exploration strategy in the A-DDQN algorithm, and expressed in Equation (12).

$$\pi(s) = \begin{cases} \arg\max_a Q(s_t, a_t), 1-\varepsilon \\ A^*, \varepsilon \end{cases} \tag{12}$$

Equation (12) indicates that in the exploration decision process of the A-DDQN algorithm, the initial exploration rate is set to 0.35, with a 0.65 probability of making decisions by the neural network, and a 0.35 probability of the A* algorithm. The frequency of A* algorithm usage is initially high and decreases as training progresses. It is crucial that the A* algorithm is discontinued at an early stage, and the decision-making process relies entirely on the neural network to ensure its eventual convergence. The evolution of the exploration rate throughout training is illustrated in Fig. 7.

4. Simulation and comparison studies

4.1. Emission estimation

The STEAM2 model is an advanced tool used for evaluating exhaust emissions in maritime traffic with a particular focus on particulate matter (PM) and carbon monoxide (CO) emissions. The model employs data from the Automatic Identification System (AIS) to offer real-time updates on a ship's location and speed, integrating with comprehensive technical information about the ship, including engine load, fuel consumption, and emission control technologies. In contrast to

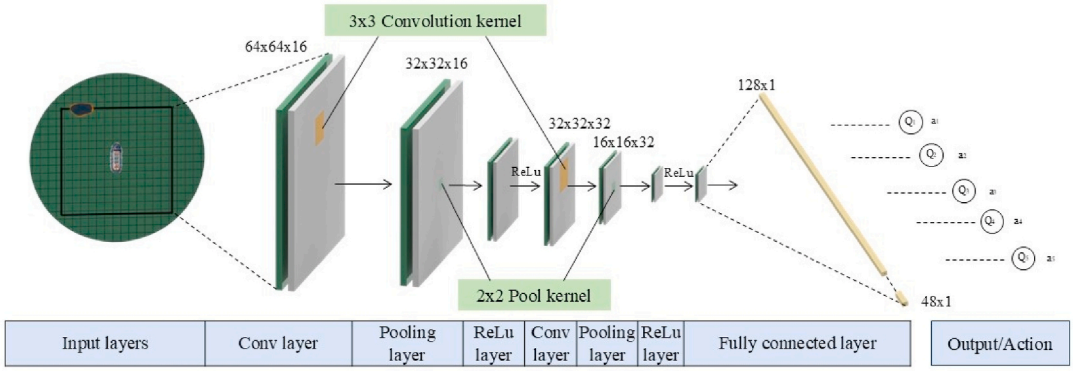


Fig. 6. Neural network structure.

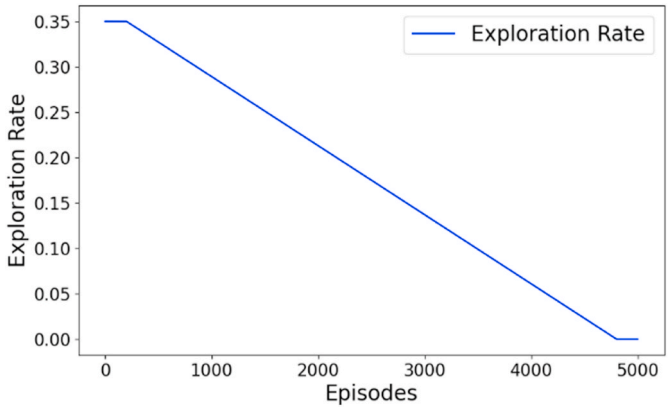


Fig. 7. Exploration rate variation process.

traditional emission models, STEAM2 accounts for dynamic factors such as ship speed, water resistance, and multi-engine configurations, allowing for a more accurate assessment of emissions. The model considers the various influences of ship speed, engine load, sulphur content in fuel, and emission reduction technologies. This integration allows for highly detailed simulations of emissions, achieving precision in both geographic and temporal distributions down to a few meters, utilizing publicly accessible ship characteristics (Grifoll et al., 2022).

To evaluate the proposed route's energy efficiency and its potential for reducing emissions, the STEAM2 method is utilized to analyse the exhaust emissions produced during the ship's voyage. The evaluation primarily considers the following data: engine-rated power, engine load, fuel consumption rate, design speed, sulphur and carbon content in the fuel, and engine revolutions per minute (RPM). According to the STEAM2 method, the ship's emissions during navigation are calculated as the sum of emissions of various pollutants, as shown in Equation (13).

$$E_{T_p} = P \cdot EL \cdot EF \cdot t \tag{13}$$

Where P the engine's average power output, measured in kilowatts (kw), EL represents the engine load, EF the emission factor associated with each pollutant, and t represents the total sailing time in hours. EL represents the engine load, is calculated as shown in Equation (14).

$$EL = \frac{P_E}{P_e \cdot n_e} \tag{14}$$

Where P_E represents the power output of the main engine, n_e represents the number of identical engines installed, and P_e represents the maximum continuous rated power of a single installed engine. The absolute fuel consumption estimate for the ship during navigation can be derived based on Equation (13), as shown in Equation (15).

$$FC = \sum_{i=0}^{n} P_i \cdot SFOC_i \backslash \Delta t_i \tag{15}$$

Where $SFOC$ refers to the fuel oil consumption specific to each unit of engine power, and Δt represents the time required to move between nodes for each segment.

$SFOC$ depends on various parameters, including engine load, age, and fuel type. In STEAM2, the model is defined as shown in Equation (16) and Equation (17).

$$SFOC = SFOC_{rel} \cdot SFOC_{base} \tag{16}$$

$$SFOC_{rel} = 0.455 \cdot EL^2 - 0.71 \cdot EL + 1.28 \tag{17}$$

Where $SFOC_{rel}$ represents the relative fuel consumption rate obtained through regression analysis, and $SFOC_{base}$ refers to the specific fuel oil consumption parameter for different engines per unit of engine power.

4.2. Comparison of algorithms

4.2.1. Training process and parameters

The size and operational characteristics of medium-sized cargo ships make them well-suited for simulating complex navigation tasks during training. Their dimensions are neither too large nor too small, preventing overly complex or unrepresentative simulation calculations. Although different types of ships may encounter varying conditions in different navigational environments, medium-sized cargo ships adequately represent the core issues in route planning and are applicable to most shipping scenarios. Simulation experiments were carried out on medium-sized cargo ships, commonly used in coastal or ocean transport, to assess the effectiveness of the algorithm proposed in this study. The modelled ship measures 225 m in length, has a deadweight of 8000 tons, and can achieve a speed of 16.1 knots. The objective of route planning is to guide the ship from its initial position to the specified destination. Table 1 summarizes the configuration and parameters of the simulation, with the starting coordinates at (5,5) and the destination at (108,108) (see Table 2).

Table 1
Training parameter settings.

Parameters	Configuration
Replay Buffer Size	1×10^6
Batch Size	128
Learning Rate	7.5×10^{-6}
Discount Factor	0.99
Initial Exploration Rate	0.35
Minimum Exploration Rate	0.05
Target Network Update Rate	1000
Maximum Iterations	5000

Table 2
Route length and navigation time for route planning based on the A-DDQN, DDQN, Improved A*, APF, TD3 and RRT algorithms.

Map size (nautical miles *nautical miles)	Model name	Route length (nautical miles)	Travel time (h)
180*180	**A-DDQN**	**199.54**	**12.75**
	DDQN	225.00	14.44
	Improved A*	202.50	12.93
	RRT	205.88	13.15
	APF	220.22	14.09
	TD3	233.85	15.10

Table 1 lists all hyperparameters along with their corresponding values. The batch size, learning rate and exploration rate are particularly critical, it's because they have a significant impact on the training process. We began with a modest batch size of 32 and progressively increased it to 128 and 256, assessing their effects on both training speed and model accuracy. Ultimately, a batch size of 128 was chosen to balance computational efficiency and training stability. For the learning rate, we started with a standard initial value of 0.001 and adjusted it based on the observed changes in the loss function during training to improve precision. As described in Section 3.3, the initial exploration rate was set to 0.35 and was gradually reduced in a linear fashion as training progressed. We set the number of iterations for each training round to 5000 and obtained the optimal hyperparameters shown in Table 1 through continuous comparison of the training results from each round.

4.2.2. Simulation experiments

The training process for 5000 iterations took a total of 20 h, utilizing the simulated navigation environment developed in this study, which consists of three primary phases. This experiment aimed to evaluate the proposed method's effectiveness within complex environments influenced by ocean currents. Table 1 provides comprehensive information on the experimental parameters, while Fig. 8 displays the simulation test results. The ship begins its journey from the initial location, steering clear of land and islands, and, influenced by ocean currents, it formulates an optimal route to the target point. The experiment incorporated various training stages, with corresponding records of route planning performance and trends in cumulative reward values. With the increase in training iterations, the cumulative reward values consistently improved, leading to a marked enhancement in route planning performance.

Fig. 8 depicts the training process of the A-DDQN algorithm. In the early phase as shown in Fig. 8(a), the ship's trajectory appears quite convoluted, resulting in a cumulative reward value of 272.62. As training progresses, as shown in Fig. 8(b), the route becomes more optimized, with some reductions in unnecessary course adjustments, although minor adjustments are introduced in a few areas, and the reward value increases to 290.69. In the 4500th iteration, as shown in Fig. 8(c), the route is further optimized with fewer course deviations, and the reward value rises to 310.91. After 5000 iterations, as shown in Fig. 8(d), the ship effectively avoids obstacles, generating an optimal route, and the reward value significantly increases to 320.68.

The experimental findings indicate that the proposed method demonstrates significant robustness in dynamic environments, including those affected by ocean currents. It effectively reduces ship navigation time and shows excellent disturbance resistance, aligning with practical application requirements.

4.2.3. Comparison of route planning based on different algorithms

Fig. 9 presents the route planning outcomes for both the conventional DDQN algorithm and the enhanced version of the algorithm. The efficiency of random exploration is notably low, especially in larger environmental models, where random exploration does not equate to effective exploration. It can sometimes cause the ship to move further away from the target area, failing to improve the neural network's fitting capability. Additionally, random exploration may increase the probability of selecting invalid manoeuvres. As indicated by the red circle in Fig. 9, during normal navigation, the ship suddenly changes course to the right due to random exploration. Considering that the sea area has neither significant winds and waves nor corresponding islands or illegal zones, the total journey distance unnecessarily increases. Similarly, in the area indicated by the yellow circle, unreasonable exploration occurs, where random exploration causes the ship to make a sharp turn during normal navigation. It is the significant steering changes that can easily lead to safety hazards during navigation.

To prevent irrational route planning as described above, this paper replaces random exploration with the A* algorithm, which not only optimizes the route but also improves the convergence ability of the model.

Fig. 10 illustrates a comparison of the proposed method with traditional global planning techniques, highlighting both route length and travel duration, which demonstrates the effectiveness of the approach. In the simulation, all ships start from a common point and proceed toward the target area within a specified marine environment. The traditional A* algorithm typically optimizes for a single objective as the shortest distance. However, ship route planning necessitates the consideration of multiple interrelated objectives in practical navigation. The improved A* algorithm incorporates multiple cost functions, integrating various optimization objectives into a unified framework. The improved A* algorithm selects the route that minimizes the total cost function $f(N)$ as shown in Equation (18).

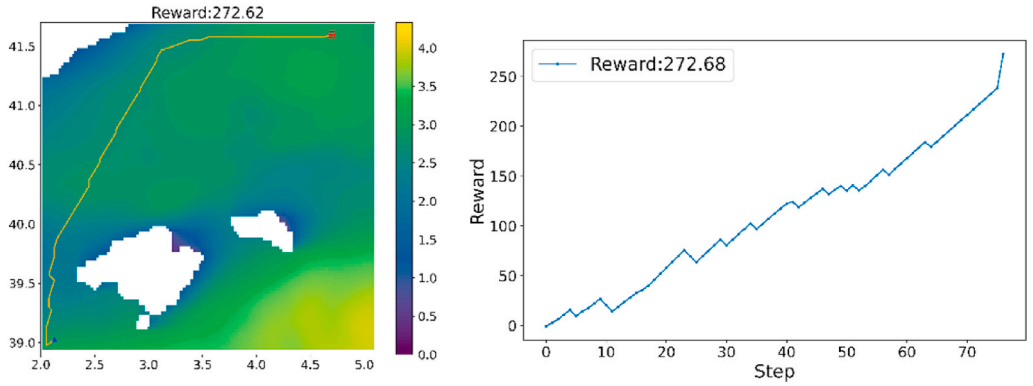
$$f(N_n) = \alpha h(N_n) + \beta \sum_{i=1}^{n} g(N_i) + \gamma g_{loss}(N) \tag{18}$$

Where α, β and γ denotes weight coefficients to balance the importance of different objectives, $h(N_n)$ is a heuristic function to estimate the minimum route cost from node n to the target area, $g(n)$ denotes the cost from node N_i to its parent node N_{i-1}, while $g_{loss}(N)$ denotes the minimum cost of ship speed loss from node n to the target area due to meteorological influences.

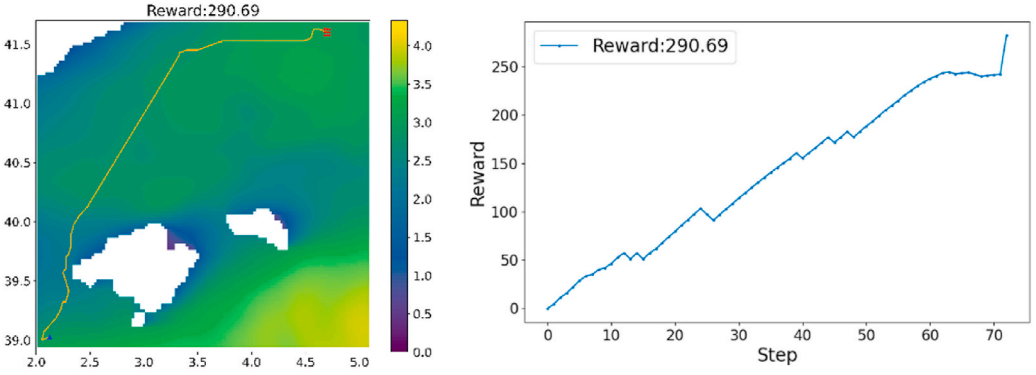
In the algorithm comparison experiment, we employed a linear interpolation method to interpolate the speed loss values of the navigation area onto the grid, assigning distance-related weights to each connection. While the improved A*, RRT, APF, TD3, DDQN and A-DDQN algorithms effectively address route planning challenges in high-dimensional spaces with complex constraints, successfully directing ships to their destinations, the key differences are evident in the efficiency of the routes and the complexity involved in the travelling. This study validates the advantages of the proposed method through comparisons with various alternative approaches.

The RRT algorithm calculates the distance from the starting position to the target area as 205.88 nautical miles. In comparison, the route generated by the DDQN algorithm is 225.00 nautical miles, whereas the route determined by the improved A* algorithm covers 202.50 nautical miles. The route generated by the APF algorithm is 220.22 nautical miles, while the route generated by the TD3 algorithm is 233.85 nautical miles. In contrast, the improved A-DDQN algorithm results in a route length of 199.54 nautical miles, indicating a notable reduction in route distance by the improved method.

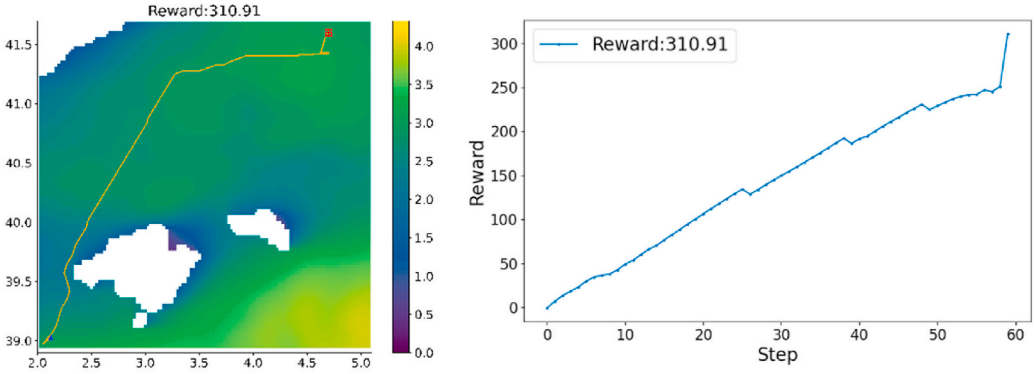
In terms of travel duration, the RRT algorithm takes 13.15 h to reach the target area from the starting point, whereas the improved A* algorithm accomplishes the same in 12.93 h. The route generated by the APF algorithm requires 14.09 h to reach the target area, whereas the TD3 algorithm accomplishes the same task in 15.10 h. The enhanced DDQN algorithm further decreases the travel time to 12.75 h. This demonstrates that the improved A-DDQN algorithm minimizes the distance and decreases the time costs in actual navigation, demonstrating higher navigation efficiency and superior route optimization capabilities.



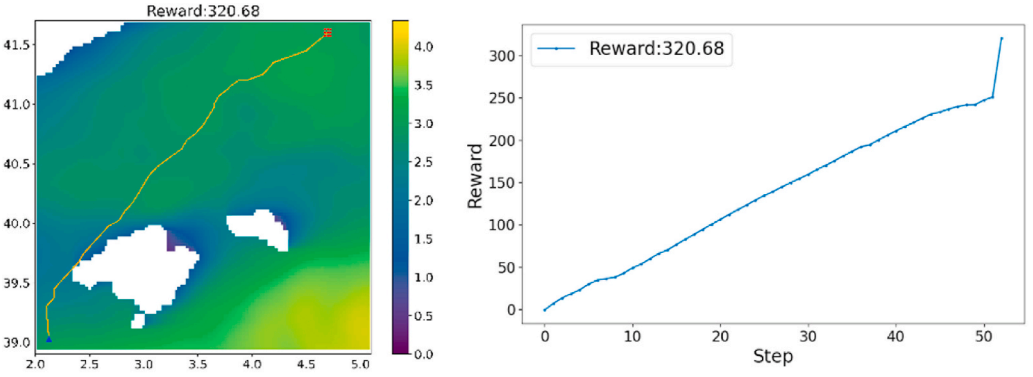
(a) Training result after 1772 iterations



(b) Training result after 3099 iterations



(c) Training result after 4558 iterations



(d) Training result after 4934 iterations

Fig. 8. Training process of the A-DDQN Algorithm.

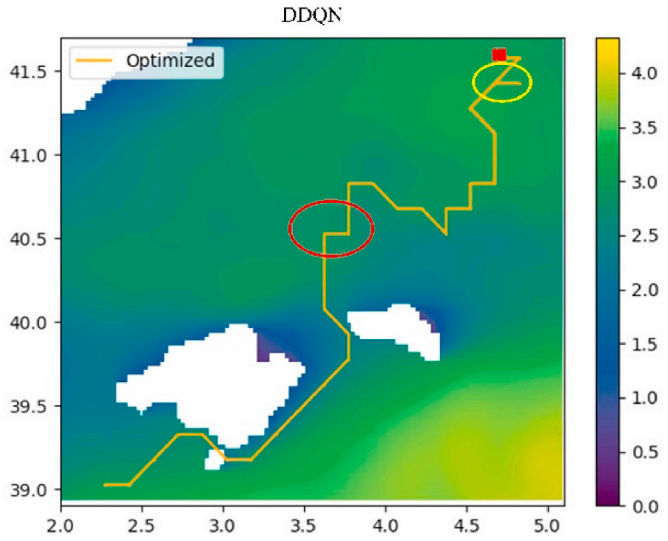


Fig. 9. Example of random exploration.

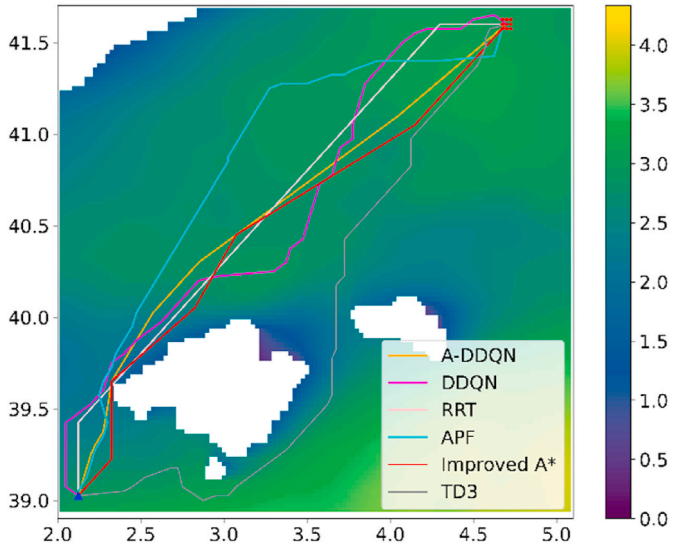


Fig. 10. Route planning results of A-DDQN, DDQN, improved A*, APF, TD3 and RRT algorithms.

4.2.4. *Emission comparison*

Based on the fuel consumption and emission calculation method outlined in Section 4.1 with the STEAM2 model, the fuel usage and emissions of the ship following the planned routes were assessed to confirm the contribution of the A-DDQN-planned route to environmentally friendly and energy-efficient navigation. Fig. 11 illustrates coloured curves that compare fuel consumption over time for routes determined by various algorithms within the same navigation environment.

It is noteworthy that the performance of the DDQN algorithm and TD3 algorithm models are worse when compared to the other algorithm models, primarily due to longer navigation routes and unnecessary course alterations resulting from random exploration. To demonstrate the effectiveness of the improved algorithm and highlight the differences between various deep reinforcement learning algorithms, the standard DDQN algorithm is used as a representative for comparison with other algorithms. From the analysis of the overall trend, the green curve representing A-DDQN remains lower than the other curves for most of the time, demonstrating that the A-DDQN algorithm yields better results in terms of energy savings and emission reductions.

To assess the energy efficiency and emission reduction of the optimal route produced by the proposed method, calculations were performed on the ship's fuel usage and exhaust emissions to validate the approach. As illustrated in Table 3, simulation results reveal that the fuel consumption for the ship navigating the route devised by the DDQN algorithm totals 42.379 tons. The fuel consumption of the route planned based on the APF algorithm is 41.260 tons, showing a slight reduction compared to the standard DDQN algorithm. The route planned based on the TD3 algorithm results in a fuel consumption of 44.465 tons, indicating a slight increase compared to the standard DDQN algorithm. In contrast, the route developed by the RRT algorithm results in a fuel consumption of 38.472 tons, indicating a notable reduction compared to the DDQN algorithm. The improved A* algorithm further lowers fuel consumption to 37.814 tons. Additionally, the A-DDQN algorithm demonstrates superior performance with a fuel consumption of just 37.325 tons, outperforming the other three algorithms.

When compared to the route designed by the DDQN algorithm, the APF algorithm achieves a 2.84 % reduction in the fuel consumption, the TD3 algorithm achieves a 5.20 % increase in the fuel consumption (the negative symbol in Table 3 indicating the TD3 model requires more fuel consumption). It is also noted that the RRT algorithm achieves a 9.22 % reduction in fuel consumption, while the improved A* algorithm shows a decrease of 10.77 %, while the A-DDQN algorithm demonstrates a 11.93 % reduction, highlighting its superior performance in terms of fuel efficiency. Regarding exhaust emissions, the A-DDQN algorithm surpasses the DDQN algorithm's emissions of 114.612 tons, as well as those from the RRT algorithm at 103.731 tons, the improved A* algorithm at 101.924 tons, the APF algorithm at 111.36 tons and the TD3 algorithm at 120.57 tons. The A-DDQN algorithm results in a 12.16 % decrease in emissions compared to the DDQN algorithm, yielding better outcomes than the other algorithms.

In summary, the A-DDQN algorithm significantly reduces fuel costs and emissions in actual navigation, enhancing environmental benefits and aligning with the demands of the modern era for green, low-carbon, energy-efficient, and environmentally friendly solutions.

4.2.5. *Comprehensive performance evaluation of the A-DDQN algorithm*

Based on the experimental results and analysis presented in Sections 4.2.3 and 4.2.4, it is evident that the A-DDQN algorithm proposed in this paper demonstrates significant advantages in optimizing multiple objectives, including route length, travel time, fuel consumption, and emissions. Comparative experiments show that A-DDQN not only achieves the shortest route length (199.54 nautical miles) and the lowest travel time (12.75 h) among the tested algorithms, but also significantly reduces fuel consumption (37.325 tons) and emissions (100.675 tons), outperforming traditional algorithms and other deep reinforcement learning-based methods. These improvements reflect A-DDQN's ability to dynamically balance route length and fuel efficiency, particularly under varying weather conditions, by integrating a multi-objective cost function that accounts for speed loss, distance, and environmental factors, resulting in superior navigation planning.

Furthermore, the experimental results validate that A-DDQN avoids the inefficiency of random exploration seen in the standard DDQN and TD3 algorithms, which lead to increased fuel consumption due to unnecessary course adjustments. By utilizing the improved A* algorithm for exploration and integrating a multi-objective cost function, A-DDQN ensures more rational and efficient route planning. This comprehensive optimization capability aligns with the demands of modern maritime practices, which seek to minimize fuel consumption and emissions while maintaining operational efficiency. The findings further demonstrate the effectiveness of A-DDQN in achieving environmentally friendly and energy-efficient navigation, making it a reliable solution for practical applications.

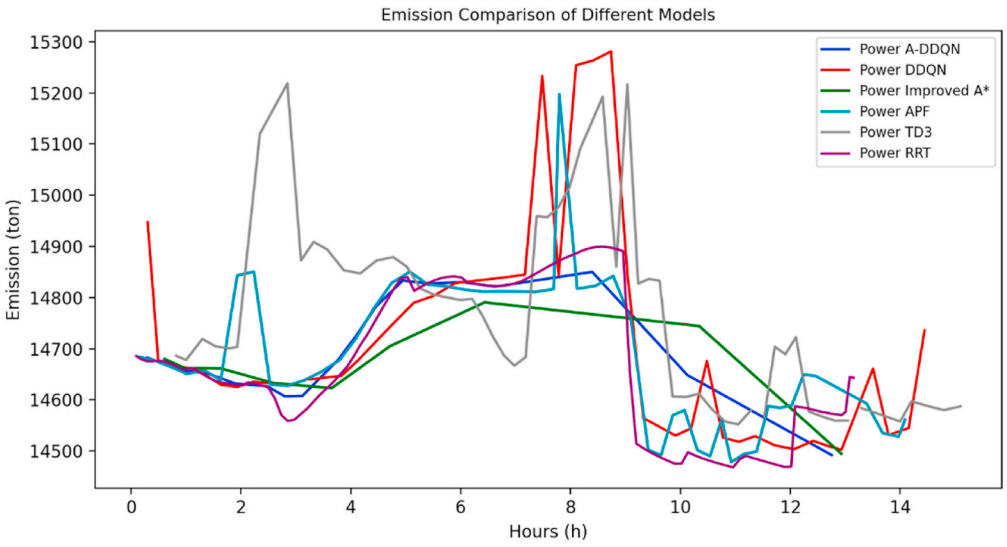


Fig. 11. Emission comparison of different models.

Table 3
Ship emission calculations under different models.

Model name	Fuel consumption (ton)	Fuel consumption reduction (vs DDQN)	Emissions (ton)	Percentage of emissions mitigation (vs DDQN)
A-DDQN	37.325	11.93 %	100.675	12.16 %
Improved A*	37.814	10.77 %	101.924	11.07 %
RRT	38.472	9.22 %	103.731	9.49 %
APF	41.260	2.64 %	111.36	2.84 %
TD3	44.465	−4.92 %	120.57	−5.20 %
DDQN	42.379	0	114.612	0

4.3. Comparison of ship route planning based on different scenarios

Meteorological conditions play a crucial role in ship route planning, affecting factors such as navigation safety, time efficiency, and energy consumption. Therefore, evaluating the performance of the proposed framework under adverse weather conditions is essential to ensure its stability and robustness. As shown in Table 4, sea waves are classified into ten different levels based on wave height, and different wave conditions have varying impacts on ships. Sea waves with a significant wave height under 2 m exhibit relatively mild surface fluctuations, allowing ships to navigate smoothly under such sea conditions. The hull may experience slight vertical motions, but this typically does not affect the ship's stability. Sea waves with a significant wave height between 2 and 4 m exhibit more pronounced surface fluctuations, and the ship may experience noticeable pitching or rolling during navigation. Under such conditions, the ship's stability is affected, and it may be necessary to reduce speed to maintain smooth navigation especially during high-speed navigation. Sea waves with a significant wave height over 4 m significantly impact the stability of ships. The ship may experience severe pitching with the risk of capsizing. Speed typically needs to be substantially reduced to minimize the impact of the waves on the hull. Large waves also increase fuel consumption, as ships often need to adopt slower and more cautious navigation strategies. Additionally, prolonged navigation under large wave conditions accelerates hull wear and structural damage, reducing the ship's service life.

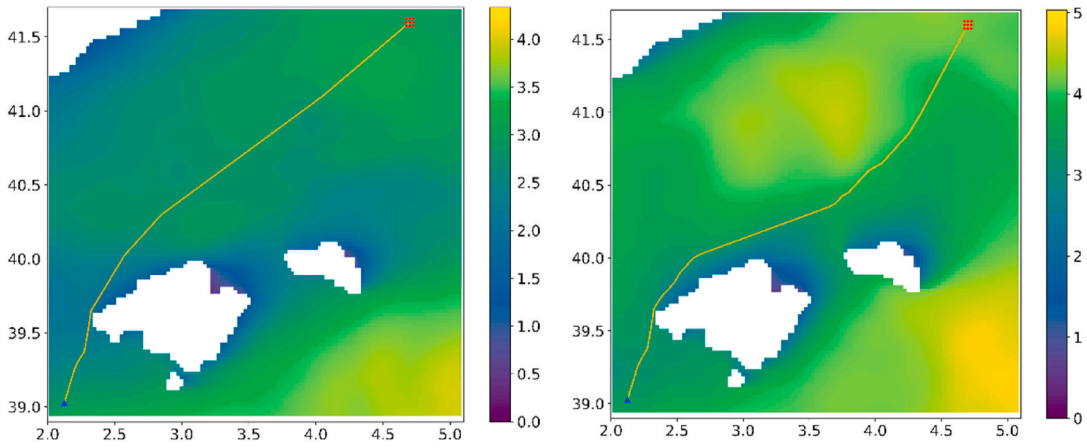
Table 4
Sea state classification scale.

Sea state classification	Significant wave height (meters)
Calm-glassy	0
Calm-rippled	0–0.1
Smooth-wavelet	0.1–0.5
Slight	0.5–1.25
Moderate	1.25–2.5
Rough	2.5–4.0
Very rough	4.0–6.0
High	6.0–9.0
Very high	9.0–14.0
Phenomenal	>14.0

We conducted a comparative experiment in two regions with the same latitude and longitude in the Mediterranean: one representing meteorological data for an area with an average sea state classification of moderate, and the other representing meteorological data for an area with an average sea state classification of very rough. As shown in Fig. 12, in the route planning results for the region with an average sea state classification of moderate in Fig. 12 (a), the route is relatively smooth, and the ship's route closely follows a straight line, resulting in lower energy consumption and shorter travel time. In contrast, in the region with an average sea state classification of very rough in Fig. 12 (b), the proposed framework accurately identifies and avoids the rough sea areas, successfully planning the route to ensure that the ship avoids high-risk areas with waves of rough or higher levels. The framework dynamically adjusts the ship's route, ensuring both safety and efficiency while avoiding rough sea areas.

5. Conclusion

Ships will experience speed loss due to the influence of oceanographic and meteorological factors during maritime navigation. Provided that ship safety is ensured, avoiding areas with severe winds and waves, reducing fuel consumption costs, and decreasing greenhouse gas emissions from fuel consumption are important development directions for the shipping industry. We present a framework for ship route planning utilizing the A-DDQN algorithm, integrating the global pathfinding strengths of the A* algorithm with the adaptive learning features of DDQN. This approach aims to decrease fuel consumption during navigation while also reducing exhaust emissions. The framework is structured into four phases: In the first phase, we determine the speed loss value of the ship in the maritime area, utilizing hydrometeorological data. Combining with geographical information of the area, we use a convolutional neural network approach to construct a ship navigation environment model. In the second phase, the ship's location information and the speed loss at that position serve as input data for the A-DDQN



(a) The result under moderate sea state conditions

(b) The result under very rough sea state conditions

Fig. 12. Results under different sea state conditions.

algorithm. In the third phase, the action strategy that yields the maximum reward value is chosen and executed from the available actions, incorporating the input state information alongside the defined reward mechanism. This iterative procedure continues to repeat the second and third phases until the ship reaches the target area. In the fourth phase, the navigation route from the starting point to the destination is determined, and the STEAM2 emission calculation model is employed to evaluate the exhaust emissions produced by the ship throughout its voyage.

The A-DDQN, RRT, and A* algorithms serve as benchmarks for comparison, demonstrating that the A-DDQN algorithm achieves a 11.93 % reduction in fuel consumption and a 12.16 % decrease in greenhouse gas emissions when compared to the traditional DDQN algorithm. Additionally, when evaluated against these other algorithms, A-DDQN exhibits significant benefits regarding both the length and smoothness of the route. This approach not only ensures the safety of maritime navigation but also minimizes energy usage and exhaust emissions, offering an effective and environmentally friendly solution for intelligent route planning. It supports the dual goals of cost reduction and carbon emission minimization in the shipping industry, underscoring its significant potential for application.

Although the Intelligent ship route planning via an A* search model enhanced Double-Deep Q-Network has achieved favourable results in reducing time and economic costs as well as lowering carbon emissions, the following directions deserve our further attentions for the purpose of enhance our study. First, the model may encounter situations with high computational complexity, necessitating improvements in computational efficiency when confronted with high-dimensional state spaces. Next, the ship's navigation strategy may be influenced by real-time environmental changes in highly dynamic environments, it is interesting to integrate other deep reinforcement learning methods to enhance performance. Then, due to time and experimental resource constraints, the current study has not conducted comprehensive testing and analysis of the method's consistency and sensitivity when applied to more complex or longer routes. We plan to address this gap and provide further examination in future research. Meanwhile, to address the limitations of the current study, future work will focus on developing a multi-objective optimization framework that simultaneously considers route and speed optimization. This framework will predict fuel consumption more accurately and dynamically adjust navigation strategies in response to real-time environmental changes. Furthermore, we will investigate the impact of ship motion states such as pitch and roll on speed optimization, aiming to improve both energy efficiency and navigation stability. These efforts will contribute to a more robust and adaptive route planning system for maritime applications. Last but not least, we can also introduce cutting-edge artificial intelligence models to predict ship fuel consumption improving the accuracy of fuel consumption predictions.

CRediT authorship contribution statement

Xinqiang Chen: Writing – review & editing, Data curation, Conceptualization. **Ruiyang Hu:** Writing – original draft, Data curation, Conceptualization. **Kai Luo:** Writing – review & editing, Funding acquisition. **Huafeng Wu:** Funding acquisition, Data curation. **Salvatore Antonio Biancardo:** Investigation, Data curation. **Yiwen Zheng:** Writing – original draft, Conceptualization. **Jiangfeng Xian:** Funding acquisition, Investigation.

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 work was jointly supported by National Natural Science Foundation of China (No. 52331012, 52472347), Open Fund of Chongqing Key Laboratory of Green Logistics Intelligent Technology (Chongqing Jiaotong University) (No.KLGLIT2024ZD001), Open Fund of Jiangxi Key Laboratory of Intelligent Robot (JXINTROB-2024-201).

References

Armstrong, V.N., 2013. Vessel optimisation for low carbon shipping. Ocean Eng. 73, 195–207.
Bai, X., et al., 2023. USV path planning algorithm based on plant growth. Ocean Eng. 273.
Charalambopoulos, N., Xidias, E., Nearchou, A., 2023. Efficient ship weather routing using probabilistic roadmaps. Ocean Eng. 273.
Chen, Y., et al., 2021. Path planning and obstacle avoiding of the USV based on improved ACO-APF hybrid algorithm with adaptive early-warning. IEEE Access 9, 40728–40742.
Chen, K., et al., 2023a. Liner shipping network design model with carbon tax, seasonal freight rate fluctuations and empty container relocation. Sustain. Horizons 8.
Chen, X., et al., 2023b. Maritime traffic situation awareness analysis via high-fidelity ship imaging trajectory. Multimed. Tool. Appl. 83 (16), 48907–48923.
Chen, X., et al., 2024. Ship visual trajectory exploitation via an ensemble instance segmentation framework. Ocean Eng. 313.

Chu, Z., et al., 2023. Path planning based on deep reinforcement learning for autonomous underwater vehicles under ocean current disturbance. IEEE Trans. Intellig. Vehicl. 8 (1), 108–120.
Cullinane, K., Yang, J., 2022. Evaluating the costs of decarbonizing the shipping industry: a review of the literature. J. Mar. Sci. Eng. 10 (7).
Du, W., et al., 2022a. Energy saving method for ship weather routing optimization. Ocean Eng. 258.
Du, W., et al., 2022b. Ship weather routing optimization based on improved fractional order particle swarm optimization. Ocean Eng. 248.
Gan, L., et al., 2025. Graph neural networks enabled accident causation prediction for maritime vessel traffic. Reliab. Eng. Syst. Saf. 257.
Gkerekos, C., Lazakis, I., 2020. A novel, data-driven heuristic framework for vessel weather routing. Ocean Eng. 197.
Grifoll, M., Borén, C., Castells-Sanabra, M., 2022. A comprehensive ship weather routing system using CMEMS products and A* algorithm. Ocean Eng. 255.
Hashali, S.D., Yang, S., Xiang, X., 2024. Route planning algorithms for unmanned surface vehicles (USVs): a comprehensive analysis. J. Mar. Sci. Eng. 12 (3).
He, Y., et al., 2021. Collision-avoidance path planning for multi-ship encounters considering ship manoeuvrability and COLREGs. Transport. Saf. Environ.
He, Z., et al., 2023. A novel model predictive artificial potential field based ship motion planning method considering COLREGs for complex encounter scenarios. ISA Trans. 134, 58–73.
Jang, D.-u., Kim, J.-s., 2022. Development of ship route-planning algorithm based on rapidly-exploring random tree (RRT*) using designated space. J. Mar. Sci. Eng. 10 (12).
Lee, S.-J., Sun, Q., Meng, Q., 2023. Vessel weather routing subject to sulfur emission regulation. Transport. Res. E Logist. Transport. Rev. 177.
Lee, H.-T., Kim, M.-K., 2024. Optimal path planning for a ship in coastal waters with deep Q network. Ocean Eng. 307.
Li, Y., et al., 2023a. A ship route planning method under the sailing time constraint. J. Mar. Sci. Eng. 11 (6).
Li, J., Huang, C., Pan, M., 2023b. Path-planning algorithms for self-driving vehicles based on improved RRT-Connect. Transport. Saf. Environ. 5 (3).
Liu, Y., Wang, T., Xu, H., 2022. PE-A* algorithm for ship route planning based on field theory. IEEE Access 10, 36490–36504.
Liu, L., et al., 2023. Data-driven framework for extracting global maritime shipping networks by machine learning. Ocean Eng. 269.
Luo, Y., et al., 2024. Reviews on the power management for shipboard energy storage systems. Sustain. Horizons 9.
Ma, D., et al., 2024. Multi-objective ship weather routing method based on the improved NSGA-III algorithm. J. Ind. Inform. Integrat. 38.
Mannarini, G., Carelli, L., 2019. VISIR-1.b: ocean surface gravity waves and currents for energy-efficient navigation. Geosci. Model Dev. (GMD) 12 (8), 3449–3480.
Novac, V., et al., 2020. Ship routing in the black sea based on Dijkstra algorithm. In: 20th International Multidisciplinary Scientific GeoConference Proceedings SGEM 2020, Informatics, Geoinformatics and Remote Sensing, pp. 301–308.
Öztürk, Ü., Akdağ, M., Ayabakan, T., 2022. A review of path planning algorithms in maritime autonomous surface ships: navigation safety perspective. Ocean Eng. 251.
Raza, Z., 2020. Effects of regulation-driven green innovations on short sea shipping's environmental and economic performance. Transport. Res. Transport Environ. 84.
Sahoo, S.P., et al., 2023. Hybrid path planning using a bionic-inspired optimization algorithm for autonomous underwater vehicles. J. Mar. Sci. Eng. 11 (4).
Shang, W.-L., et al., 2023. Spatio-temporal analysis of carbon footprints for urban public transport systems based on smart card data. Appl. Energy 352 (36).
Shang, W.-L., et al., 2024. Can financial subsidy increase electric vehicle (EV) penetration—evidence from a quasi-natural experiment. Renew. Sustain. Energy Rev. 190.
Shu, Y., et al., 2023. Path planning for ships assisted by the icebreaker in ice-covered waters in the Northern Sea Route based on optimal control. Ocean Eng. 267.
Shu, Y., et al., 2024. Reference path for ships in ports and waterways based on optimal control. Ocean Coast Manag. 253.
Silva Junior, A.G.d.S., et al., 2020. High-level path planning for an autonomous sailboat Robot using Q-learning. Sensors 20 (6).
Tadros, M., Ventura, M., Soares, C.G., 2023. Review of current regulations, available technologies, and future trends in the green shipping industry. Ocean Eng. 280.
Ventikos, N.P., Sotiralis, P., Annetis, E., 2022. A combined risk-based and condition monitoring approach: developing a dynamic model for the case of marine engine lubrication. Transport. Saf. Environ. 4 (3).
Wang, X., et al., 2024. Deep reinforcement learning: a survey. IEEE Transact. Neural Networks Learn. Syst. 35 (4), 5064–5078.
Wu, A., et al., 2022. Routeview: an intelligent route planning system for ships sailing through Arctic ice zones based on big Earth data. Int. J. Dig. Earth 15 (1), 1588–1613.
Wu, Y., Wang, T., Liu, S., 2024. A review of path planning methods for marine autonomous surface vehicles. J. Mar. Sci. Eng. 12 (5).
Xiaofei, Y., et al., 2022. Global path planning algorithm based on double DQN for multi-tasks amphibious unmanned surface vehicle. Ocean Eng. 266.
Xie, H., et al., 2021. Connectivity-aware 3D UAV path design with deep reinforcement learning. IEEE Trans. Veh. Technol. 70 (12), 13022–13034.
Xing, B., et al., 2023. A review of path planning for unmanned surface vehicles. J. Mar. Sci. Eng. 11 (8).
Xu, X., Yan, X., Zhang, D., 2022. Introduction to special issue on maritime safety and smart shipping. Transport. Saf. Environ. 4 (4).
Zaccone, R., et al., 2018. Ship voyage optimization for safe and energy-efficient navigation: a dynamic programming approach. Ocean Eng. 153, 215–224.
Zhang, L., et al., 2021. Path planning for autonomous ships: a hybrid approach based on improved APF and modified VO methods. J. Mar. Sci. Eng. 9 (7).
Zhang, Y., Wen, Y., Tu, H., 2023. A method for ship route planning fusing the ant colony algorithm and the A* search algorithm. IEEE Access 11, 15109–15118.
Zhang, G., et al., 2025. Autonomous navigation and control for a sustainable vessel: a wind-assisted strategy. Sustain. Horizons 13.
Zheng, M., et al., 2024. An improved VO method for collision avoidance of ships in open sea. J. Mar. Sci. Eng. 12 (3).
Zis, T.P.V., Psaraftis, H.N., Ding, L., 2020. Ship weather routing: a taxonomy and survey. Ocean Eng. 213.