Strategies to improve the isochrone algorithm for ship voyage optimisation

Citation for the original published paper (version of record):
Chen, Y., Tian, W., Mao, W. (2024). Strategies to improve the isochrone algorithm for ship voyage optimisation. Ships and Offshore Structures, 19(12): 2137-2149.
http://dx.doi.org/10.1080/17445302.2024.2329011

Ships and Offshore Structures

ISSN: (Print) (Online) Journal homepage: www.tandfonline.com/journals/tsos20

Strategies to improve the isochrone algorithm for ship voyage optimisation

Yuhan Chen, Wuliu Tian & Wengang Mao

To cite this article: Yuhan Chen, Wuliu Tian & Wengang Mao (14 Mar 2024): Strategies to improve the isochrone algorithm for ship voyage optimisation, Ships and Offshore Structures, DOI: 10.1080/17445302.2024.2329011

To link to this article: https://doi.org/10.1080/17445302.2024.2329011

Published online: 14 Mar 2024.

SHIPS AND OFFSHORE STRUCTURES
https://doi.org/10.1080/17445302.2024.2329011

Strategies to improve the isochrone algorithm for ship voyage optimisation

Yuhan Chen [a], Wuliu Tian [b] and Wengang Mao [a]

[a]Department of Mechanics and Maritime Sciences, Chalmers University of Technology, Gothenburg, Sweden; [b]School of Navigation, Beibu Gulf University, Guangxi, People's Republic of China

ABSTRACT

An optimisation algorithm is an essential part of voyage planning systems to achieve autonomous and intelligent operations. Various algorithms have been proposed for voyage planning to minimise fuel consumption and increase punctuality. Among them, the isochrone method has been recognised for its robustness and efficiency in voyage optimizations. This paper improves it to overcome its incompetence in multi-objective optimisation and reliable route convergence towards the destination. Five improved methods are proposed and compared, to investigate the most effective enhancing strategy to achieve robustness and practicality in real-time application. By changing search space after the middle stage of the voyage, and formulating an augmented cost function to refine search criteria according to different optimisation objectives, one improved isochrone method (named Isochrone-A*) shows more competitive capability, with potential in real-time implementations. The effectiveness and efficiency of these five improved strategies are compared using four ocean-crossing voyages collected by a chemical tanker.

ARTICLE HISTORY
Received 18 October 2023
Accepted 25 January 2024

KEYWORDS
Energy efficiency; isochrone algorithm; searching criteria; voyage optimisation

1. Introduction

To mitigate climate change resulting from maritime transport, the International Maritime Organization (IMO) requires ship owners and technical managers to continually enhance the energy efficiency of their ship operations (Poulsen et al. 2022). Shipping companies typically also strive to maximise their revenue while reducing sailing expenses, including fuel consumption. A voyage planning system, as an essential solution for e-navigation, can be implemented to reduce air emissions from shipping, improve cost efficiency, and assist autonomous ship navigation (Zhao et al. 2023; Zhao and Bai 2023). A proper optimisation algorithm is a crucial component in a ship's voyage planning systems, to achieve specific predefined objectives (Simonsen et al. 2015). Many voyage optimisation algorithms have been widely implemented and are under research, including recently developed using advanced AI/machine learning techniques (Xue 2022; Xue and Qian 2023). For actual sailing, commonly employed algorithms are the so-called deterministic optimisation algorithms rather than the stochastic ones such as genetic algorithms. Depending on how the searching area is discretized in time, the optimisation algorithms can be divided into dynamic and static grid-based voyage optimizations (Wang et al. 2021).

For dynamic grid-based optimisation algorithms, searching waypoints/grid is dynamically developed during the optimisation process, e.g. the isochrone algorithm first proposed by (James 1957) and originally used to assist manual navigation planning on maps. It was further improved by (Hagiwara 1989) in computer-assisted voyage planning to consider fast arrival time. The irregular shape of the isochrone caused by the non-convexity of a ship's performance at sea may cause 'isochrone loop' as in (Wisniewski 1991). (Roh 2013) improved the isochrone algorithm by considering the ship performance impact at sea. Those methods were further modified by changing equal travelling time to equal power consumption as Isochrone, named as Isopone algorithms (Klompstra et al. 1992) and Isocost (Topaj et al. 2019). Moreover, to enhance the performance of 2D isochrone method, (Lin et al. 2013) proposed a 3D isochrone method and applied it with Particle Swarm Optimization (PSO) in (Lin 2018). Similar approaches were also investigated such as (Sasa et al. 2021) to further consider the involuntary speed loss and maneuverability for ship voyage planning. Another typical example of a dynamic grid searching method is the Dividing Rectangles (DIRECT) algorithm by (Larsson et al. 2015).

The static grid-based methods predefine all searching waypoints around a ship's sailing area at the initial optimisation stage, and optimal waypoints can only be chosen from this static grid. If a ship's speed is fixed along a voyage, they are often named as two-dimensional (2D) voyage optimisation methods, such as the dynamic programming algorithms in (Chen 1978), (De Wit 1990) and (Calvert et al. 1991) to optimise a ship's route/voyage with fixed speed or power setting based on the theory of (Bellman 1952). Conventional Dijkstra algorithm (Dijkstra 1959), A* algorithm (Hart et al. 1968) and their recent development, such as in (Silveira et al. 2019), (Ma et al. 2020), (Pennino et al. 2020), (Shin et al. 2020), and (Grifoll et al. 2022), (Bahrami and Siadatmousavi 2023) etc., are other commonly used voyage planning methods. If a ship's speed/power is configured to vary along the voyage, it is recognised as a three-dimensional(3D) voyage optimisation method. For example, the 2D dynamic programming method that was further developed by (Wang and Meng 2012), (Zaccone and Figari 2017; Zaccone et al. 2018), and (Wang et al. 2019) to allow speed variation.

However, voyage optimisation systems should always consider the balance between computation efficiency and the effectiveness of optimisation algorithms for practical operations. In addition, frequent change in a ship's sailing status or engine settings is not

CONTACT Wengang Mao wengang.mao@chalmers.se Division of Marine Technology, Department of Mechanics and Maritime Sciences, 412 96, Gothenburg, Sweden

preferred by operators. Sophisticated methods such as 3D optimisation methods and AI-based reinforcement learning algorithms are generally too complicated to solve voyage optimisation problems within a reasonable time range due to many variables and their ambiguous dependencies. According to the market survey by (Simonsen et al. 2015), shipping companies expect the runtime of the voyage optimisation algorithm to be at most 1 min, preferably within 15 s. Meanwhile, maintaining accurate Estimated Time of Arrival (ETA) are widely considered as an energy-efficient measure while also avoiding sailing risks (Turna 2023). Therefore, in this paper, the isochrone voyage optimisation method, well-known for its computation efficiency and characteristic for ensuring ETA, is further developed. Hagiwara (1989) improved the traditional isochrone method from a manual navigation method to an efficient voyage planning algorithm. However, it may result in irregular routes with unrealistic shapes, and is not suitable for multi-objective optimisation. Other researchers refined the isochrone method by including the speed variation and employing it in advanced machine learning algorithms to enhance the optimisation capability, but also increasing its complexity. To inherit the computational efficiency to make it practical for real operations, while also enhancing the voyage optimisation capability, this paper introduces different improvement strategies from the aspects of improving the search space and convergency during the optimisation process. A general overview of the isochrone voyage optimisation algorithm is presented in Section 2. In Section 3, different improvement strategies are described in detail. Section 4 compares the results of the voyage optimisation with those improved methods in terms of their effectiveness and efficiency using full-scale measurement data from a chemical tanker. It is followed by conclusions in Section 5.

2. Overview of the isochrone algorithm for voyage optimisation

A computer-assisted ship voyage planning/optimisation system normally contains several components as listed in Figure 1. For example, the sailing constraints, e.g. land avoidance, no-go zones, traffic separation scheme, etc., are used to define and generate a waypoint grid for optimal route searching. The metocean environment and ship performance models are used to estimate the cost by the cost functions corresponding to various optimisation objectives, such as ETA, minimum fuel consumption, etc. The waypoint grid may contain huge amounts of waypoints required to estimate their associated sailing costs. Thus, for a voyage optimisation system, the requirement of computational effort can easily exceed a computer's capacity. Consequently, the critical component in the voyage planning system is a proper optimisation algorithm.

In this study, the objective of the voyage optimisation algorithm is to achieve energy-efficient sailing with arrival punctuality. Other important factors related to practical ship navigations, such as ship motions, collision avoidances, maritime service fees, etc., with proper models to describe the cost functions associated with those factors, the proposed isochrone method should have the capability to consider this multi-objective voyage optimisation problem. However, they are outside the scope of this study and will be investigated in our future research activities. In addition, detailed navigation plannings, such as the traffic separation zones, keel clearance, or real-time navigation warnings from the ECDIS system, etc., are only partly covered in the method development. But they can be smoothly implemented in the method by considering the bathymetric maps and tidal currents, as well as setting proper reference routes. Those implementation issues for actual maritime applications are not fully considered in the current study.

2.1. Overview of isochrone voyage optimisation

For the isochrone optimisation methods in this paper, the word 'isochrone' indicates the contour lines that the ship can reach with equal sailing time. The isochrone voyage optimisation method was initially proposed by (James 1957) for manual use, to help ships navigate to the destination as soon as possible, or on the accurate ETA. The general procedures of the isochrone voyage optimisation method are as follows. Firstly, a reference route, which can be either the shortest route (the great circle route) or a typical sailing route by experience, should be chosen to guide the search space. Then, a ship's voyage is divided into different time stages. For each time stage starting from the departure, every waypoint moves forward to the next time stage (of equal sailing time) with several candidate options (subsequent waypoints) as in Figure 2. And from the next time stage, the whole process is repeated in such a recursive way until the destination is reached.

However, for each time stage, if all generated waypoints for the next stage are kept, the total number of potential waypoints will grow exponentially. Overcoming the 'curse of dimension' motivates researchers to improve the isochrone method for actual ship voyage

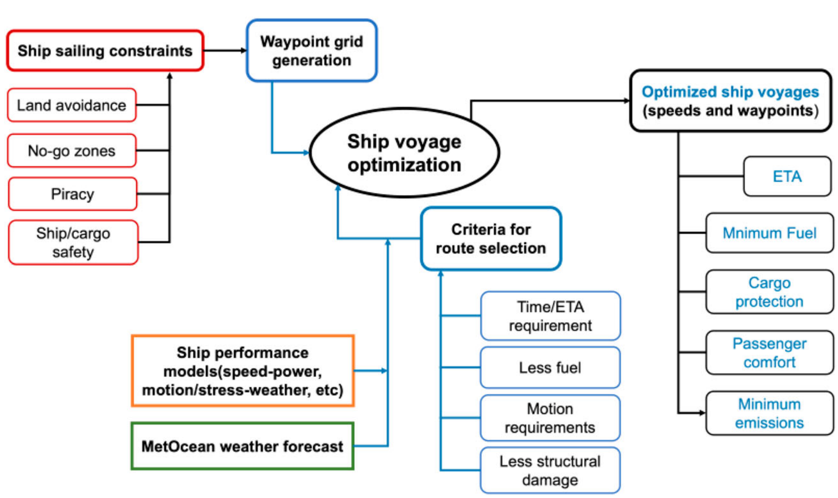

Figure 1. An overall scheme of a typical ship voyage planning/optimisation system (This figure is available in colour online.).

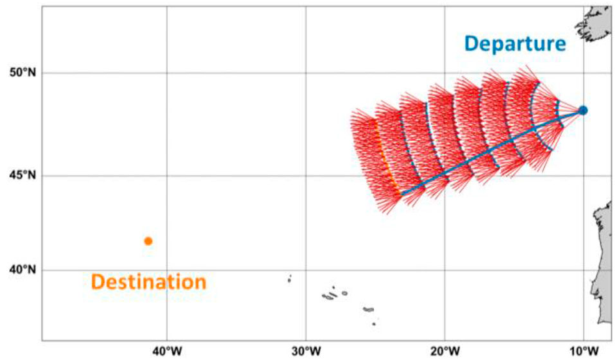

Figure 2. A waypoint grid system generated by an isochrone voyage optimisation method (This figure is available in colour online.).

optimisation applications. (Hagiwara 1989) introduced the concept of subsectors helping to select several optimal waypoints at each time stage. The subsectors are defined as parallel lines of equal spacing on both sides of the reference route. At each stage, only the best waypoint is chosen in each subsector to compose the next isochrone/time stage. The number of waypoints is restricted, and a ship is also supposed to move within the area defined by subsectors.

2.2. Procedures of the isochrone voyage optimisation algorithm

The waypoint/grid generation procedure of the original isochrone voyage optimisation method (Hagiwara 1989) (denoted as Isochrone method hereafter) is illustrated in Figure 3, where the ship speed is assumed to be fixed along the voyage unless encountering harsh weather conditions (with involuntary speed reduction). The parameters used to define the grid system are listed in Table 1. Let the departure point denote by $X_0 = [x_0, y_0]$, the destination by $X_f = [x_f, y_f]$, and the great circle route between X_0 and X_f is chosen as the reference route as in Figure 3. Let a ship's service speed denote by V, the isochrone voyage optimisation can be conducted as follows:

(1) From $X_0 = [x_0, y_0]$ at the departure time T_0, a ship sails to the 1st time stage ($T_0+\Delta t$), following $2m + 1$ course headings $C_{ref} \pm i{\cdot}\Delta C$ ($i = 0, 1, \dots, m$) respectively, where C_{ref} is the course heading at X_0 along the reference route.
(2) The newly generated waypoints should be checked with the sailing constraints as in Figure 1: land-crossing/shallow water, no-go zones, or safety factors (such as sea ice in arctic sailing). Then, all feasible waypoints form the first isochrone as $\{\boldsymbol{X}_1(i), i = 1, 2, \dots, 2m + 1\}$.
(3) For each (i-th) waypoint in the current isochrone $\{\boldsymbol{X}_1(i)\}$, repeat Step (1) to get new waypoints for the 2nd time stage $\{\boldsymbol{X}_2(i, j), j = 1, 2, \dots, 2m + 1\}$, and evaluate the cost of all generated new waypoints.
(4) Define a set of subsectors, starting from X_0 following $2k + 1$ course headings $C_{ref} \pm s{\cdot}\Delta S_n$ ($s = 0, 1, \dots, k$), here $n = 2$ represents the 2nd time stage, as in Figure 3. The subsector range ΔS_n ($n = 2$) is defined by:

$$\Delta S_n = \frac{c \cdot \Delta D}{\sin(c \cdot d_n)}, \; C = \frac{\pi}{60 \times 180}, \tag{1}$$

where d_n is the expected sailing distance in the n^{th} time stage of isochrone after $n{\cdot}\Delta t$ hours:

$$d_n = n \cdot \Delta t \cdot V. \tag{2}$$

The subsectors $\{S_n(s), n = 2, s = 1, 2, \dots, 2k\}$ are represented by the area between each pair of adjacent headings $C_{ref} + (s\text{-}k\text{-}1) \cdot\Delta S_n$ and $C_{ref} + (s\text{-}k) \cdot\Delta S_n$ ($n = 2$; $s = 1, 2, \dots, 2k$).
(5) In each (s-th) subsector $S_n(s)$ ($n = 2$), preserve only one optimal waypoint according to the costs associated with all the generated waypoints $\{\boldsymbol{X}_n(i, j), n = 2\}$. All optimal waypoints in each subsector compose the 2nd/next isochrone $\{\boldsymbol{X}_n(s), n = 2, s = 1, 2, \dots, 2k\}$.
(6) Repeat Steps (2) and (3) for the n-th time stage ($n = 3, 4, \dots$), recursively, to first generate all waypoints $\{\boldsymbol{X}_n(i, j), i = 1, 2, \dots, 2k, j = 1, 2, \dots, 2m + 1\}$. Then from all those waypoints, select the n-th isochrone as $\{\boldsymbol{X}_n(s), s = 1, 2, \dots, 2k\}$. When the geographical distance from the current isochrone to X_f is less than $\Delta t{\cdot}V$, connect all the waypoints directly to $X_f = [x_f, y_f]$.
(7) X_f can be reached through a set of potential routes from X_0. For energy-efficient sailing, the optimal route is identified as the route with the minimum fuel consumption. The weather impacts are considered as changed fuel costs, to either avoid harsh weather or utilise the ocean current.

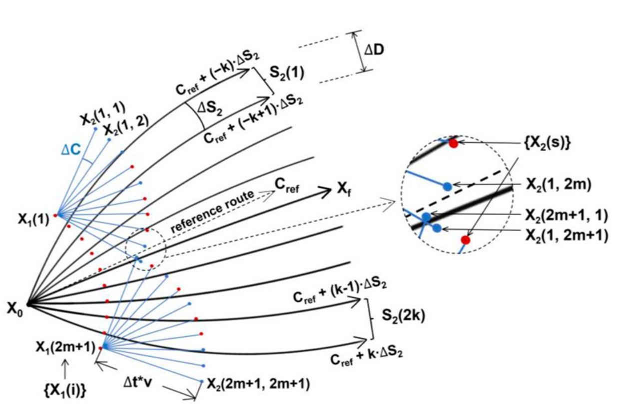

Figure 3. Graphic interpretation for the generation of the second isochrone$\{\boldsymbol{X}_2(s)\}$ in the Isochrone method (This figure is available in colour online.).

Table 1. Parameter of isochrone algorithm.

Δt	Sailing time between two adjacent time stages
ΔC	The increment of heading angles between two adjacent sub-routes from each of the current 'optimal' waypoints at each time stage.
$2m + 1$	Number of successor waypoints for each waypoint at current stage
$2k$	Number of subsectors
ΔD	The width of the searching limit within each local subsector

2.3. Parameter sensitivity in the isochrone method

In this method, five essential parameters in Table 1 should be well specified, as they control the generation of waypoints in the search grid, thereby significantly influencing voyage optimisation results,

- ***Δt***: It controls the looseness of the search grid along the direction toward the destination, i.e. the step size of the dynamic route-finding process. Large Δt may avoid local optimums as well as converge faster towards the destination. However, it may cause abrupt turns in the candidate routes.
- ***ΔC***: It specifies the step size for generating the successors from the current waypoints. When ΔC is larger, newly generated waypoints will expand wider. However, since new waypoints can also reach more subsectors, too large values of ΔC may lead to locally optimised predecessor waypoints, as other candidate

waypoints in nearby subsectors may be ruled out, and finally lead to partly overlapped routes.

- ***m***: Corresponding to the number of successors for the current waypoint. If m is small, the search area ahead of each current waypoint might not be efficiently covered by enough successors. And if it is large, the search performance might be significantly reduced, since great computation effort is needed to search big areas/waypoints.
- ***k***: It defines the number of subsectors. Similar to m, larger k (more subsectors) covers a broader search area when the whole voyage search progresses. It may improve performance by generating more potentially feasible routes but may also increase the computational effort of the process.
- ***ΔD***: It defines the width of the single subsector and, hence controls the width of the search grid along the voyage. Small ΔD indicates a narrow searching range perpendicular to the reference route which may be insufficient for optimal route-finding, while large ΔD can lead to sharp turns in routes toward the destination.

All those parameters should be chosen within a reasonable range to allow for efficient voyage optimisation. Their values are generally dependent on the length of the voyage and weather dynamics. Increasing the values of those parameters may have a positive impact on optimisation results, but a trade-off between performance and computational efficiency should also be considered. Generally, the Isochrone voyage optimisation method by (Hagiwara 1989) can eliminate the phenomenon of the 'curse of dimension'. However, some shortcomings can still be observed. For example, sub-routes will continuously widen as they progress, and grow to cover a vast search range, as seen in Figure 3. When nearing the vicinity of the endpoint, waypoints are directly connected to the destination. Thus, routes with sharp turns around the destination would easily appear, as shown in Figure 4. Most of those candidate routes with abrupt changes in direction are not realistic for practical voyage planning. In this study, several strategies to improve the Isochrone method are investigated and discussed as follows.

3. Strategies for improvement

For a ship voyage optimisation method, two key components, i.e. the waypoint grid generation and the cost/weight functions associated with each waypoint, can significantly influence its result and performance. In this study, five strategies to modify those two components are investigated for their capability to improve the original Isochrone method in ship voyage optimisation. A flowchart for different modification strategies is presented in Figure 5.

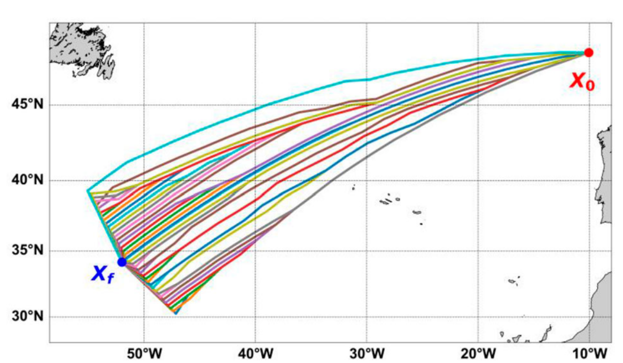

Figure 4. Potential optimal routes generated by the Isochrone method (Hagiwara 1989) (This figure is available in colour online.).

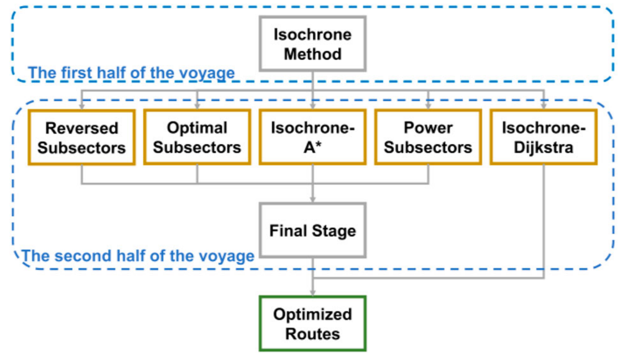

Figure 5. Concepts for different improvement strategies of the Isochrone method for voyage optimisation (This figure is available in colour online.).

In all the modification methods, the Isochrone optimisation process divides a ship's voyage planning into two parts from the departure to the destination. In the first half voyage, the Isochrone voyage optimisation method by (Hagiwara 1989) is used. Then five strategies are introduced in the second half voyage to,

(1) avoid the route convergence problem as presented in Figure 4,
(2) maintain fast optimisation/computation efficiency,
(3) improved optimisation results.

The five improvement strategies are briefly summarised in Table 2 and will be presented respectively in the following subsections.

3.1. Reversed subsectors

The sharp turns near the destination as in Figure 4 are caused by the fact that the waypoints searching process, in the original Isochrone voyage optimisation method Hagiwara (1989), involves only expansion without a convergence process. The subsectors are defined as a monotonically increasing function of sailing distance d_n, i.e. distance from departure to the n-th time stage as in Equation (2). Since the subsectors guide the search areas during the voyage optimisation process, reconstructing subsectors in the late stages of a voyage is proposed in this study to resolve the convergence problem. In the second half of the voyage, the sailing distance from the departure X_0, i.e. d_n is replaced by the distance to X_f (denoted as d_{NS}). It is then used to define the width of the following subsectors in the second half of a voyage as,

$$\Delta S_{NS} = \frac{c \cdot \Delta D}{\sin(c \cdot d_{NS})},\ d_{NS} = d_{total} - d_n, \tag{3}$$

where d_{total} is the total distance from X_0 to X_f along the reference route, ΔD is the maximum local subsector width (i.e. resolution of the isochrone as in Figure 3) and indicates the width limit for each subsector in the distance, and c is a constant defined in Equation (2). Note that subsectors are given by intervals of the headings, hence ΔS_{NS} represents the range for each subsector in headings at the n^{th} stage. The subsectors in the first half voyage, together with the reversed subsectors in the second half are shown in Figure 6. A reversed and symmetric subsector set is generated, which gradually reduces its range when approaching the endpoint Figure 7.

Finally, the complete procedure for this modified isochrone method can be executed as follows:

(1) Follow the steps in Section 2.2 for the first half of a voyage.
(2) When the distance from the current time stage to X_f is less than half of the total distance d_{total}, reversed subsectors are

Table 2. Methods used in different parts of the voyage planning/optimisation process.

Name of modification	First half voyage	Second half voyage		Final Stage
		Subsector	Searching method	
Reversed subsectors	Isochrone method	Reversed subsectors	Isochrone method	ΔC = ΔC*10%
Optimal subsectors			Reserve suboptimal nodes	
Isochrone-A*			Augmented heuristic function	
Power subsectors		-	Optimal power greedy search	
Isochrone-Dijkstra		Dijkstra algorithm		

constructed using the back course heading angles as $C_{inv} \pm s{\cdot}\Delta S_{NS}$ ($s = 0, 1, \dots, k$), where C_{inv} is the azimuth angle of the back course at X_f, i.e. from X_f to X_0 along the reference route, and ΔS_{NS} is defined in (3). See Figure 6 (right part) for references of reversed subsectors.

(3) Then, $2k$ reversed subsectors $\{S_{NS}(s), s = 1, 2, \dots, 2k\}$ are represented by the area between each pair of the adjacent back course heading angles $C_{inv} + (s\text{-}k\text{-}1) \cdot\Delta S_{NS}$ and $C_{inv} + (s\text{-}k) \cdot\Delta S_{NS}$ ($s = 1, 2, \dots, 2k$).

(4) From each isochrone waypoint in the current time stage, i.e. $\{\boldsymbol{X}_{n\text{-}1}(s), s = 1, 2, \dots, 2k\}$, generate new waypoints for the n-th time stage towards X_f following course headings $C_{ni} \pm i{\cdot}\Delta C$ ($i = 0, 1, \dots, m$), where C_{ni} denotes the initial course at the current waypoint $\{X_{n\text{-}1}(s), s = 1, 2, \dots, 2k\}$, along the great circle route from $X_{n\text{-}1}(s)$ to X_f. The new waypoints at the n-th time stage are denoted by $\{\boldsymbol{X}_n\ (i, j), i = 1, 2, \dots, 2k; j = 1, 2, \dots, 2m + 1\}$.

(5) Evaluate the cost of each waypoint $\{\boldsymbol{X}_n\ (i, j)\}$. In each ($s$-th) subsector $S_{NS}(s)$, choose the one with the optimal cost to form the isochrone $\{\boldsymbol{X}_n(s)\}$ at the n-th time stage, as a waypoint member. The optimal waypoint selection criterion is the shortest distance to the destination, to avoid the detour leading to a high fuel consumption.

(6) Repeat steps (3) and (4). Once the current distance to X_f is less than $3{\cdot}\Delta t{\cdot}V$, decrease ΔC to $0.1{\cdot}\Delta C$, since the reverse subsectors progressively narrow down and will be compact around X_f.

(7) When the distance to X_f is less than $\Delta t{\cdot}V$, connect current waypoints to X_f as the final sub-route.

3.2. Optimal subsectors

As illustrated in Figure 6, the convergence problem of Isochrone method in Figure 4 might be solved by constructing reversed subsectors. However, when a voyage optimisation process approaches the destination, subsectors at the very late time stages can become significantly narrow. Subsequently, very few of the current isochrone waypoints, i.e. $\{\boldsymbol{X}_n(s), s = 1, 2, .., 2k$ and k is very small$\}$, might be chosen with successor waypoints for forming the next isochrone $\{\boldsymbol{X}_{n+1}(s)\}$, since waypoints with lower costs from the current stage will have temporary optimality (such as the waypoint $X_n(a)$) in the illustration in Figure 8. A great part of waypoints in the next isochrone $\{\boldsymbol{X}_{n+1}(s)\}$ are successors of $X_n(a)$. Therefore, sub-routes generated afterward may all originate from very limited waypoints $\{\boldsymbol{X}_n(s)\}$. This phenomenon could be considered as the route search being trapped in a local optimisation. However, an ideal optimisation algorithm should be able to search candidate routes covering sufficient sailing areas.

To overcome the possible problem from the above modification, the following procedures named optimal subsections are proposed as follows, with a corresponding flowchart in Figure 9:

(1) The generation of the waypoint grid will be carried out as in Section 3.1, with the optimal waypoint selection criterion as the shortest distance to the destination.

(2) In the latter half voyage, the number of nodes/waypoints preserved in each subsector is increased and controlled by an extra parameter to prevent the waypoints from an exponential increase in quantity. It is set to 3 for the case study voyages in the following analysis.

(3) Every predecessor waypoint is only allowed to keep a limited number of its successors, avoiding its domination, i.e. for a

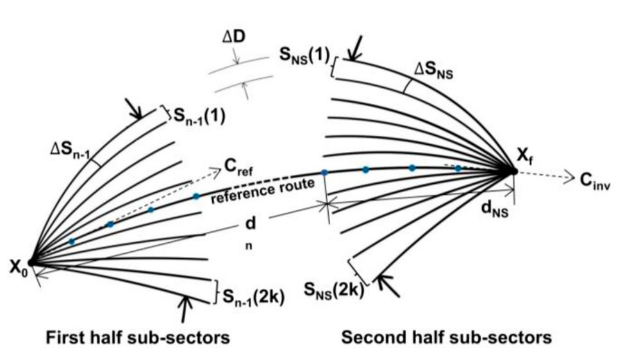

Figure 6. Definition of subsectors along the voyage (This figure is available in colour online.).

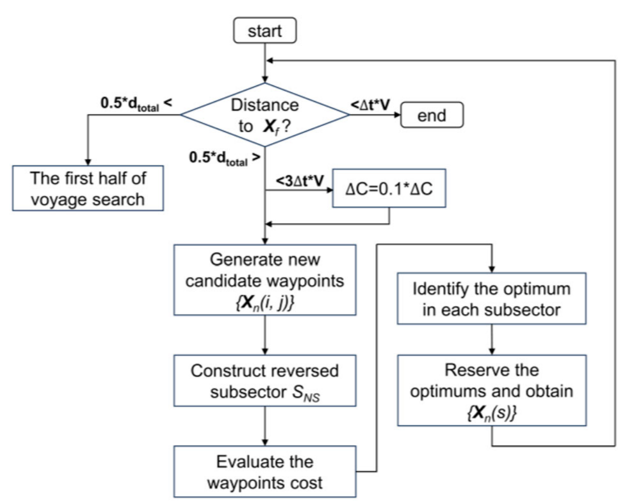

Figure 7. The flowchart of Reversed subsector method (This figure is available in colour online.).

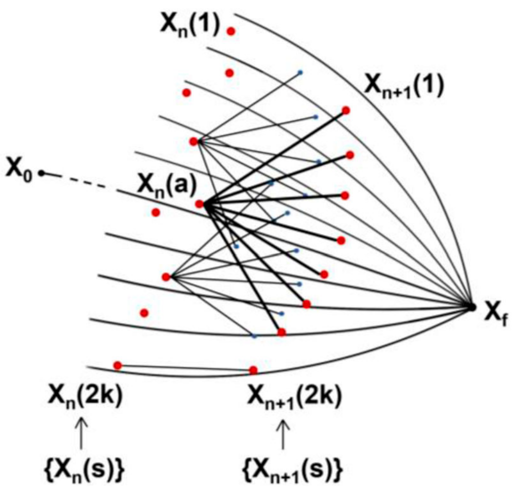

Figure 8. Illustration of a waypoint with temporary optimality (This figure is available in colour online.).

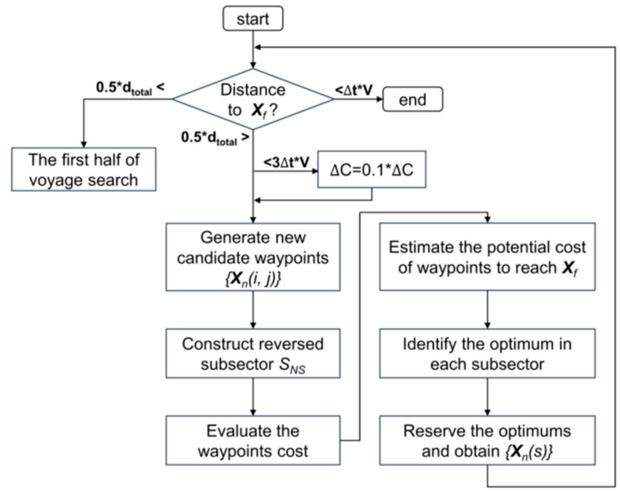

Figure 10. The flowchart of Isochrone-A* method (This figure is available in colour online.).

preceding waypoint, the number of successors preserved in the next stage is restricted. It is set to 5 for the case study.

This approach provides candidate waypoints with a higher probability of surviving till the final time stages of a voyage, while also restricting the domination of one waypoint at its following time stage. Thus, this approach is proposed to prevent temporary optimality of choosing only a few dominating waypoints.

3.3. Isochrone-A* method

In addition to modifying the waypoints grid system by the redefinition of subsectors, another approach is to explore various criteria that are used for selecting optimal waypoints at each subsector corresponding to certain cost functions. Most of today's selection criteria for isochrone voyage optimisation methods are defined as either the shortest distance to the destination, or minimum fuel consumed to the current waypoints. They encompass solely the information from the previous part of the voyage. This study will investigate how to augment a heuristic term to also encompass future considerations.

The ideas of A* algorithms are implemented in this improvement strategy to consider the future cost of potential candidate sub-routes. A* is a conventional and widely used graph-searching algorithm. It is an informed search algorithm and its evaluation function for selection criteria involves both forward and backward cost estimation along searching sub-routes:

$$f(n) = g(n) + h(n), \tag{4}$$

where $g(n)$ is the cost from the departure, and $h(n)$ is the heuristic term to estimate the cost to reach the destination. Within the proposed Isochrone-A* method, the waypoint grid system is generated by the same approach as Section 3.1. And the changes are taken in formulating the evaluation criterion:

(1) In the first half of the voyage, the evaluation criterion $f(n)$ for optimal selection is kept as the shortest distance to the destination, to avoid detours in the early stages.
(2) In the second half voyage, the evaluation criterion $f(n)$ is formulated to involve the heuristic term $h(n)$:
- $g(n)$: The accumulative fuel consumption from departure.
- $h(n)$: The estimation of the fuel consumption to the destination. Assuming departing from the current waypoint towards X_f through the great circle route, the fuel cost is estimated based on dynamic weather updates at each time stage.
- $f(n)$: The estimated fuel consumption for the entire voyage.

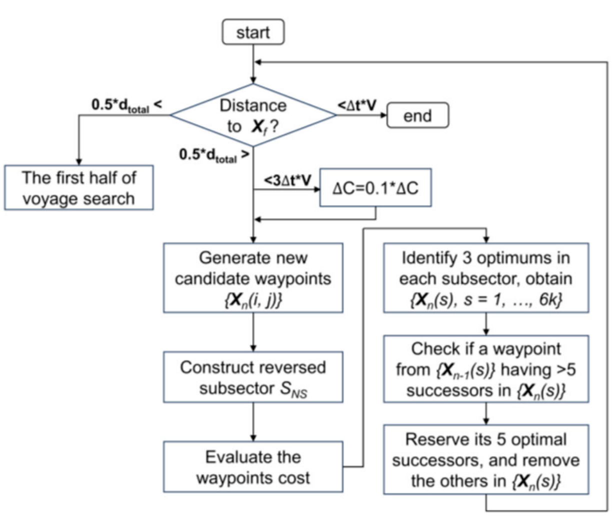

Figure 9. The flowchart of Optimal subsector method (This figure is available in colour online.).

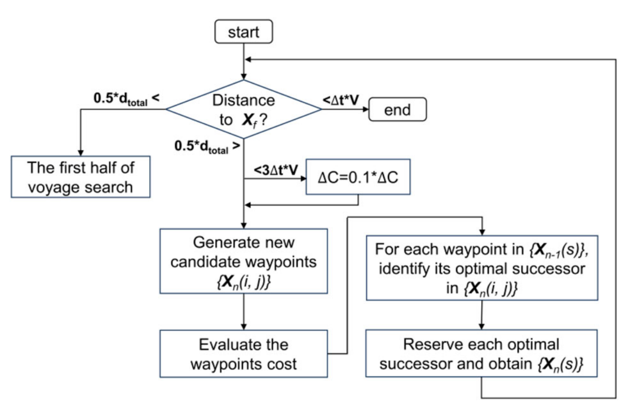

Figure 11. The flowchart of Power subsector method (This figure is available in colour online.).

The overall procedures are presented in the flowchart in Figure 10. Thus, by taking an overall view of the whole route in the waypoint selection, this approach improves the Isochrone method to avoid the local optimum, based on the strategy in Section 3.1.

3.4. Power subsectors

To avoid the problems of temporary suboptimality as in Figure 8, an alternative way of limiting the number of waypoints in the latter half of a voyage is considered, i.e. the greedy dynamic search method. It is named as 'power subsectors' here, which select the optimal point among the successors for each waypoint in the current isochrone. This indicates each of the preceding waypoints will form a feasible route to reach X_f. This approach can be carried out as follows, as in Figure 11.

Firstly, in the first half of the voyage, an optimal point in each subsector is chosen as the one with the shortest distance to the destination. Secondly, for the latter half of the voyage, every waypoint proceeds towards X_f following the heading $C_{ni} \pm j{\cdot}\Delta C$ ($j = 0, 1, \ldots, m$). Then, among all $2m + 1$ generated successors, select the optimal point with the lowest fuel consumption and append it to the grid as the next generation. Continue until the destination. The dynamic grid of isochrone in the first half will reach its widest in the middle, thereby avoiding local optimum.

3.5. Isochrone-Dijkstra method

In the isochrone algorithm, sub-routes connecting a waypoint at the current stage are solely associated with the connecting waypoints at the previous time stage. Once a waypoint at a current stage is removed, all its predecessors will be automatically deleted. It may significantly reduce the search area when a voyage converges to its destination. On the other side, allowing a large spread of searching as in the original Isochrone optimisation method leads to unrealistic route planning as in Figure 4. Therefore, this study explores the implementation of Dijkstra's algorithm for the second half of a voyage to solve those two issues.

The Dijkstra method is a graph-searching algorithm, where a static grid is first established based on the sailing area. Edges are connected between waypoints in adjacent time stages, and the cost of these edges (or sub-routes) are assigned corresponding to e.g. fuel consumption along the edges. Distinguished from the isochrone algorithm, there is no natural binding between waypoints. The Dijkstra methods progressively explore adjacent waypoints and update the lowest cost to reach each waypoint based on the cumulative cost. With the static grid, the Dijkstra algorithm can find the lowest cost route between

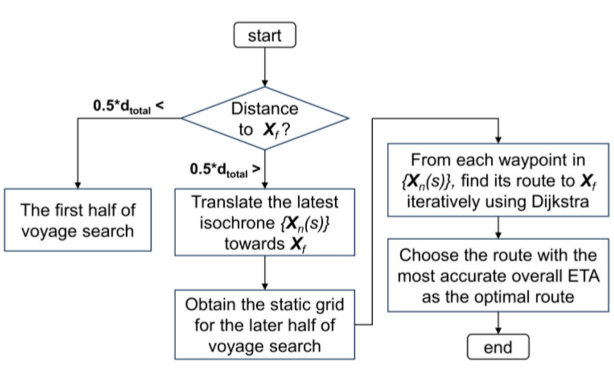

Figure 13. The flowchart of Isochrone-Dijkstra method (This figure is available in colour online.).

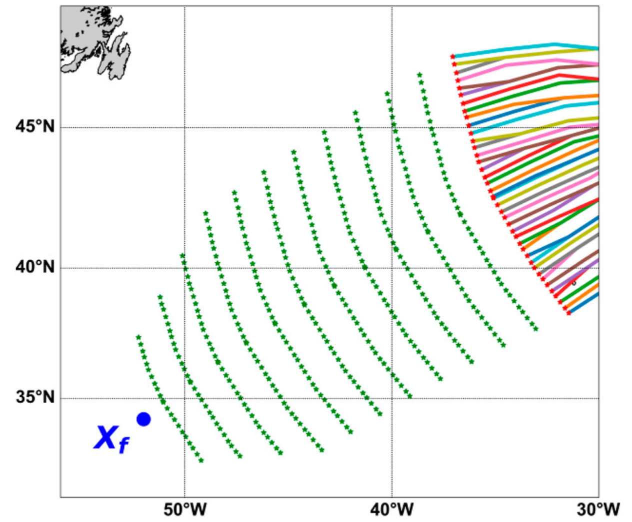

Figure 12. Static grid generated to deploy the Dijkstra algorithm (This figure is available in colour online.).

two waypoints since it enumerates every possibility in its predefined grid. This approach can be executed followed by the flowchart in Figure 13.

(1) In the first half of a voyage, choose an optimal waypoint in each subsector with the shortest distance to the destination to compose the new isochrones.
(2) When reaching the second half of the voyage, a static grid is generated based on the waypoints in the latest isochrone, as illustrated in Figure 12. The candidate waypoints in the subsequent stages are obtained by translating the latest isochrone along the direction of the GC_{ref} towards X_f, for the route convergence.
(3) For each waypoint in the latest isochrone as the starting point, assign cost to the estimated fuel consumption for all sub-routes, and search for the lowest cost route to X_f, using the Dijkstra algorithm. Then, several potential sailing routes will be obtained.
(4) These candidate routes would possess different times of arrival, since the travelling distance varies in sub-routes, and sailing speed is assumed constant. In this approach, the optimal route will be chosen as the route with the most accurate ETA as required.

4. Comparison of improved isochrone methods

A chemical tanker with full-scale measurement is used in the case study, to compare the capability in voyage optimisation using the isochrone methods improved by the five proposed strategies. A conventional weather routing system was installed on the ship to provide guidance on her voyage planning. Combined with the ship master's experience, the actual sailing routes are supposed to be more efficient than ordinary voyage planning systems. Two winter and two summer voyages of the ship, measured from 2015 to 2016, are selected as the case study voyages for comparison, as shown in Figure 14. To be more representative, the four voyages cover two westbound and eastbound voyages with diverse environmental conditions, i.e. in harsh, typical, and calm sea states respectively.

In this study, the objective of voyage optimisation is set as the minimum fuel consumption along a voyage, and meanwhile, the

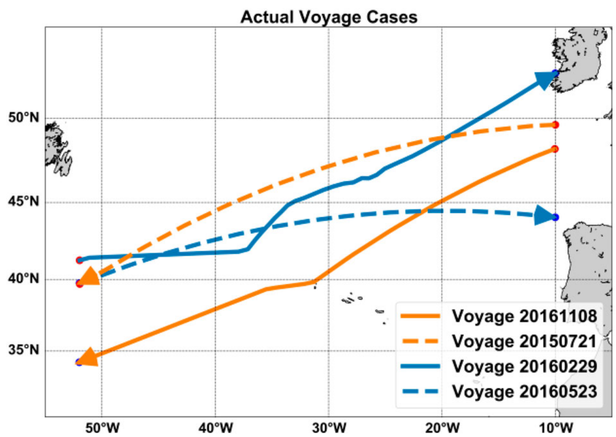

Figure 14. Case study voyages used for comparison (This figure is available in colour online.).

ETA of voyage planning is set the same as the selected case study voyages. The cost function for fuel estimation demands all encountered MetOcean environmental data inputs (wind, wave, and current), as well as a ship performance model, which provides the ship's speed and fuel consumption relationship. In this case study, related MetOcean parameters are extracted from ECMWF ERA-5 (2019) dataset for wind and wave, and ocean current data is acquired from Copernicus 2019 server. Furthermore, the ship energy performance model, which is a semi-empirical model developed by (Lang and Mao 2020; Lang and Mao 2021), is used in the Isochrone optimisation methods.

To get the best optimisation results, the parameters in the Isochrone optimisation algorithm are selected based on hyper-parameter investigation. The values are listed in Table 3. Generally, the parameter Δt can initially be determined to divide the voyage into 20 time-stages. Following that, ΔC, m, ΔD, and k can be chosen referring to the actual voyage sailing range/width, and the general sea state during the voyage. As in Section 2.3, the search range for a single waypoint is defined by m multiplies ΔC, and the overall search width is restricted by k multiplies ΔD. ΔC and ΔD indicate the step size. Calm sea environments sailing tends to be near the great circle route to save distance, thus its search range can be assigned small, with looser/larger step sizes, as Voyage20150721 and 20160523; otherwise, the values can be set to allow a wider range search in smaller steps, as Voyage20161108 and 20160229.

4.1. Optimisation results for the westbound voyages

Two westbound case study voyages, one during winter Voyage20161108 and one during summer Voyage20150721, are investigated in this section. The optimisation results in terms of fuel consumption, sailing time (ETA), and sailing distance are listed in Table 4. The corresponding voyages are presented in Figure 15. For both westbound voyages, the sea states encountered by their actual route are generally mild, i.e. H_s (the significant wave height) is less than 3.5 metres. This should be attributed to the careful planning by the experienced captain assisted by an onboard

Table 3. Parameters of Isochrone algorithm for the voyages.

Voyage name	ΔC [degree]	m	Δt [h]	ΔD	k
Voyage20161108	0.8	10	8	9	10
Voyage20150721	0.3	15	7	5	15
Voyage20160229	0.5	30	8	6	30
Voyage20160523	0.2	10	8	6	10

routing system. Meanwhile, actual routes do not deviate much from the great circle route, thereby their total sailing distances are near the shortest achievable distance. It may indicate the actual routes were well optimised.

For the Voyage20150721, all the Isochrone based optimisation methods could further reduce the fuel consumption than the actual route, from 3.9% to 5.7%. But for the Voyage20161108, some modified Isochrone methods work better, such as Isochrone-A*, Power subsectors, and Isochrone-Dijkstra methods. Specifically, Isochrone-A* shows the best performance with 2.6% energy improvement. From these two westbound cases, Isochrone-A* method provides the most energy-efficient route in voyage optimisation. On the contrary, the performance of the original Isochrone method is not very promising. It gives the most fuel consumption with a long sailing distance, and sharp turns also appear in its route when approaching the destination in both cases in Figure 15. In addition, Optimal subsectors behaves nearly identically to Reversed subsectors method due to their greatly resembled routes and close amount of fuel expense in two cases.

To further investigate the optimisation results, the details of the winter Voyage20161108 are presented in Figures 15–17. During this voyage, the sailing conditions are quite calm with the highest wave of 2.6 metres. As shown in Figure 15, the optimal routes provided by the three methods, i.e. Isochrone-A*, Isochrone-Dijkstra, and Power subsectors, are similar in most of the parts with minor deviations. While the other two methods give almost identical results. This coincides with the optimisation results given in Table 4. For encountered H_s shown in Figure 16, the most noticeable weather variations occur between longitude −35°W to −30°W for all optimised voyages.

In Figure 17, high waves are observed in the northern part of the sailing area. The actual route differs from the great circle route and has altered its course twice, to stay in the calm sea state. However, its detour causes a longer sailing distance and slightly higher average speed to meet the same ETA. Three routes sail closer to high waves area, i.e. routes from Reversed subsectors, Optimal subsectors, and Isochrone method. Since high waves are in the vicinity of the great circle route, it implies that these routes prioritise short distances over avoiding high. However, they show more fuel consumption than the actual route. Finally, Isochrone-A*, Power subsectors, and Isochrone-Dijkstra methods can give improved results than the actual route. The other Voyage 20150721 shows similar results as sailing in relatively calm sea environments, with the highest encountered H_s of 3.5 metres. The actual route follows the great circle route. Meanwhile, routes suggested by other optimisation methods diverge slightly toward north at around longitude −30°W. They have encountered similar sea weather showing more similar optimisation results. While the Isochrone-A* method gives the best optimisation results with 5.7% fuel saving.

4.2. Optimisation results for the eastbound voyages

The optimisation results for two eastbound cases, Voyage 20160229 and 20160523, are presented in this section to show the rough and ordinary sailing environments respectively. The optimised routes obtained by different approaches are presented in Figure 18. The corresponding ETA, fuel consumption, and travel distance are listed in Table 5.

The chosen eastbound voyages have encountered harsh sea conditions when sailing in the North Atlantic, especially the winter Voyage20160229 representing a voyage sailing in extremely rough weather conditions. Its actual sailing encountered H_s of 9 metres. It was planned to avoid the storm by first heading a bit south of

Table 4. Results of modified Isochrone algorithms, actual routes and 2DDA, for the two westbound voyages.

	Voyage20161108				Voyage20150721			
Optimisation methods	ETA [h]	Fuel [ton]	Distance [km]	Average Speed [knot]	ETA [h]	Fuel [ton]	Distance [km]	Average Speed [knot]
Actual Route	164.3	159.7	3877.5	12.8	139.8	177.7	3453.6	13.4
2DDA	172.5	154.2	4024.3	12.6	146.5	161.6	3660.9	13.5
Isochrone method	167.8	162.0	3896.1	12.5	142.4	170.8	3533.5	13.4
Reversed subsectors	164.8	163.2	3807.2	12.5	139.8	168.5	3474.3	13.4
Optimal subsectors	164.4	162.9	3798.5	12.5	139.8	168.5	3474.3	13.4
Power subsectors	165.4	156.1	3840.7	12.5	139.9	168.3	3482.3	13.4
Isochrone-A*	165.1	155.6	3836.1	12.5	140.0	167.5	3487.1	13.5
Isochrone-Dijkstra	165.1	155.7	3834.3	12.5	140.0	168.7	3478.0	13.4

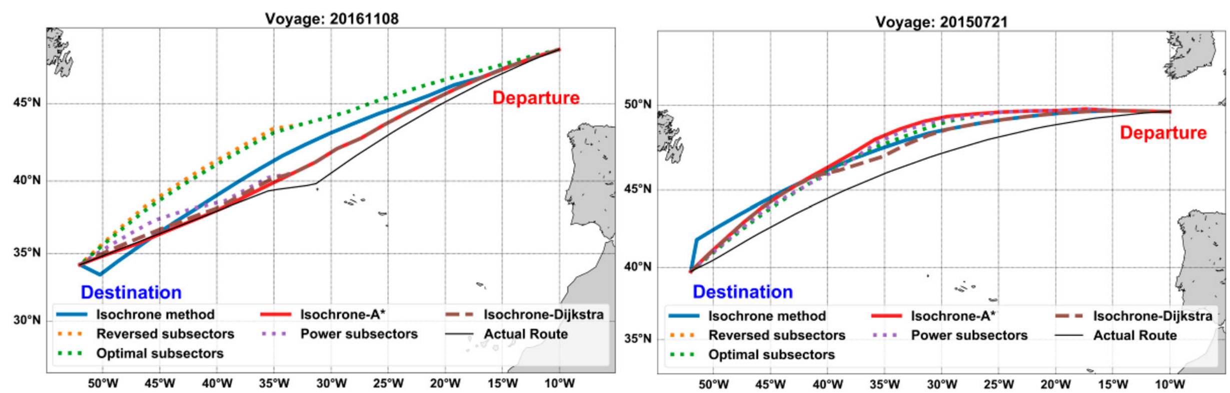

Figure 15. Optimised routes of Voyage 20161108 (left) and 20150721 (right) by different methods.

the great circle route, then turning back and heading toward the destination. Another relatively normal sailing environment is also introduced as the summer Voyage20160523, where the highest H_s reaches around 5 metres. This is rather a common sailing environment in the North Atlantic. For sailing conditions in these cases, the fuel consumption can change dramatically if the voyage planning is not efficient. However, all the proposed methods could show considerable improvements in comparison with the actual routes. As shown in Table 5 for the winter Voyage20160229, Isochrone-A* saves the most fuel up to about 9.0% than the actual route. And for the summer Voyage 20160523, all the optimisation methods present a similar fuel consumption, among which the Isochrone-A* method results in the most savings at 3.8%. Only the original Isochrone method shows higher fuel consumption than the actual route.

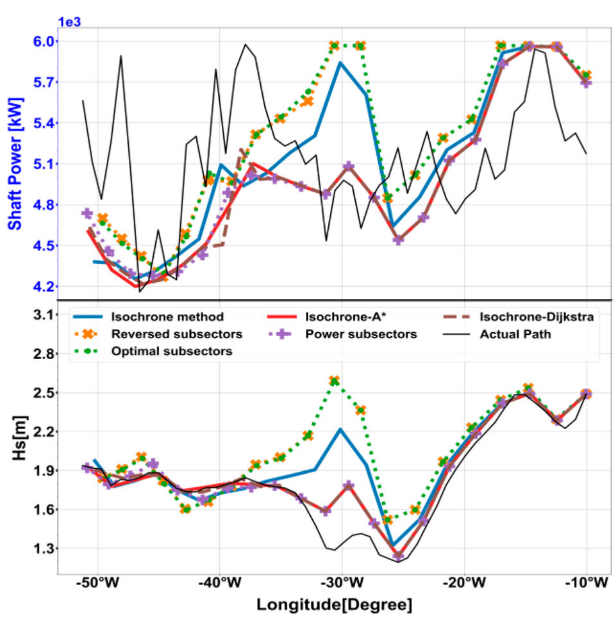

Figure 16. Shaft power and encountered H_s in Voyage 20161108 (This figure is available in colour online.).

More details of the winter voyage 20160229 are presented in Figures 19 and 20, as this case involves severe weather changes. During this voyage, the ship experienced two major storms. The first storm appeared between longitude −40°W to −30°W. To avoid encountering it, four planning methods, i.e. Isochrone method, Reversed subsector, Optimal subsectors, and Isochrone-Dijkstra, suggest sailing head up-north first, while the other three methods recommend moving south. All planned trajectories are split into two groups and advance in two directions. And Figure 19 shows that these northern voyages come across lower H_s as intended and avoid more of the first storm compared with southern voyages. For the second storm between longitude −30°W to −20°W, however, three southern voyages mostly avoid its centre region and could maintain the constant speed with lower shaft power of engine. However, for four northern voyages, the impact of the second storm is greater, as they must employ more engine power in higher H_s to hold a constant speed, from longitude −25°W to the destination.

This dynamic voyage planning process is shown with the contour plot of H_s in Figure 20. The second storm moves from south to north, passes in front of three southern routes, and then hits four northern routes. Therefore, two southern routes, Isochrone-A* and Power subsectors, are more energy efficient compared with the planned northern routes. The actual route, although first heading south to consider the impact of the second storm, does not completely avoid high H_s from the first storm while also running a long distance, thus is not fuel-efficient. There are two peaks in the shaft power plot of the actual route, while the shaft power of other voyages does not increase greatly due to the

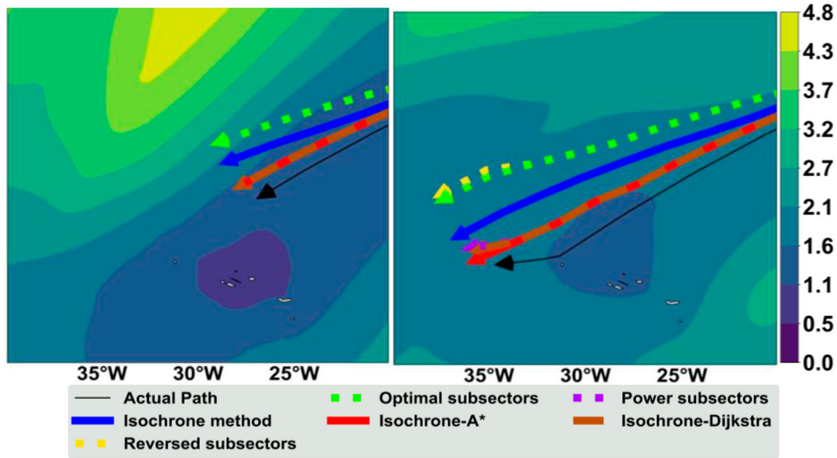

Figure 17. Weather conditions along the evolution for different optimal planning of Voyage 20161108 (This figure is available in colour online.).

constraint on the engine; thus, their sailing speeds are involuntarily reduced at different levels. Again, the Isochrone method generates a route with an obvious abrupt turn near the destination which is impractical for operation. In this case, all strategies can improve the Isochrone method to not only suggest a smooth route, but also meet the objectives of more fuel saving and accurate ETA. Isochrone-A* shows the maximum fuel saving with a 9.0% reduction.

For the eastbound summer Voyage20160523, the main difference arises between longitude −30°W to −25°W with the appearance of the highest H_s during the voyage. The sea states encountered within this area lead to different final fuel consumptions. The overall encountering time is not long; thus, the result of every method is close as in Table 5. The actual route opts for short distances for fuel saving. By slightly deviating in routes, voyages from optimisation methods achieve a certain level of fuel reduction. The four improved Isochrone methods give similar results with around 3% savings, and Isochrone-A* method still provides the best result, a 3.8% fuel reduction.

4.3. Optimisation comparison with 2DDA

The Dijkstra algorithm is a well-developed algorithm for routing challenges and has been used widely in today's applications of voyage planning. It is renowned for its efficiency in determining the shortest route between two waypoints within its pre-discretized grid, and its application can also be extended to optimising routes for various objectives. In this case study, it is chosen as the fuel efficiency. The 2D Dijkstra algorithm (2DDA) is included in the comparison to further demonstrate the effectiveness of the proposed improved isochrone methods. As a two-dimensional method,

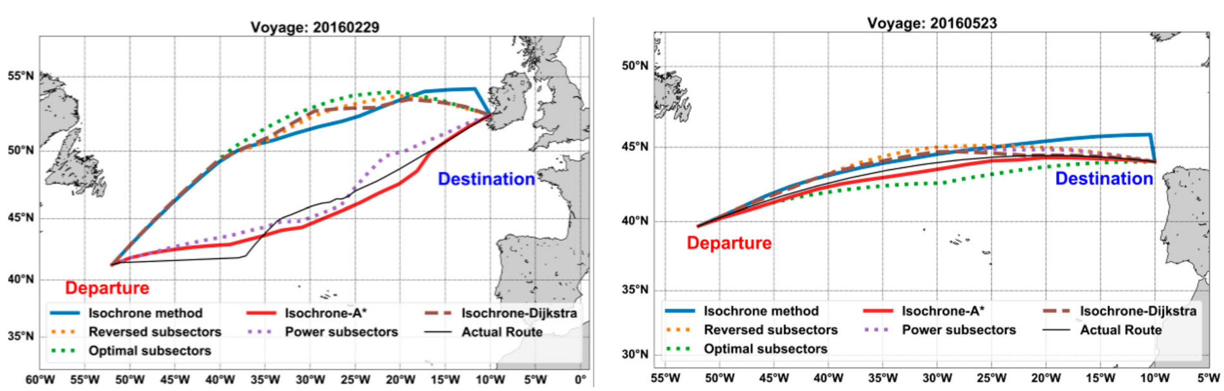

Figure 18. Optimised routes of Voyage20160229 (left) and 20160523 (right) by different methods (This figure is available in colour online.).

Table 5. Results of modified Isochrone algorithms, actual routes and 2DDA, for the two eastbound voyages.

	Voyage20160229				Voyage20160523			
Optimisation methods	ETA [h]	Fuel [ton]	Distance [km]	Average Speed [knot]	ETA [h]	Fuel [ton]	Distance [km]	Average Speed [knot]
Actual Route	159.0	171.5	3624.9	12.3	144.5	156.2	3476.8	13.0
2DDA	161.2	153.2	3589.7	12.0	145.7	150.4	3476.7	12.9
Isochrone method	161.7	164.2	3539.0	11.8	152.3	156.8	3628.8	12.9
Reversed subsectors	159.6	164.6	3450.0	11.7	146.6	151.7	3484.9	12.8
Optimal subsectors	161.4	168.5	3465.6	11.6	146.6	151.1	3491.2	12.9
Power subsectors	159.6	158.9	3549.4	12.0	146.4	151.0	3481.7	12.8
Isochrone-A*	159.9	156.1	3586.5	12.1	145.8	150.2	3479.3	12.9
Isochrone-Dijkstra	159.9	162.6	3453.1	11.7	146.5	151.2	3483.6	12.8

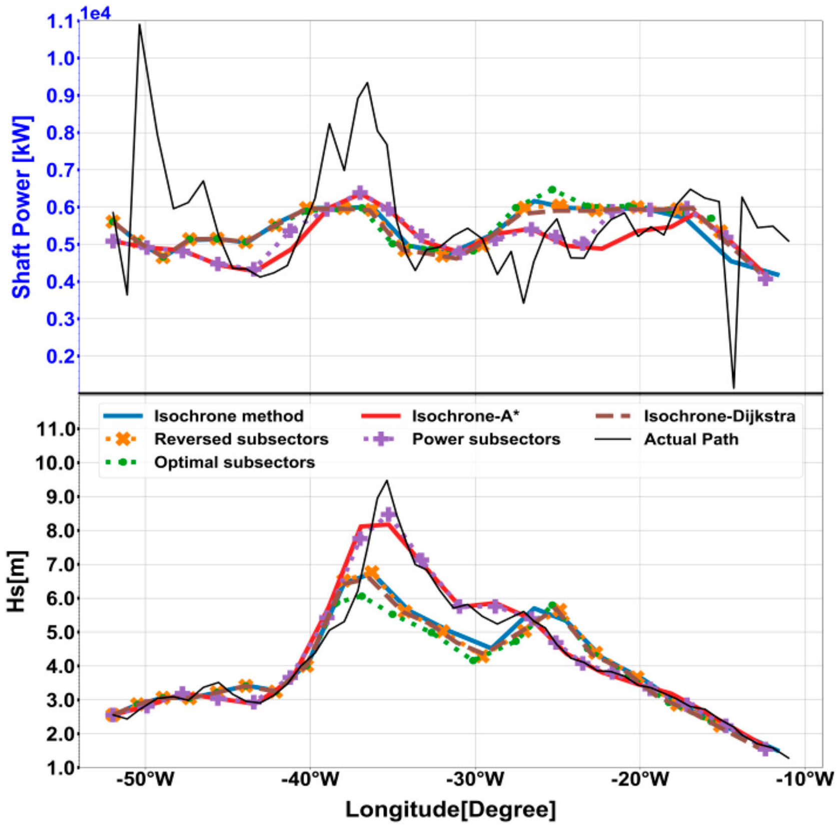

Figure 19. Shaft power and encountered H_s in Voyage 20160229 (This figure is available in colour online.).

2DDA also employs constant sailing speed in sub-routes during the process, unless extreme weather is encountered. Since it requires a pre-defined grid for initialisation, for the four voyages included in this case study, its grid is generated with the same number of time stages as isochrone methods and contains the same number of candidate waypoints at each time stage.

The optimisation results are presented in Tables 4 and 5, respectively. From the aspect of fuel usage, it could be seen that 2DDA provides the lowest results for all four cases. This phenomenon is consistent with its capability in ensuring to enumerate the lowest cost route within the grid. However, when it comes to the ETA, it is hard to provide a route that meets the requirement of accuracy. For two eastbound cases, the ETAs of the optimised routes are near the actual route. For the two westbound cases, i.e. Voyage 20161108 and Voyage 20150721, it shows both apparent delays in arrival, with around 5 and 8 h late respectively. The reasons for the delay could be found by its longer sailing distance, indicating that it is taking a detour to avoid the unfavoured environmental impact, thereby encountering the calm sea states to reduce fuel consumption. On the contrary, the isochrone methods, especially Isochrone-A* method, can present a close result in fuel consumption as 2DDA. The differences in fuel saving between 2DDA and Isochrone-A* method are 0.8%, 3.3%, 1.7%, and −0.1% for four voyages respectively, compared with the actual routes. However, the planned ETA of the Isochrone-A* method is as required in the actual route, with no obvious deviations.

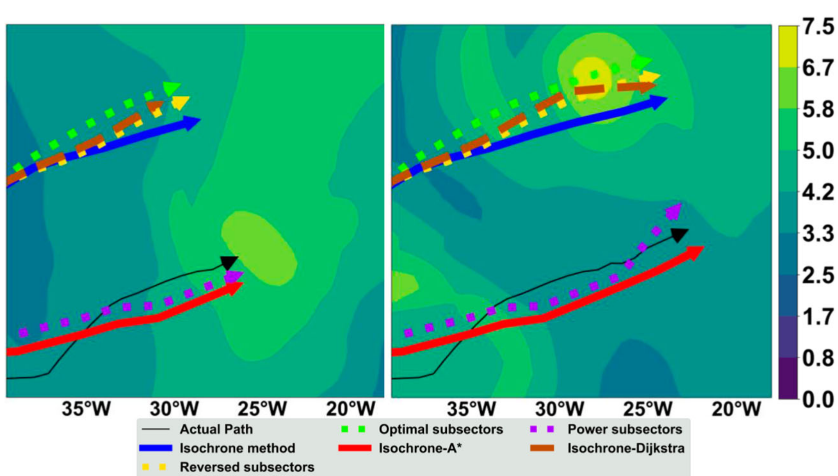

Figure 20. Weather conditions along the evolution for different optimal planning of Voyage 20160229 (This figure is available in colour online.).

Table 6. Comparison and summary for different improvement strategies.

Optimisation methods	Improvements	Limitations
Reversed subsectors	Improved route convergency	Local optimisation as overlapped routes
Optimal subsectors	Improved route convergency, overlapped waypoints removal	Local optimisation as overlapped routes
Isochrone-A*	Improved route convergency, refined cost function	Slightly increased computational load compared with other isochrone methods
Power subsectors	Improved route convergency, separated route-finding for different waypoint to avoid overlapped routes	Limited by greedy search method, cannot achieve a good trade-off between various influencing factors for optimisation objectives
Isochrone-Dijkstra	Improved route convergency, combined static search grid to avoid local optimisation	Unable to ensure ETA, longer overall runtime

4.4. Discussions of the optimisation results

From the optimisation results of the four case study voyages, all five modified Isochrone algorithms deliver improved results compared with the Isochrone method under various weather situations. Moreover, the effectiveness of their improvement strategies can be further investigated by comparing their optimisation performances.

As introduced in Section 3, Optimal subsectors, and Isochrone-A* are improvements based on Reversed subsectors method. However, Optimal subsectors shows nearly identical performance with Reversed subsectors method in all cases. Since it is set to demonstrate the effectiveness of keeping an optimal waypoint set in subsectors, instead of the single optimal waypoint, this result implies that this approach has a negligible effect in improving the optimisation performance. Meanwhile, Isochrone-A* method can deliver significantly improved results under different sailing conditions. It could show at least 2.6% fuel saving for calm sea sailings, and 9.0% for harsh sea sailings. On average, the proposed Isochrone-A* method could help to reduce 5.3% fuel consumption compared with the actual routes. Therefore, it proves that refining the evaluation function with a heuristic term, as in Isochrone-A*, is an effective approach for enhancing the optimisation performance in the isochrone algorithm. Moreover, the result comparison with all five strategies suggests that Isochrone-A* employs the most promising improvement strategy.

Besides, Power subsectors method also provides good results in case studies but could not give the globally optimised result with robustness. It is referred to the greedy search algorithm, which assumes that the optimal solutions of divided sub-tasks can lead to a global optimisation for the complete task. And the result implies that this assumption is invalid for isochrone algorithm in the complicated voyage optimisation problems. Isochrone-Dijkstra method also shows notable improvement in optimisation results. However, for the Dijkstra algorithm, it is hard to get an accurate ETA as naturally as isochrone algorithms when applying a constant speed. Therefore, Isochrone-Dijkstra method could not guarantee to provide the lowest possible cost route as the original Dijkstra algorithm, since it must also meet the requirement of accurate ETA. A similar situation can be observed in the result comparison with 2DDA. To achieve the highest fuel efficiency, it shows notable delays in voyage arrivals. And for the static grid of the Dijkstra method, it is hard to ensure the ETA is as required by employing the constant speed. However, for actual sailing, the accurate ETA is a significant objective to achieve effective planning and delays as notable deviations from the planned schedule can lead to problems in transportation efficiency Table 6.

5. Conclusion

Five different approaches to improve the original Isochrone voyage optimisation method are explored to avoid sharp turning and local grid convergence in a ship's second half of voyage planning. By changing the grid searching process, the so-called 'Reversed subsectors', 'Power subsectors' and 'Isochrone-Dijkstra' methods are proposed. And based on 'Reversed subsectors', 'Optimal subsectors' and 'Isochrone-A*' methods are further proposed to consider the evaluation function during the searching process. A chemical tanker with full-scale measurement is used as the case study ship, and their effectiveness is demonstrated by four actual voyages representing diverse sailing conditions, and the comparison with the 2D Dijkstra algorithm (2DDA). Several key contributions are concluded as:

- Abrupt turns in voyage planning by the original Isochrone method are efficiently resolved by using reserved subsectors in the second half of voyages.
- Moreover, all five improved Isochrone methods possess an improved capability in energy-efficient voyage optimisation with accurate ETA considering dynamic weather, compared with original Isochrone method. By keeping the same ETA as the actual routes, all the proposed methods could suggest voyages converge smoothly towards the destination, with notable improvement in energy efficiency in various sailing states.
- Specifically, the Isochrone-A* method shows the most significant potential and robustness among all strategies, with at least 2.6% fuel saving for calm sea sailings, 9.0% for harsh sea sailings, and on average 5.3% for all cases. In comparison with 2DDA, the results from Isochrone-A* method also show no significant difference in fuel consumption, while ensuring the ETA is as required.
- In addition, the characteristic of computation efficiency from the traditional isochrone method is inherited for the proposed strategies.

Finally, the overall runtime for all improved Isochrone methods is no more than 1 min, remaining at the same level as original Isochrone method, except Isochrone-Dijkstra. It is more time-consuming than other isochrone-type methods since Dijkstra algorithm in the latter half voyage requires more computation time than isochrone methods. Within those methods, there are five parameters needed to construct the search grid, which have a great impact on the optimisation results, and require further study with possible quantified and formulated instructions.

Disclosure statement

No potential conflict of interest was reported by the author(s).

Funding

The authors would like to acknowledge the funding from the project AUTO-Barge, European Union's EU Framework Program for Research and Innovation Horizon 2020 under Grant Agreement No. 955768, the Swedish Vinnova project 2021-02768, the Lighthouse Sustainable Shipping Program project of FP4_2020.

Data Availability Statement

The data used in this study cannot be made publicly available due to the NDA with the ship owner who provided us with the data.

ORCID

Yuhan Chen http://orcid.org/0009-0001-8510-3935
Wuliu Tian http://orcid.org/0000-0003-4168-2549
Wengang Mao http://orcid.org/0000-0002-7126-1254

References

Bahrami N, Siadatmousavi SM. 2023. Ship voyage optimisation considering environmental forces using the iterative Dijkstra's algorithm. Ships Offsh Struct. 1–8. doi:10.1080/17445302.2023.2231200.

Bellman R. 1952. On the theory of dynamic programming. Proc Natl Acad Sci USA. 38:716–719. doi:10.1073/pnas.38.8.716.

Calvert S, Deakins E, Motte R. 1991. A dynamic system for fuel optimization trans-ocean. The Journal of Navigation. 44:233–265. doi:10.1017/S0373463300009978.

Chen H. 1978. A dynamic program for minimum cost ship routing under uncertainty Massachusetts Institute of Technology.

De Wit C. 1990. Proposal for low cost ocean weather routeing. The Journal of Navigation. 43:428–439. doi:10.1017/S0373463300014053.

Dijkstra EW. 1959. A note on two problems in connexion with graphs. Numer Math. 1:269–271. doi:10.1007/BF01386390.

Grifoll M, Borén C, Castells-Sanabra M. 2022. A comprehensive ship weather routing system using CMEMS products and A* algorithm. Ocean Eng. 255:111427. doi:10.1016/j.oceaneng.2022.111427.

Hagiwara H. 1989. Weather routing of (sail-assisted) motor vessels. Ph D thesis. Delft Univ Tech.

Hart PE, Nilsson NJ, Raphael B. 1968. A formal basis for the heuristic determination of minimum cost paths. IEEE Trans Syst Sci Cybern. 4:100–107. doi:10.1109/TSSC.1968.300136.

James RW. 1957. Application of wave forecast to marine navigation. Washington: New York University.

Klompstra MB, Olsder G, Van Brunschot P. 1992. The isopone method in optimal control. Dyn Control. 2:281–301. doi:10.1007/BF02169518.

Lang X, Mao W. 2020. A semi-empirical model for ship speed loss prediction at head sea and its validation by full-scale measurements. Ocean Eng. 209:107494. doi:10.1016/j.oceaneng.2020.107494.

Lang X, Mao W. 2021. A practical speed loss prediction model at arbitrary wave heading for ship voyage optimization. J Mar Sci Appl. 20:410–425. doi:10.1007/s11804-021-00224-z.

Larsson E, Simonsen MH, Mao W. 2015. Direct optimization algorithm in weather routing of ships. Proceedings of the ISOPE International Ocean and Polar Engineering Conference; ISOPE.

Lin Y-H. 2018. The simulation of east-bound transoceanic voyages according to ocean-current sailing based on particle swarm optimization in the weather routing system. Mar Struct. 59:219–236. doi:10.1016/j.marstruc.2018.02.001.

Lin Y-H, Fang M-C, Yeung RW. 2013. The optimization of ship weather-routing algorithm based on the composite influence of multi-dynamic elements. Appl Ocean Res. 43:184–194. doi:10.1016/j.apor.2013.07.010.

Ma D, Ma W, Jin S, Ma X. 2020. Method for simultaneously optimizing ship route and speed with emission control areas. Ocean Eng. 202:107170. doi:10.1016/j.oceaneng.2020.107170.

Pennino S, Gaglione S, Innac A, Piscopo V, Scamardella A. 2020. Development of a new ship adaptive weather routing model based on seakeeping analysis and optimization. J Mar Sci Eng. 8:270. doi:10.3390/jmse8040270.

Poulsen RT, Viktorelius M, Varvne H, Rasmussen HB, von Knorring H. 2022. Energy efficiency in ship operations-exploring voyage decisions and decision-makers. Trans Res Part D: Trans Environ. 102:103120. doi:10.1016/j.trd.2021.103120.

Roh M-I. 2013. Determination of an economical shipping route considering the effects of sea state for lower fuel consumption. Int J Naval Arch Ocean Eng. 5:246–262. doi:10.2478/IJNAOE-2013-0130.

Sasa K, Chen C, Fujimatsu T, Shoji R, Maki A. 2021. Speed loss analysis and rough wave avoidance algorithms for optimal ship routing simulation of 28,000-DWT bulk carrier. Ocean Eng. 228:108800. doi:10.1016/j.oceaneng.2021.108800.

Shin YW, Abebe M, Noh Y, Lee S, Lee I, Kim D, Bae J, Kim KC. 2020. Near-optimal weather routing by using improved A* algorithm. Applied Sciences. 10:6010. doi:10.3390/app10176010.

Silveira P, Teixeira A, Guedes-Soares C. 2019. AIS based shipping routes using the Dijkstra algorithm. TransNav: Int J Mar Navig Saf Sea Transp. 13:565–571. doi:10.12716/1001.13.03.11.

Simonsen MH, Larsson E, Mao W, Ringsberg JW. 2015. State-of-the-art within ship weather routing. In: Proceedings of the ASME 2015 34th International Conference on Ocean, Offshore and Arctic Engineering. Volume 3: Structures, Safety and Reliability. May 31–June 5, 2015; St. John's, Newfoundland, Canada. ASME. V003T02A053. doi:10.1115/OMAE2015-41939.

Topaj A, Tarovik O, Bakharev A, Kondratenko A. 2019. Optimal ice routing of a ship with icebreaker assistance. Appl Ocean Res. 86:177–187. doi:10.1016/j.apor.2019.02.021.

Turna İ. 2023. A Fuzzy Bayesian approach for 'Appraisal' of ship voyage plans. Ships Offsh Struct. 18:859–866. doi:10.1080/17445302.2022.2077279. Review of Maritime Transport 2021 (UNCTAD/RMT/2021) https://unctad.org/system/files/official-document/rmt2021_en_0.pdf: United Nations Publication.

Wang H, Lang X, Mao W. 2021. Voyage optimization combining genetic algorithm and dynamic programming for fuel/emissions reduction. Transp Res Part D: Transp Environ. 90:102670. doi:10.1016/j.trd.2020.102670.

Wang H, Mao W, Eriksson L. 2019. A three-dimensional Dijkstra's algorithm for multi-objective ship voyage optimization. Ocean Eng. 186:106131. doi:10.1016/j.oceaneng.2019.106131.

Wang S, Meng Q. 2012. Sailing speed optimization for container ships in a liner shipping network. Transp Res Part E: Log Transp Rev. 48:701–714. doi:10.1016/j.tre.2011.12.003.

Wisniewski B. 1991. Methods of route selection for a sea going vessel. Gdansk: Wydawnictwo Morskie.

Xue H. 2022. A quasi-reflection based SC-PSO for ship path planning with grounding avoidance. Ocean Eng. 247:110772. doi:10.1016/j.oceaneng.2022.110772.

Xue H, Qian K. 2023. Improved BSO for ship path planning with grounding alert using interpolated Delaunay triangulation. Ships Offsh Struct. 18:1275–1285. doi:10.1080/17445302.2022.2111924.

Zaccone R, Figari M. 2017. Energy efficient ship voyage planning by 3D dynamic programming. J Ocean Techn. 12:49–71.

Zaccone R, Ottaviani E, Figari M, Altosole M. 2018. Ship voyage optimization for safe and energy-efficient navigation: A dynamic programming approach. Ocean Eng. 153:215–224. doi:10.1016/j.oceaneng.2018.01.100.

Zhao L, Bai Y. 2023. Data harvesting in uncharted waters: interactive learning empowered path planning for USV-assisted maritime data collection under fully unknown environments. Ocean Eng. 287:115781. doi:10.1016/j.oceaneng.2023.115781.

Zhao L, Wang F, Bai Y. 2023. Route planning for autonomous vessels based on improved artificial fish swarm algorithm. Ships Offsh Struct. 18:897–906. doi:10.1080/17445302.2022.2081423.