Automated Graph-Based Path Planning for Navigating Maritime Traffic

Jianghui Li, *Member, IEEE*, Jinliang Liu, Chunshan Liu, *Member, IEEE*, and Peng Jiang

***Abstract*—Graph representations convert large-scale Automatic Identification System (AIS) trajectories into a routable network, enabling maritime path planning to be posed as a constrained shortest-path problem. In this work, we propose an automated maritime path planning framework that integrates high-accuracy navigation chart extraction with global AIS-derived graph-based path planning and local path refinement. The navigation chart is modeled as a directed graph network, in which nodes and edges, derived from historical AIS trajectories, denote waypoints and major navigational paths connecting them. Based on graph theory, we formulate custom cost functions to compute optimal paths, considering probability, distance, and time. Environmental factors, including water depth and electronic navigational charts, are incorporated to enhance routing accuracy and navigational safety. Additionally, a localized path refinement mechanism is implemented to automatically avoid non-navigable areas and smooth the path to optimize navigational feasibility. Using trajectory data from various types of vessels and draft configurations, the proposed method is validated against both historical averages and public platform-provided benchmarks, with a reduction of up to 8% in travel time and up to 4% in distance. The proposed method also demonstrates consistent distance reductions with preserved navigational feasibility against representative state-of-the-art route planning algorithms.**

***Index Terms*—AIS data, graph network, ship path planning, maritime navigation, path localization.**

I. Introduction

OVER the past decades, maritime traffic has become the predominant mode of global trade, with steady growth observed [1]. The safety of maritime transportation has been a critical concern, as approximately 85% maritime accidents involve navigation incidents such as collisions and groundings [2]. Consequently, research efforts have increasingly concentrated on developing safe navigation strategies [3]. By introducing e-navigation as a coordinated framework for the management of marine information both onboard and ashore, the International Maritime Organization (IMO) has underscored the importance of improving navigational safety [4]. In practice, advanced ports such as the Port of Singapore employ Vessel Traffic Management Systems (VTMS) that fuse real-time data to monitor vessel movements and mitigate safety risks [5].

The Automatic Identification System (AIS) serves as the primary source of vessel dynamic information. Its messages typically contain identification (MMSI number), GPS or BeiDou coordinates (latitude, longitude), speed over ground and course over ground, destination and additional vessel and voyage details, which is widely applied in trajectory prediction [6], [7], [8], [9], behavior analysis [10], path optimization [11], [12] and anomaly detection [13], [14] to enhance maritime domain awareness (MDA). However, fully exploiting AIS for maritime navigation remains challenging: AIS data are often noisy, and their spatial density and quality vary across regions [15]. This highlights the necessity for an automated system capable of analyzing data and charting courses for vessels. Moreover, because AIS was originally designed for identification and tracking, it lacks metadata (e.g., quality, reliability, uncertainty), which further complicates its direct use for path planning.

Vessel path planning has been extensively investigated from different perspectives [16], [17], yet most existing studies either rely on real-time navigation information or assume simplified, scenario-specific environments [18], [19], with limited exploitation of large-scale historical traffic data. Consequently, recurrent navigation patterns and long-term risk characteristics embedded in historical AIS trajectories remain underutilized. These limitations motivate methods that explicitly incorporate historical vessel behavior into the path planning process to improve route safety and efficiency at the level of maritime transportation systems.

This paper presents a data-driven maritime path planning framework that fuses large-scale historical AIS trajectories with environmental constraints, such as water depth and electronic navigation charts (ENCs). Historical vessel tracks are preprocessed and clustered to extract main traffic lanes and representative waypoints, which are then abstracted into a directed maritime network where nodes denote waypoints and edges denote feasible sailing legs shaped by regulations and operational practice. Environmental constraints are used to remove edges crossing non-navigable waters and

Received 24 December 2024; revised 25 April 2025, 15 November 2025, and 2 February 2026; accepted 22 March 2026. This work was supported in part by the National Major Science and Technology Projects of China under Grant 2024ZD1406604, in part by the Natural Science Foundation of Xiamen Municipality under Grant 3502Z202373006, and in part by the Science and Technology Projects of Fujian Province under Grant 2025H0001. The Associate Editor for this article was H. L. Vu. *(Corresponding author: Chunshan Liu.)*

Jianghui Li and Jinliang Liu are with the State Key Laboratory of Marine Environmental Science and the Key Laboratory of Underwater Acoustic Communication and Marine Information Technology, Ministry of Education, Xiamen University, Xiamen 361102, China (e-mail: jli@xmu.edu.cn).

Chunshan Liu and Peng Jiang are with the School of Communication Engineering, Hangzhou Dianzi University, Hangzhou 310018, China (e-mail: chunshan.liu@hdu.edu.cn).

Digital Object Identifier 10.1109/TITS.2026.3678113

to assign safety-related penalties. Path planning between an origin–destination pair is formulated as a graph search problem with three alternative cost functions that encode different planning priorities. Localized adjustments are applied to refine the path identified from the graph network, so as to avoid non-navigable areas and ensure smooth navigation.

The main contributions of this work are summarized as follows:

- We propose a graph-construction scheme that transforms raw AIS data streams into a waypoint–edge maritime network to capture underlying navigation patterns.
- Within this graph-search framework, we design an automated path-planning method equipped with three alternative cost functions, i.e., *probability-priority*, *distance-priority*, and *time-priority* based cost functions, to accommodate different planning pReferences.
- We develop an automated local path refinement algorithm, combining non-navigable-region avoidance with route smoothing, which accelerates graph-based path planning and improves routes for both long- and short-distance traffic.

The rest of this paper is organized as follows. Sec. II provides an overview of the related work in maritime path planning. Sec. III elaborates on the construction of the AIS-derived graph network and the global path planning scheme. Local path refinement strategies are discussed in Sec. IV. Sec. V presents the dataset and the experimental results of the proposed approach. Finally, conclusions and future perspectives are summarized in Sec. VI.

II. Related Work

Driven by advancements in artificial intelligence, path planning has been extensively studied, with applications across fields such as robotics, autonomous vehicles, and uncrewed aerial vehicles [20], [21]. In maritime settings, path planning has been effectively applied in Autonomous Underwater Vehicles (AUVs), Autonomous Surface Vehicles (ASVs), and Uncrewed Surface Vehicles (USVs) [22], [23]. However, its application to ships remains challenging due to complex environmental dynamics, navigation safety, and regulatory compliance.

Depending on whether complete environmental information is available for modeling the entire environment, path planning can be generally classified into two categories: global and local path planning [24]. Global path planning aims for global optimality, while local path planning prioritizes flexibility but may result in only locally optimal paths. In practical scenarios, systems often integrate global planning for overall optimization with local planning/refinement for obstacle avoidance [25]. The overall workflow of path planning includes three primary steps: environmental modeling, pattern extraction, and path planning.

A. Environmental Modeling

Environmental modeling aims to convert real-world environments into processable formats for autonomous systems. Typical modeling methods include geometric modeling approaches, such as visibility graphs and Voronoi diagrams [26], [27], [28], cell decomposition methods, such as regular grids [29], and probabilistic roadmap (PRM) techniques [24]. Visibility graphs and Voronoi diagrams represent routes by connecting visible vertices or outlining equidistant paths between obstacles, respectively, which facilitates optimal pathfinding results but can be computationally complex in intricate environments. The cell decomposition method replaces original obstacles with decomposed grids, transforming obstacles into regular polygonal shapes, thus simplifying complex environments and making path planning more computationally manageable. PRM samples nodes in high-dimensional configuration spaces, constructing probabilistic graphs where nodes represent collision-free configurations and edges denote feasible connections [30], making PRM particularly effective for high-dimensional spaces.

B. Vessel Motion Pattern Extraction

Vessel motion pattern and route extraction offer critical insights into complex maritime traffic and significantly support path planning and safety analysis. To identify motion patterns from raw AIS data, techniques such as machine learning [31], [32], and statistical methods [33] are commonly employed. Vector-based methods have been developed to extract network nodes (e.g., waypoints, ports, and other fixed locations) and edges (route segments) from navigation paths to construct a maritime traffic network that captures vessel motion patterns [34]. A low-dimensional maritime route representation method based on graph theory has been proposed to capture the spatial and temporal characteristics of vessel movement, enhancing the simplicity and efficiency of vessel motion pattern extraction [35]. This graph-based structure represents connections between waypoints as network edges, enabling a broader perspective on route connectivity and traffic patterns. Vector-based methods are highly effective in structured, high-traffic areas due to their reliance on clear boundaries and extracted traffic patterns. However, they are hard to situate in less-regulated regions, where small obstacles and unpredictable conditions may reduce their adaptability and accuracy [36].

C. Path Planning Algorithms

Specific optimization criteria, such as minimizing distance, maximizing safety and efficiency, or time-optimal, are defined to guide the path planning process after environmental modeling. To achieve these goals, a path-searching algorithm is employed to identify a collision-free route from the starting point to the destination, adhering to the established criteria to ensure an efficient and safe voyage. These algorithms can generally be categorized into three kinds: graph-based methods, sampling-based methods, and artificial intelligence-based methods.

1) Graph-Based Methods: Graph-based algorithms are widely employed to identify optimal paths within graph or grid structures. These methods represent the environment as nodes and edges, where weights–such as distance, time, or energy costs–are assigned to optimize specific path-planning

objectives. Dijkstra's algorithm [37] is an effective single-source shortest path algorithm that determines the global shortest route by expanding paths from the departure node and exploring edges to unvisited nodes in a weighted, directed, or undirected graph system [38]. A constrained Dijkstra method is proposed for optimal path planning of USVs in obstacle-rich environments [39], using a circular boundary around the USV as a safety distance constraint to ensure the path is both optimal and collision-free. The A* algorithm, developed to enhance Dijkstra's method, incorporates a heuristic cost function to accelerate the search process [40]. Similarly, a constrained A* algorithm is designed to account for environmental factors, thereby generating safer waypoints and reducing search time [41]. While graph-based algorithms are effective for path planning and can accommodate constraints such as traversal time and distance, their applications in maritime navigation remain relatively limited.

2) Sampling-Based Methods: Sampling-based motion planning methods, such as the Probabilistic Roadmap (PRM) and Rapidly Exploring Random Tree (RRT), are widely used for their probabilistic completeness and scalability in high-dimensional, cluttered environments [42]. These algorithms explore the configuration space via random sampling and incrementally build collision-free paths without requiring exhaustive discretization. For autonomous vessels, RRT*-based schemes have been applied to generate trajectories that satisfy the Convention on the International Regulations for Preventing Collisions at Sea (COLREGs) while preserving dynamic feasibility [43]. Owing to its inherent Voronoi bias, RRT can rapidly expand toward unexplored regions, which is advantageous for kinodynamic planning in state spaces with differential constraints [44], [45].

Despite these advantages, classical RRT exhibits several well-known limitations. First, its reliance on extensive random sampling leads to substantial computational cost in high-dimensional state spaces [46], [47]. Second, RRT often struggles in narrow passages or scenarios requiring precise maneuvers, where sparse samples can hinder the discovery of high-quality feasible trajectories and may yield suboptimal or even infeasible paths [45], [48]. Third, its performance is sensitive to parameter choices, including sampling density, step size, and the maximum number of iterations and nodes [47]. Although numerous variants have been proposed to improve efficiency and robustness in specific applications [49], [50], these approaches still inherit the fundamental sampling and parameter-sensitivity issues of the original RRT framework.

3) AI-Based Methods: A prominent category within AI-based methods is bio-inspired algorithms, which include Ant Colony Optimization (ACO) and Genetic Algorithms (GA). The ACO algorithm employs pheromone accumulation produced by ants to guide pathfinding [51]. For path planning and obstacle avoidance, ACO can be integrated with strategies such as the Artificial Potential Field (APF) to achieve global and local path planning in complex environments [52], [53]. As an adaptive search method, GA excels in optimization and flexibility, making it a valuable tool for enhancing the safety and efficiency of maritime path planning and collision avoidance [54]. While the ACO algorithm demonstrates robust search capabilities, it is hindered by premature convergence issues and low convergence rates. Likewise, heuristic algorithms such as GA rely on randomly generated initial values, which can result in considerable variability in outcomes [55].

Reinforcement Learning develops optimal strategies through interactions with the environment. In path planning, reinforcement learning algorithms iteratively refine strategies to find the optimal path, proving particularly effective for real-time planning in dynamic settings. A collision avoidance method based on Deep Reinforcement Learning (DRL) is proposed, which monitors vessel positions, speeds, and directions in real-time to assess collision risks and generate avoidance paths that comply with COLREGs [56].

Overall, although existing maritime path planning methods have employed various optimization techniques, they usually overlook the valuable navigational patterns embedded in historical AIS trajectory data. Graph theory provides several advantages for path planning, including efficient pathfinding algorithms (e.g., Dijkstra's and A*), adaptability to diverse optimization objectives (e.g., time, distance, or risk), and rigorous mathematical guarantees of optimality [57], yet its application to maritime navigation remains limited. This is primarily due to the absence of structured road networks and the inherently unstructured nature of open sea environments. These challenges highlight the need for a framework capable of extracting and formalizing vessel movement patterns from historical data, thereby enabling the construction of a navigable maritime route network. In the following section, we present a data-driven approach that addresses this gap by integrating trajectory-based learning with probabilistic graph modeling.

III. Graph Network Construction

This section outlines the construction of a directed graph from spatiotemporal trajectory points, denoted as {t, *coord*}, where t represents the timestamp and *coord* indicates the geographic coordinates of the vessel. The graph is constructed by extracting waypoints at shipping lane crossings and their connections from historical ship trajectories, which are classified as nodes and edges. In this directed graph, a junction accessible from another may not allow reverse access. To quantify pathfinding results, three automated path-planning methods are introduced: probability-priority, time-priority, and distance-priority, each utilizing probability-based, time-based, and distance-based traversal costs, respectively. After obtaining the traversal costs for all edges, a pathfinding algorithm is employed to determine the optimal path within the maritime environment.

A. Trajectory Generalization and Aggregation

Vessels typically travel in straight lines and maintain consistent navigation statuses for extended periods, with occasional changes in course or speed. Thus, identifying these irregular events–such as stops and turns–is critical for analyzing maritime traffic patterns. To capture vessel motion patterns, a three-step approach is proposed: significant point extraction, point clustering, and flow generation.

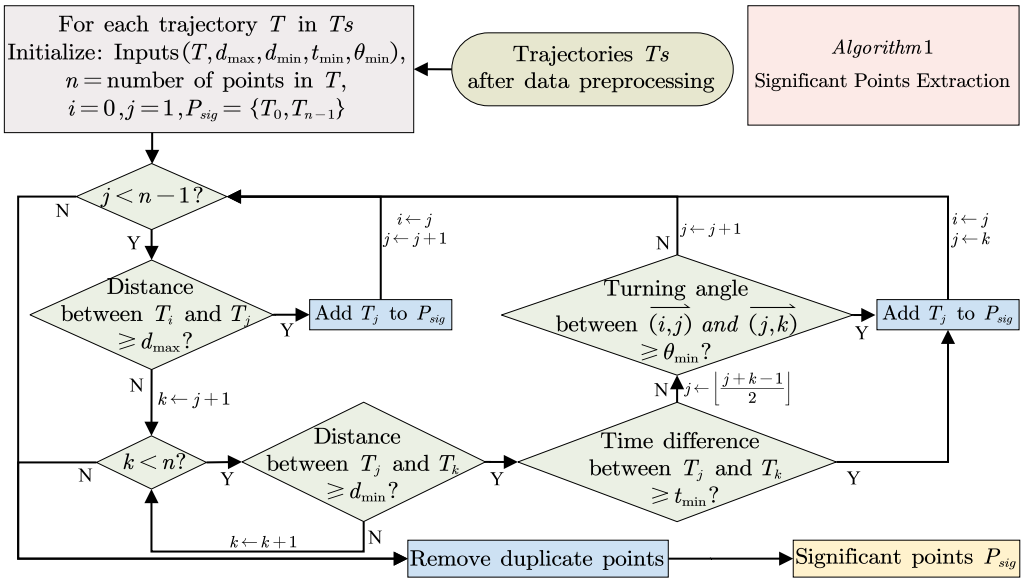


Fig. 1. Flowchart for extracting significant points from AIS tracks. Besides trajectories' start and end points, P_{sig} includes three types of significant points: significant distance points P_{sd} where the distance from the previous point exceeds $d_{\max}$, significant turning points P_{st} where the turning angle exceeds $\theta_{\min}$, and significant stopping points P_{ss} where the stop duration within a certain area exceeds $t_{\min}$.

First, significant points are extracted, which include turning points P_{st}, stopping points P_{ss}, and both the starting and ending points of trajectories. Additionally, representative points from long, straight segments are included to capture trajectory details comprehensively. To ensure even distribution, additional points P_{sd} are selected based on a maximum distance threshold ($d_{\max}$) between adjacent key points. Significant stops are identified using a minimum distance threshold ($d_{\min}$) and a minimum docking duration ($t_{\min}$). Specifically, points within $d_{\min}$ are considered as the same location, while stops exceeding $t_{\min}$ are flagged as significant. Significant turning points are detected using a minimum turning angle ($\theta_{\min}$). The details of the significant point extraction algorithm are shown in Fig. 1.

To further generalize these key points, a density-based approach inspired by grid-based DBSCAN (Density-Based Spatial Clustering of Applications with Noise) is applied. The algorithm organizes points into convex clusters to ensure an even distribution of centroids across the study area. Cluster centroids are calculated by averaging the coordinates of all points within each cluster.

As demonstrated in Fig. 2, the point clustering algorithm consists of two primary steps: grid clustering and centroid recalculation. First, the area is divided into grids of a predefined size (in this case, $d_{\max}$), and all significant points are assigned to grids based on their spatial locations. Each point is checked against centroids in its eight neighboring grid cells. If centroids exist in these surrounding grids, the point is assigned to the nearest one, and the centroid's coordinates are updated. If no nearby centroid is found, a new cluster is created with the current point as the initial centroid.

In the centroid recalculation step, centroids are fixed, and the cluster composition is refined. Points are reassigned to clusters based on proximity to the fixed centroids, and the number of points in each cluster is recorded as the weight of the corresponding centroid. The pseudocode can be found in the Supplementary Material.

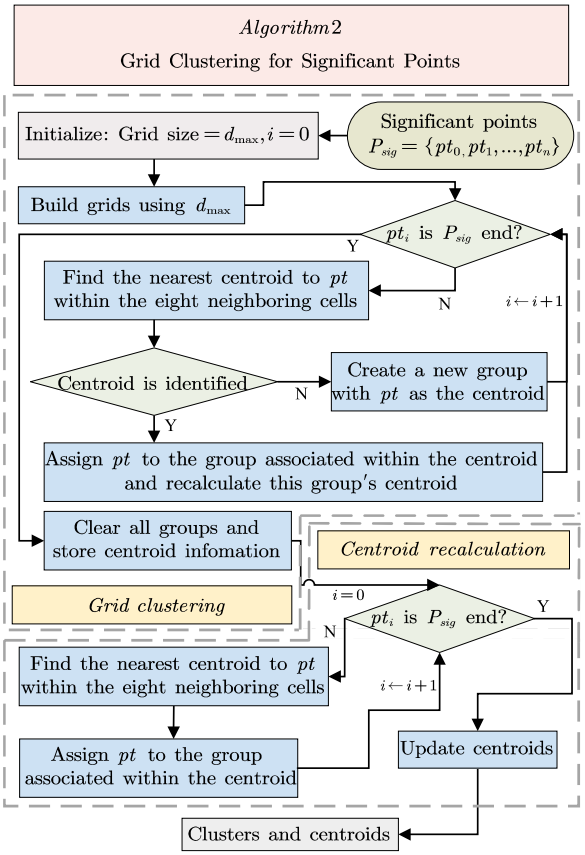


Fig. 2. Flowchart for clustering significant points using grids: first, map each point to a predefined grid and subsequently update the groups and the coordinates of the centroids. Continue this process until all points in P_{sig} are processed and the centroids stabilize. Then, regroup all points based on the updated centroids to obtain the final clusters and centroids.

The centroids then serve as waypoints in the maritime route network. Connections between waypoints are established by analyzing trajectory segments. Each trajectory is represented as a sequence of points linked in chronological order, which is mapped to a series of centroids. The nearest waypoint for each trajectory point is identified, and transitions between waypoints are recorded. For every new waypoint visited, the algorithm logs the current and previous waypoints and increments a passage count, a measure of traffic density initialized to zero. If the traversal order between two waypoints reverses, it is treated as a distinct path. By aggregating the recorded passages, a network that represents movement dynamics across the study area is constructed.

B. Cost Calculation

In this section, we establish a cost framework on the graph network and propose three distinct cost calculation methods: probability-based cost, distance-based cost, and time-based cost. Since real shipping lanes are directed, the traversal cost from point A to point B differs from that in the reverse direction. After calculating traversal costs, Dijkstra's algorithm is applied for pathfinding.

1) Probability-Based Cost: Vessels have varying probabilities of choosing different routes at junctions in real-world scenarios. By analyzing recorded passage counts, selection ratios for alternative paths at junctions can be computed. With a sufficiently large dataset, these ratios approximate the probabilities of route choice. The probability of transitioning between two nodes, known as the flow rate, is determined by dividing the passage frequency of the current edge by the total passage frequency of all outgoing edges from the originating node.

To identify the most probable path in a graph, this problem is reformulated as a shortest path problem using Dijkstra's algorithm. We apply the NetworkX library's implementation of Dijkstra's algorithm [58] to find the shortest path by iteratively selecting nodes with the lowest cost until reaching the destination node. Since the pathfinding weights are represented by probabilities (ranging from [0,1]), these probabilities are converted into cost values to fit the requirements of Dijkstra's algorithm.

In a graph G, let the probability between two nodes (u, v) be denoted as $w(u, v)$. We define $w_0(u, v) = -\log(w(u, v))$. The shortest path from node A to node B is determined by finding the shortest path in the graph using w_0 as the weight function. For a path $v_1 \to v_2 \to \cdots \to v_n$, the probability is given by $w(v_1, v_2) \times w(v_2, v_3) \times \cdots \times w(v_{n-1}, v_n)$. Using w_0 as the weight, the cost of this path is:

$$\begin{aligned} d(v_1, v_n) &= w_0(v_1, v_2) + w_0(v_2, v_3) + \cdots + w_0(v_{n-1}, v_n) \\ &= [-\log(w(v_1, v_2))] + [-\log(w(v_2, v_3))] + \cdots \\ &\quad + [-\log(w(v_{n-1}, v_n))] \\ &= -1 \times [\log(w(v_1, v_2)) + \log(w(v_2, v_3)) + \cdots \\ &\quad + \log(w(v_{n-1}, v_n))] \\ &= -1 \times \log(w(v_1, v_2) \times w(v_2, v_3) \times \cdots \\ &\quad \times w(v_{n-1}, v_n)). \end{aligned} \tag{1}$$

Eq. (1) provides the minimal path cost from A to B. Consequently, this path minimizes the value of:

$$\begin{aligned} s(A, B) = -1 \times \log(w(A, v_2) \times w(v_2, v_3) \times \\ \cdots \times w(v_{n-1}, B)) \end{aligned} \tag{2}$$

Since the logarithm function is monotonic, the minimal value of $-1 \times w(A, v_2) \times w(v_2, v_3) \times \cdots \times w(v_{n-1}, B)$ corresponds to the maximum value of $w(A, v_2) \times w(v_2, v_3) \times \cdots \times w(v_{n-1}, B)$. Thus, we obtain the path of the highest probability in the graph network between two vertices (u, v), which has the lowest traversal cost of probability among all possible paths. After identifying the nearest waypoints to both the current vessel location and the destination, the final path is constructed by connecting the current location to its nearest waypoint, following the highest probability path between nodes, and ultimately reaching the destination.

2) Distance-Based Cost: For distance-based pathfinding, the geographic distance cost between waypoints is calculated using the Haversine formula, which provides the great circle distance between two points on the Earth's surface. The formula is given by:

$$\begin{aligned} d = 2 \\ \cdot R \cdot \operatorname{atan2}\Bigg(\sqrt{\sin^2\left(\frac{\Delta\phi}{2}\right) + \cos(\phi_1) \cdot \cos(\phi_2) \cdot \sin^2\left(\frac{\Delta\lambda}{2}\right)}, \\ \sqrt{1 - \left[\sin^2\left(\frac{\Delta\phi}{2}\right) + \cos(\phi_1) \cdot \cos(\phi_2) \cdot \sin^2\left(\frac{\Delta\lambda}{2}\right)\right]} \Bigg) \end{aligned} \tag{3}$$

where ϕ_1 and ϕ_2 are the latitudes of the two points in radians, $\Delta\phi$ is the difference in latitude, $\Delta\lambda$ is the longitude difference, R is Earth's radius (approximately 6,371 km), and d is the distance in kilometers between the points.

Using Dijkstra's algorithm with this distance metric enables effective determination of the shortest path between waypoints based on physical distance.

3) Time-Based Cost: By adopting the distance information obtained in Sec. III-B.2, travel time can be estimated given a required traversal speed. Due to the lack of direct speed information from the aggregated network, we exploit a location-based speed estimation method that approximates vessel speeds by analyzing trajectory segments with similar geographic locations and orientations relative to the corresponding network edges.

To identify these similar trajectory segments, two circular regions are generated around the endpoints of each edge in the network, each with a radius equal to half the edge length. These circles serve as spatial filters for locating relevant trajectory points. Trajectories containing points within both regions are retained, ensuring spatial coverage near both endpoints. For each qualifying trajectory, the points closest to each endpoint are identified and connected in temporal order to form trajectory segments. Repeating this process across all trajectories yields a set of candidate trajectory segments.

A direction-matching criterion is then applied to further refine these segments. Specifically, if the angle between a trajectory segment's direction and the edge direction exceeds a specified value (30° in this case), the segment is excluded as it

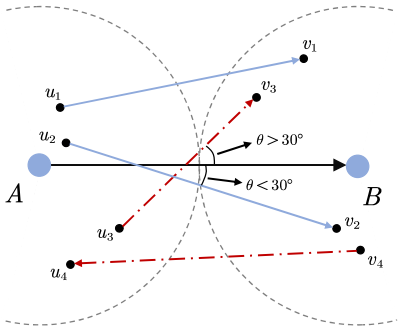


Fig. 3. Process for selecting similar trajectory segments for speed estimation: First, circular regions centered at nodes A and B are defined to facilitate comparison. Next, trajectories intersecting both regions are identified. The closest points to A and B on these trajectories are then selected to form segments that deviate by no more than 30° from the direction of edge (A, B). The blue segments fulfill the criteria for speed estimation, while the red dashed segments do not, highlighting the importance of directionality for accurate speed assessments.

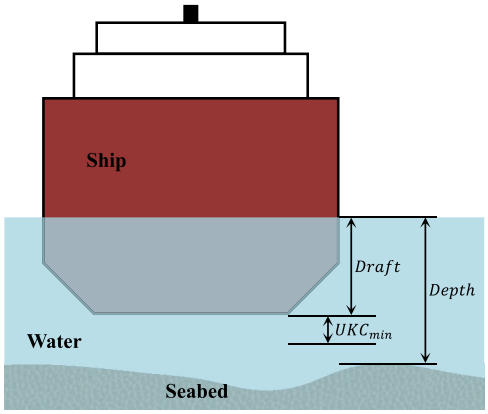


Fig. 4. Minimum Under-Keel Clearance requirements for safe navigation.

is not adequately aligned with the edge. This criterion ensures alignment between the estimated speed and the direction of the edge, as illustrated in Fig. 3.

The average speed for each edge is computed based on the remaining trajectory segments. Travel time for each edge is subsequently obtained by dividing the edge distance by the estimated speed, and the total traversal time for a path is calculated as the sum of the travel times across all constituent edges. This approach produces realistic travel time estimates consistent with typical vessel speeds, providing a practical basis for time-based pathfinding using Dijkstra's algorithm.

C. Environmental Constraints

In practical ship navigation, environmental constraints—such as water depth, which limits a vessel's allowable draft—must be carefully considered. Moreover, electronic navigational charts (ENCs) offer essential route planning information, including restricted areas and traffic separation schemes. These factors are fundamental to our approach for generating safe and efficient navigation paths.

1) Water Depth: Under Keel Clearance (UKC) is a critical parameter for ensuring navigational safety by maintaining a minimum vertical distance between the keel of a vessel and the seabed, as illustrated in Fig. 4. It is mathematically defined as [59]:

$$UKC = Depth - Draft \geq UKC_{\min} \tag{4}$$

where $Depth$ represents the water depth, $Draft$ denotes ship draught (draft), and $UKC_{\min}$ is the minimum required clearance. The value of $UKC_{\min}$ is influenced by several factors, including maximum vessel draught, navigation speed, and seabed composition. Under stable environmental conditions, $UKC_{\min}$ is often approximated as a constant proportion k of the ship's maximum draught [60]. Thus, the safe water depth (SWD) can be derived as:

$$\begin{aligned} SWD &= Draft_{\max} + UKC_{\min} \\ &= Draft_{\max} \cdot (1 + k) \end{aligned} \tag{5}$$

To construct a digital depth model, bathymetric data from, e.g., the European Marine Observation and Data Network (EMODnet) [61], can be utilized. For vessels with known draft parameters, a navigability mask layer is generated through raster-based geospatial analysis to distinguish navigable and non-navigable zones, as illustrated in Fig. 5. The mask is subsequently applied to eliminate graph edges that do not satisfy the minimum depth requirements for safe navigation.

2) Electronic Navigational Charts: ENCs present essential sailing rules through clear symbols that highlight navigational constraints [62]. Table I summarizes the three types of restricted areas considered in this study: danger areas, traffic separation schemes (TSS), and prohibited or restricted areas.

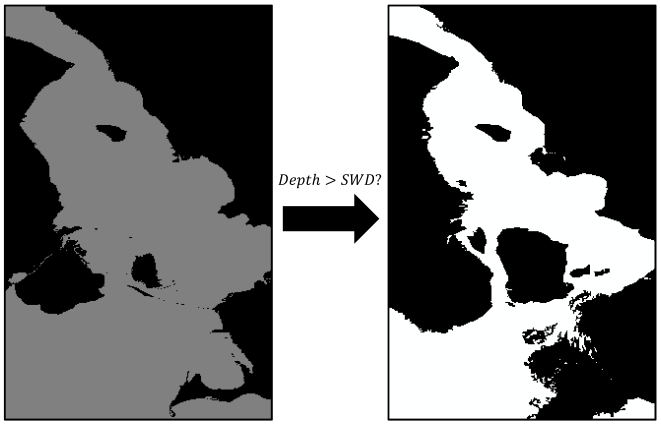


Fig. 5. Workflow for generating a binary navigability mask from EMODnet bathymetric data. (Left) High-resolution water depth raster derived from EMODnet. (Right) Binarized navigability mask classifying regions as navigable (white) or non-navigable (black) through thresholding at SWD ($SWD = Draft_{\max} + UKC_{\min}$). The arrow indicates the geospatial processing pipeline.

TABLE I
THREE TYPES OF RESTRICTED AREAS DEFINED IN THE ELECTRONIC NAVIGATIONAL CHART

Symbols of S-57 ENC	Symbols used here	Explanation
		Transparent danger highlight for mariner's use
		Traffic separation scheme area
		Prohibited or restricted area

To ensure navigational safety and regulatory compliance, the path planning process must strictly adhere to these charted constraints. Vessels are required to avoid danger areas and prohibited zones to reduce risks such as grounding or collision. Within TSS regions, crossings should be made at angles as close to 90 degrees as possible, following international maritime regulations, to minimize interference with main traffic flows.

These constraints are integrated into the network model by removing edges intersecting prohibited zones and adjusting paths near danger areas and TSS boundaries based on prescribed safety margins. This refinement ensures that the resulting routes are both feasible and compliant with navigational safety standards.

Following trajectory clustering, cost calculation, and graph optimization under environmental constraints, Dijkstra's algorithm is used to present the path with minimum cost. To further reduce unnecessary changes in path direction, waypoints are filtered according to their associated weights. Specifically, waypoints located in open-sea regions with weights below a predefined threshold are removed from the planned path. This threshold is determined in accordance with the overall distribution and weight characteristics of waypoints in the network, ensuring alignment with typical vessel navigation patterns, as discussed in Sec. V. If the removal of a waypoint results in the path entering a non-navigable area, the previous configuration is restored.

IV. Local Path Refinement

The graph network construction and global path search processes do not account for the impact of landmasses and islands on path planning, which may lead to routes that extend through non-navigable regions. To address these issues at a finer spatial scale, we introduce a local path refinement module that combines a non-navigable-region avoidance algorithm with path smoothing.

When a planned path intersects a non-navigable region, the avoidance algorithm initiates to generate alternative routes. A buffer zone with a width of d_{buffer} is established along the boundary of non-navigable regions to maintain a safe clearance between the path and obstacles. Any gaps within the buffer are closed to create a continuous barrier. The value of the clearance is determined by environmental complexity or regulatory requirements; in the study, d_{buffer} is set to 200 meters to ensure safe navigation.

To create a bypass route, the perpendicular bisector of the intersecting segment between the original path and the buffered obstacles is used to locate new intermediate nodes, with the search direction oriented toward the closest obstacle boundary. The search occurs within a preset distance, d_{search}, of 200 meters. The process is repeated until the path no longer intersects any buffer zones. To avoid excessive deviation from the original route, the number of search attempts is limited to five, with a cumulative search distance of up to 1000 meters. If no suitable node is identified within the maximum attempts, the last identified intermediate node is retained to create two new edges, and the search process is reinitiated. The procedure is illustrated in Fig. 6.

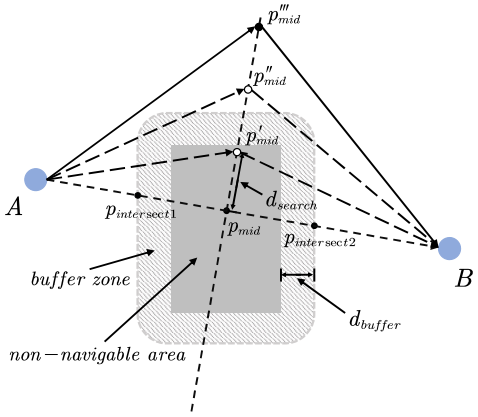


Fig. 6. Process for identifying a new intermediate point in the intersection segment between the edge from the graph network and the non-navigable area: First, a buffer zone of d_{buffer} is established along the edge of the non-navigable area. Second, the intersection points between edge (A, B) and the buffer are identified, and the perpendicular bisector is determined. Third, new intermediate nodes are searched along the bisector within a predefined searching distance d_{search}, directing them closer to the boundary. This ensures connections between edge nodes and the intermediate node avoid the non-navigable zone and maintain a safe distance.

After applying the avoidance algorithm, the path undergoes a smoothing process to reduce sharp turns and enhance navigational feasibility. Specifically, B-spline interpolation is applied to create a smoother curve. However, long segments or sharp turns may cause the interpolation to produce over-smoothed curves at junctions, deviating from the intended route. To mitigate this, additional control points are inserted within extended edges before interpolation, shortening each interval to better conform to the original route. The traversal distance and time are then recalculated to ensure the optimized route meets all navigational requirements.

V. Experimental Results

A. Dataset

In this paper, we utilize AIS data provided by the Danish Maritime Authority (DMA) [63]. While the explicit information in AIS data may not fully satisfy the requirements of path planning, additional latent information is extracted through data filtering and other preprocessing techniques to better capture vessel behavior.

Initially, data with null values or logically inconsistent value ranges are excluded. Next, AIS data are classified by vessel type, segregating trajectory data for different categories. Specific vessel types are then selected for trajectory extraction. Thereafter, trajectories are down-sampled via linear interpolation at fixed time intervals (e.g., every 5 minutes) to balance data precision and volume. Data resolution is reduced to 0.01° for longitude and latitude, 1 knot for speed over ground (SOG), and 1° for course over ground (COG) to further reduce data volume. The resulting trajectories T, grouped by MMSI number with associated temporal attributes $\{t, coord\}$, are obtained and stored.

For this analysis, the region of interest (ROI) encompasses the waters surrounding Denmark, ranging from (53° N, 4° E) to (59° N, 17° E), and is used to construct the graph network.

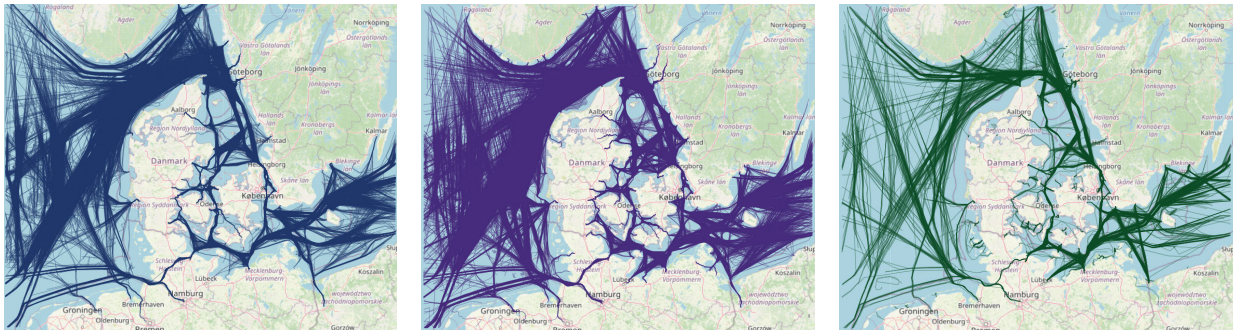


Fig. 7. Vessel trajectories extracted from AIS messages: (a) Tankers (January 1–December 31, 2023). (b) Cargo ships (January 1–June 30, 2023). (c) Passenger ships (January 1–December 31, 2023).

AIS data from January 1 to December 31, 2023, is utilized for tankers and passenger ships, while data from January 1 to June 30, 2023, is employed for cargo ships. The dataset includes over 359.2 million AIS messages from tankers, more than 410.8 million from cargo ships, and over 491.8 million from passenger ships. In total, 9,750 trajectories for tankers, 27,541 for cargo ships, and 2,621 for passenger ships are extracted for further analysis, as shown in Fig. 7.

The geographic data used in this study are sourced from OpenStreetMap [64], which provides critical boundary information such as coastlines, land-water borders, administrative boundaries, and base maps for visualization. Since we focus exclusively on maritime routes and sea traffic conditions, data related to inland waterways is excluded. Importantly, the Kiel Canal–one of the busiest artificial waterways globally–connects the North Sea and the Baltic Sea directly, substantially reducing transit distance and time for maritime transport. However, since it is not depicted as a land boundary in standard geographic layers, the data are specifically processed and integrated into the graph network to reflect its strategic importance in maritime navigation.

Based on ENCs information and official publications from the DMA [65], three primary routes connect the Baltic Sea to the Kattegat: the Sound Strait, the Great Belt, and the Little Belt. Among these, the Sound Strait and the Great Belt are commonly used by medium and large vessels, and they differ significantly in navigational characteristics. The Sound Strait has a minimum depth of approximately 7.5 meters, while the Great Belt reaches around 15 meters. Given this depth disparity and the wide range of sizes (in terms of draft) within the tanker and cargo ship categories, two representative draft values are selected for each vessel type to support the route planning analysis. Considering the constant $Draft_{\max}$ ratio of $k = 0.3$ is typically used for coastal waters [60], we adopt $k = 0.2$ to determine $UKC_{\min}$ and SWD, where $SWD \geq Draft_{\max} \cdot (1 + k)$ is required to ensure navigational safety. Accordingly, tankers and cargo ships with drafts of 6 meters and 12 meters, as well as passenger ships with a draft of 8 meters, are included in our analysis to ensure both representativeness and practical relevance.

Additionally, based on ENCs and public data from the Bundesamt für Seeschifffahrt und Hydrographie (BSH) [66], the average water depth of the Kiel Canal is approximately 9.5 meters. Therefore, only vessels with relatively small drafts are considered suitable for route planning through it.

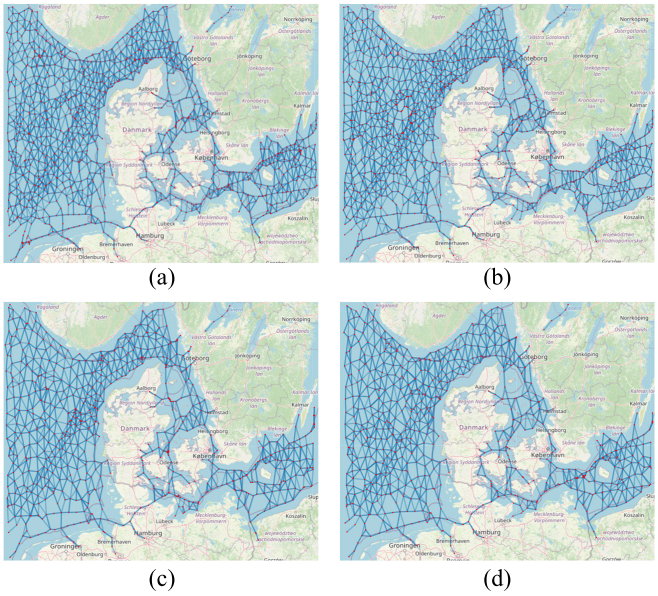


Fig. 8. Centroid points and flows derived from the tanker dataset through Algorithm 1 and Algorithm 2, employing different parameter configurations of $d_{\min}$ and $d_{\max}$: (a) $d_{\min} = 3$ km, $d_{\max} = 20$ km. (b) $d_{\min} = 4$ km, $d_{\max} = 20$ km. (c) $d_{\min} = 3$ km, $d_{\max} = 25$ km. (d) $d_{\min} = 4$ km, $d_{\max} = 25$ km.

B. Results of Global Graph-Based Path Planning

1) Results of Graph Construction: Before constructing the graph network, it is critical to select appropriate parameters to minimize errors and ensure that the network accurately reflects primary shipping routes. As shown in Fig. 8, the parameters $d_{\min}$ and $d_{\max}$ for the tanker dataset are determined by assessing the network's ability to reproduce trajectories under various parameter combinations, with particular emphasis on significant straits. Similarly, optimal parameter configurations for the other two datasets are carefully selected and summarized in Table II, and further details on parameter selection are discussed in Sec. VI.

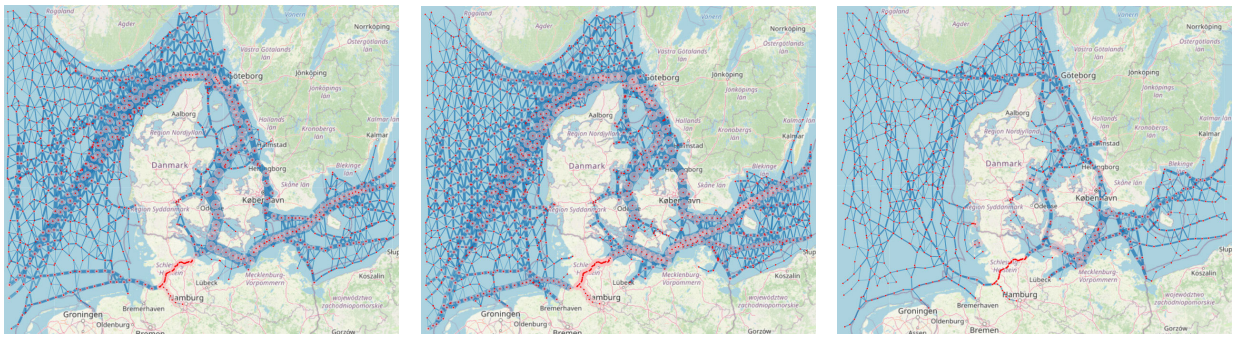


Fig. 9. Graph network architecture of AIS tracks (with updated global local path planning results): (a) tanker. (b) cargo ship. (c) passenger ship; red nodes represent the clustered waypoints, and blue lines represent the flows between waypoints; the size of red nodes indicates the congestion level at a junction, and the width of blue lines indicates the flow rate between adjacent junctions.

TABLE II

PARAMETER CONFIGURATIONS ACROSS VESSEL TYPES: TANKERS, CARGO SHIPS, AND PASSENGER SHIPS

Ship type	$d_{\min}$(km)	$d_{\max}$(km)
Tanker	4	20
Cargo	3	20
Passenger	4	20

TABLE III

ORIGIN-DESTINATION (OD) PAIRS AND CORRESPONDING COORDINATES USED FOR PLANNED ROUTE EVALUATION

OD cases	Dataset	Origin	Destination
OD1	Tanker	(55.9° N, 16° E)	(57.75° N, 10.88° E)
OD2	Tanker	(53.72° N, 4.3° E)	(57.75° N, 10.88° E)
OD3	Tanker	(54.3° N, 12° E)	(57.75° N, 10.88° E)
OD4	Cargo	(55.65° N, 15.4° E)	(53.88° N, 9.125° E)
OD5	Cargo	(54.3° N, 12° E)	(57.75° N, 10.96° E)
OD6	Cargo	(53.4° N, 4.7° E)	(57.75° N, 10.96° E)
OD7	Passenger	(54.5° N, 10.3° E)	(57.63° N, 11.67° E)

Moreover, the parameters $t_{\min}$ and $\theta_{\min}$ are derived from empirical observations. The minimum docking duration $t_{\min}$ is set to 5 mins, and the minimum turning angle $\theta_{\min}$ is set to 45°. Using these parameters, three graph networks are constructed, with nodes and edges representing shipping patterns (Fig. 9). Edge costs, as defined in Sec. III-B, are calculated and stored as attributes for the subsequent pathfinding process.

2) Results of Global Path Planning: To evaluate the effectiveness of the proposed global planner, we first examine the pathfinding results on the AIS-derived graphs *without* applying the local refinement module in Sec. IV. The origin-destination (OD) pairs used in this analysis are summarized in Tab. III, which covers tanker, cargo, and passenger vessels. For each OD case, the planner computes three paths under the probability-based, distance-based, and time-based cost metrics, and the experiments are conducted under two representative draft settings, i.e., 6 m and 12 m.

For the tanker dataset, Fig. 10 and Fig. 11 illustrate representative global routes for OD1–OD3 under the two draft depths. In both cases, the planned routes closely follow the main shipping lanes extracted from historical AIS data. This

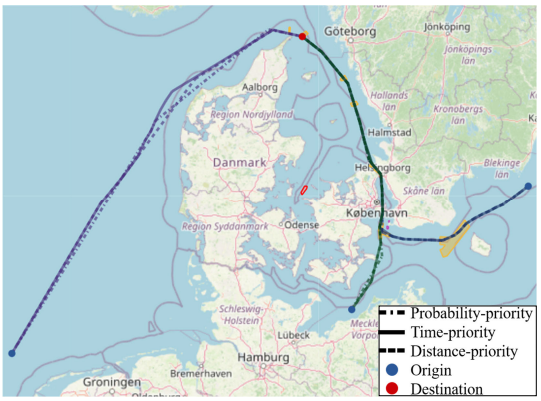


Fig. 10. Pathfinding results without local refinement for the tanker and $Draft$ = 6 m (OD1: blue; OD2: purple; OD3: green).

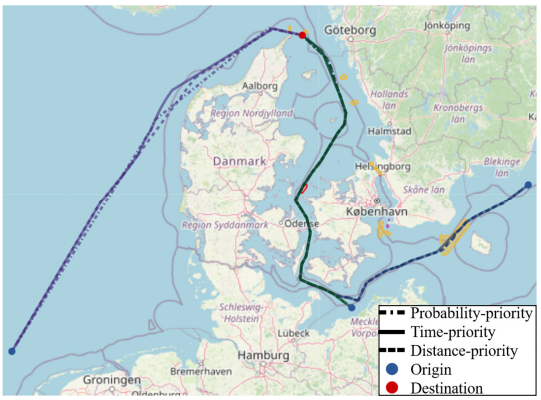


Fig. 11. Pathfinding results without local refinement for the tanker and $Draft$ = 12 m (OD1: blue; OD2: purple; OD3: green).

indicates that the AIS-derived graph successfully captures the backbone structure of real traffic corridors. The probability-based cost tends to adhere more strictly to high-traffic routes and reproduce the dominant historical corridors, as expected. The distance-based cost slightly shortens the geometric length by cutting across less frequently used segments when feasible. The time-based cost usually lies between the two: by taking into account the average speed along each edge, it prefers routes where vessels can sail faster (e.g., deeper channels and

TABLE IV
TIME AND DISTANCE OPTIMIZATION BY VESSEL DRAFT DEPTH

Draft	OD case	Time reduction	Distance reduction
6m	OD1	7.37%	≈0%
	OD2	44.82%	1.61%
	OD3	6.13%	2.33%
	OD4	4.48%	≈0%
	OD5	7.89%	1.27%
	OD6	4.14%	≈0%
12m	OD1	4.34%	≈0%
	OD2	7.89%	1.35%
	OD3	6.13%	2.33%
	OD5	6.96%	1.35%
	OD6	4.14%	≈0%
8m	OD7	1.18%	1.94%

less congested routes), even if the overall geometric distance is marginally longer. Comparing the two draft settings, deeper-draft tankers are more confined to deep-water corridors, while shallower-draft tankers have more flexibility to exploit shorter alternatives in coastal areas. Similar routing behaviors are observed for the cargo and passenger vessels (OD4–OD7), for which the planned routes are omitted for brevity.

The quantitative impact of the draft depth on route efficiency is summarized in Tab. IV, which compares the planned routes with the historical AIS trajectories in terms of travel time and travelled distance for all vessel types and OD pairs. For most cases, the global planner reduces the estimated travel time by about 4%–8% while keeping the travelled distance within 0–2.5% of the historical routes (e.g., OD1: 7.37% time reduction with almost no change in distance; OD5: 7.89% time reduction with only 1.27% increase in distance). The only extreme case is OD2, where the time reduction reaches 44.82% at the cost of a modest 1.61% increase in distance, because the AIS-derived graph reveals a deep-water corridor that allows a higher average sailing speed than the route most frequently used in the historical data. These results indicate that the global graph-based planner can systematically exploit the structure of the AIS network and the draft-dependent edge feasibility to obtain more time-efficient routes, without significantly altering the overall sailing distance.

C. Results of the Complete Framework With Local Refinement

We next evaluate the complete planning framework, i.e., the global AIS-derived graph planner followed by the local refinement procedure described in Sec. IV. The proposed method is compared with the PI-DP-RRT planner presented in [50], which also leverages AIS data. To avoid draft-induced confounding of the feasible set, the comparison is restricted to OD1–OD6 under a 6 m draft.

To ensure a rigorous evaluation, the PI-DP-RRT algorithm was tuned via sensitivity analyses. In practice, we use the same tuned parameter set for OD1–OD3 and OD5–OD6, and a separate set for OD4. Specifically, we optimized the branch extension step size to maximize navigational granularity in narrow straits while minimizing computational overhead, thereby avoiding coarse segmentation that reduces feasibility in constrained waters. Simultaneously, the guided sampling probability was fine-tuned to balance stochastic exploration with adherence to the guide path, thereby preventing both the increased search effort caused by excessive randomness

TABLE V
PARAMETER SETTING FOR PI-DP-RRT [50]

Parameters	OD4	Other OD cases
Branch extension length (km)	1.0	1.0
Sampling probability	0.2	0.3
Goal proximity threshold (km)	10.0	8.0
Guide region side length (km)	5.0	5.0
Maximum number of iterations	160,000	
K-means cluster number	1211	

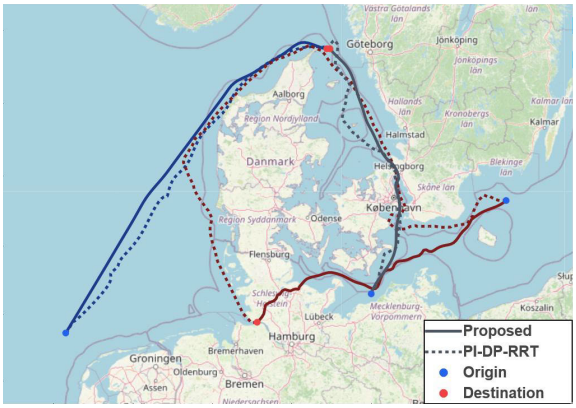


Fig. 12. Representative routes produced by the proposed framework (with local refinement) and PI-DP-RRT for OD2 (blue), OD4 (red), and OD5 (grey).

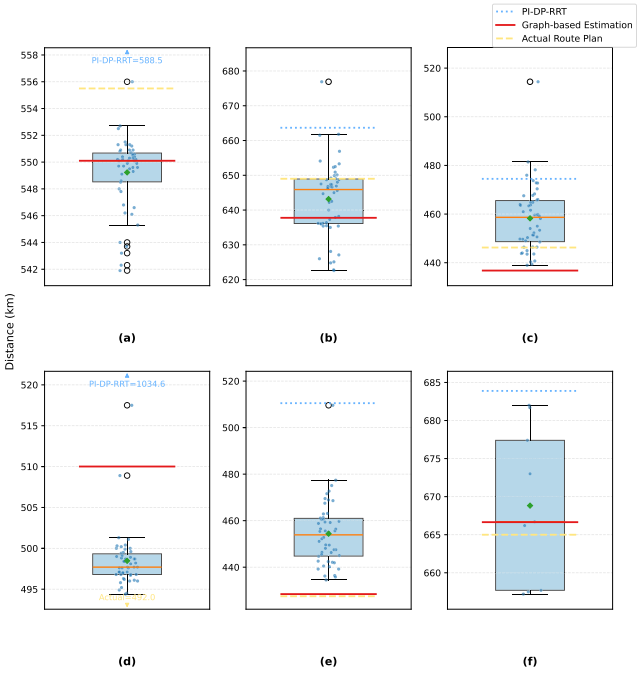
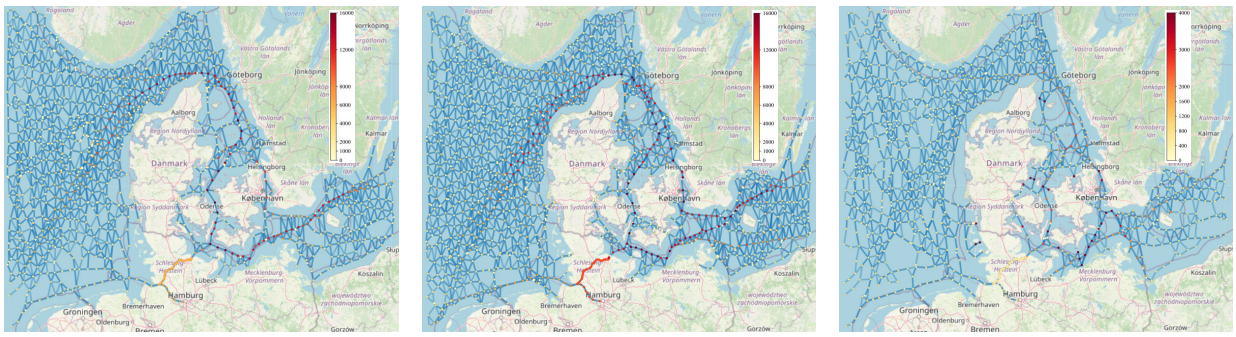


Fig. 13. Box-and-whisker plots of traveled distance for historical trajectories with similar origin and destination to OD1 to OD6. The boxes show the AIS historical distributions. Horizontal markers indicate the distances of the proposed framework, PI-DP-RRT, and (optionally) the VesselFinder plan. Panel titles (a–f) correspond to OD1–OD6.



(a) (b) (c)

Fig. 14. Heatmap of waypoints with intensity representing the weight magnitude of each waypoint: (a) Tanker dataset; (b) Cargo ship dataset; (c) Passenger ship dataset;.

and the obstacle collisions resulting from over-constrained guidance. Furthermore, to prevent premature termination, a "run-to-completion" strategy was employed by setting a relaxed maximum iteration threshold (see Tab. V). Cubic B-spline interpolation is also applied for path smoothing. Since PI-DP-RRT does not provide reliable speed modeling, distance is used as the primary metric for comparison.

To assess how the planned routes behave relative to real operations, we construct an empirical baseline from historical AIS data. For each OD pair, we collect historical trajectories whose origins and destinations lie within the same port regions as the OD under consideration, and treat their travelled distances as a reference distribution. Voyage plans generated by the VesselFinder platform [67] are also included as an external benchmark. To ensure a fair comparison regarding navigational safety, these routes were obtained using the identical OD pairs as in our simulation, with water depth constraints strictly enforced by manually inserting 1–2 intermediate waypoints to guide the route selection through navigable deep-water channels. Both the proposed framework and PI-DP-RRT use the same set of OD pairs and the same reference data in all experiments.

Fig. 12 overlays the planned routes for three representative ODs (OD2, OD4, and OD5), where solid lines denote the routes planned by the proposed framework under the distance-based metric, and dashed lines denote those generated by PI-DP-RRT. In all three cases, the proposed routes adhere to the main AIS-derived corridors, traverse the deep-water channels between Denmark and Sweden where appropriate, and smoothly avoid coastlines and islands. The paths remain within the envelope of historical AIS trajectories and exhibit gentle curvatures thanks to the spline-based refinement. Compared with the raw global paths, the locally refined routes preserve the large-scale structure selected by the graph planner while removing small-scale collisions and jagged segments near shorelines and port areas, without materially changing the overall length.

In contrast, PI-DP-RRT can deviate significantly from typical traffic patterns. As shown on OD2, it drifts offshore into a long detour outside the dense AIS corridor; on OD5, it introduces unnecessary loops or zigzag segments in coastal narrowings. The difference is most evident on OD4, where the feasible corridor runs through a narrow inland waterway. In this case, the proposed route correctly enters and follows this channel, whereas PI-DP-RRT often fails to enter the channel and detours along the coast.

It is worth noting that PI-DP-RRT still produces irregular paths even after the parameter tuning described earlier. This behavior stems from the Douglas–Peucker simplification itself: it tends to keep many guide points in high-curvature regions while leaving long straight sections sparsely sampled. As a result, the guide points are spatially non-uniform, and a fixed set of sampling parameters cannot simultaneously support fine maneuvering in dense turns and efficient exploration across long, under-constrained segments. This structural limitation makes it difficult for PI-DP-RRT to reach the same level of overall path smoothness as the proposed topological approach.

A more thorough comparison of all ODs is shown in Fig. 13, where box–whisker plots depict the historical AIS distance distributions and horizontal markers indicate the distances of the proposed framework produced under the distance-based metric, PI-DP-RRT, and the VesselFinder plan. As shown in Fig. 13, the planned path from VesselFinder closely tracks the AIS medians from historical trajectories. Similarly, the proposed routes consistently sit at or below the AIS median and within the interquartile band. In contrast, PI-DP-RRT typically lies above the median and often approaches or exceeds the upper whisker, indicating longer and less realistic paths.

For concrete readings from Fig. 13, on OD2, the proposed route is 1.26% shorter than the AIS median and 3.90% shorter than PI-DP-RRT, and it is also 1.73% shorter than VesselFinder. On OD5, the proposed route is 5.61% below the AIS median and 16.08% shorter than PI-DP-RRT. These results show that the complete framework produces routes that align with historical practice and public voyage plans while remaining materially shorter than the PI-DP-RRT baseline.

VI. Conclusion

This paper presented an AIS-based vessel path planning framework grounded in graph theory. Historical AIS trajectories are transformed into a directed maritime route network by extracting and clustering significant points into waypoints and connecting them via aggregated trajectory segments,

thereby capturing prevalent traffic flows and navigation patterns. Within this network, three traversal cost formulations, i.e., probability-priority, distance-priority, and time-priority, are designed to support different planning objectives, while environmental constraints from water depth and ENCs are used to remove edges in non-navigable waters and penalize unsafe regions. A local path refinement module further integrates a non-navigable-area avoidance algorithm with B-spline interpolation to generate smoother, more feasible trajectories. Experiments on AIS datasets for tankers, cargo ships, and passenger ships (with multiple draft configurations) show that the proposed method produces routes that closely follow real trajectories and voyage plans, and often yields shorter or faster paths.

Despite these promising results, several aspects of the framework warrant further development. The current design primarily relies on historical traffic patterns and static chart information, without explicitly incorporating real-time environmental factors such as weather, waves, and ocean currents. In addition, validation is restricted to a limited set of ports and coastal regions, and the graph construction requires manual parameter tuning for each vessel class. Future research can therefore focus on integrating dynamic environmental data for adaptive replanning, extending validation to more diverse port environments and regulatory regimes, and exploring more systematic parameter-learning or uncertainty-aware formulations to further improve robustness and generalizability.

APPENDIX
DISCUSSION OF PARAMETER SETTINGS FOR GRAPH CONSTRUCTION

Parameter settings for graph construction, including $d_{\max}$ and $d_{\min}$, are optimized based on the maritime environment and data distribution. These parameters vary with factors such as shipping lane density and geographic features. For instance, in open maritime areas, larger $d_{\max}$ values are used, whereas smaller values capture the complexity of narrow channels. Additionally, the density maps provided on maritime electronic chart websites (i.e., https://shipfinder.co/ and https://www.marinetraffic.com/en/ais/) serve as valuable references for parameter selection. For example, between northern Denmark and southern Norway (approximately 100 km apart), density maps suggest a spacing of 20–25 km between routes, making this range a suitable choice for $d_{\max}$. Furthermore, the relative consistency of large vessels' navigation states over longer distances informs the selection of $d_{\min}$, typically set at 3–4 km. Minor parameter adjustments, within a 10% range, have minimal impact, as tested on the passenger ship dataset, where planned route variations in travel time and distance remain under 0.5 hours and 5 km, respectively.

In local path planning, parameter adjustments are essential to accommodate varying geographic complexities, where adjustments to the non-navigable area algorithm's search step and buffer width improve efficiency. Heatmaps assist in determining weight thresholds for filtering intermediate points by analyzing waypoint significance within their geographic context, as shown in Fig. 14. Specifically, heatmaps are generated based on the recorded weights of waypoints in the graph to analyze their geographic context and relative significance. By integrating historical trajectories and routes, we assess whether these points should be retained, thus determining an appropriate weight threshold for different datasets. Overall, parameter selection primarily depends on vessel type and environment, with suboptimal settings potentially leading to excessive traversal distances and times compared to real-world averages.

According to the results, the proposed approach is promising. To apply it in real scenarios, it still requires further refinement, particularly in dynamic maritime environments. Future implementations should incorporate real-time environmental factors such as weather conditions and ocean currents, which are essential for ensuring safe and adaptive maritime navigation. Collision avoidance in shared waters and seasonal variations in route usage also warrant attention. Additionally, further validation of the proposed methodology in diverse port environments and coastal waters—characterized by varying traffic patterns and regulatory constraints—is planned to demonstrate its practical applicability and scalability. Accounting for the above real-world conditions might yield routes that diverge from those planned by the proposed method.

Additionally, while the method suits vessels with clear navigation patterns, such as tankers, cargo ships, and passenger ships, its applicability to fishing vessels is limited due to their irregular trajectories influenced by fishing activities. The success of this approach relies on the regularity of navigation behavior, making it less effective for analyzing vessels with unpredictable movement patterns. Nonetheless, this method demonstrates significant potential and could, in the future, assist not only in manned shipping but also in advancing uncrewed shipping operations.

REFERENCES

[1] Z. Wan, J. Chen, A. E. Makhloufi, D. Sperling, and Y. Chen, "Four routes to better maritime governance," *Nature*, vol. 540, no. 7631, pp. 27–29, Dec. 2016.

[2] K. Wróbel, J. Montewka, and P. Kujala, "Towards the assessment of potential impact of unmanned vessels on maritime transportation safety," *Rel. Eng. Syst. Saf.*, vol. 165, pp. 155–169, Sep. 2017.

[3] P. Zhou, W. Zhao, J. Li, A. Li, W. Du, and S. Wen, "Massive maritime path planning: A contextual online learning approach," *IEEE Trans. Cybern.*, vol. 51, no. 12, pp. 6262–6273, Dec. 2021.

[4] *The Eighty-First Session of Maritime Safety Committee (MSC)*, IMO, Strategy for the development and implementation of e-navigation (MSC 85/26/Add.1, annex 20), 2006, p. 1.

[5] E. Tu, G. Zhang, L. Rachmawati, E. Rajabally, and G.-B. Huang, "Exploiting AIS data for intelligent maritime navigation: A comprehensive survey from data to methodology," *IEEE Trans. Intell. Transp. Syst.*, vol. 19, no. 5, pp. 1559–1582, May 2018.

[6] S. Hexeberg, A. L. Flåten, B. H. Eriksen, and E. F. Brekke, "AIS-based vessel trajectory prediction," in *Proc. 20th Int. Conf. Inf. Fusion (Fusion)*, Jul. 2017, pp. 1–8.

[7] X. Xu, C. Liu, J. Li, and Y. Miao, "Trajectory clustering for SVR-based time of arrival estimation," *Ocean Eng.*, vol. 259, Sep. 2022, Art. no. 111930.

[8] X. Xu, C. Liu, J. Li, Y. Miao, and L. Zhao, "Long-term trajectory prediction for oil tankers via grid-based clustering," *J. Mar. Sci. Eng.*, vol. 11, no. 6, p. 1211, Jun. 2023.

[9] X. Li, C. Liu, J. Li, L. Zhao, and Z. Du, "Advancing ship trajectory prediction: Integrating deep learning with enhanced reference trajectory correction techniques," *Ocean Eng.*, vol. 311, Nov. 2024, Art. no. 118880.

[10] Y. Zhou, W. Daamen, T. Vellinga, and S. P. Hoogendoorn, "Ship classification based on ship behavior clustering from AIS data," *Ocean Eng.*, vol. 175, pp. 176–187, Mar. 2019.

[11] S.-K. Zhang, G.-Y. Shi, Z.-J. Liu, Z.-W. Zhao, and Z.-L. Wu, "Data-driven based automatic maritime routing from massive AIS trajectories in the face of disparity," *Ocean Eng.*, vol. 155, pp. 240–250, May 2018.

[12] H. Yu et al., "Ship path optimization that accounts for geographical traffic characteristics to increase maritime port safety," *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 6, pp. 5765–5776, Jun. 2022.

[13] D. Nguyen, R. Vadaine, G. Hajduch, R. Garello, and R. Fablet, "GeoTrackNet—A maritime anomaly detector using probabilistic neural network representation of AIS tracks and a contrario detection," *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 6, pp. 5655–5667, Jun. 2022.

[14] J. Liu, J. Li, and C. Liu, "AIS-based kinematic anomaly classification for maritime surveillance," *Ocean Eng.*, vol. 305, Aug. 2024, Art. no. 118026.

[15] G. Wang, J. Meng, and Y. Han, "Extraction of maritime road networks from large-scale AIS data," *IEEE Access*, vol. 7, pp. 123035–123048, 2019.

[16] X.-P. Yan, S.-W. Wang, F. Ma, Y.-C. Liu, and J. Wang, "A novel path planning approach for smart cargo ships based on anisotropic fast marching," *Expert Syst. Appl.*, vol. 159, Nov. 2020, Art. no. 113558.

[17] H. Xue, "A quasi-reflection based SC-PSO for ship path planning with grounding avoidance," *Ocean Eng.*, vol. 247, Mar. 2022, Art. no. 110772.

[18] C. Liu, Q. Mao, X. Chu, and S. Xie, "An improved A-star algorithm considering water current, traffic separation and berthing for vessel path planning," *Appl. Sci.*, vol. 9, no. 6, p. 1057, Mar. 2019.

[19] G. Mannarini, D. N. Subramani, P. F. J. Lermusiaux, and N. Pinardi, "Graph-search and differential equations for time-optimal vessel route planning in dynamic ocean waves," *IEEE Trans. Intell. Transp. Syst.*, vol. 21, no. 8, pp. 3581–3593, Aug. 2020.

[20] K. Karur, N. Sharma, C. Dharmatti, and J. E. Siegel, "A survey of path planning algorithms for mobile robots," *Vehicles*, vol. 3, no. 3, pp. 448–468, Aug. 2021.

[21] M. Jun and R. D'Andrea, "Path planning for unmanned aerial vehicles in uncertain and adversarial environments," in *Cooperative Control: Models, Applications and Algorithms* (Cooperative Systems), vol. 1. Boston, MA, USA: Springer, 2003, pp. 95–110.

[22] J. Tisdale, Z. Kim, and J. K. Hedrick, "Autonomous UAV path planning and estimation," in *Proc. IEEE Robot. Autom. Mag.*, vol. 16, Jun. 2009, pp. 35–42.

[23] Y. Liu and R. Bucknall, "Path planning algorithm for unmanned surface vehicle formations in a practical maritime environment," *Ocean Eng.*, vol. 97, pp. 126–144, Mar. 2015.

[24] H.-y. Zhang, W. Lin, and A. Chen, "Path planning for the mobile robot: A review," *Symmetry-Basel*, vol. 10, no. 10, p. 450, Oct. 2018.

[25] L. Liu, X. Wang, X. Yang, H. Liu, J. Li, and P. Wang, "Path planning techniques for mobile robots: Review and prospect," *Expert Syst. Appl.*, vol. 227, Oct. 2023, Art. no. 120254.

[26] M. Candeloro, A. M. Lekkas, A. J. Sørensen, and T. I. Fossen, "Continuous curvature path planning using Voronoi diagrams and Fermat's spirals," in *Proc. 9th IFAC Conf. Control Appl. Mar. Syst.*, vol. 46, 2013, pp. 132–137.

[27] H. Niu, Y. Lu, A. Savvaris, and A. Tsourdos, "Efficient path planning algorithms for unmanned surface vehicle," *IFAC-PapersOnLine*, vol. 49, no. 23, pp. 121–126, 2016.

[28] Y. Shin and E. Kim, "Hybrid path planning using positioning risk and artificial potential fields," *Aerosp. Sci. Technol.*, vol. 112, May 2021, Art. no. 106640.

[29] X. Liu, Y. Li, J. Zhang, J. Zheng, and C. Yang, "Self-adaptive dynamic obstacle avoidance and path planning for USV under complex maritime environment," in *Proc. IEEE Access*, vol. 7, Aug. 2019, pp. 114945–114954.

[30] L. E. Kavraki, P. Švestka, J. C. Latombe, and M. H. Overmars, "Probabilistic roadmaps for path planning in high-dimensional configuration spaces," in *Proc. IEEE Trans. Robot. Autom.*, vol. 12, Aug. 1996, pp. 566–580.

[31] G. K. D. de Vries and M. van Someren, "Machine learning for vessel trajectories using compression, alignments and domain knowledge," *Expert Syst. Appl.*, vol. 39, no. 18, pp. 13426–13439, Dec. 2012.

[32] Z. Xiao, L. Ponnambalam, X. Fu, and W. Zhang, "Maritime traffic probabilistic forecasting based on vessels' waterway patterns and motion behaviors," *IEEE Trans. Intell. Transp. Syst.*, vol. 18, no. 11, pp. 3122–3134, Nov. 2017.

[33] H. Wang, A. Kläser, C. Schmid, and C.-L. Liu, "Dense trajectories and motion boundary descriptors for action recognition," *Int. J. Comput. Vis.*, vol. 103, no. 1, pp. 60–79, May 2013.

[34] P. Kaluza, A. Kölzsch, M. T. Gastner, and B. Blasius, "The complex network of global cargo ship movements," *J. R. Soc. Interface*, vol. 7, no. 48, pp. 1093–1103, Jul. 2010.

[35] Z. Yan et al., "Exploring AIS data for intelligent maritime routes extraction," *Appl. Ocean Res.*, vol. 101, Aug. 2020, Art. no. 102271.

[36] H. Rong, A. P. Teixeira, and C. Guedes Soares, "Maritime traffic probabilistic prediction based on ship motion pattern extraction," *Rel. Eng. Syst. Saf.*, vol. 217, Jan. 2022, Art. no. 108061.

[37] E. W. Dijkstra, "A note on two problems in connexion with graphs," *Numerische Math.*, vol. 1, no. 1, pp. 269–271, Dec. 1959.

[38] H. Wang, W. Mao, and L. Eriksson, "A three-dimensional Dijkstra's algorithm for multi-objective ship voyage optimization," *Ocean Eng.*, vol. 186, Aug. 2019, Art. no. 106131.

[39] Y. Singh, S. Sharma, R. Sutton, D. C. Hatton, and A. Khan, "Feasibility study of a constrained Dijkstra approach for optimal path planning of an unmanned surface vehicle in a dynamic maritime environment," in *Proc. ICARSC*, Apr. 2018, pp. 117–122.

[40] P. Hart, N. Nilsson, and B. Raphael, "A formal basis for the heuristic determination of minimum cost paths," *IEEE Trans. Syst. Sci. Cybern.*, vol. SSC-4, no. 2, pp. 100–107, Jul. 1968.

[41] Y. Singh, S. Sharma, R. Sutton, D. C. Hatton, and A. Khan, "A constrained A approach towards optimal path planning for an unmanned surface vehicle in a maritime environment containing dynamic obstacles and ocean currents," in *Proc. Ocean Eng.*, vol. 169, Dec. 2018, pp. 187–201.

[42] Y. Li, F. Zhang, D. Xu, and J. Dai, "Liveness-based RRT algorithm for autonomous underwater vehicles motion planning," *J. Adv. Transp.*, vol. 2017, pp. 1–10, Oct. 2017.

[43] R. Zaccone, "COLREG-compliant optimal path planning for real-time guidance and control of autonomous ships," *J. Mar. Sci. Eng.*, vol. 9, no. 4, p. 405, Apr. 2021.

[44] S. LaValle, *Rapidly-exploring Random Trees: A New Tool for Path Planning*, Standard 9811, Oct. 1998.

[45] S. M. LaValle and J. J. Kuffner, "Randomized kinodynamic planning," *Int. J. Robot. Res.*, vol. 20, no. 5, pp. 378–400, May 2001.

[46] H.-T.-L. Chiang and L. Tapia, "COLREG-RRT: An RRT-based COLREGS-compliant motion planner for surface vehicle navigation," *IEEE Robot. Autom. Lett.*, vol. 3, no. 3, pp. 2024–2031, Jul. 2018.

[47] D.-U. Jang and J.-S. Kim, "Development of ship route-planning algorithm based on rapidly-exploring random tree (RRT) using designated space," *J. Mar. Sci. Eng.*, vol. 10, no. 12, p. 1800, Nov. 2022.

[48] W. Hu, S. Chen, Z. Liu, X. Luo, and J. Xu, "HA-RRT: A heuristic and adaptive RRT algorithm for ship path planning," *Ocean Eng.*, vol. 316, Jan. 2025, Art. no. 119906.

[49] J. Zhang, H. Zhang, J. Liu, D. Wu, and C. G. Soares, "A two-stage path planning algorithm based on rapid-exploring random tree for ships navigating in multi-obstacle water areas considering COLREGs," *J. Mar. Sci. Eng.*, vol. 10, no. 10, p. 1441, Oct. 2022.

[50] Q. Gu, R. Zhen, J. Liu, and C. Li, "An improved RRT algorithm based on prior AIS information and DP compression for ship path planning," *Ocean Eng.*, vol. 279, Jul. 2023, Art. no. 114595.

[51] J. Luo and Y. Su, "Path planning for multi-USV target coverage in complex environments," *Ocean Eng.*, vol. 312, Nov. 2024, Art. no. 119090.

[52] P. Wu et al., "Autonomous obstacle avoidance of an unmanned surface vehicle based on cooperative manoeuvring," in *Proc. Ind. Robot.*, vol. 44, 2017, pp. 64–74.

[53] Y. Chen, G. Bai, Y. Zhan, X. Hu, and J. Liu, "Path planning and obstacle avoiding of the USV based on improved ACO-APF hybrid algorithm with adaptive early-warning," *IEEE Access*, vol. 9, pp. 40728–40742, 2021.

[54] J. Xin, J. Zhong, F. Yang, Y. Cui, and J. Sheng, "An improved genetic algorithm for path-planning of unmanned surface vehicle," *Sensors*, vol. 19, no. 11, p. 2640, Jun. 2019.

[55] H. Yang, J. Qi, Y. Miao, H. Sun, and J. Li, "A new robot navigation algorithm based on a double-layer ant algorithm and trajectory optimization," *IEEE Trans. Ind. Electron.*, vol. 66, no. 11, pp. 8557–8566, Nov. 2019.

[56] D.-H. Chun, M.-I. Roh, H.-W. Lee, J. Ha, and D. Yu, "Deep reinforcement learning-based collision avoidance for an autonomous ship," *Ocean Eng.*, vol. 234, 2021, Art. no. 109216.

[57] A. V. Goldberg and C. Harrelson, "Computing the shortest path: A search meets graph theory," *SODA*, pp. 156–165, 2005.

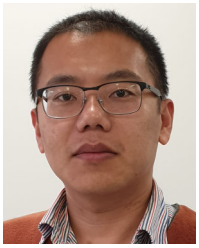
[58] A. Hagberg, P. J. Swart, and D. A. Schult, "Exploring network structure, dynamics, and function using NetworkX," Dept. Los Alamos Nat. Lab. (LANL), Los Alamos, NM, USA, Tech. Rep. LA-UR-08-05495, 2008.

[59] A. Weintrit, *Marine Navigation and Safety of Sea Transportation*. Boca Raton, FL, USA: CRC Press, 2009.

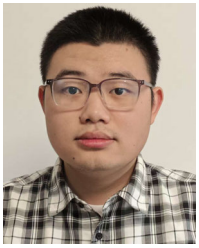
[60] H.-T. Lee and M.-K. Kim, "Optimal path planning for a ship in coastal waters with deep q network," *Ocean Eng.*, vol. 307, Sep. 2024, Art. no. 118193.
[61] E. Commission. (2025). *Emodnet Bathymetry*. [Online]. Available: https://emodnet.ec.europa.eu/en
[62] H.-T. Lee, H.-M. Choi, J.-S. Lee, H. Yang, and I.-S. Cho, "Generation of ship's passage plan using data-driven shortest path algorithms," *IEEE Access*, vol. 10, pp. 126217–126231, 2022.
[63] D. M. Authority. (2024). *Ais Data*. [Online]. Available: https://dma.dk/safety-at-sea/navigational-information/ais-data
[64] OpenStreetMap.(2024). *Openstreetmap Data*. [Online]. Available: https://osmdata.openstreetmap.de/data/coastlines.html
[65] D. M. Authority. (2022). *Navigation Through Danish Waters*. [Online]. Available: https://www.soefartsstyrelsen.dk/Media/E/B/Navigation_through_Danish_waters.pdf
[66] Bundesamtfürseeschifffahrt and Hydrographie. (2015).*German Traffic Regulations for Navigable Maritime Waterways*. [Online]. Available: http://waterways.cz/documents/SeeschStrO_engl.pdf
[67] (Oct. 2025). *Vessel Finder*. [Online]. Available: https://route.vesselfinder.com/

Jianghui Li (Member, IEEE) received the B.S. degree in communications engineering from the Huazhong University of Science and Technology, Wuhan, China, in 2011, and the M.Sc. degree in communications engineering and the Ph.D. degree in electronics engineering from the University of York, York, England, U.K., in 2013 and 2017, respectively. From 2017 to 2021, he was a Research Fellow with the University of Southampton, U.K. Since 2021, he has been a Professor with the State Key Laboratory of Marine Environmental Science, Xiamen University, China. His current research interests include offshore carbon capture, utilization and storage (CCUS), underwater acoustics, adaptive signal processing, and artificial intelligence in ocean and maritime engineering.

Jinliang Liu received the B.S. degree in science and technology of remote sensing from Wuhan University, Wuhan, China, in 2022, and the M.Sc. degree in marine physics from Xiamen University, Xiamen, China in 2025. Since then, he has been working as a Researcher at the 710 Research Institute of China State Shipbuilding Corporation Ltd. His research interests include AIS data analysis, intelligent shipping and intelligent uncrewed underwater vehicle (UUV) manufacturing.



Chunshan Liu (Member, IEEE) received the B.S. degree in applied physics from the University of Science and Technology of China, in 2007, the M.Sc. degree in communication engineering, and the Ph.D. degree in electronic engineering from the University of York, York, U.K., in 2008 and 2012, respectively. He was a Post-Doctoral Research Associate with the University of York from April to June 2012 and a research fellow at Macquarie University, Australia, between 2012 and 2019. Since late 2019, he has been a Professor with the School of Communication Engineering, Hangzhou Dianzi University, Hangzhou, China. His research interests include signal processing in wireless communications and underwater acoustics, and big data analysis. He received the Discovery Early Career Researcher Award from the Australian Research Council (ARC DECRA) in November 2017 and the Kathleen Mary Stott Memorial Prize for Excellence in Research from the University of York in 2012.



Peng Jiang received the B.S. degree in communication engineering from Hangzhou Dianzi University, Hangzhou, China, in 2025, where he is currently pursuing the M.S. degree. His research interests include AIS data analysis and maritime traffic intelligence.