Contents lists available at ScienceDirect

International Journal of Naval Architecture and Ocean Engineering

journal homepage: http://www.journals.elsevier.com/international-journal-of-naval-architecture-and-ocean-engineering/

Ship collision avoidance route planning using CRI-based A* algorithm

Chanhee Seo [a, c], Yoojeong Noh [a, *], Misganaw Abebe [b], Young-Jin Kang [b], Sunyoung Park [a], Cheolhyeon Kwon [c]

[a] School of Mechanical Engineering, Pusan National University, Busan, Republic of Korea
[b] Research Institute of Mechanical Technology, Pusan National University, Busan, Republic of Korea
[c] Department of Mechanical Engineering, Ulsan National Institute of Science and Technology, Ulsan, Republic of Korea

ARTICLE INFO

Article history:
Received 15 March 2023
Received in revised form
18 August 2023
Accepted 7 September 2023
Available online 9 September 2023

Keywords:
A* algorithm
Ship collision
COLREGs
Collision risk index
Ship domain
Route planning

ABSTRACT

This study presents a novel ship route planning algorithm that takes into account both operational economy and safety by integrating the A* algorithm with a collision avoidance algorithm that evaluates the Collision Risk Index (CRI) between the own ship and the target ship. The CRI-based A* algorithm defines a penalty zone, allowing the own ship to explore safe routes based on the International Regulations for Preventing Collisions at Sea 1972 (COLREGs) and performs an adaptive and effective node search on an extended local map grid according to various encounter situations. The proposed algorithm is validated through simulations of head-on, fine-broad crossing, converging crossing, and overtaking encounters, indicating an economical and safe optimum route compared to conventional ship domain-based route planning.

1. Introduction

Smart ship technologies are evolving based on Information and Communication Technology (ICT) and Artificial Intelligence (AI) technology, and the biggest advantages of unmanned vessels, economics, and stability, must be considered in the development of unmanned vessels.

The economic feasibility of ships can be improved by exploring routes that minimize Fuel Oil Consumption (FOC), which is also linked to marine environmental regulations; therefore, route optimization is a very important issue. At the same time, collision avoidance is crucial for the safety of ships, particularly in operating systems such as autonomous vessels. It is highly important to select safe avoidance routes by assessing the risk of collision with other vessels based on maritime regulations such as COLREGs. To achieve a balance between ship safety and economic considerations, it is necessary to develop algorithms that minimize the risk of collision while maximizing operational cost-effectiveness.

In this study, we propose a route planning method that considers both the safety and optimality of the route by applying a CRI and adhering to COLREGs regulations using an enhanced A* algorithm. The proposed algorithm explores nodes in the area suitable for collision avoidance between ships. The CRI takes into account factors such as Distance Closest Point of Approach (DCPA), Time Closest Point of Approach (TCPA), relative distance, relative angle, and relative velocity, representing the level of risk on a scale from 0 to 1. Additionally, a penalty zone is defined based on the Target Ship (TS) to ensure that the A* algorithm adheres to the COLREGs regulations. These two factors are incorporated in the A* algorithm and iteratively updated, influencing the search space of the algorithm. The proposed method has been validated through 2000 simulations in various encounter scenarios, and the results show that it efficiently explores economic routes while maintaining a low risk of collision compared to other ship domain models such as Goodwin, Fuji, and Szlapczynski.

2. Literature survey

There have been many previous studies on technologies to find economical routes for ships. For example, many studies proposed a route search method that minimizes the Estimated Time of Arrival (ETA) or traveling distances. For this, various meta-heuristic

[*] Corresponding author.
E-mail addresses: cksgml456@naver.com (C. Seo), yoonoh@pusan.ac.kr (Y. Noh), misge98@gmail.com (M. Abebe), zmanx@pusan.ac.kr (Y.-J. Kang), ppsy17@naver.com (S. Park), kwonc@unist.ac.kr (C. Kwon).
Peer review under responsibility of The Society of Naval Architects of Korea.

https://doi.org/10.1016/j.ijnaoe.2023.100551

Nomenclature

S_O	Location of own ship
S_T	Location of target ship
V_O	Velocity of own ship
V_T	Velocity of target ship
φ_O	Course of own ship
φ_T	Course of target ship
D_R	Relative distance
V_R	Relative velocity
φ_R	Relative course
θ_T	Relative bearing
u_{DCPA}	Membership function of DCPA
u_{TCPA}	Membership function of TCPA
u_{D_R}	Membership function of relative distance
u_{θ_T}	Membership function of relative bearing
u_{ε}	Membership function of V_T/ V_O
s	Start node
u	Current node
v	Next node
R	Range of penalty zone
d	Cost between s and u
r	Binary variable about CRI
c	Binary variable about COLREGs rule
$\widehat{t}$	Estimated progress time

Abbreviations

CRI	Collision risk index
OS	Own ship
TS	Target ship
SOG	Speed over ground
DCPA	Distance at closest point of approach
TCPA	Time at closest point of approach
ETA	Estimated time of arrival

optimization algorithms such as Genetic Algorithm (GA) (Wang et al., 2020), simulated annealing algorithm (Vlachos, 2004), and anti-colony algorithm (Tsou and Cheng, 2013) to find the optimal route (see Table 1). A dynamic programming was used for a ship voyage optimization (Zaccone et al., 2018) and combined with a genetic algorithm to optimize ship engine powers and reduce fuel and air emissions (Wang et al., 2021). Most representative heuristic optimization algorithms used in path finding, A* algorithm, have been used. Water current, traffic separating, and berthing were considered to predict FOC models for path finding (Liu et al., 2019) and combined with 2-D grid-based Gas (Tanakitkorn et al., 2014). Ship voyage optimization utilized dynamic programming (Zaccone et al., 2018) and combined it with a genetic algorithm to optimize ship engine powers, reducing fuel and air emissions (Wang et al., 2021). In path finding, popular heuristic optimization algorithms like the A* algorithm were employed. An improved A* algorithm

Table 1
Summary of literature surveys.

Category	Algorithms		Authors
Ship route optimization	Meta-heuristic algorithm	Genetic algorithm	Wang et al. (2020)
		Anti-colony algorithm	Tsou and Cheng (2013)
		Ant colony algorithm	Zhang et al. (2021a)
		Simulated annealing algorithm	Vlachos (2004)
		NSGA-II algorithm	Lee et al. (2018)
	Heuristic algorithm	A* algorithm	Liu et al. (2019)
		Improved A* algorithm	Shin et al. (2020)
		GA + A* algorithm	Tanakitkorn et al. (2014)
	Machine learning	Reinforcement learning	Moradi et al. (2022)
		Deep Q network algorithm	Guo et al. (2021)
	Dynamic algorithm	3D dynamic programming	Zaccone et al. (2018)
		GA + Dynamic programming	Wang et al. (2021)
		GA + Velocity obstacle	Wang et al. (2023)
Collision avoidance	Meta-heuristic algorithm	Optimal reciprocal algorithm	Zhao et al. (2016)
	Rule-based algorithm	Fuzzy logic + COLREG	Hu et al. (2020)
	Machine learning	Deep reinforced learning based on COLREG	Zhao and Roh (2019), Ning et al. (2020), Chun et al. (2021)
		Deep reinforced learning with LSTM	Sawada et al. (2021)
		NN + fuzzy logic	Ahn et al. (2012)
		Attention-mechanism based on deep reinforcement learning	Jiang et al. (2022)
		Hybrid ARIMA-LSTM	Abebe et al. (2022)
	Others	Deterministic optimization algorithm for multi-ship encounter situations	Yu et al. (2022)
		Predictive probability using Kalman filter	Kim et al. (2022)
Path optimization + Safety	Meta-heuristic algorithm	GA + COLREG for multi-encounter situations	Ning et al. (2020)
		NSGA-II + COLREG	Li and Pang (2013)
		Modified potential field ant colony algorithm	Gao et al. (2023)
	Heuristic algorithm	A* algorithm + COLREG	Blaich et al. (2012)
		Dynamic anti-collision A* algorithm for multi-ship encounter situations	He et al. (2022)
	Machine learning	Deep Q + COLREGs	Guo et al. (2021) Xu et al. (2022)
	Others	Danger immune algorithm + COLREG	Xu, Q. (2014)
		3D ship motion model + Beta antennae search algorithm	Xie et al. (2019)

considering water current, traffic separating, and berthing factors was proposed (Liu et al., 2019) and combined with a 2-D grid-based GA (Tanakitkorn et al., 2014).

In recent days, various studies have applied machine learning models for ship route optimization. Zhang et al., 2021a employed an ANN for FOC prediction and an enhanced ant colony optimization algorithm for multi-objective ship route optimization. Shin et al. (2020) introduced an improved A* algorithm that addresses the limitations of traditional A* algorithms and enables smoother route exploration. Recently, reinforced learning has emerged as a development for ship route optimization. Guo et al. (2021) proposed optimized deep Q network algorithm for coastal ship path planning. Similarly, Moradi et al. (2022) utilized ANN for FOC prediction and optimized deep Q network algorithm for ship route optimization.

Ship safety is vital in operational planning, considering economic feasibility and collision risks. CRIs play a key role in quantifying collision risks. Studies have proposed evaluation methods and avoidance algorithms based on CRIs and COLREGs. Kearon (1977) introduced risk estimation methods using parameters like DCPA and TCPA. Xu et al. (2010), Bukhari et al. (2013), and Ahn et al. (2012) employed fuzzy methods to statistically consider various factors affecting collision risks. Abebe et al. (2021) enhanced CRI evaluation using the Dampster-Shafer theory and gradient boosting regressor. In terms of collision avoidance algorithms, Zhao and Roh (2019) proposed a strategy based on deep reinforcement learning and COLREGs, while Zhao et al. (2016) introduced a fuzzy theory-based CRI evaluation method and a collision avoidance algorithm using velocity obstacles. Nguyen et al. (2018) integrated a CRI evaluation model into AIS for small vessels. Abebe et al. (2022) employed a hybrid ARIMA-LSTM model to predict ship trajectories for collision avoidance. Additionally, Hu et al. (2020); Yu et al. (2022); Xie et al. (2020); and Sawada et al. (2021) developed collision avoidance algorithms for multi-ship encounters.

Previous studies focused on either economic path optimality or collision risk assessment and avoidance. Recent research has aimed to simultaneously consider both path optimization and safety. Various algorithms such as fast grid-based collision avoidance (Blaich et al., 2012), danger immune algorithms (Xu et al., 2010), optimal collision avoidance control theory (Xie et al., 2019), GA (Ning et al., 2020), NSGA-II (Li et al., 2019), and deep Q network (Guo et al., 2021; Xu et al., 2022) have been proposed. A* algorithm for dynamic anti-collision was proposed for multi-ship encounter situations (He et al., 2022). However, many of these studies lack consideration of both COLREGs and collision risk indices (CRIs) together. They also do not account for weather effects and lack general applicability due to limited simulation tests. Advanced route optimization methods based on reinforcement learning have potential, but their lengthy optimization process hinders real-time route prediction.

However, most of the studies have proposed the ship's trajectory planning considering ship collision avoidance that uses only CRI without COLREGs or COLREGs without CRIs. They use distance-based path cost to explore economic routes while reducing the risk of collision, which may be difficult to calculate accurate route costs because they do not consider weather effects such as waves and wind. Simulation tests considering specific encounter situations and ships' operating conditions are insufficient to show the general applicability of the proposed algorithms. Recently, advanced route optimization methods based on reinforcement learning have been actively developed. However, these methods often require a significant amount of time for optimization, limiting their practical application for real-time route prediction.

Compared to previous studies, this study makes the following main contributions: (1) It proposes a CRI-based pathfinding algorithm that incorporates the CRI and COLREGs into the improved A* algorithm. The improved A* algorithm utilizes weather-based ETA costs, allowing for more accurate evaluation of route costs compared to distance-based methods by considering weather factors and providing more precise sailing times. By evaluating the CRI and finding an economic path that adheres to COLREGs, the algorithm enables the identification of economical, safe, and reliable paths. (2) An adaptive node search method, which adjusts according to encounter situations, reduces the computational time required for the existing node search in the implemented A* algorithm. (3) The introduction of a penalty zone prevents unnecessary route searches as the ship follows COLREGs. (4) By continuously calculating CRI values during navigation, the algorithm ensures a more economical and safer route compared to other ship domain models. (5) The dataset, comprising 2000 data points generated through Latin hypercube sampling, includes high-risk encounter situations and operating conditions, providing validation for the safety and economics of the proposed algorithm.

3. CRI-based A* algorithm

3.1. Overview of the proposed algorithm

When the Own Ship (OS) encounters the Target Ship (TS), it determines whether to give way after collecting AIS data for the target ship. According to the COLREGs, if the position of the target ship is included within a radius of 6 nm within an angle of $-5^\circ - 112.5^\circ$ based on the own ship's moving direction, the own ship is obliged to give way. When the own ship needs to give way, the low-resolution global map grids are subdivided into high-resolution local map grids to define the start and end nodes in the dense grid. Once the local map grid is generated between the own ship and the target ship, the process of searching for avoidance paths through the proposed CRI-based A* algorithm, as shown in Fig. 1, begins.

Starting from the start node, the path cost (f cost) at the current node and violations of the CRI value and COLREGs are first checked. If the current node is not the same as the end node, an edge relaxation process is conducted to calculate the path cost of adjacent nodes and update the path cost when moving to the adjacent nodes. After checking for violations of the CRI value and COLREGs at the adjacent node, the algorithm returns to the first step, and the node with the smallest path cost becomes the current node.

Section 3.2 explains how to evaluate the f cost at search nodes in the A* algorithm, where the f cost represents the ETA calculated using the SOG model. In the A* algorithm, an adaptive node search direction is proposed based on the avoidance situation for efficient search for paths within the dense grid map generated during collision avoidance. Section 3.3 explains the CRI evaluation method based on fuzzy theory, which incorporates parameters such as DCPA, TCPA, relative distance, relative angle, and relative speed. The collision risk levels can be divided into low ($0 \leq \text{CRI}<0.4$), medium ($0.4 \leq \text{CRI}<0.7$), and high ($0.7 \leq \text{CRI}<1.0$) (Zhou and Wu, 2004). A CRI value exceeding 0.7 indicates a high risk of collision, hence 0.7 is set as the collision avoidance criterion. Users have the flexibility to set a lower CRI value for safer collision avoidance, depending on their preference. Section 3.4 describes the definition of the penalty zone, which represents an area where the target ship is obligated to give way based on COLREGs. The CRI values and COLREGs are utilized as constraints in the route optimization process using the A* algorithm. Section 3.5 outlines the path search process that adaptively avoids collisions using the CRI and penalty zones within the A* algorithm. Section 4 introduces the ship domain models used in previous research and compares the navigation efficiency and safety between the proposed method and existing model-based

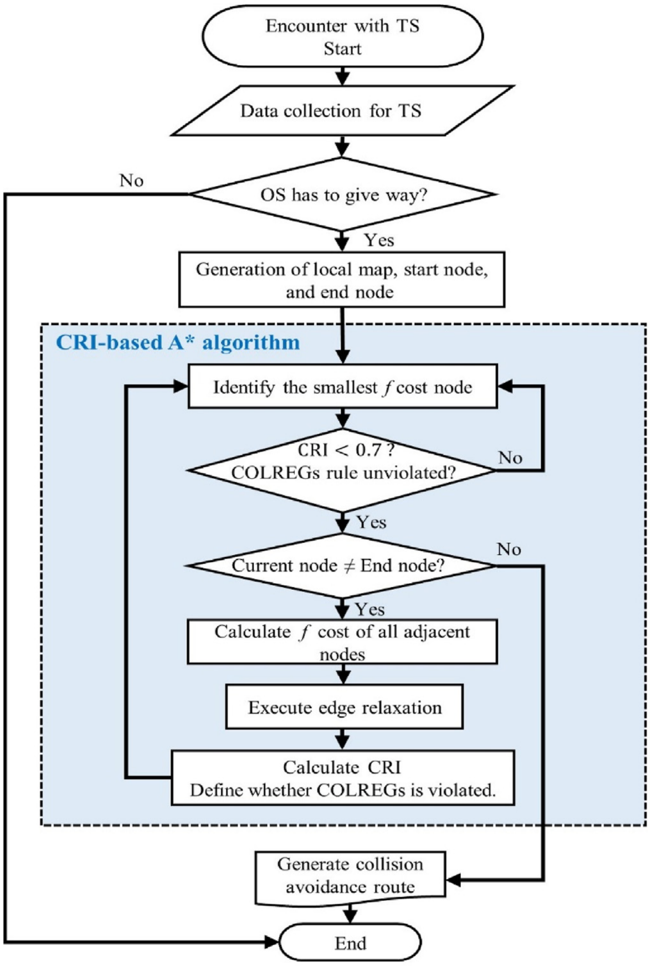


Fig. 1. Flow chart of collision avoidance system.

methods through simulation tests involving various encounter situations between the own ship and the target ship.

3.2. A* algorithm using SOG prediction model

The A* algorithm, proposed by Hart et al. (1968), applied a cost-evaluation methodology called "Heuristic" to address the inefficiency of the Djikstra algorithm, which explores all directions equally. The heuristic method reduces the calculation of unnecessary nodes, enabling quicker determination of the destination. The A* algorithm aims to find the path from a given start node to the end node with the minimum cost, as depicted in Eq. (1).

$$Minimize\, f(u) = g(u) + h(u) \tag{1}$$

Here, $g(u)$ is the actual distance cost from the start node to the current node (u), and $h(u)$ is the minimum expected cost from the current node to the end node as a heuristic function. The objective function is mainly used as cost; $f(u)$ is the distance or time. In this study, the Estimated Time of Arrival (ETA) was used as the cost. The ETA is defined as

$$ETA = \frac{2 \times D(u,v)}{SOG(u) + SOG(v)} \tag{2}$$

where u and v are the current and next nodes, respectively. $D(u,v)$ is the great-circle distance between two nodes calculated using a haversine function, and SOGs need to be obtained from the ship speeds at the current and next nodes. In this study, SOG was predicted based on AIS and weather data using XGBR, where 13 variables are used as input variables through variation inflation factor analysis. The AIS-related variables include COG and draught; the weather-related variables include total wave height, total wave direction, total wave period, pressure surface, sea surface salinity, sea surface temperature, wind speed, wind angle, current speed, and current angle; gross tonnage was also considered as an additional variable. The XGBR was verified as accurate and robust for predicting the speed (Chen and Guestrin, 2016; Shin et al., 2020; Abebe et al., 2020).

Each time a node is searched in the A* algorithm, the above calculation is repeated, and when the current node becomes the end node, the loop ends, and the path from the end node to the start node is derived. Among the algorithms that search for paths using nodes, A* is the most popular because it generates a small number of search nodes and efficiently finds the optimal path. If a path is determined by searching for fewer nodes than that of A*, the path may not be optimal (Dechter and Pearl, 1985).

However, traditional A* algorithms are disadvantageous in that they present non-smooth paths in the grid node map. To solve this problem, Shin et al. (2020) proposed a method of reducing computational time and obtaining a smooth path by increasing the search nodes of the existing 8-way method algorithm to 16-way. Smooth pathfinding and fast calculation time are important for making fast decisions and taking action in situations where there is a real risk of collision. Therefore, this study proposes an improved A* algorithm for collision avoidance that efficiently searches nodes in consideration of the ship's encounter angle and maneuvering characteristics according to COLREGs.

Fig. 2 shows various encounter situations; head-on, overtaking, fine-broad crossing, and converging crossing, according to the relative angles between the own ship and the target ship. According to each encounter situation, the A* algorithm performs two types of node searches, as shown in Fig. 3. Traditional A* algorithms search for nodes located in 45° directions in all directions from the current node, as shown in Fig. 3(a). However, in the case of head-on or fine-broad crossing situations within ±60° of the relative angle, as shown in Fig. 3(b), it is sufficient to conservatively search for nodes only within the most extreme boundaries in fine-broad crossing. On the other hand, for converging crossing situations, as shown in Fig. 3(c), collisions can be avoided even if only the node is searched up to the boundary within ±112.5°. Once the search space is defined based on the current node, the current node selects the node with the lowest path cost within the search space. Then, the selected node becomes the current node and repeatedly searches

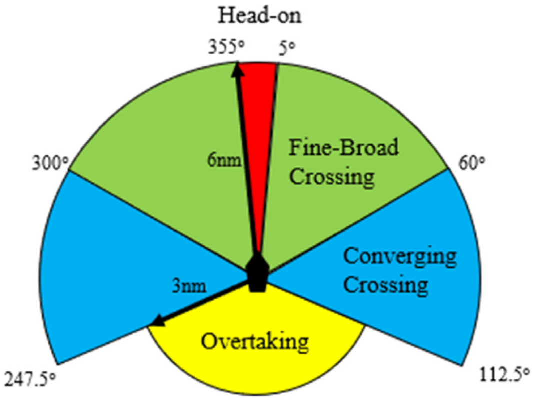


Fig. 2. Ship encounter situation.

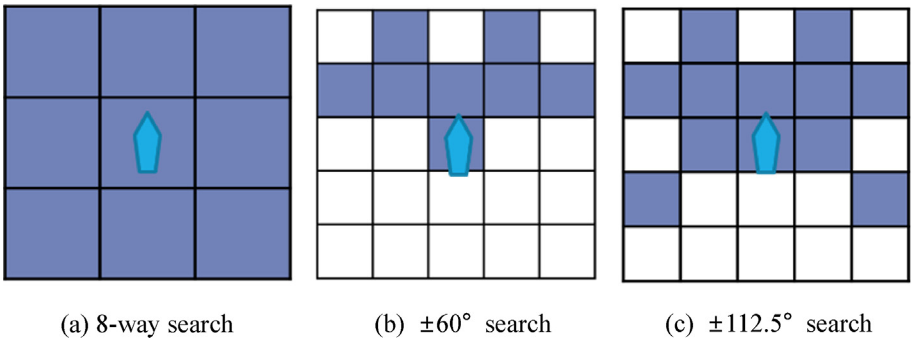

(a) 8-way search (b) ±60° search (c) ±112.5° search

Fig. 3. Traditional node search and adaptive node searches.

the node with the lowest path cost until it reaches the target node. In the case of an overtaking situation, the own ship does not significantly change the direction; thus, the own ship only searches for nodes within ±60° as with the head-on and fine-broad crossing situations.

The adaptive node search, which changes the node search area according to the encounter situation, efficiently searches nodes toward the direction of the end node, unlike traditional path search algorithms that search in all directions. This method is based on the assumption that the own ship is sailing in a great sea and there are no major obstacles between ships and non-holonomic characteristics in ship maneuvering. In addition, it can make the path smoother and more realistic by enabling node search at a narrower angle than the 8-way search method. Calculating the forward-biased area of the ship standard to reduce computational time can allow the proposed algorithm to quickly find the optimal path in the event of a collision risk situation. The proposed method was developed for single-vessel collision scenarios; however, in situations involving multiple vessel encounters, the selection of the path considers the lowest collision risk and cost based on the CRI calculation between the own ship and target ships. Modifications to the selected area for node exploration may be necessary.

3.3. Collision risk assessment

To avoid the collision of ships, various factors, such as relative angles and directions, should be considered in addition to the distance between the own ship and the target ship. The CRI evaluates the degree of collision risk by comprehensively considering the DCPA, TCPA, relative distance, relative bearing, and the relative velocity between two ships. Among these, DCPA and TCPA correspond to factors with large weights in determining the CRI values. As shown in Fig. 4, DCPA and TCPA can be obtained through geometric calculations of ship encounter situations where the dashed-dotted lines represent the extended line of the relative velocity, V_R. The location coordinates, velocity, and course of the own ship are $S_O(x_O, y_O)$, V_O, and φ_O, and those of the target ship are $S_T(x_T, y_T)$, V_T, and φ_T. D_R, V_R, and φ_R are the relative distance, velocity, and course, respectively. a_T is the azimuth of the target ship, and θ_T is the relative bearing.

The equations for DCPA, TCPA, D_R, and θ_T are defined in Eqs. (3)–(9).

$$DCPA = D_R \times \sin(\varphi_R - a_T - \pi) \tag{3}$$

$$TCPA = D_R \times \cos(\varphi_R - a_T - \pi) \, / \, V_R \tag{4}$$

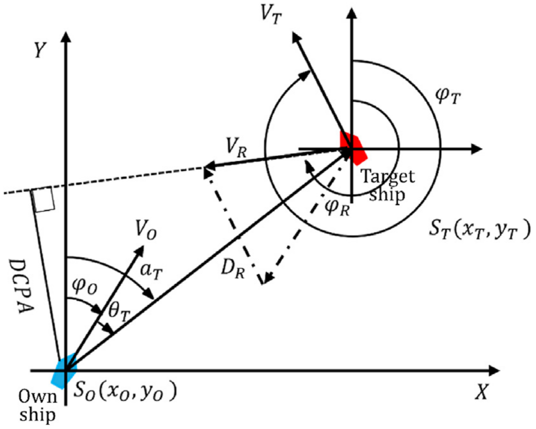


Fig. 4. Diagram of the two-ship encounter geometry.

$$D_R = \sqrt{(x_T - x_O)^2 + (y_T - y_O)^2} \tag{5}$$

$$\varphi_R = \begin{cases} \tan^{-1}(V_{Rx}/V_{Ry}), & V_{Rx} \geq 0, V_{Ry} \geq 0 \\ \tan^{-1}(V_{Ry}/V_{Rx}) + 90^\circ, & V_{Rx} \geq 0, V_{Ry} \leq 0 \\ \tan^{-1}(V_{Rx}/V_{Ry}) + 180^\circ, & V_{Rx} \leq 0, V_{Ry} \leq 0 \\ \tan^{-1}(V_{Ry}/V_{Rx}) + 270^\circ, & V_{Rx} \leq 0, V_{Ry} \geq 0 \end{cases} \tag{6}$$

$$\begin{cases} V_{Rx} = V_{Tx} - V_{Ox} \\ V_{Ry} = V_{Ty} - V_{Oy} \end{cases}, \begin{cases} V_{Ox} = V_O \sin \varphi_O \\ V_{Oy} = V_O \cos \varphi_O \end{cases}, \begin{cases} V_{Tx} = V_T \sin \varphi_T \\ V_{Ty} = V_T \cos \varphi_T \end{cases} \tag{7}$$

$$V_R = \sqrt{{V_{Rx}}^2 + {V_{Ry}}^2} \tag{8}$$

$$\theta_T = a_T - \varphi_O \tag{9}$$

The factors introduced earlier (DCPA, TCPA, D_R, θ_T, and V_R) are important for evaluating the collision risk index. However, it is difficult to evaluate CRI deterministically because the relationship between each factor is complex, and ambiguous and each factor has a different effect on evaluating the risk of collision depending on the encounter situation. Therefore, a recent study was conducted to predict CRI through fuzzy inference using membership functions for various factors, including DCPA and TCPA. Fuzzy inference can reflect the influence of each factor in CRI evaluation through membership functions according to various encounter situations. It has been widely used in recent years because the degree of collision risk can be expressed as a 0–1 value, which is independent of the scales of each factor. In this study, the CRI was calculated using the membership functions of five factors by applying the weights used in previous studies (Xie et al., 2019).

$$CRI = \mathbf{W} \cdot \mathbf{U} = (0.4, 0.367, 0.133, 0.067, 0.033)\begin{bmatrix} u_{DCPA} \\ u_{TCPA} \\ u_{D_R} \\ u_{\theta_T} \\ u_{\varepsilon} \end{bmatrix} \tag{10}$$

where **W** is the weight vector, and the weight values above are typically used values that are estimated based on the difficulty of ship avoidance in various collision situations using navigation simulations (Xie et al., 2019; Zhang et al., 2021b). **U** is the membership matrix, and ε is the velocity ratio of the target ship to the own ship, $\varepsilon = V_T/V_O$. The membership functions for each factor are defined in Eqs. (11)–(15). The parameters used in these equations are listed in Table 2 (Li and Pang, 2013; Gang et al., 2016; Hu et al., 2020). In Table 2, d_1 and d_2 refer to the minimum distance and safest encounter distance between two ships, respectively. The factor '2' between d_1 and d_2 reflects the instability and incoordination of the ships. t_1 and t_2 are the time taken by the give-away ship from the start of avoidance action to the nearest point of approach and the time of taking action to the target ship, respectively. D_1 and D_2 are the distance to be avoided by the give-away ship taking avoidance action and the safe distance for the give-way ship to take collision avoidance action, respectively. The formula for D_2 is obtained by the radius of the marine power model proposed by Davis et al. (1980) considering the relative bearing between the own ship and target ship.

$$u_{DCPA} = \begin{cases} 1, & |DCPA| \le d_1 \\ 0.5 - 0.5\sin\left[\dfrac{\pi}{d_2 - d_1}\left(|DCPA| - \dfrac{d_2 + d_1}{2}\right)\right], & d_1 < |DCPA| \le d_2 \\ 0, & d_2 < |DCPA| \end{cases} \tag{11}$$

$$u_{TCPA} = \begin{cases} 1, & t_1 \ge |TCPA| \\ \left(\dfrac{t_2 - |TCPA|}{t_2 - t_1}\right)^2, & t_1 < |TCPA| \le t_2 \\ 0, & t_2 < |TCPA| \end{cases} \tag{12}$$

$$u_{D_R} = \begin{cases} 1, & 0 \le D_R < D_1 \\ \left(\dfrac{D_2 - D_R}{D_2 - D_1}\right)^2, & D_1 \le D_R \le D_2 \\ 0, & D_2 < D_R \end{cases} \tag{13}$$

$$u_{\theta_T} = 0.5\left[\cos(\theta_T - 19^\circ) + \sqrt{\frac{440}{289} + \cos^2(\theta_T - 19^\circ)}\right] - \frac{5}{17} \tag{14}$$

$$u_{\varepsilon} = \frac{1}{1 + \frac{2}{\varepsilon\sqrt{\varepsilon^2 + 1 + 2\varepsilon \sin C}}} \tag{15}$$

As can be seen from Eqs. (11)–(15) above, CRI has a value between 0 and 1 for the degree of collision risk by simultaneously reflecting various conditions for two ships. The closer it is to 1, the higher the collision risk. Because the risk of collision is generally considered high when the CRI is greater than 0.7 (Nguyen et al., 2018; Gang et al., 2016), an immediate collision avoidance maneuver is required when having a high CRI value in a collision situation. The CRI-based A* algorithm prevents a node having a CRI of 0.7 or more from being selected as the next node when searching for a path. This enables a safer, more robust, and more efficient route search than determining the safety area by considering only the relative distance between ships. The advantage of the proposed method over the existing ship domain-based path search is described in detail in Section 4.

3.4. COLREGs compliant collision avoidance method

In situations where there is a risk of collision between two ships at sea, if there is confusion in determining the direction of avoiding each other, the risk of collision increases and can lead to accidents.

Table 2
Explanation and calculation of relevant parameters.

Variables	Formulas	Explanation
d_1	$\begin{cases} 1.1 - \dfrac{0.2\theta_T{}^\circ}{180}, & 0 \le \theta_T < 112.5^\circ \\ 1.0 - \dfrac{0.4\theta_T{}^\circ}{180}, & 112.5^\circ \le \theta_T < 180^\circ \\ 1.0 - 0.4(360 - \theta_T)/180^\circ, & 180^\circ \le \theta_T < 247.5^\circ \\ 1.1 - \dfrac{0.2(360 - \theta_T)}{180^\circ}, & 247.5^\circ \le \theta_T \le 360^\circ \end{cases}$	Smallest encounter distance
d_2	$d_2 = 2d_1$	Absolute safest encounter distance
t_1	$t_1 \begin{cases} \dfrac{\sqrt{D_1^2 - DCPA^2}}{V_R}, & \|DCPA\| \le D_1 \\ \dfrac{D_1 - \|DCPA\|}{V_R}, & \|DCPA\| > D_1 \end{cases}$	Ship collision time
t_2	$\dfrac{\sqrt{12^2 - DCPA^2}}{V_R}$	Avoidance time
D_1	$(8–12)L$	Distance of last action
D_2	$1.7\cos(\theta_T - 19^\circ) + \sqrt{4.4 + 2.89\cos^2(\theta_T - 19^\circ)}$	Distance of action
C	$\cos^{-1}\left(\dfrac{V_{Ox}V_{Tx} + V_{Oy}V_{Ty}}{\sqrt{(V_{Ox}{}^2 + V_{Oy}{}^2)(V_{Tx}{}^2 + V_{Ty}{}^2)}}\right)$	Collision angle ($0^\circ \le C \le 180^\circ$)

L: length of the own ship.

Therefore, the IMO enacted the COLREGs in 1972 to prevent this. The COLREGs consist of five parts from A to E, of which Part B, a rule related to navigation, is an important part of determining the direction of collision avoidance. Zaccone et al. (2019) dealt with collision avoidance decisions considering Part B. In this study, the proposed algorithm performed a safe path search by applying the regulations on Part B's collision avoidance direction and give-way according to the COLREGs.

To avoid a collision between two ships, it is important to ensure that the own ship does not enter the area (penalty zone). Here, the encounter angle required to give way is between −5° and 112.5° based on course over ground of the target ship, and the radius R that determines the range of the penalty zone is 12 min away from the SOG of the target ship. Fig. 5 shows the penalty zone with radius R and is defined by reference to the radius used in the look head function of the Electronic Chart Display and Information System (ECDIS), an electronic map used by ships at sea. The look head is a range based on the speed of the target ship and generally uses rectangular and cone shapes. In ECDIS, dangerous areas when sailing are determined according to the set range of look heads. Although the criteria for setting the range of look heads used in ECDIS are different depending on the shipping company operated by the ship, it is generally used based on the distance of 12 min based on the SOG of the target ship, and the range R of the penalty zone is also determined using the same criteria.

$$R = SOG_T \times 12/60 \ \ [nm] \tag{16}$$

When selectively searching for nodes in the direction of the target node for economic operations described in Section 3.1, the penalty zone is defined as areas where it is impossible to avoid collisions. Because the proposed CRI-based A* algorithm selectively searches for nodes that do not belong to the penalty zone, it searches for paths that comply with the COLREGs.

3.5. Collision avoidance routing

Most path search algorithms, including typical A* algorithms, aim to find paths on a set map, and obstacles to avoid are marked on the map in advance, making it impossible to consider dynamically moving objects. However, if the path and time of the object's movement can be known in advance, dynamic objects can be reflected in the path search using the tree structure of the A* algorithm. This section describes how the proposed algorithm reflects the dynamic characteristics of the target ship, excludes collision risk nodes, and searches for safe routes.

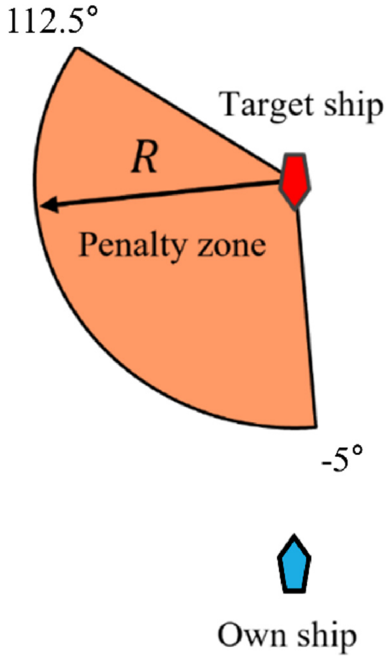


Fig. 5. Penalty zone.

The procedure of the proposed A* algorithm for ship collisions is shown in Fig. 1 and described as follows:

```
i.   For all nodes $\in$ N in graph G provided, initialize with
     $d(node) = \infty$ and $r[node], c[node] = False$
ii.  For source $s$, initialize $d(s) = 0$ and $f(s) = d(s) + heuristic(s)$, then put into
     Queue as key with value of $f$.
iii. Extract minimum-valued node $u$ from Queue.
iv.  Checking safety and regulation, If $r[u]$ or $c[u]$ is True then repeat iii.
v.   Execute edge relaxation and examining CRI and COLREGs based on $d$ at
     adjacent node $v$, putting newly visited nodes into Queue.
vi.  Repeat iii., iv., and v. until the path reaches the target node $t$.
```

where N is the set of all nodes, G is the map graph, and d is the ETA between the start node (s) and a node.

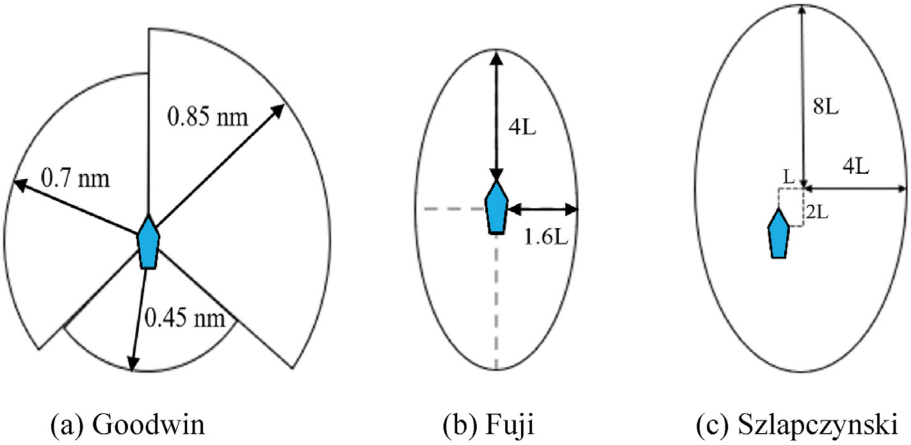
In step i, $d(node)$ is the ETA between the start node and all nodes in **N**. r is a bool value that determines whether there is a risk of collision according to the CRI value and has only a True or False value. It has a True value for CRI<0.7 and False for CRI≥0.7 (Zhou and Wu 2004; Hu et al., 2020). c is a bool variable that determines whether the own ship enters the penalty zone. When the own ship enters the penalty zone, c is True; otherwise, False. In the proposed algorithm, to search for a path that satisfies both collision safety and the COLREGs simultaneously, if either is violated, node v is put in a closed set and excluded from the next node candidate. The proposed algorithm is similar to the traditional A* algorithm dealing with an obstacle zone, but there is a difference in the calculation of the avoidance paths based on the dynamic motion of the target ship.

The A* algorithm has a heap-type nonlinear graph structure based on a complete binary tree, which means that the data is hierarchically structured, unlike a linear structure. As a feature of the heap structure, all search nodes except the start node have one parent node. Therefore, in the process of node search through the algorithm, the A* algorithm can find the parent node of each search node, and the parent node of the parent node can also be known. Through this repeated parent node search, the path from the current node to the start node can be calculated, and the distance and time of the calculated path can be measured. This method is used in the A* algorithm to calculate the path when the current node arrives at the end node, which is used in this study to determine the time taken from the start node to each current node to predict the location of the target ship. However, because the A* algorithm performs a one-time route calculation, the following two assumptions are required to predict the location of the relative ship.

First, the speeds of the own ship and the target ship do not change. Second, the target ship does not give way. In other words, COG is assumed to be constant. The own ship is assumed to be a power-driven vessel, and the target ship is assumed to be a vessel restricted in its ability. The estimated progress time $(\hat{t})$ through the distance from the start node to the current node can be determined using the above two assumptions. As shown in Eqs. (17)–(19), the expected location (latitude and longitude) of the target ship that changes for each node can be found.

Table 3
Pseudocode of the CRI-based A* algorithm.

```
function A* ($\boldsymbol{G}(\boldsymbol{V},\boldsymbol{E}),\boldsymbol{W},s,t,\boldsymbol{OS},\boldsymbol{TS}$)
    for all $u \in V-\{s\}$ do
        $d[v]\infty, pred[v]nil$
        $r[v], c[v] \leftarrow False$
    $d[s] \leftarrow HEURISTIC(s,t)$                              : Heuristic function
    $\boldsymbol{Queue} \leftarrow \{s\}, \boldsymbol{S} \leftarrow \emptyset$
    while $\boldsymbol{Queue} = \emptyset$ do
        $u \leftarrow Extract\_min_d(\boldsymbol{Queue})$
        if $\{r[u]\ or\ c[u]\} = True$ then
            Continue
        $\boldsymbol{S} \xleftarrow{update} \boldsymbol{S} \cup \{u\}$
        if $u = t$ then
            return $d, MAKEPATH(t,pred)$                      : Tracking backward
        else if $pred[u] \neq \emptyset$ then
            $Calculate\ ETA(s,u)$
        for all $e \in \{e_{u,v} \in E\ |u \in V, v \in \boldsymbol{adjacent}[u]\}\ \}$ do
            $OS \xleftarrow{update} OS + ETA(s,u) + \Delta ETA(u,v)$        : Examining safety and regulation
            $TS \xleftarrow{update} TS + ETA(s,u) + \Delta ETA(u,v)$
            $r[v] \xleftarrow{update} CRI(OS,TS)$
            $c[v] \xleftarrow{update} COLREG(OS,TS)$
            if $d[v] > d[u] + \boldsymbol{W}(e)$ then                     : Edge relaxation
                $d[v] \xleftarrow{update} d[u] + \boldsymbol{W}(e) + HEURISTIC(v,t)$
                $pred[v] \xleftarrow{update} u$
                $\boldsymbol{Queue} \leftarrow \boldsymbol{Queue} \cup \{u\}$
```



(a) Goodwin (b) Fuji (c) Szlapczynski

Fig. 6. Ship domain models.

Table 4
Initial data of simulation ships.

		Latitude [deg]	Longitude [deg]	SOG [knot]	COG [deg]
Own ship		37	131	15	0
Target ship	Head-on	37.1	131	15	180
	Fine-Broad Crossing	37.085	131.035	15	225
	Converging Crossing	37.065	131.065	16	265
	Overtaking	37.06	131	5	0

$$\hat{t} = Distance_v(s,v)\ /\ SOG_{OS} \tag{17}$$

$$Latitude_{TS}(v) = Latitude_{TS}(s) + SOG_{TS} \cdot \cos(COG_{TS}) \cdot \hat{t} \tag{18}$$

$$Longitude_{TS}(v) = Longitude_{TS}(s) + SOG_{TS} \cdot \sin(COG_{TS}) \cdot \hat{t} \tag{19}$$

Therefore, to apply the CRI and penalty zone to the proposed A* algorithm, in the process of inputting $d(v)$ of the adjacent node$[u]$ as a key value, the previously calculated CRI value and the target ship are input to the position of $r[v]$ and the Boolean value of $c[v]$, respectively. Then, the node selected as the smallest f cost is checked to satisfy the two constraints (CRI and COLREGs). Otherwise, the node that satisfies the two constraints and has the next low cost is selected. In the preceding iterative manner, when the current node(u) is equal to the end node(t), the final path is generated, and the algorithm is terminated. The end node is

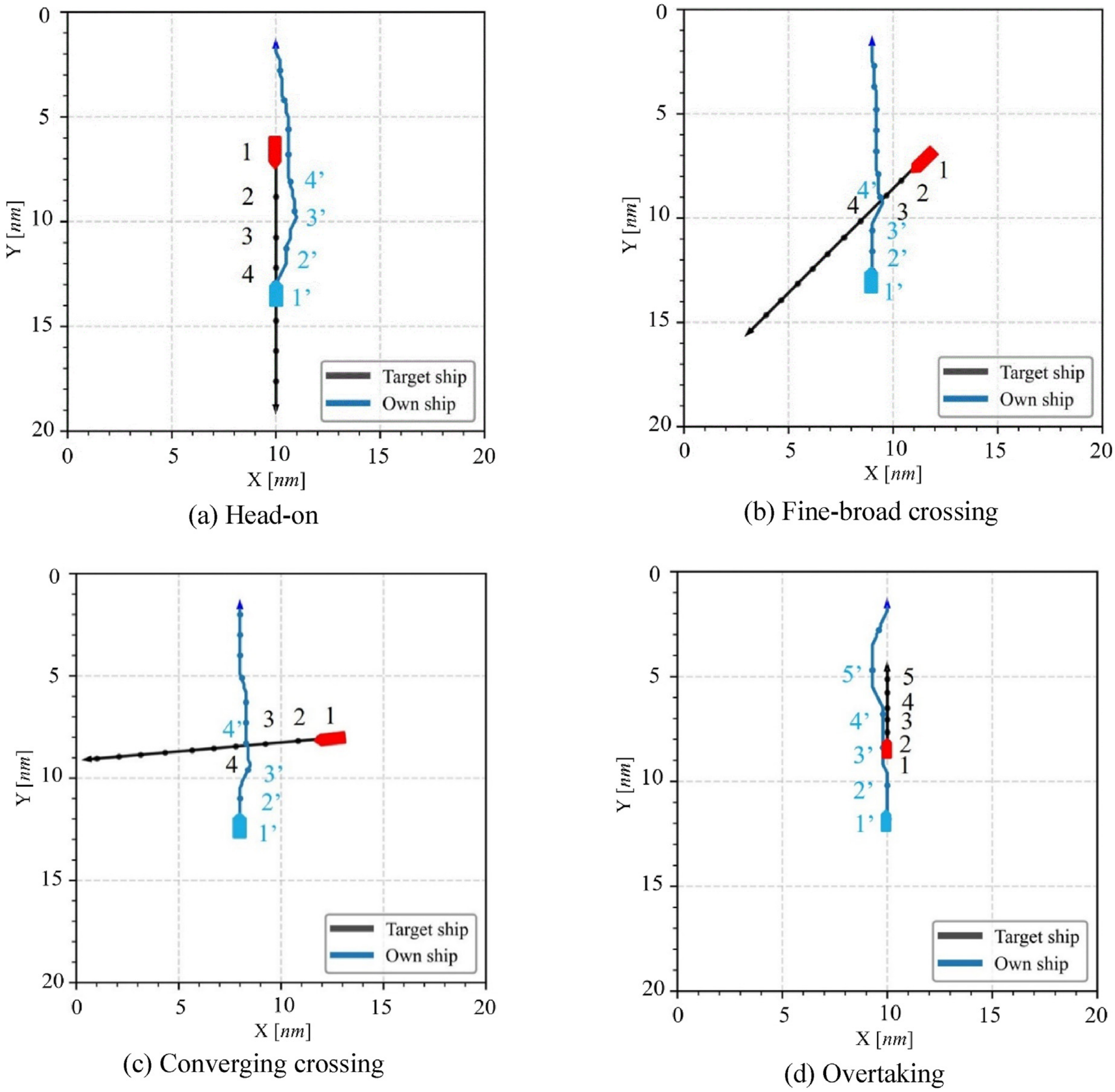


Fig. 7. Collision avoidance simulations of the proposed algorithm on local map grids.

defined as the intersection where the ship's route is extended to meet the edge of the local map, allowing the own ship to follow the existing economic route after taking a collision avoidance action. The pseudocode of the proposed algorithm is presented in Table 3. The CRI-based A* algorithm ensures safer operation, by lowering the risk of collision than conventional A* algorithms in that it rediscovers collision avoidance paths under COLREGs when potential collisions are detected within the penalty zone.

4. Comparison with ship domain

Sections 2 and 3 discuss in detail the map grid configuration in which the algorithm is performed, the A* algorithm using SOG prediction models, CRI, and how to apply the COLREGs to search for paths capable of collision avoidance based on the A* algorithm. This section examines how the CRI-based A* algorithm works compared with other methods based on traditional ship domains. First, we examined whether the proposed algorithm can generate routes that avoid the target ship's access from various angles. Next, the existing ship domain-based and CRI-based collision avoidance algorithms are applied equally to the A* algorithm to evaluate the avoidance maneuvering performance of the own ship in various encounter situations.

4.1. Building the environmental map

Before testing the performance of the proposed algorithm, the grid size of the map used was reduced, and its resolution was increased. The global map and prediction models used in previous studies were suitable for route planning in a large sea, and it takes a long time to compute as the size of the map is large. However, in routing for collision avoidance between ships, the distance between nodes must be short to search for detailed avoidance paths, and the calculation time should be reduced to establish immediate avoidance path planning. Therefore, in this study, the grid-based global map was split into a local map, and the edges between nodes were divided at 1/10 intervals so that the grid scale was 0.1 nm. The intersection points were used as nodes so that the algorithm could search for more detailed paths. Linear interpolation was performed at each node to apply the predicted SOG parameter,

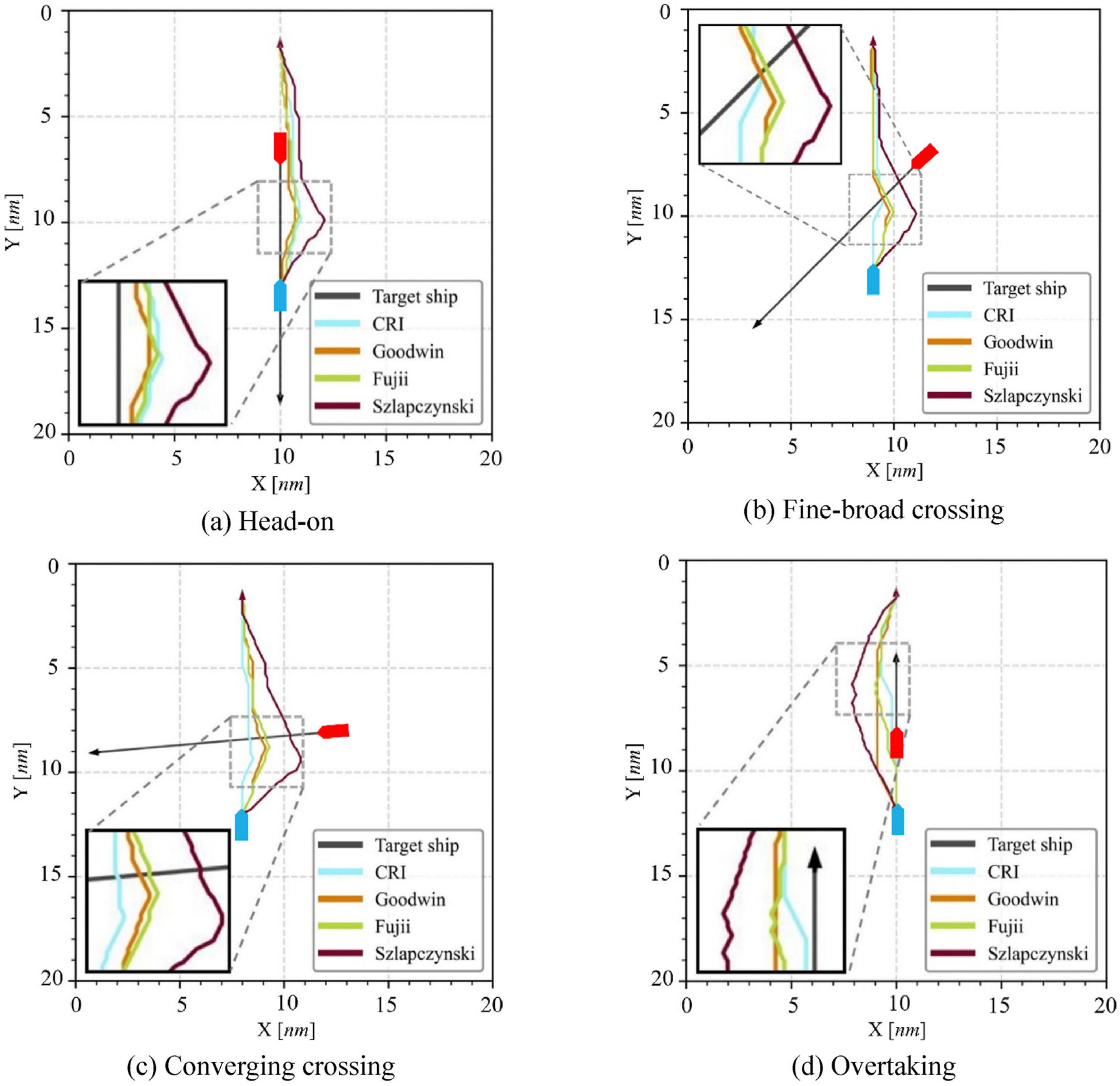


Fig. 8. Collision avoidance path calculated in various situations on local map grids.

Table 5
The result table of measuring the maximum CRI and distance for each encountered situation.

		CRI-based model	Goodwin	Fujii	Szlapczynski
Head-on	Max. CRI	0.57	0.92	0.81	0.55
	Distance	11.67	11.53	11.62	12.45
Fine-Broad crossing	Max. CRI	0.65	0.93	0.85	0.30
	Distance	12.20	12.38	12.43	13.34
Converging crossing	Max. CRI	0.62	0.27	0.22	0.21
	Distance	11.67	11.96	12.06	13.54
Overtaking	Max. CRI	0.63	0.94	0.95	0.57
	Distance	11.50	11.62	11.72	12.62

Distance unit: *nm*.

which is the *g*-cost used in the A* algorithm, to the local map. While this study used a grid scale of 0.1 nm, using a denser grid scale can ensure a smoother path and allow for more accurate calculation of CRI.

4.2. Collision avoidance based on ship domain

The ship domain method, in which the safety area of a ship is determined by the length over all (LOA), is generally used in collision avoidance and maritime transportation research. Goodwin (1975) proposed a circular ship domain based on a ship's traffic history, and different safety distances were determined according to the angle of an encounter between ships. Fujii and Tanaka (1971) and Coldwell (1983) proposed an elliptical ship domain; similarly, the safety distance is measured in proportion to the LOA. In a recent study, Szlapczynski et al. (2018) adopted the existing elliptical ship

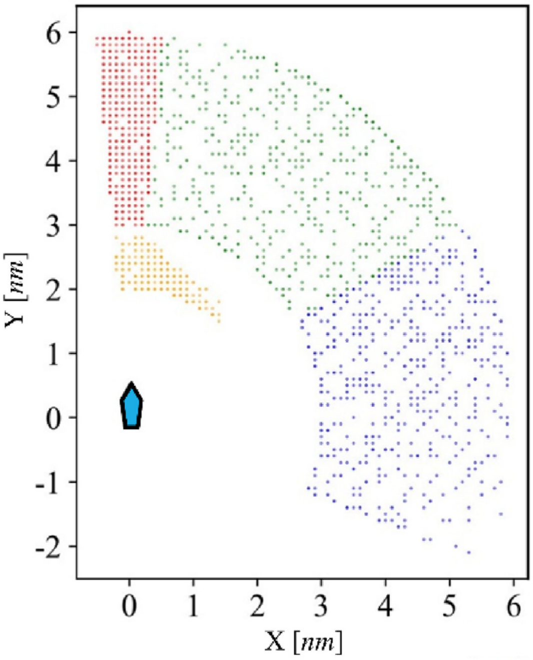


Fig. 9. Sampled data for various encountered situations.

domain but proposed that the safety distance is different by setting the center of the ship domain differently depending on the encounter situation. In addition, AIS data-based ship domain (Hansen et al., 2013) and ship domain considering speed and human factors (Wang and Chin, 2016) were studied. In this study, Goodwin and Fujii models are the most commonly used ship domains, and Szlapczynski's relatively recently studied models were selected and compared with collision avoidance models using CRI. The CRI-based A* algorithm simultaneously evaluates the CRI and path cost when performing a path search and selects a node with a CRI of 0.7 or less. On the other hand, the ship domain-based path search algorithm selects the node with the lowest path cost where the target ship does not enter the ship domain based on the defined safe distance. The paths calculated through comparison with each ship domain were evaluated in terms of path efficiency and safety. Fig. 6 shows the ship domains selected as a comparative model.

4.3. Simulation results and discussion

To confirm the validity of the proposed algorithm, paths were searched for various encounter situations. Table 4 shows the maneuvering characteristics of the own ship and the target ship used in the simulation. It is assumed the visibility between the own ship and the target ship is 6 nm based on COLREGs rule 22, and thus, the distance at which the ship must start collision avoidance is approximately 6 nm. In the situation of head-on and crossing, it started at a distance of approximately 6 nm based on the own ship, and overtaking was set at a distance of 3–4 nm. Based on the own ship, the route was calculated by setting a risk-level target ship in four situations: head-on, fine-broad crossing, converging crossing and overtaking. As shown in Fig. 7, the route calculated for each encounter situation bypasses a specific collision risk region to avoid a dangerous route. The path marked in blue is the path of the own ship and is marked as 1′–4′ according to the time order of the path finding. The black straight path is the path of the target ship and is marked 1 to 4 in the time order. Looking at the path of the own ship and the target ship, it can be seen that the path direction of the ship changes before the collision occurs. In addition, paths facing each other's port side were calculated according to the COLREGs. In the case of an overtaking situation, because there is no regulation to avoid in a specific direction, the fastest path was calculated based on a criterion with a low risk of collision. Therefore, the proposed algorithm calculates an appropriate avoidance path for four different encounters.

However, to compare the CRI-based algorithm and ship domain models, the path calculation was performed by applying each of the four models under the same conditions for the previous four situations. Fig. 8 shows the path calculation results obtained using the ship domain models and the proposed algorithm. As shown in the figure, the calculated path results depend on each collision avoidance model; in general, the path results using CRI, Goodwin, and Fujii show less path change, whereas the Szlapczynski model calculates an avoidance path that is relatively far from the target ship. Long relative distances can yield safe routes that lower the risk of collision but are not good in terms of path efficiency. On the other hand, a short relative distance can increase path efficiency, but the risk of collision can increase; therefore, it is necessary to explore economic routes under conditions where the risk of collision is not high. The CRI-based model shows path distances similar to those of the other models while maintaining a CRI value below 0.7.

Table 5 shows the calculated path length (distance [*nm*]) and maximum CRI values of each model performed in each encounter situation. As shown in the table, the CRI-based algorithm does not have a maximum CRI value exceeding 0.7 in all four cases and has a short path distance. In the case of Goodwin and Fujii models, the calculated maximum CRI values vary greatly depending on the encountered situation, and the CRI values are also estimated to be 0.7 or higher, indicating a high risk of collisions. This means that the safety areas defined by the Goodwin and Fujii models can be evaluated differently depending on the angle and speed of the encounter. Szlapczynski's ship domain defines a larger safety area than that of Goodwin and Fujii models, indicating that the path efficiency is generally evaluated as the lowest CRI among the four collision avoidance models, but the distance to the route differs by approximately 1 nm compared to the proposed model.

A COLREGs-based dataset was created to test the collision avoidance performance of the proposed algorithm in most areas where avoidance obligations for various encounter situations are required. The range of the generated data is defined as $5 < \mathrm{SOG}<24$, $|\mathrm{DCPA}|<1$, and $\mathrm{TCPA}<0.4$ by the COLREGs, which is the case where there is a risk of collision based on the operating state of the target ship and the latitude, longitude, speed, and orientation of the own ship. In addition, to simplify the simulation and uniform generation of data, Latin Hypercube Sampling (LHS) was used to sample the population sufficiently with 500 datasets for each area of head-on, fine-broad crossing, converging crossing, and overtaking. The LHS is the statistical sampling technique used to spread the sample data more evenly in a defined region than random sampling; thus, it simulates various encounter situations and operation conditions. In actual maritime operations, there are often situations where other vessels violate COLREGs or make unexpected speed and direction changes. However, in this study, it was assumed that both the own ship and target ships are unmanned ships, and encounters or collisions that occur unexpectedly were not explicitly considered. Fig. 9 shows the data sampled using the LHS method.

To verify the suitability of the CRI-applied A* algorithm using 2000 data points simulating four different encountered situations, Goodwin, Fujii, and Szlapczynski models were applied to the A* algorithm. Figs. 10 and 11 are box plots showing the maximum CRI values and the distances of the paths among the avoidance paths according to each model and situation. As shown in Fig. 10, the Goodwin and Fujii models show that the collision risk is calculated

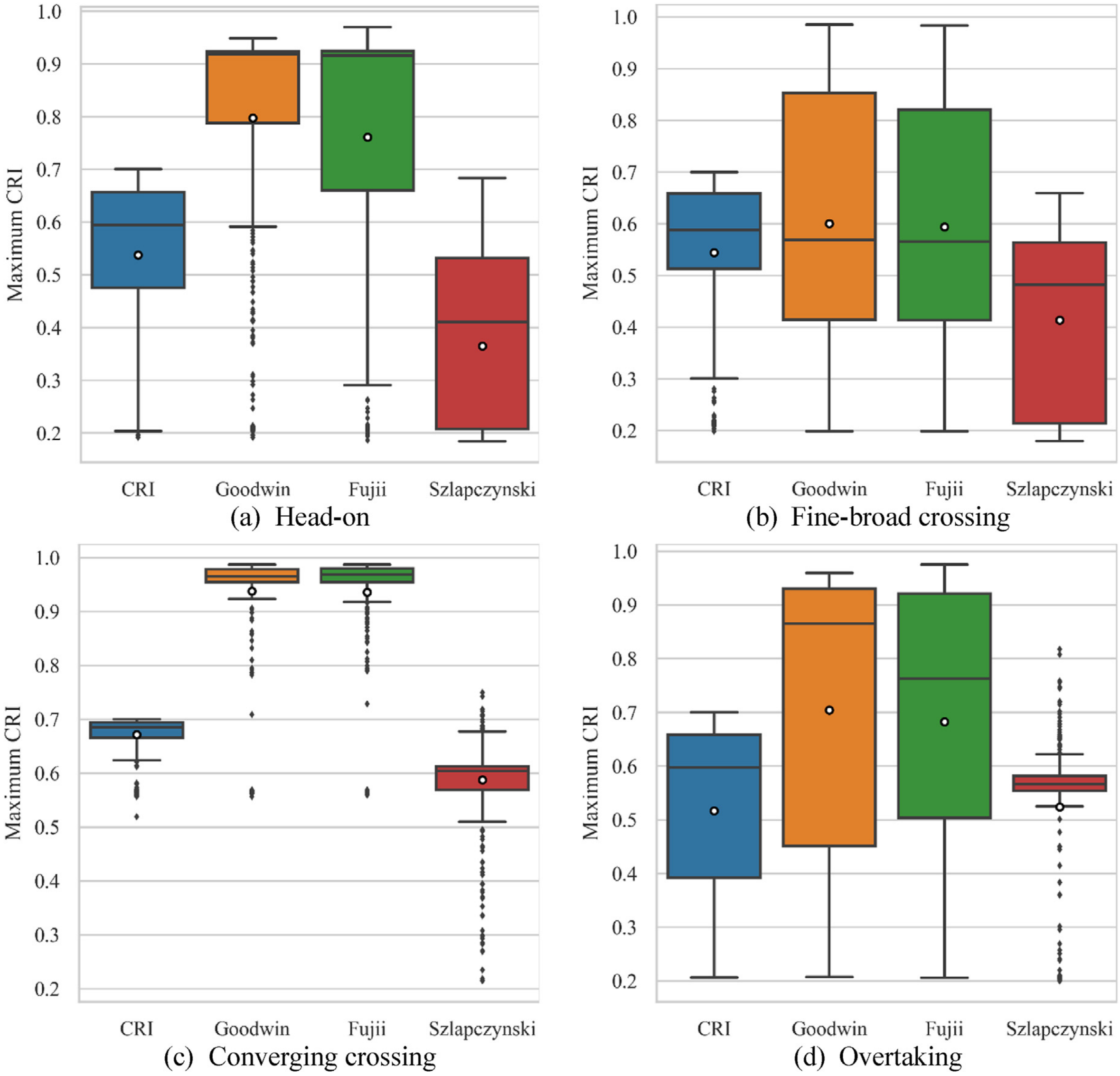


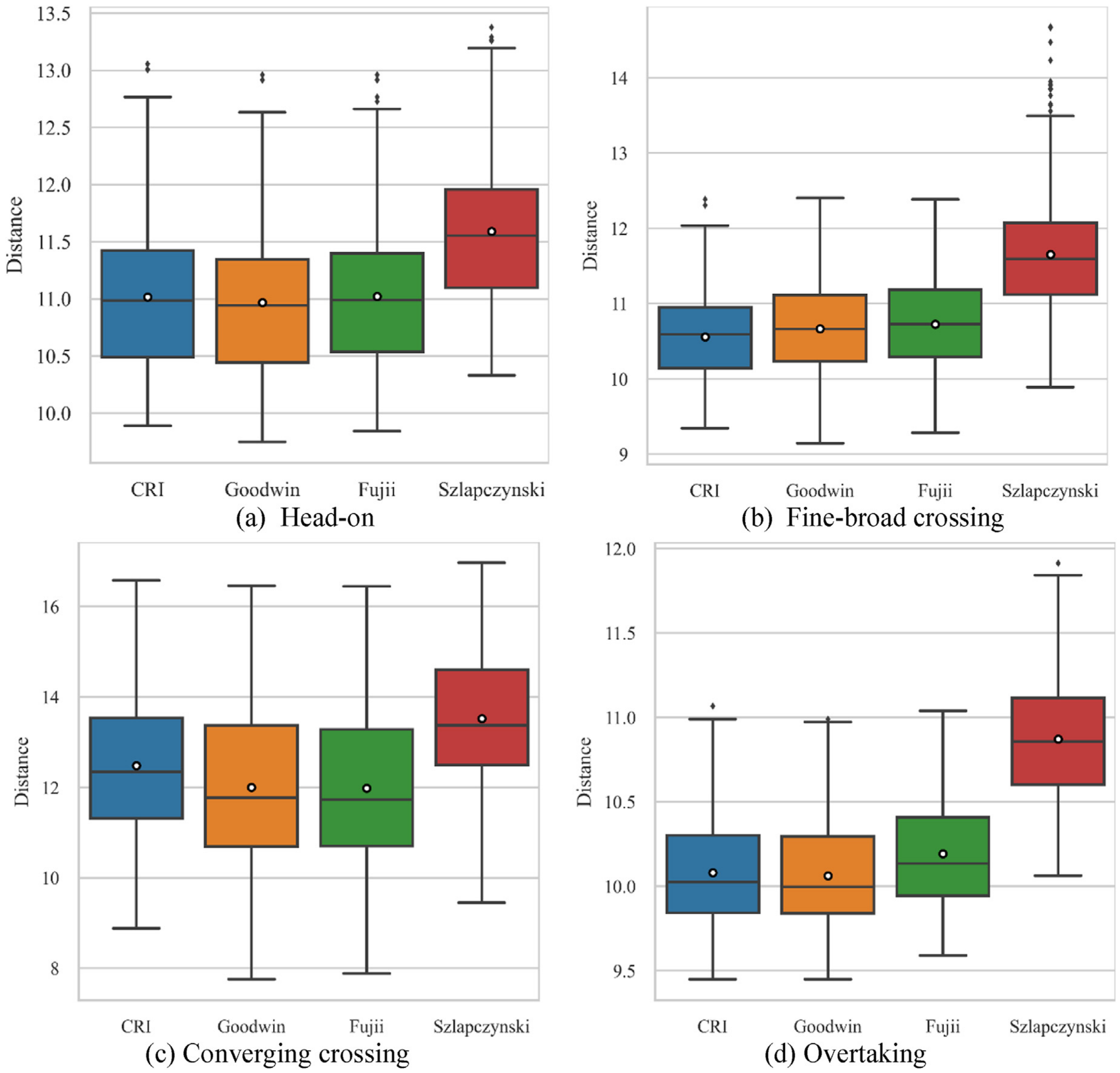
Fig. 10. Comparison of maximum CRI results.

to be high on average in all situations. This is because the two models use only distance-based ship domains and do not consider relative speed and bearings, so there is a risk that own ships may not take collision avoidance actions despite the high risk of the actual collision. On the other hand, the Szlapczynski model has a wider ship domain region than other models, making it mandatory to take action to avoid collisions even when the risk of collision is not necessarily high. Therefore, the Szlapczynski model has low CRI values on average, which increases the travel distance owing to the effect of reducing the economic feasibility of the route. However, in the case of the CRI-based algorithm, as shown in Figs. 10 and 11, the distance of the calculated path compared to those of the Goodwin and Fuji models does not exceed the risk level of CRI but also has a CRI value of 0.7 or less at the same time.

In addition, the CRI-based algorithm has a relatively low variation in maximum CRI values compared to Goodwin and Fujii models. The reason for this is that the proposed method selectively searches only for nodes with a CRI value not exceeding 0.7, while Goodwin and Fujii have more diverse CRI values because they generate various paths depending on the dynamic data of the target ships due to their small ship domain region. In the conversing crossing situation, the angle of the encounter between the own ship and the target ship is narrow, and as shown in Fig. 7(c), most own ships go around the target ship, so the variations in the maximum CRI values also decrease due to the similarity of the avoidance path of all models. However, the Szlapczynski model causes the own ship to avoid the target ship excessively, resulting in various CRI values ranging from 0.2 to 0.75. On the other hand, in the overtaking situation, the Szlapczynski model, which has a high safety distance, avoids the target ship far away, so most of the maximum CRI values are concentrated around 0.5–0.6. However, compared to the proposed algorithm, the Szlapczynski model still has an excessively high or low maximum CRI value.

The CRI-based algorithm considers various factors affecting the risk of collision when evaluating the CRI; thus, the safety of the ship can be accurately evaluated using operation data for the target ship when searching for nodes of the A*algorithm. Thus, the proposed algorithm is safe and economical compared to other ship domains.

Table 6 shows the mean values of CRI and the distance obtained from the four different models for the four encountered situations.



(a) Head-on

(b) Fine-broad crossing

(c) Converging crossing

(d) Overtaking

Fig. 11. Comparison of distance results.

Table 6
Mean values of maximum CRI and distance of four models.

Encounter	Measures	CRI-based model	Goodwin	Fujii	Szlapczynski
Head-on	Max. CRI	0.6	0.8	0.75	0.38
	Distance	11	11	11	11.5
Fine-broad crossing	Max. CRI	0.55	0.6	0.6	0.4
	Distance	10.5	10.5	10.5	11.5
Conversing crossing	Max. CRI	0.68	0.95	0.95	0.6
	Distance	12.2	12	12	14
Overtaking	Max. CRI	0.5	0.7	0.7	0.5
	Distance	10.1	10.1	10.2	10.8
Total	Max. CRI	0.567	0.760	0.743	0.472
	Distance	11.03	10.92	10.98	11.91

Distance unit: *nm*.

The proposed method has a lower average value than the Goodwin and Fujii models, with the maximum CRI value not exceeding 0.7. On the other hand, the Szlapczynski model has a low CRI value owing to its conservative ship domain. The proposed method calculates paths of generally similar distances to the Goodwin and Fujii models, while there is a large distance difference to the Szlapczynski model. This means that the own ship using the Szlapczynski model bypasses a long distance for collision avoidance, which increases the total path distance, resulting in poor path efficiency. In particular, it can be seen that in encountered situations with a high risk of collisions, such as head-on and converging crossing situations, the CRI-based algorithm shows significantly better results than other models. Therefore, the A* algorithm applied with the CRI and COLREGs ensures the safety of the route and efficient ship operation in avoiding collisions. If encountered with the other ships during actual operation, the data from the target ship and own ship can be used to efficiently calculate avoidable routes, and SOG values used in algorithms can be updated using weather and AIS data collected in real time. In addition, because the last location of the collision avoidance path is on the line of the optimized route by the A* algorithm, the own ship can follow the optimum path immediately after collision avoidance.

5. Conclusion

This study proposed an A* algorithm for collision avoidance based on CRI and COLREGs. The proposed CRI-based A* algorithm for collision avoidance finds the optimized path in consideration of global map-based economic feasibility when there is no risk of collision, but when the own ship enters the penalty zone defined based on the COLREGs, it performs a CRI-based safe path search. To this end, the global map can be subdivided into a local map, enabling a more precise and accurate collision risk assessment and improving the collision avoidance maneuvering of the own ship. In addition, a more efficient route search is possible by designating the area of the search node set according to various encountered situations during the route search. To verify the performance of the proposed method, 2000 data points for the target ship were generated using LHS for four encounter situations (head-on, fine-broad crossing, converging crossing, overtaking). The CRI-based A* algorithm ensures the safety and economic feasibility by exploring paths close to optimal in most collision risk situations compared to Goodwin, Fujii, and Szlapczynski models using conventional ship domains. The Goodwin and Fujii models, although resulting in shorter path distances, had a higher risk of collision due to their reliance on distance-based ship domains. On the other hand, the Szlapczynski model, with its wider ship domain, had lower collision risk but sacrificed path efficiency. In contrast, the proposed algorithm maintained a CRI value below 0.7, ensuring both safety and efficiency with similar path distances to the Goodwin and Fujii models. Overall, the proposed algorithm demonstrated robustness, providing a balanced solution with low variability in both CRI and distance. In this study, the collision avoidance route was explored in a situation where the speed and direction of the target ship did not change. The size and maneuvering characteristics of ships and multi-ship encounter situations were not considered, so that it could not be applicable when the own ship or target ships are large in size or sudden collision avoidance actions are taken. In future studies, we will study a more advanced route search method in consideration of the various aforementioned cases such as multiple encounter situations. In addition, an algorithm for real-time route optimization and collision avoidance will be developed so that it can be applied to actual maritime operations where inter-ship communication is possible.

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.

Acknowledgement

This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT) (No. 2020R1A5A8018822), National Innovation Cluster Program (P0006887, Build on Cloud Intelligence Platform based Marine Data) funded by the Ministry of Trade, Industry & Energy (MOTIE) and Korea Institute for Advancement of Technology (KIAT) and Korea Institute of Energy Technology Evaluation and Planning(KETEP) grant funded by the Korea government(MOTIE) (20214000000140).

References

Abebe, M., Shin, Y., Noh, Y., Lee, S., Lee, I., 2020. Machine learning approaches for ship speed prediction towards energy efficient shipping. Appl. Sci. 10, 2325.

Abebe, M., Noh, Y., Seo, C., Kim, D., Lee, I., 2021. Developing a ship collision risk index estimation model based on Dempster-Shafer theory. Appl. Ocean Res. 113, 102735.

Abebe, M., Noh, Y., Kang, Y.-J., Seo, C., Kim, D., Seo, J., 2022. Ship trajectory planning for collision avoidance using hybrid ARIMA-LSTM models. Ocean Eng. 256, 11527.

Ahn, J.H., Rhee, K.P., You, Y.J., 2012. A study on the collision avoidance of a ship using neural networks and fuzzy logic. Appl. Ocean Res. 37, 162–173.

Blaich, M., Rosenfelder, M., Schuster, M., Bittel, O., Reuter, J., 2012. Fast grid based collision avoidance for vessels using A* search algorithm. In: Proceedings of the 17th International Conference on Methods & Models in Automation &Robotics (MMAR), pp. 27–30. Miedzyzdroje, Poland.

Bukhari, A.C., Tusseyeva, I., Kim, Y.G., 2013. An intelligent real-time multi-vessel collision risk assessment system from VTS view point based on fuzzy inference system. Expert Syst. Appl. 40 (4), 1220–1230.

Chen, T., Guestrin, C., 2016. Xgboost: a scalable tree boosting system. In: Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 785–794. San Francisco, California, USA.

Chun, D.-H., Roh, M.-I., Lee, H.-W., Ha, J., Yu, D., 2021. Deep reinforcement learning-based collision avoidance for an autonomous ship. Ocean Eng. 234, 109216.

Coldwell, T.G., 1983. Marine traffic behaviour in restricted waters. J. Navig. 36, 430–444.

Davis, P.V., Dove, M.J., Stockel, C.T., 1980. A computer simulation of marine traffic using domains and arenas. J. Nav. 33 (2), 215–222.

Dechter, R., Pearl, J., 1985. Generalized best-first search strategies and the optimality of A. J. ACM 32, 505–536.

Fujii, Y., Tanaka, K., 1971. Traffic capacity. J. Navig. 24, 543–552.

Gang, L., Wang, Y., Sun, Y., Zhou, L., Zhang, M., 2016. Estimation of vessel collision risk index based on support vector machine. Adv. Mech. Eng. 8 (11), 1–10.

Gao, P., Zhou, L., Zhao, X., Shao, B., 2023. Research on ship collision avoidance path planning based on modified potential field ant colony algorithm. Ocean Coast Manag. 235, 106482.

Goodwin, E.M., 1975. A statistical study of ship domains. J. Navig. 28, 328–344.

Guo, S., Zhang, X., Du, Y., Zheng, Y., Cao, Z., 2021. Path planning of coastal ships based on optimized DQN reward function. J. Mar. Sci. Eng. 9 (2), 210.

Hansen, M.G., Jensen, T.K., Lehn-Schiøler, T., Melchild, K., Rasmussen, F.M., Ennemark, F., 2013. Empirical ship domain based on AIS data. J. Navig. 66, 931–940.

Hart, P.E., Nilsson, N.J., Raphael, B., 1968. A formal basis for the heuristic determination of minimum cost paths. IEEE Trans. Syst. Man Cybern. 4, 100–107.

He, Z., Liu, C., Chu, X., Negenborn, R.R., Wu, Q., 2022. Dynamic anti-collision A-star algorithm for multi-ship encounter situations. Appl. Ocean Res. 118, 102995.

Hu, Y., Zhang, A., Tian, W., Zhang, J., Hou, Z., 2020. Multi-ship collision avoidance decision-making based on collision risk index. J. Mar. Sci. Eng. 8, 640.

Jiang, L., An, L., Zhang, X., Wang, C., Wang, X., 2022. A human-like collision avoidance method for autonomous ship with attention-based deep reinforcement learning. Ocean Eng. 264, 112378.

Kearon, J., 1977. Computer program for collision avoidance and track keeping. In: Proceedings of the International Conference on Mathematics Aspects of Marine Traffic, pp. 229–242. London, UK.

Kim, J.H., Lee, S., Jin, E.S., 2022. Collision avoidance based on predictive probability using Kalman filter. Int. J. Nav. Archit. Ocean Eng. 14, 100438, 2092-6782.

Lee, S.-M., Roh, M.-I., Kim, K.-S., Jung, H., Park, J.J., 2018. Method for a simultaneous determination of the path and the speed for ship route planning problems. Ocean Eng. 157, 301–312.

Li, B., Pang, F.W., 2013. An approach of vessel collision risk assessment based on the D–S evidence theory. Ocean Eng. 74, 16–21.

Liu, C., Mao, Q., Chu, X., Xie, S., 2019. An improved A-star algorithm considering water current, traffic separation and berthing for vessel path planning. Appl. Sci. 9, 1057.

Li, J., Wang, H., Zhao, W., Xue, Y., 2019. Ship's trajectory planning based on improved multiobjective algorithm for collision avoidance. J. Adv. Transport. 2019, 4068783.

Moradi, M.H., Brutsche, M., Wenig, M., Wagner, U., Koch, T., 2022. Marine route optimization using reinforcement learning approach to reduce fuel consumption and consequently minimize CO2 emissions. Ocean Eng. 259, 111882.

Nguyen, M., Zhang, S., Wang, X., 2018. A novel method for risk assessment and simulation of collision avoidance for vessels based on AIS. Algorithms 11, 204.

Ning, J., Chen, H., Li, T., Li, W., Li, C., 2020. COLREGs-Compliant unmanned surface vehicles collision avoidance based on multi-objective genetic algorithm. IEEE

Access 8, 190367–190377.

Sawada, R., Sato, K., Majima, T., 2021. Automatic ship collision avoidance using deep reinforcement learning with LSTM in continuous action spaces. JMST 26, 509–524.

Shin, Y.W., Abebe, M., Noh, Y., Lee, S., Lee, I., Kim, D., Kim, K.C., 2020. Near-optimal weather routing by using improved A* algorithm. Appl. Sci. 10, 6010.

Szlapczynski, R., Krata, P., Szlapczynska, J., 2018. Ship domain applied to determining distances for collision avoidance maneuvers in give-way situations. Ocean Eng. 165, 43–54.

Tanakitkorn, K., Wilson, P.A., Turnock, S.R., Phillips, A.B., 2014. Grid-based GA path planning with improved cost function for an over-actuated hover-capable AUV. In: Proceedings of the 2014 IEEE/OES Autonomous Underwater Vehicles (AUV), pp. 1–8. Oxford, MS, USA.

Tsou, M.C., Cheng, H.C., 2013. An ant colony algorithm for efficient ship routing. Pol. Marit. Res. 20, 28–38.

Vlachos, D.S., 2004. Optimal ship routing based on wind and wave forecasts. Appl. Numer. Anal. Comput. Math. 1, 547–551.

Wang, Y., Chin, H.C., 2016. An empirically-calibrated ship domain as a safety criterion for navigation in confined waters. J. Navig. 69, 257–276.

Wang, L.P., Zhang, Z., Zhu, Q.D., Ma, S., 2020. Ship route planning based on double-cycling genetic algorithm considering ship maneuverability constraint. IEEE Access 8, 190746–190759.

Wang, H., Lang, X., Mao, W., 2021. Voyage optimization combining algorithm and dynamic programming for fuel/emissions reduction. Transport. Res. Transport Environ. 90, 102670.

Wang, J., Li, S., Li, B., Zhao, C., Cui, Y., 2023. An energy-efficient hierarchical algorithm of dynamic obstacle avoidance for unmanned surface vehicle. Int. J. Nav. Archit. Ocean Eng. 15, 100528.

Xie, S., Chu, X., Zheng, M., Liu, C., 2019. Ship predictive collision avoidance method based on an improved beetle antennae search algorithm. Ocean Eng. 192, 106542.

Xie, S., Chu, X., Zheng, M., Liu, C., 2020. A composite learning method for multi-ship collision avoidance based on reinforcement learning and inverse control. Neurocomputing 411, 375–392.

Xu, Q., 2014. Collision avoidance strategy optimization based on danger immune algorithm. Comput. Ind. Eng. 76, 268–279.

Xu, Q., Meng, X., Want, N., 2010. Intelligent evaluation system of ship management. J. Mar. Nav. Safety Sea Transp. 4, 479–482.

Xu, X., Cai, P., Ahmed, Z., Yellapu, V.S., Zhang, W., 2022. Path planning and dynamic collision avoidance algorithm under COLREGs via deep reinforcement learning. Neurocomputing 468, 181–197.

Yu, J., Liu, Z., Zhang, X., 2022. DCA-based collision avoidance path planning for marine vehicles in presence of the multi-ship encounter situation. J. Mar. Sci. Eng. 2022, 529.

Zaccone, R. Ottaviani, Figari, E., Altosole, M., 2018. Ship voyage optimization for safe and energy-efficient navigation: a dynamic programming approach. Ocean Eng. 153, 215–224.

Zaccone, R., Martelli, M., Figari, M., 2019. A COLREG-Compliant ship collision avoidance algorithm. In: Proceedings of the 2019 18th European Control Conference (ECC), pp. 2530–2535. Napoles, Italy.

Zhang, M., Montewka, J., Manderbacka, T., Kujala, P., Hirdaris, S., 2021b. A big data analytics method for the evaluation of ship – ship collision risk reflecting hydrometeorological conditions. Reliab. Eng. Syst. Saf. 213, 107674.

Zhang, G., Wang, H., Zhao, W., Guan, Z., Li, P., 2021a. Application of improved multi-objective ant colony optimization algorithm in ship weather routing. J. Ocean Univ. China 20, 45–55.

Zhao, L., Roh, M.-I., 2019. COLREGs-compliant multiship collision avoidance based on deep reinforcement learning. Ocean Eng. 191, 106436.

Zhao, Y., Li, W., Shi, P., 2016. A real-time collision avoidance learning system for unmanned surface vessels. Neurocomputing 182, 255–266.

Zhou, J., Wu, C., 2004. Construction of the collision risk factor model. J. Ningbo Univ. 170, 61–65.