Vegard Nitter Vestad

Automatic and practical route planning for ships

Master's thesis in Cybernetics and Robotics
Supervisor: Morten Breivik
June 2019

Master's thesis

NTNU
Norwegian University of Science and Technology
Faculty of Information Technology and Electrical Engineering
Department of Engineering Cybernetics

NTNU
Norwegian University of
Science and Technology

Vegard Nitter Vestad

Automatic and practical route planning for ships

Master's thesis in Cybernetics and Robotics
Supervisor: Morten Breivik
June 2019

Norwegian University of Science and Technology
Faculty of Information Technology and Electrical Engineering
Department of Engineering Cybernetics

NTNU
Norwegian University of
Science and Technology

Preface

After studying optimal control and path planning in Japan for two semesters from 2017 to 2018, I had naturally grown a strong interest in applying theoretical knowledge of path planning and optimal control to a practical problem. It simply became too tempting to take advantage of the opportunity to write my Master Thesis on the subject of route planning for ships in cooporation with Kongsberg Maritime, an industry leader in marine robotics.

This thesis is written as part of an M.Sc. degree in Cybernetics and Robotics at the Department of Engineering Cybernetics, Norwegian Unitversity of Science and Technology (NTNU) in cooperation with Kongsberg Maritime. I would like to thank my supervisor Morten Breivik and all my co-supervisors Glenn Ivan Bitar from NTNU, and Jon Bernhard Høstmark and Even Ødegaard from Kongsberg Maritime, for their valuable guidance and feedback throughout the project. Further thanks go to Thor Olsen from Kongsberg Maritime at Grilstad who has been extraordinary helpful at teaching me navigation and how to use K-Sim. I would also like to thank the rest of the staff at Grilstad for their hospitality, even after spending many days there conducting experiments.

The goal of the project was to solve a practical route planning problem. During the semester the supervisors have contributed with guidance through an hourly biweekly follow-up meeting where the progress of the thesis and other related topics were discussed. My supervisors from NTNU have helped me with exploration of relevant theory, while my supervisors from Kongsberg have helped me keep focus on the practical problem and helped define the problem itself. In addition to this, Thor Olsen has helped with the usage of the K-Sim and taught me the fundamental navigation principles.

The work is based upon previous work done by Glenn Ivan Bitar in path planning theory. The analysis was further extended and adapted by me during the the pre-project work during the autumn 2018. It was suggested by Jon Bernhard Høstmark from the beginning to use compliance to electronic chart displays as a requirement and K-Sim for benchmarking. All other requirements were loosely defined, and were continuously defined together in the start phase of the project. In order to solve the problem, effort was put into getting access to approved nautical charts from Kartverket.

My focus on route planning has taken many directions since the start of the work. The work on route planning started out as a project meant as a pre-study for the Master thesis. It started as a theoretical exploration of route planning methods for ships, but has developed to become much more practical since then. The project also started out with a focus on autonomous ships, but ended up focusing primarily on manned ships. Also, the focus changed from finding optimal routes to prioritize finding routes satisfying a set of requirements. The combination of the theoretical and practical aspects has significantly improved the quality of this thesis. The work done in this thesis and the pre-project also resulted in a publication written in cooperation with my co-supervisor Glenn Ivan Bitar for the IFAC CAMS 2019 conference. Contributing to a publication during the semester has given an additional workload, but the resulting experience and knowledge with academic work has been a rewarding involvement.

Vegard Nitter Vestad
Trondheim, June 10, 2019

Abstract

If a ship is to transit from one location to another, a route is usually created manually by an experienced navigator, even though automatic planners exist. The reason why many automatic planners are outperformed in practice by manual planning, is that there are so many industry requirements to a route that needs to be fulfilled, a variety of inputs such as weather to consider and a good idea of what a good route is.

This thesis explores automatic route planning methods and proposes a method which originates in theory and is usable in practice. The routes are generated based on a mission, electronic nautical charts, sea/weather forecasts and knowledge of the ship. The generated routes are at least sailable, meet a set deadline, is compliant with basic electronic chart requirements, is possible to navigate through optical means, and adheres to traffic separation schemes. The planner is then designed to find routes of short travel distance, short travel time, low energy consumption, low rudder wear and tear and little turning.

The method is implemented and applied to a scenario from Stavanger to Håkonsvern by cargo ship during night and normal weather and sea conditions. The routes generated from the implemented method is validated according to the requirments and then benchmarked using the Kongsberg Maritime K-Sim simulator environment. The performance is compared to four existing, manual-based solutions. Three solutions are different manual routes and one is a manually modified route from an existing automatic route planner implemented in the Kongsberg chart interface. The implemented method of this thesis is shown to outperform the existing solutions.

From the analysis in the specific scenario, it was also concluded that weather and sea has a great impact on the performance of the routes. Also, it was concluded that fast and energy-efficent routes are similar, routes of low rudder wear and tear and little turning are similar, and that fast and energy-efficent routes are not similar to routes of low rudder wear and tear and little turning. Without weather, the shortest route is probably the fastest route and also the most energy efficient route. With weather, sea currents greatly impact the travel time and also energy consumption. Thus, the shortest route is probably not the fastest nor most energy-

efficient route. It was also identified a performance metric based on travel time for finding a fast and energy-efficent route, and a performance metric based on number of turns for finding routes of low rudder wear and tear and turning.

Sammendrag

Hvis et skip skal seile fra en lokasjon til en annen, er en rute vanligvis laget av en erfaren navigatør, selv om automatiske ruteplanleggere allerede eksisterer. Årsaken til at mange automatiske ruteplanleggere er utkonkurrert i praksis, er at der er så mange industrielle krav til en rute som må bli tilfredsstilt, en variasjon av input slik som vær som må tilfredsstilles, og en god idé om hva en god rute er.

Denne avhandlingen utforsker automatisk ruteplanleggingsmetoder og foreslår en metode som har utspring i teori og er brukbar i praksis. Rutene er generert basert på et oppdrag, elektronisk nautiske kart, sjø- og værmeldinger og kunnskap om skipet. De genererte rutene er ihvertfall seilbare, tilfredsstiller en satt tidsfrist, er kompatibel med grunnleggende krav i elektroniske kart, er mulig å navigere gjennom optiske midler, og følger trafikkseperasjonssystemer. Planleggeren er dermed designet for å finne ruter av lav reisedistanse, lav reisetid, lav energiforbruk, lav rorslitasje og lite svinging.

Metoden er implementert og anvendt på et scenario fra Stavanger til Håkonsvern med lasteskip om natten og under normale vær- og sjø forhold. Rutene som er generert fra den implementert metoden er validert i henhold til kravene og dermed benchmarket ved å bruke Kongsberg Maritime K-Sim simulator miljøet. Ytelsen er sammenligned med fire eksisterende, manuell-baserte løsninger. Tre løsninger er forskjellige manuelle ruter og én er en manuelt modifisert rute fra en eksisterende automatisk ruteplanelgger som er implementert i Kongsberg sitt kart interface. Den implementerte metoden i denne avhandlingen er vist å utkonkurrere de eksisterende løsningene.

Fra analysen i det spesifikke scenarioet, ble det konkludert at vær og sjø har stor påvirkning på ytelsen til rutene. Det ble også konkludert at raske og energieffektive ruter er lignende, ruter av lav rorslitasje og lite svinging er ligenende, og raske og energi-effektive ruter er ikke lignende på ruter av lav rorslitasje og lite svinging. Uten vær, er den korteste ruten sannsynligvis den raskest og også den mest energi-effektive ruten. Med vær, påvirker strømninger i havet seiletiden og også energi-forbruket. Derfor er den korteste ruten ikke nødvendigvis den raskeste eller den mest energi-effektive. Det ble også identifisert en ytelsesmetrikk basert på seiletid for å finne den raskeste og mest energi-effektive ruten, og en ytelsesmetrikk

basert på antall svingninger for å finne ruter av lav rorslitasje og lite svinging.

Contents

List of Figures

List of Tables

Abbreviations

DOF Degree of freedom.

ECDIS Electronic Chart Display and Information System.

ENC Electronic nautical charts.

TSS Traffic separation scheme.

Nomenclature

N Yaw moment.

X Surge force.

Y Sway force.

ψ Yaw angle.

r Yaw angular velocity.

u Surge linear velocity.

v Sway linear velocity.

x North position in NED frame.

y East position in NED frame.

Chapter 1

Introduction

1.1 Motivation

If a ship is to transit from location A to B safely and efficient, it is comon practice, and usually a necessity for larger ships, to plan a route before transit. This task is usually done manually by an experienced navigator and may take some hours effort. Tools that could aid this process such as automatic route planners are rarely used, even though new and better methods are proposed by the academic community every year. The first reason for this is that there are so many requirements the navigator has to consider, such as traffic rules and lighthouses, that most route planners are just not good enough. If the automatic route planner does not consider all requirements, the navigator may just as well do it himself. The second reason is that many automatic route planners are not adaptive enough. An experienced navigator knows that the best route is dependent on local weather such as wind, waves and sea currents. A third reason is that many automatic route planners do not know what a good route is. A good route is not necessarily the shortest travel distance or even the shortest travel time. An experienced navigator knows that a good route also saves energy, reduces wear and tear and uses as few turns as possible. Because of these constraints, it has been hard to automate the route planning process for ships. However, if an automatic route planner can be made to satisfy the navigators requirements such that it is accepted by the industry, many possibilities open up to outperform the human in terms of performance.

Three important motivating factors for generating better routes, especially in the ferry and shipping industry, are safety, environment and economics. It is believed that an automatic route planner may contribute to safer, more environmental and more economical journeys than a human could do alone. An advantage of automated route planning is to save time. The planner may completely take over the task of planning, or a human may only need to spend some time going over to validate the route. Another advantage is to generate more efficient routes. As before established, the shortest or fastest route at sea is not necessarily the best. An automatic route planner may solve complex optimalization problems in order to generate routes based on optimality of any performance, such as energy consumption. An automatic planner may also remove the guesswork of the planning and may thus contribute to safer routes. The planner may for example know about the wave height in an area that the navigator did not know about or forgot to consider.

A motivation for validating routes using the DNV approved ECDIS from Kongsberg Maritime is to gain practical use of the routes in the industry. Every year there are many optimal route planning algorithms proposed by the academic community, but if they can not be validated according to industrial standards, they are difficult to trust in practice. If the routes can be validated, planning becomes easier for the navigator and new solution are more acceptable.

A motivation for benchmarking by using a high-fidelity industrial simulation environment such as K-Sim, is to increase the credibility of optimality. Getting acceptance for optimality with the use of empiric data or simple ship models has proven hard. The complexity of empiric data and lack of knowing all disturbances gives questionable results. Also, simple ship models may lack important dynamics which gives questionable results. Thus, it is desirable to test performance with a deterministic simulation environment that replicates the real dynamics of the system, such that optimal routes can be benchmarked to manual routes.

1.2 Previous work

A lot of work and existing services exist that lay the fundation of automatic, practical route planning for ships. Previous work on route planning can be categorized as shown in Figure 1.1 into practical, automatic and ship knowledge. Automatic, practical route planning for ships is based on all these aspects, so they should all be considered.

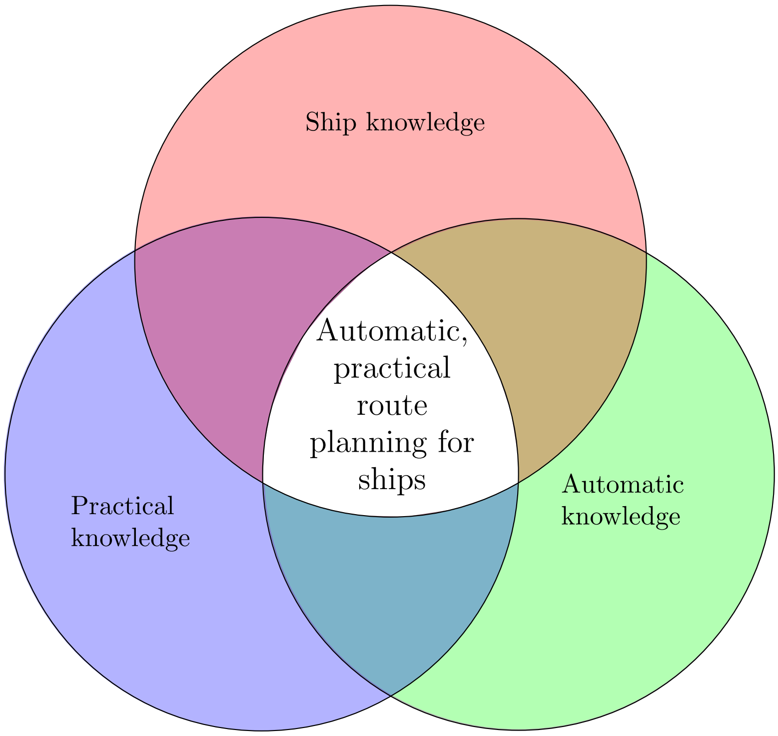


Figure 1.1: Previous work categorized

The practical aspect of route planning contains all practical knowledge related to planning manually. This aspect generally includes all industry requirements and best-practices that are established from experience. An experienced navigator may have more knowledge than any academic person when it comes to route planning, and is argubly more versatile and adaptable to unknown situations. Thus, it is a great advantage to consider the experience accumulated by the practical and manual use of route planning.

The automatic aspect of route planning contains all theoretical knowledge related to planning automatically. Many generic algorithms exist for finding paths, such as searches based on roadmaps, RRT* and other optimality based methods to name a few. A lot of work has been done on the topic of theoretical path planning for robotics in general. Every year new theoretical methods are published in the university environment. In order to create order, extensive work has been carried out by people such as LaValle in the field of categorizing and describing these planning algorithms.

The ship aspect of route planning contains all knowledge related to the ship. Many models for ships exists, from simpler based on turn radius to more complicated based on mass-spring-damper models or higher fidelity models. Knowing the ship that the routes are plans for, is usually a precondition for better results.

Considering automatic route planning methods with a ship model has been seen many times in most published route planners for ships, i.e. combining automatic and ship knowledge. The focus is generally to generate better routes, but the methods may not be valid in practice. Also making simulators from ship knowledge to improve manual route planning has been done before, i.e. combining ship and practical knowledge. However, combining all ship, practical and automatic knowledge has been done much less before. This will be the focus of this thesis.

1.3 Problem description

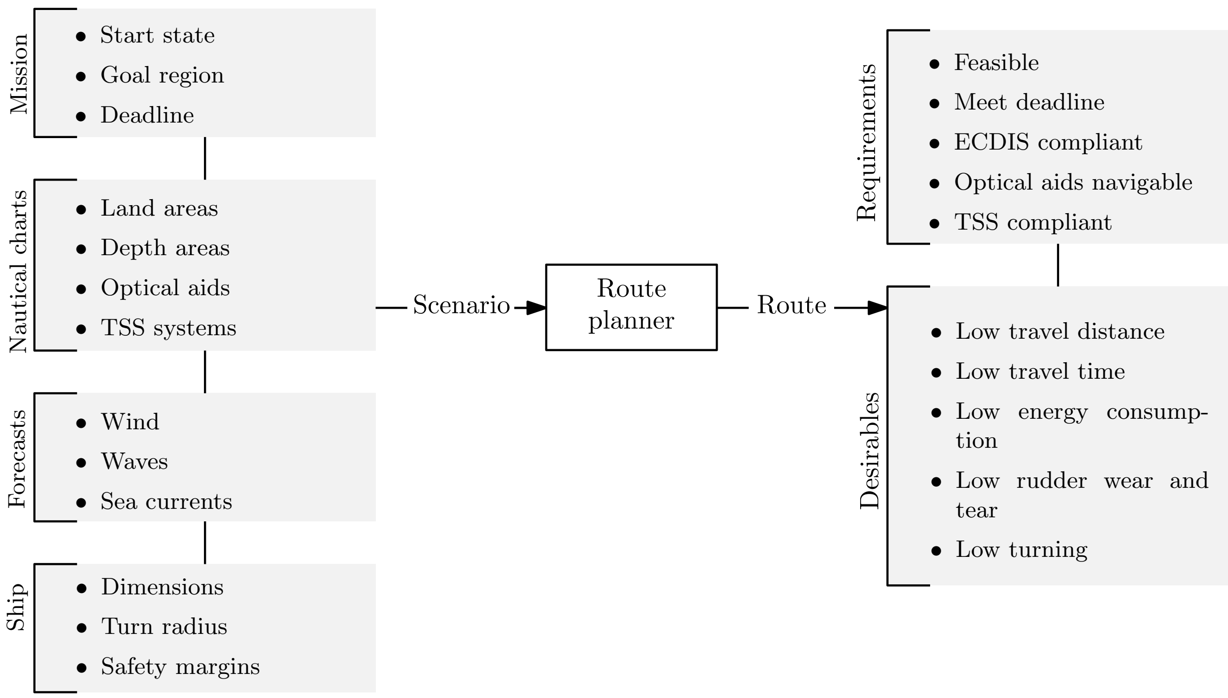


Figure 1.2: Overview of the problem description.

The main goal of this research project is to explore, understand and implement a closed loop optimal route planning solution for the problem described in Figure 1.2 for a a distance from Stavanger to Håkonsvern. That is, it shall be developed an automatic method for generating routes based on a predefined set of information, where the routes satisfy a set of requirements and are optimized on a set of desired behaviours. The information available for the planner is

- A mission describing where the ship is initially at and to where it shall sail within a given time frame defined by a deadline.

- Electronic nautical charts (ENC) describing land and depth areas, optical aids such as lighthouses, and traffic separation systems (TSS).
- Weather forecasts for wind, waves and sea currents.
- Knowledge of the ship such as its dimensions, turn radius and safety margins.

The generated routes are required to at least be

- Feasible, i.e. possible for the ship to follow.
- Meet the set deadline.
- Compliant with existing electronic chart systems (ECDIS) that are installed on ships.
- Safe to navigate by only using optical aids such as lights if e.g. all other navigational aids should fail.
- Compliant with traffic separation schemes (TSS), i.e. follow the regulated traffic flow where implemented.

The generated routes should then to the best of its ability yield

- Low travel distance.
- Low travel time.
- Low energy consumption.
- Low rudder wear and tear.
- Low turning.

The theory and implementation is applied to the following scenario: A manned, underactuated ship that is approximately 70 meters long of Kongsberg Njord shall transport some cargo from Stavanger to Håkonsvern within a given deadline during night and during normal weather and sea conditions. The distance is shown in Figure 1.3. Heading and speed is controlled by human or by autopilot/speed to follow the route.

Figure 1.3: Scenario from Stavanger to Håkonsvern

The implemented method is then benchmarked to existing solutions using the maritime industrial simulator K-Sim.

1.4 Contributions

The main contribution of this thesis is a practical and optimal route planning method for solving the defined problem. Some highlights are

- A polygon reduction method specialized to reducing electronic nautical charts (ENCs).
- A roadmap generation method for traffic separation systems (TSS).
- A safe configuration space used for safe optical navigating.
- Implementation and performance analysis using the K-Sim environment.

1.5 Outline

Chapter 2 presents theoretical background for fully understanding the problem description and presents a literature study on optimal route planners which is used as a basis for developing the solution. Chapter 3 presents the implemented route planning method and Chapter 4 presents and discusses the simulation results. Finally, Chapter 5 concludes the research with suggestions for further work.

Chapter 2

Preliminaries

2.1 Ship notations

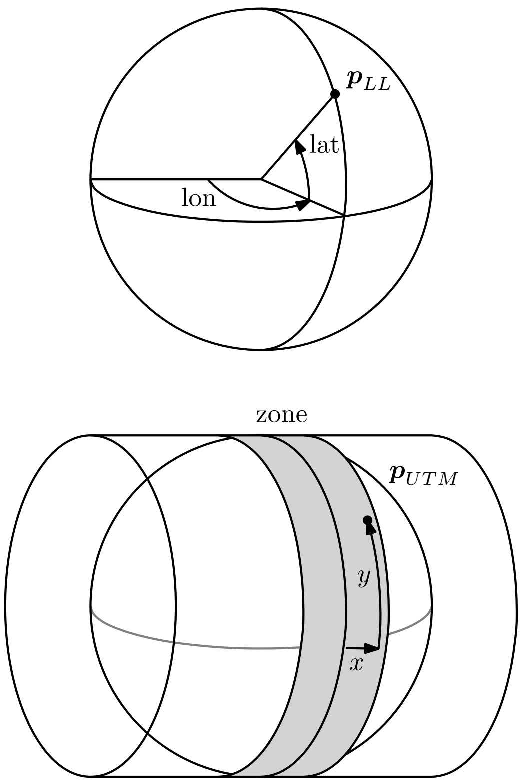
Common notations for the ship's pose, velocity and control input based on SNAME (Fossen 2011) are given in Table 2.1. The pose, velocity and control input of the ship are advantageously collected in $\boldsymbol{\eta} = [x, y, \psi]^T$, $\boldsymbol{\nu} = [u, v, r]^T$ and $\boldsymbol{\tau} = [X, Y, N]^T$, respectively. It should be emphasized that pose is given relative to a North-East-Down (NED) reference frame, and velocity and control input are given in the body, i.e. ship's, frame (Fossen 2011). That means that pose is the absolute position and orientation in a two dimensional tangent on the Earth surface oriented towards north, while velocity and control input is relative to the orientation of the ship.

	Forces and moments	**Linear and angular velocities**	**Positions and Euler angle**
Motion in x-direction (surge)	X	u	x
Motion in y-direction (sway)	Y	v	y
Rotation about z-axis (yaw)	N	r	ψ

Table 2.1: Notations for vessel. Based on SNAME (1950) (Fossen 2011).

2.2 Coordinate systems

Two coordinate systems will be used for planning, namely the latitude-longitude coordinate system and the universal traverse mercator coordinate system as shown in Figure 2.1.



a) Latitude-Longitude (LL)

b) Universal Traverse Mercator (UTM)

Figure 2.1: Coordinate systems

2.2.1 Latitude-Longitude (LL)

Latitude-Longitude is based on two coordinates, latitude and longitude, describing the position on the Earth surface in degrees. This coordiante system is usually used for for describing global positions, but is not convenient for describing local positions because they are represented in degrees.

2.2.2 Universal Traverse Mercator (UTM)

The Universal Transverse Mercator (UTM) conformal projection uses a 2-dimensional Cartesian coordinate system to give locations on the surface of the Earth. Like the traditional method of latitude and longitude, it is a horizontal position representation, i.e. it is used to identify locations on the Earth independently of altitude.

The UTM system is not a single map projection. The system instead divides the Earth into sixty zones, each being a six-degree band of longitude, and uses a secant transverse Mercator projection in each zone. For this thesis this separation will not be a problem because the whole scenario falls into zone 32.

2.3 Kongsberg Maritime ECDIS

A general Electronic Chart Display and Information System (ECDIS) is a geographic information system used for nautical navigation that complies with International Maritime Organization (IMO) regulations as an alternative to paper nautical charts (Bhattacharjee 2019). Even though ECDIS is primarly designed to replace paper nautical charts by displaying digital nautical charts, its functionality is increasing. Examples of what ECDIS can do other than displaying charts, is displaying the position and heading of the ship, plotting routes, give warnings and alerts depending on how well the ship performs. E.g. if the ship is on collision course, an alert may notify the navigator to change course.

It is not uncommon that different distributors of ECDIS, such as Kongsberg, extend the functionality even further to become more competitive in the market. Examples of extensions of Kongsbergs ECDIS, is autopilot, route validation and automatic route planning.

2.3.1 Route

In Kongsberg ECDIS, a route is a sequence of waypoints k defining a geographical passage as illustrated in Figure 2.2. Each waypoint is based on a position $\mathbf{p}_k$ where the passage is split into a straight part called the *leg* and a turn part based on a turn radius $R_{turn,k}$. The same definition will be used for the rest of this thesis because the route shall be ECDIS compliant. In the original ECDIS, a cross-track-margin defines the width of the passage. However, because parts of the ship may be outside the cross-track-margin even when the origin of the ship is inside, a safety margin m_c slightly larger than the cross-track-margin is used instead in the Kongsberg ECDIS.

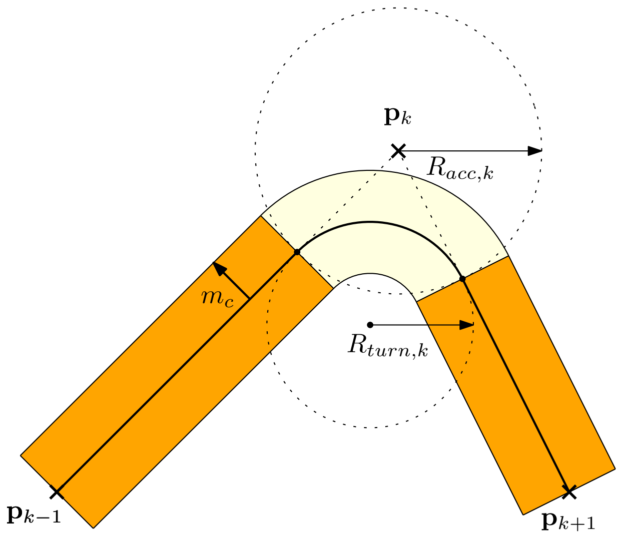


Figure 2.2: Route with waypoints, turn radius and legs as defined in Kongsberg ECDIS.

2.3.2 Route validation

Kongsberg ECDIS is an industrial validation system. It validates routes by presenting errors and warnings for each waypoint and its corresponding leg to the navigator. Examples of warnings are that the route overlaps with an area marked with high voltage cables or that the route overlaps with an area marked for airplane sea landing. It is practically impossible to generate a route which is warning free, and normally these are ignored because they pose no real danger. Error are given if there is a chance that the route may lead to collision with the coast, hitting the sea bed or any other real dangers. Thus, a valid route is defined as a route which ECDIS does not generate errors on.

In order for a route to be valid according to Kongsberg ECDIS, the route must be feasible and not hit the coast or sea bed, based on the capacity of the ship (draft, turn radius) and chart data (depth contour lines, coast lines). The route is verified according to the following rules.

1. The turn radius $R_{turn,k}$ for each waypoint k should should be larger than the turn radius of the ship $R_{turn,min}$, i.e. $R_{turn,k} \geq R_{turn,min} \ \forall \ k$

2. The route should not cross contour areas where the depth may be shallower than the safety depth m_d illustrated in Figure 2.3.

3. The route should not cross points where the depth may be shallower than the safety depth m_d.

The route validation is illustrated by example in Figure 2.3. The given route for a ship with safety depth $m_d = 8\ m$ is not valid because the route crosses contour lines where the depth may be less than 8 m, and also crosses a point where the depth is less than 8 m.

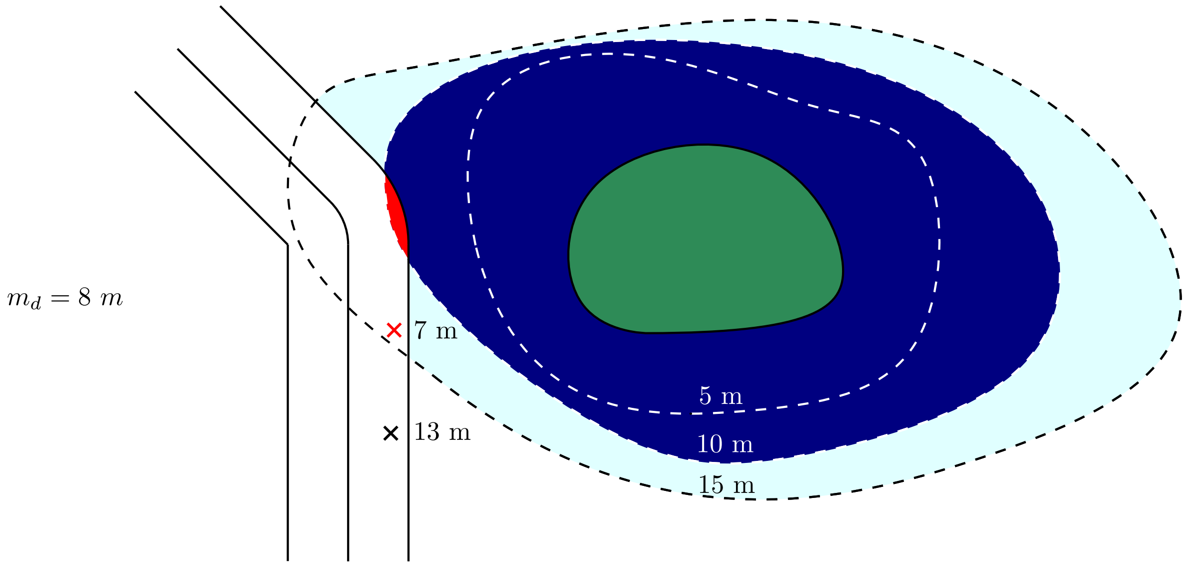


Figure 2.3: Example of Kongsberg ECDIS route validation of route

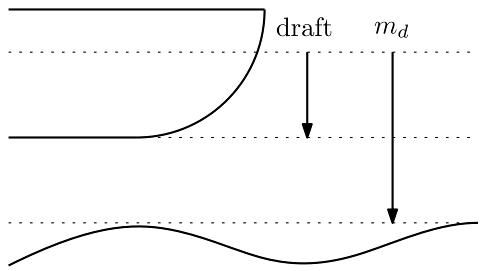


Figure 2.4: Definition of safety depth which is normally slightly larger than the draft of the ship.

2.4 Traffic separation scheme (TSS)

A Traffic Separation Scheme (TSS) is a traffic-management route-system ruled by the International Maritime Organization or IMO. The aim of TSSs is to regulate the traffic at busy, confined waterways or around capes. This is done by separation of opposing streams of traffic by appropriate means and by the establishment of traffic lanes. The most relevant TSS for the given mission, from Stavanger and Bergen, is the one north of Randaberg shown in Figure 2.5.

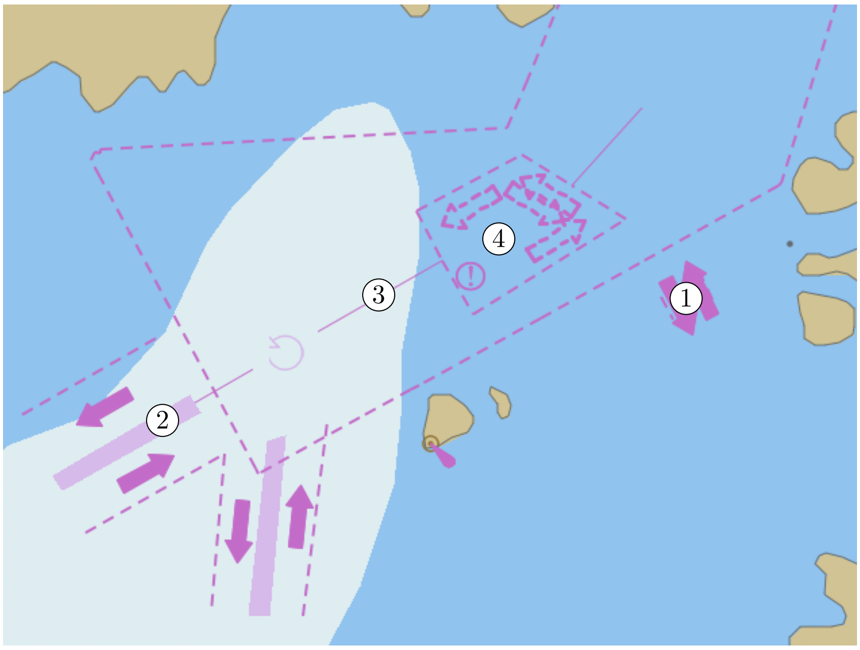


Figure 2.5: The traffic seperation scheme outside of Stavanger consisting of the 1: Traffic lane. 2: Separation zone. 3: Separation line. 4: Roundabout

A general TSS may include the following elements. Note that only the first bolded elements are present and thus relevant in the given scenario.

1. **Traffic lane**: An area (or clearway) within defined limits in which one-way traffic is established. Natural obstacles, including those forming separation zones, may constitute a boundary.

2. **Separation zone**: A zone separating traffic lanes in which ships are proceeding in opposite or nearly opposite directions; or separating a traffic lane from the adjacent sea area; or separating traffic lanes designated for particular classes of ship proceeding in the same direction

3. **Separation line**: A line separating traffic lanes in the same manner as the separation zone.

4. **Roundabout**: A separation point or circular separation zone and a circular traffic lane within defined limits.

5. Inshore traffic zone: A designated area between the landward boundary of a traffic separation scheme and the adjacent coast.

6. Recommended route: A route of undefined width, for the convenience of ships in transit, which is often marked by centreline buoys.
7. Deep-water route: A route within defined limits which has been accurately surveyed for clearance of sea bottom and submerged articles.
8. Precautionary area: An area within defined limits where ships must navigate with particular caution and within which the direction of flow of traffic may be recommended.
9. Area to be avoided: An area within defined limits in which either navigation is particularly hazardous or it is exceptionally important to avoid casualties and which should be avoided by all ships, or by certain classes of ships.

2.5 S-57 ENC format

The S-57 standard ENC format contains the following information:

- Land areas which are represented as polygons and points.
- Depth countour lines which are represented as polygons.
- Lights where position and sectors are described.
- TSS.

Figure 2.6 shows an example of a light in the ENC format and the surrounding land areas and depth contour lines.

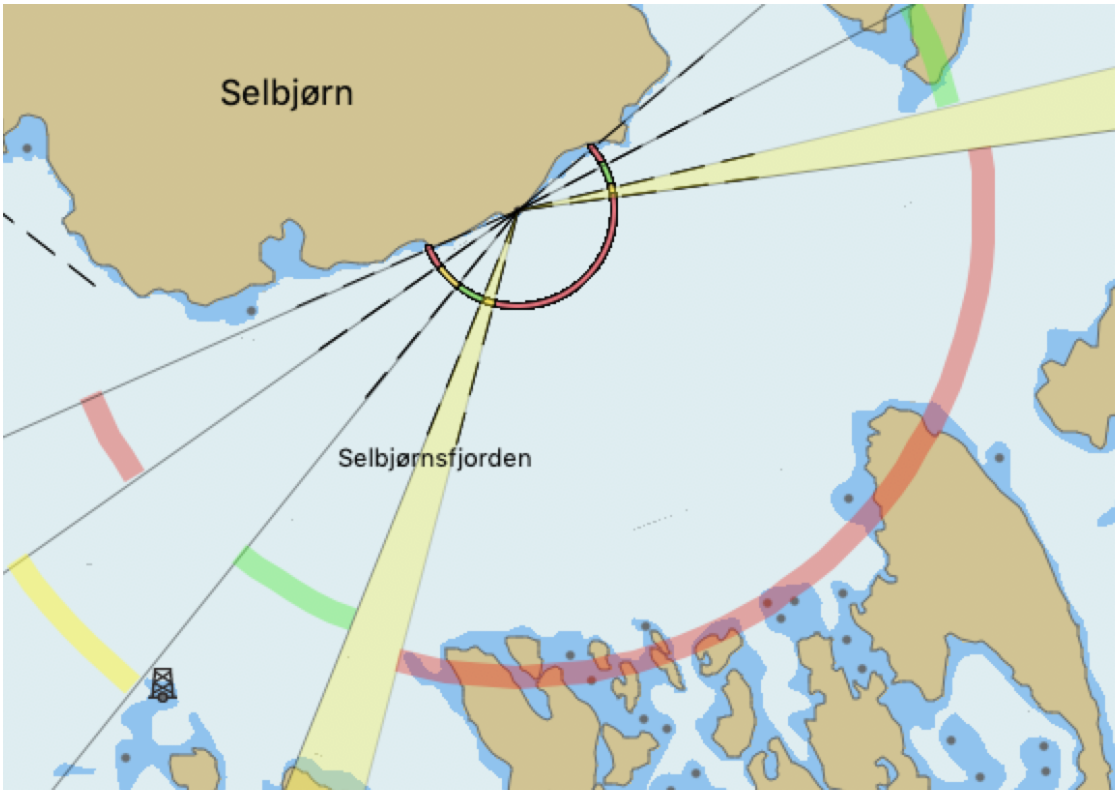


Figure 2.6: Example of light in scenario visualized.

Electronic nautical charts (ENC) that are approved in regards to S-57 are stored across multiple files with naming convention

$$CCPRRRRR.EEE$$

where CC is the producer code, P is the navigational purpose identifier, $RRRRR$ is the waterway code and waterway distance (kilometre) or identification of the equivalent paper chart number and EEE is the update number. ENC data is compiled for a variety of navigational purposes P as shown in Table 2.2.

P	Purpose	Description
1	Overview	For route planning and oceanic crossing.
2	General	For navigating oceans, approaching coasts and route planning.
3	Coastal	For navigating along the coastline, either inshore or offshore.
4	Approach	Navigating the approaches to ports or mayor channels or through intricate or congested waters.
5	Harbour	Navigating within ports, harbours, bays, rivers and canals, for anchorages.
6	Berthing	Detailed data to aid berthing.

Table 2.2: Navigation purposes

An example is $NO4Z1109.000$ which is the non-updated ENC produced by Norway for approaching ports etc. in area $Z1109$.

S-57 ENC data can be viewed by free, open-source software such as $OpenCPN$, but for easier usage in programatic environments such as MATALB it should be converted to a more readable format such as ESRI shapefiles (.shp), GeoJSON (.json) or Geography Markup Language (GML) (.gml).

2.6 Ship propulsion model

A propulsion model, from fuel to propulsion, includes all important parameters affecting the effeciency of the vessel propulsion during voyage. A variant of a propulsion model based on Pedersen and Larsen 2009 and modified to fit the considered ship is shown in Figure 2.7. The model can be used to analyse the fuel efficiency of the voyage, i.e. how to lower fuel usage and increase propulsion. It can also be used to analyze the wear and tear efficiency of the voyage, i.e. how to lower wear and tear and increase propulsion. Propulsion models have gained usefulness because modeling of these important loading conditions are fundamental for achieving efficiency during voyage, including route planning. In fact, the model is already used in the voyage planning system SeaPlanner by FORCE Technology in order to reduce fuel consumption (Schack 2019).

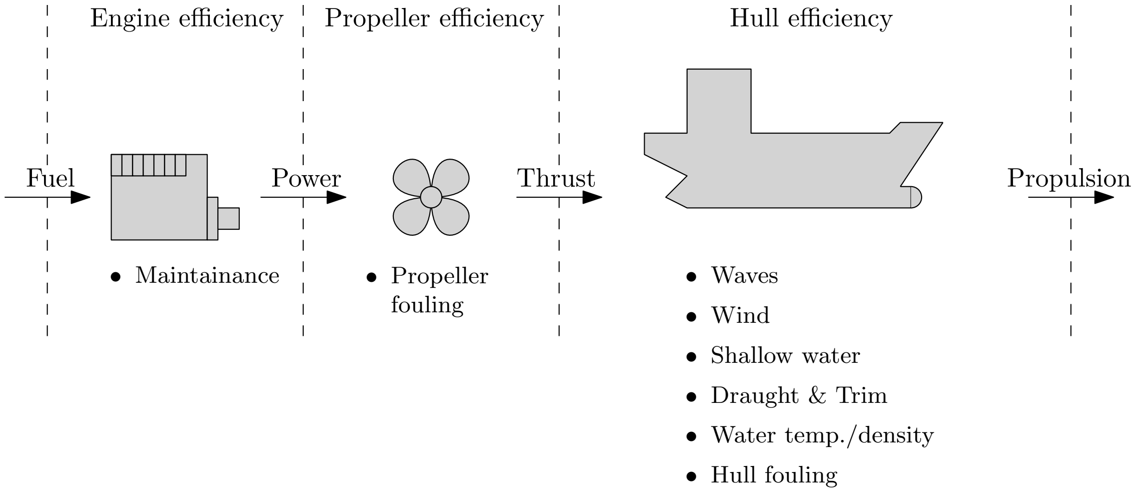


Figure 2.7: Propulsion model

The introduced propulsion model separates efficiency into engine, propeller and hull efficiency. Even though this model can be used to explain many aspects of how to increase efficiency of the propulsion, e.g. maintainance and mechanical aspects, this description focuses on the route planning aspect of increasing efficiency.

The engine efficiency is usually higher for smaller changes in power because most vessels, including the one in scope, use diesel engine configurations. Diesel engines perform most efficient at a given RPM based on the load. If the load, i.e. power required, is changing a lot, the RPM is not given enough time to reach its optimum. If the engine configuration is based on multiple engines, starting and stopping of the motors may decrease efficiency even further. This means that routes that require rapid and large changes in speed, i.e. power, should be avoided in order to increase engine efficiency.

2.6.1 Marine craft

A general ship model in both three and six degrees of freedom (DOF) is presented in (Fossen 2011). For path-planning on a two dimensional map, the 3-DOF model is sufficient, such that only forward motion, sideway motion and rotational motion need to be considered, i.e. surge, sway and yaw respectively.

That means that the simplification of only considering a NED reference frame is not valid for global path-planning around the globe. However, global path-planning considered in this project is more local because only areas are considered, and thus the model is sufficient. Then we have that x is the position in north direction, y is the position in east direction, ψ is the heading angle of the ship relative to north, u is the surge velocity, v is the sway velocity, r is the yaw angular velocity, X is the surge force, Y is the sway force and N is the moment about the z-axis. In addition, the total speed of the ship is defined as $U = \sqrt{u^2 + v^2}$. The course angle χ is defined as the angle of the velocity vector relative to north, and the sideslip β is defined as the angle between the ship's heading and course $\chi = \psi + \beta$. With this model, rotations from body frame to NED frame are carried out utillizing the rotational matrix

$$\boldsymbol{R}(\psi) = \begin{bmatrix} \cos\psi & -\sin\psi & 0 \\ \sin\psi & \cos\psi & 0 \\ 0 & 0 & 1 \end{bmatrix}. \tag{2.1}$$

Instead of a disturbance acting on the actuation, ocean currents are considered as a velocity reference $\boldsymbol{\nu}_c^n \in \mathbb{R}^3$ in NED, such that $\boldsymbol{\nu}_r(t) = \boldsymbol{\nu}(t) - \boldsymbol{\nu}_c(t)$ where

$$\boldsymbol{\nu}_c(t) = \boldsymbol{R}^T(\psi(t))\boldsymbol{\nu}_c^n. \tag{2.2}$$

The resulting kinematics and kinetics are then given by

$$\dot{\boldsymbol{\eta}} = \boldsymbol{R}(\psi)\boldsymbol{\nu} \tag{2.3a}$$

$$\boldsymbol{M}\dot{\boldsymbol{\nu}}_r + \boldsymbol{C}(\boldsymbol{\nu}_r)\boldsymbol{\nu}_r + \boldsymbol{D}(\boldsymbol{\nu}_r)\boldsymbol{\nu}_r = \boldsymbol{\tau} \tag{2.3b}$$

where $\boldsymbol{M}$, $\boldsymbol{C}(\boldsymbol{\nu})$ and $\boldsymbol{D}(\boldsymbol{\nu})$ are model matrices denoted inertia, Coriolis and damping, respectively. A disadvantage of using these simplified dynamics is that they may be inaccurate in harsh sea state, e.g. high waves, when roll and pitch are no longer approximately zero.

For simulating the ship dynamics, the states are transformed to state-space form

$$\dot{\boldsymbol{x}} = \boldsymbol{f}(\boldsymbol{x}, \boldsymbol{\tau}) = \begin{bmatrix} \boldsymbol{R}(\psi)\boldsymbol{\nu} \\ \boldsymbol{M}^{-1}(\boldsymbol{\tau} - \boldsymbol{C}(\boldsymbol{\nu}_r)\boldsymbol{\nu}_r - \boldsymbol{D}(\boldsymbol{\nu}_r)\boldsymbol{\nu}_r) + \dot{\boldsymbol{\nu}}_c \end{bmatrix} \tag{2.4}$$

where $\boldsymbol{x} = [\boldsymbol{\eta}^T, \boldsymbol{\nu}^T]^T$.

2.6.2 Optical aids navigation

All ships should, and most ship are required to know the position of the vessel at all times. Even though most ships are equipped with modern navigational technology such as radars to estimate the ship position, the position of the ship should also be estimatable from optical aids. The motivation is to increase redundancy, such that the ship position can be estimated even if the radar should fail. This means that the route should be navigable with common optical aids, such as lighthouses, buoys and day beacons. Since the ship in the considered scenario is sailing during night, only lights are reliable sources for optical navigation.

The ship position is normally estimated in three accuracies as illustrated in Figure 2.8.

1. Exact
2. Line of Position (LoP)
3. Area of Position (AoP)

The best accuracy is naturally to estimate the position exactly. However, this may not always be possible in all waters and not always needed. In many cases it is possible to estimate the position to a line or an area which may be good enough for the given situation. The position can be estimated by methods such as sector navigating and bearing.

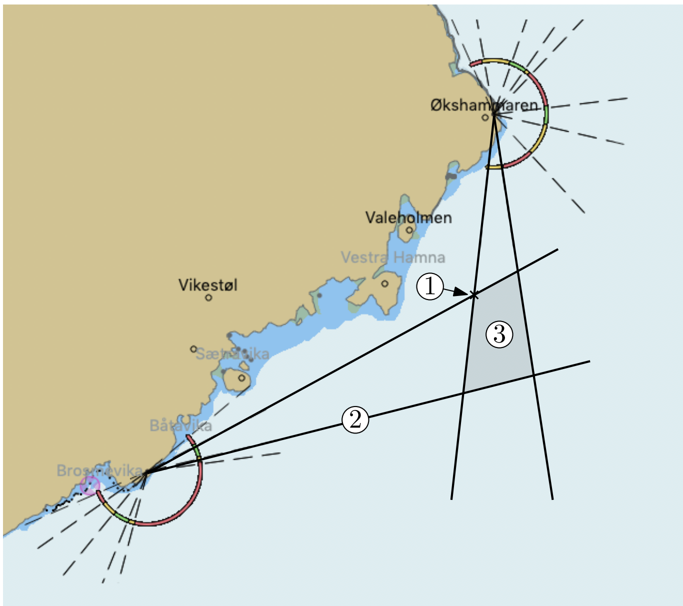


Figure 2.8: Accuracy. 1: Exact. 2: Line of Position. 3: Area of Position.

2.6.3 TSS behaviour

The rules for behaving in a TSS are incorporated qualitatively in the International Regulations for Preventing Collisions at Sea (COLREGs) under Part B, Section I, Rule 10. Ignoring the rules related to anchoring, fishing and exceptions based on inmanoeuvreability, the collection of the traffic rules that are relevant for route planning are:

Rule 10: Traffic separation schemes (TSS)

a. A vessel using a traffic separation scheme shall:

(i) proceed in the appropriate traffic lane in the general direction of traffic flow for that lane;

(ii) so far as practicable keep clear of a traffic separation line or separation zone;

(iii) normally join or leave a traffic lane at the termination of the lane, but when joining or leaving from either side shall do so at as small an angle to the general direction of traffic flow as practicable.

b. A vessel shall, so far as practicable, avoid crossing traffic lanes but if obliged to do so shall cross on a heading as nearly as practicable at right angles to the general direction of traffic flow.

c. (i) A vessel shall not use an inshore traffic zone when she can safely use the appropriate traffic lane within the adjacent traffic separation scheme. However, vessels of less than 20 metres in length, sailing vessels and vessels engaged in fishing may use the inshore traffic zone.

(ii) Notwithstanding sub-paragraph d. (i), a vessel may use an inshore traffic zone when en route to or from a port, offshore installation or structure, pilot station or any other place situated within the inshore traffic zone, or to avoid immediate danger.

d. A vessel other than a crossing vessel or a vessel joining or leaving a lane shall not normally enter a separation zone or cross a separation line except:

(i) in cases of emergency to avoid immediate danger;

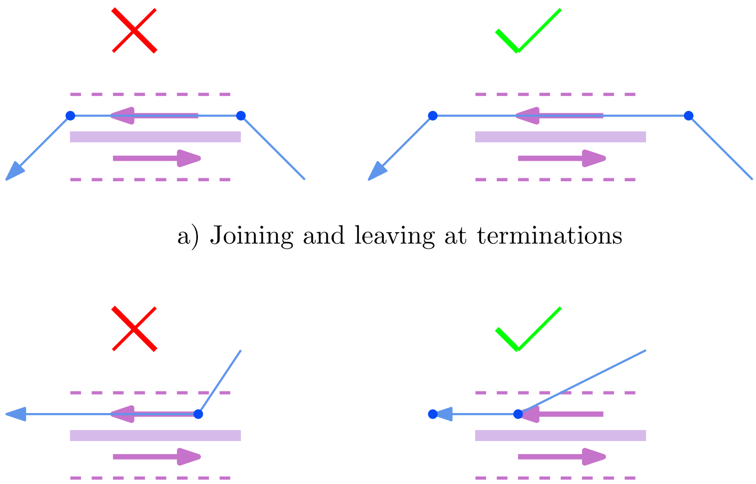
(ii) to engage in fishing within a separation zone.

e. A vessel navigating in areas near the terminations of traffic separation schemes shall do so with particular caution.

f. A vessel not using a traffic separation scheme shall avoid it by as wide a margin as is practicable.

g. A vessel of less than 20 metres in length or a sailing vessel shall not impede the safe passage of a power-driven vessel following a traffic lane.

Rule 10 is one of the longest rules and one of the hardest to understand. Two common mistakes are illustrated in Figure 2.9. When joining and leaving a TSS at terminations, one should do this in a timely manner in order to comply with rule 10. When joining and leaving midway, one should follow the flow as best as possible in order to comply with rule 10.



a) Joining and leaving at terminations

b) Joining and leaving midway

Figure 2.9: TSS rules

2.7 Spaces

Many operating spaces are used to represent different aspects of route planning, or path planning in general. The most common spaces are the workspace and configuration space (LaValle 2006), which separates where the control objective is defined, and where the pose is defined. The workspace $\mathcal{W}$ is where the ship and obstacle geometry are described. The workspace is the physical volume the ship can cover. The configuration space $\mathcal{C}$ is where the motion is represented and contains all possible values the configuration can take.

2.8 Route planning classification

There are many path planning methods in the academic environment, and many ways to classify them. One way to classify route planning methods is based on the classification suggested by (Bitar 2017). This classification separates roadmap-based methods and complete path-based methods. Complete path methods considers the whole space when planning and can be done analytically or approximately. Analytical solutions are hard to solve, and especially time consuming if there are many constraints. That is why approximate solutions such as multiple shooting methods are preferred over analytical. Roadmap-based methods are based on the structure of roadmaps. Combinatorial approaches to motion planning find paths through the continuous configuration space without resorting to approximations.

Due to this property, they are alternatively referred to as exact algorithms. This is in contrast to the sampling-based motion planning algorithms. The following are some examples of planning methods without going into much detail.

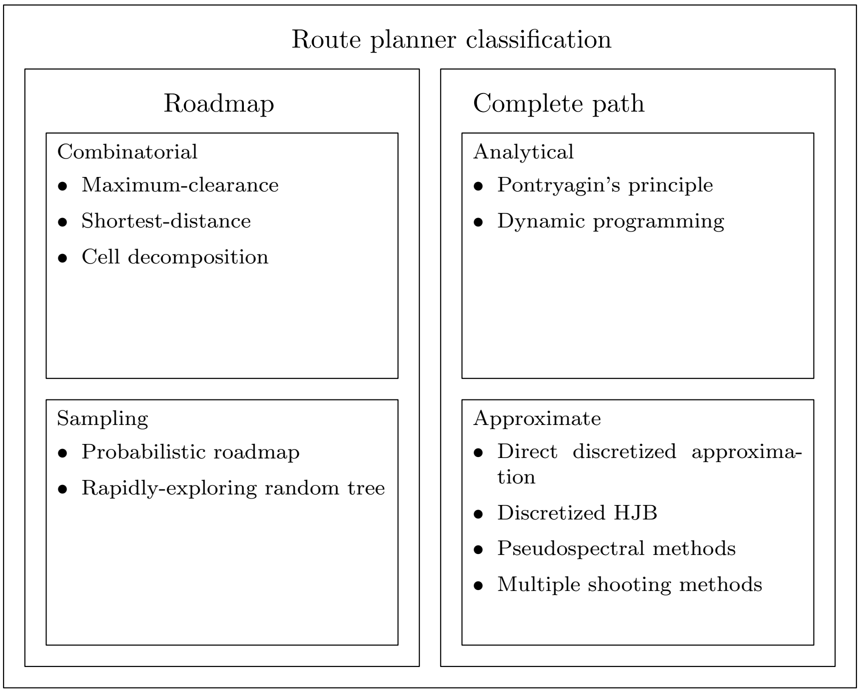


Figure 2.10: Classification of planners (Bitar 2017).

A shortest-distance roadmap is a set of paths in an environment that represent shortest-distance between vertices of obstacles and termination points as shown in Figure 2.11. The shortest-distance roadmap is also known as shortest-path roadmap and visibility graph.

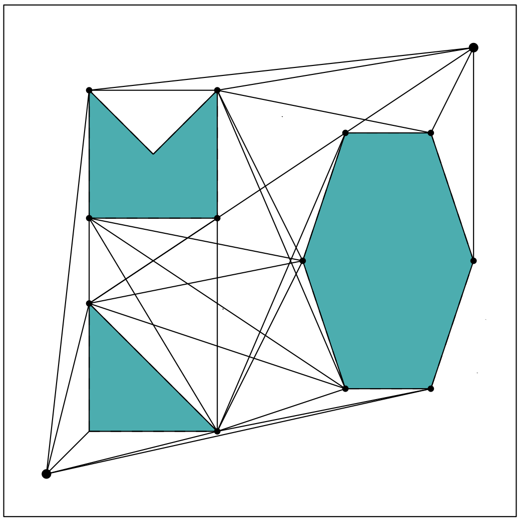


Figure 2.11: Example of shortest-distance roadmap

A maximum-clearance roadmap is a set of paths in an environment that represent maximum clearance between obstacles as illustrated in Figure 2.12. The maximum-clearance roadmap also goes by the names generalized Voronoi diagram, maximum distance roadmap and retraction method. The main objective of the maximum-clearance roadmap is to keep as far away as possible from obstacles. This means that each point along a roadmap edge is equidistant from two points on the boundary of the obstacle space. Each roadmap vertex corresponds to the intersection of two or more roadmap edges and is therefore equidistant from three or more points along the boundary of the obstacles. Using this roadmap is sometimes preferred in mobile robotics applications because it is difficult to measure and control the precise position of a mobile robot. Traveling along the maximum-clearance roadmap reduces the chances of collisions due to these uncertanties.

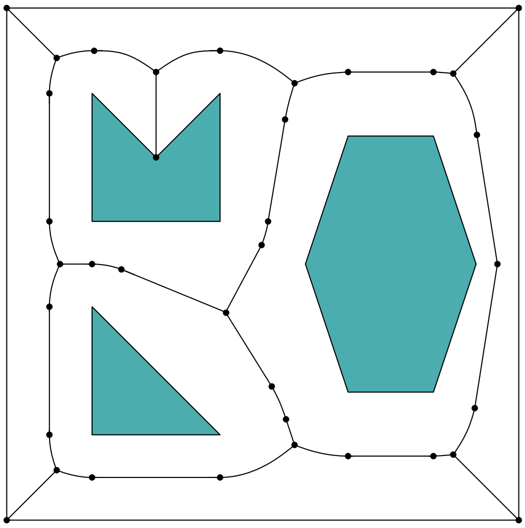


Figure 2.12: Example of maximum-clearance roadmap

The introduction of incremental sampling-based planners, such as probabilistic roadmaps (PRM) and rapidly-exploring random trees (RRT) enabled solving planning problems in high-dimensional state spaces in reasonable computation time, even though the problem is known to be PSPACE-hard (Latombe 1991). PRM and RRT also posess theoretical guarantees such as probabilistically complete, which means that a feasible solution will be found, if one exists, with a probability approaching one if one lets the algorithm run long enough. While RRT provides efficient exploration of high-dimensional state spaces, dynamically feasible trajectories, and demonstrated applicability to complex motion planning applications (Kuwata et al. 2009), it has also been shown to converge almost surely to non-optimal solutions (Karaman and Frazzoli 2011). Consequently, many variations of RRT have been developed in different directions in order to increase performance.

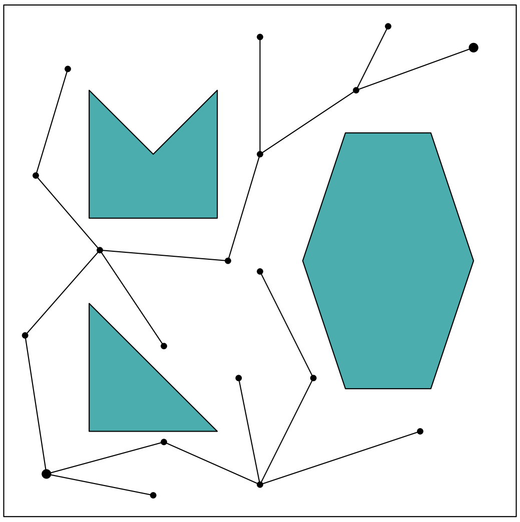


Figure 2.13: Example of rapidly-exploring random tree roadmap

One of the more popular extensions of RRT called RRT* was introduced by Karaman and Frazzoli 2011 that additionally achieves asymptotic optimality, i.e. an optimal solution based on a cost will be found with a probability approaching one. While RRT* provides efficient exploration in high-dimensional state spaces, probabilistic completeness and asymptotic optimality, it may not converge to the optimum in feasible time in practice. Consequently, many extensions of RRT* try to improve the convergence speed.

Chapter 3

Route planning implementation

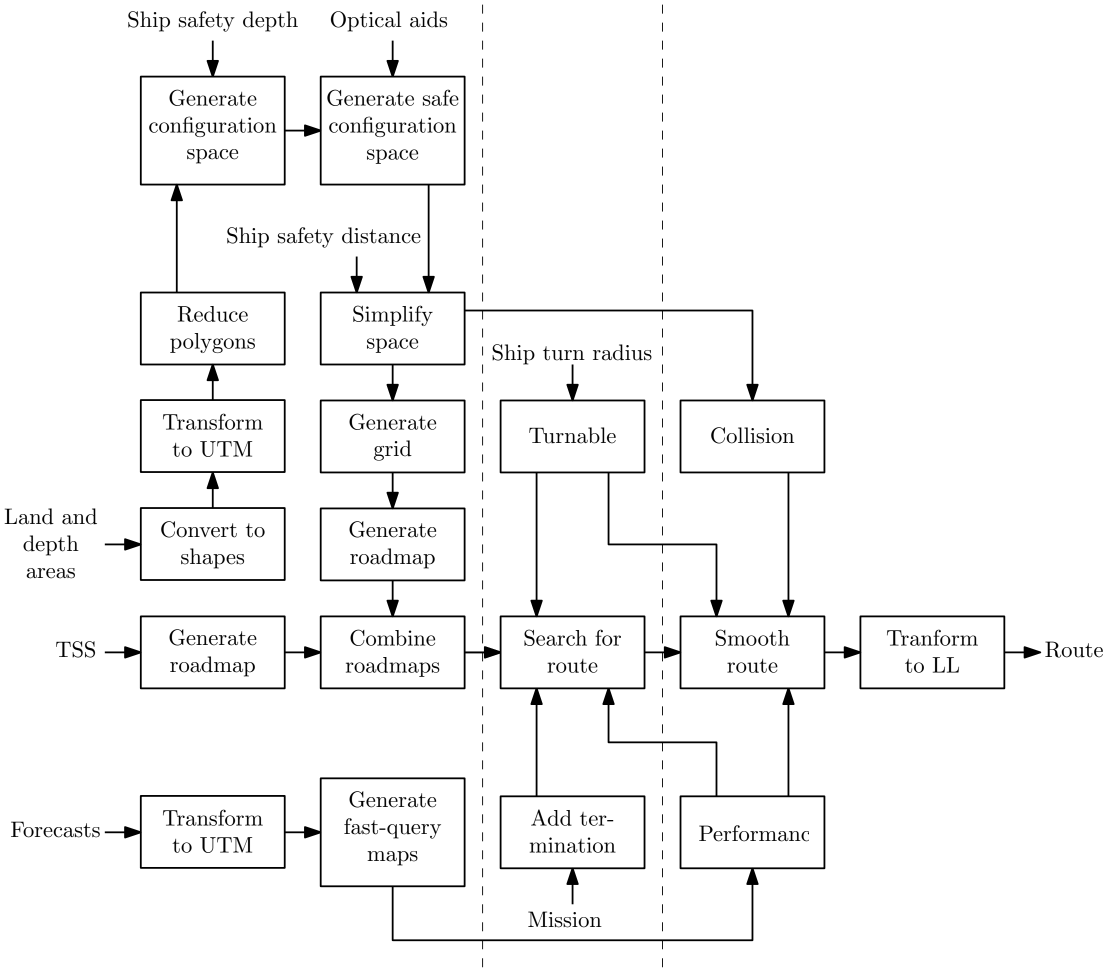


Figure 3.1: Overview of the complete implemented route planning method

Figure 3.1 shows an overview of the complete implemented route planning method. The implementation is designed to solve the problem stated in the introduction, and this chapter is dedicated to describe all the components and the complete model in detail.

The implemented method is based on an A* search on a combinatorial roadmap made from a variable sized grid, combined with a TSS roadmap. There are multiple reasons to why this method was chosen, even though this is only one of many methods to solve path planning problems. The main reason for choosing a roadmap-based method, is because of constraints such as TSS and ECDIS. This makes it hard to apply complete-path planning. Because of constraints, analytical complete-path methods fail completely. Then there is the approximate complete-path methods which is possible to use in the open ocean by using a method such as multiple shooting. However, the complexity of the obstacles makes it too computative to calculate the violation even though it is possible. Even determining if a point is in collision is a computative task. There is also the problem of the TSS which is even more constrained. The reason for applying a combinatorial-based method instead of a sampling-based method is based on repeatability. All sampling based methods are stochastic by nature and may produce different results based on randomness. Sampling-based methods may arguably yield higher performance, but for this purpose a deterministic approach is used to produce consistent result. Maximum-clearance and shortest-distance roadmaps are not used because they are naturally biased towards safety and distance, respectively. In order to incorporate time or energy, other roadmaps such as Deluancy-based and grid-based can be used to give higher flexilibty. The advantages of Delaunay-based and grid-based methods is that the density can be adjusted to be more dense around obstacles. This is an important property because there is little optimization that can be done on the open ocean and alot that can be done by choosing how to maneuver around obstacles. Delaunay-based give more flexibility in regards to the density of the roadmap because the probability distribution can be chosen freely. For a grid-based method, the cell size should match up to neighbour cells, which makes it less flexible in terms of density. However, to receive consistent results, the grid-based is preferred over the probabilistic Delaunay-based method.

The ship and electronical nautical charts are not expected to change for each query of the method, while forecasts and mission are expected to change for each query. This means that it is convenient to divided the method into three stages based on change of mission and forecasts, the pre-processing, query and post-processing phase. In the pre-processing stage, all computation that are independent of the mission are computed. This include processing charts and generating roadmaps. These computations can be done ahead of time, and ideally only once if the ship parameters and charts stay unchanged. A clear motivation is to move as much of the work into the pre-computing phase, because this will decrease computation time for each query. In the query phase, the mission and forecasts are used to compute a route. This typically includes a search, which may be believed to be the main component of any path planning method. However, the search is only a small part of the whole method. At this stage the route is usable, but there are still improvements that can be made. In the post-processing stage, a route is improved to increase performance and obtain feasibility if it is not already feasible. This typically includes smoothing of the route and corrections if needed. The transition between these stages are displayed in Figure 3.1 with vertical lines.

The further description is divided into steps to be computed in order to fulfill the method. Step 1 to X describes the pre-computation phase, step X to Y describes the query phase and Y to Z describes the post-processing stage.

3.1 Convert to shapes

The electronic nautical charts are approved and stored in a standard S-57 format. In this scenario they are stored in multiple charts that can be one of 6 levels of detail. Only level 4, which is the approach purpose, is considered because it gives enough details to be able to sail most routes in the scenario. There are two challenges with these charts. They are not stored in a readable format which is usable by most applications. If they are to be used directly, a parser is needed. However, it is easier to convert to a more readable format. Another challenge is that the charts are not unified in the scenario, meaning that there are multiple chart chattered around to make up the complete chart of the scenario. At some point these needs to be merged together. Because of the complexity of the charts, it is faster to merge the charts together at a later stage, after they have been simplified.

It is convenient to convert to a more user-friendly format such as ESRI shapefile. In order to convert S57-ENC to ESRI shapefile using the GDAL package, install GDAL on the system. For MAC OS, GDAL can be installed in the *Homebrew* environment using

```
>> brew install gdal
```

Then to obtain polygons, execute

```
>> ogr2ogr -f "ESRI Shapefile" -skipfailures CCPRRRRR path/
   to/file/CCPRRRRR.000 -nlt POLYGON
```

and to obtain lines execute

```
>> ogr2ogr -f "ESRI Shapefile" -skipfailures CCPRRRRR path/
   to/file/CCPRRRRR.000 -nlt LINESTRING
```

and to obtain points execute

```
>> ogr2ogr -f "ESRI Shapefile" -skipfailures CCPRRRRR path/
   to/file/CCPRRRRR.000 -nlt POINT
```

3.2 Transform to UTM and LL

Positions in the orignal charts are represented in longitude-latitude (LL) coordinates. A problem with LL coordinates is that distances are not inherited. Calculating a distance between two points is not a trivial task even though it can be approximated for small distances. It requires more calculation with longitude-latitutde coordinates than traditional x-y coordinates. Because distance is such an important performance metric and will be computed many times, computation time in the query phase can be saved by converting to flat coordinates such as UTM. At a later stage, the solution can be converted back to LL coordinates. Conversion from LL to UTM is done using the WGS84 reference coordinate system.

3.3 Reduce polygons

ENC charts use polygons to represent the real-life obstacles. This approximation should satisfy three important properties:

- The polygons should cover all obstacles
- The polygons should have similar shape as the real obstacles
- The polygons should be represented by the minimum amounth of vertices

The first one is a strict requirement which ensures that no collision is possible, while the other two may vary depending on the quality. Similarity of shapes ensures that more usable areas are not marked as obstacle, and few vertices ensures fast computing. For route planning, the outer corners are arguably the most important usable areas, because good routes tend to keep close to the shore in order to minimize distance. Approved maps are generally represented using a high number of vertices to get their status of approval. This high number of vertices adds high complexity to the route planning problem even though this type of quality is not necessarily needed. Thus, the polygons that represent the data of the map, should be downsampled into a similar curve with fewer points that still covers all obstacles.

Reducing polygons by removing vertices is not a new phenomenom, and there exists many methods for this. Two popular methods are the Ramer–Douglas–Peucker and Visvalingam–Whyatt algorithm. There are two issues with standard reduction methods for ENC charts. One issue of these algorithms is that the reduced polygon after reduction may no longer cover the obstacle. This is an issue for the given problem because then the polygons may no longer guarantee no collision. Another problem is that all well-known reduction methods focus on maintaining the overall shape. For ENC maps there are some areas where details are more important than other areas, namely the outer corners. This motivates the need for a new and bette method for reducing ENC charts.

For the purpose of this thesis, a polygon reduction method is developed specifically to reduce ENC charts. The method maintains the property that polygons should cover all obstacles while keeping the important chapes and removing the most unnecessary vertices.

The main principle of the method is presented in Figure 3.2. The algorithm is based on removing one vertex at the time, and by always removing the best vertex at that time. This is a greedy-optimal approach, meaning that the violation of removing each vertex needs to be computed before choosing to remove the vertex that yields lowest violation. The method repeats this behaviour until the violation of removing the best vertex is too large, i.e. $c_k > c_{max}$. There are other termination possibilities such as to stop after a given number of iterations or a given number of removed vertices. However, to achieve similar and fair reduction between independent charts, the termination criteria based on violation is used. The detils of the implemented method is displayed in Algorithm 1.

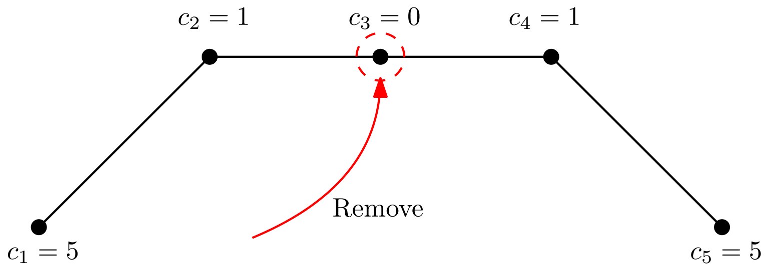


Figure 3.2: The main concept of the polygon reduction method. For each iteration, calculate the cost of removing all vertices and remove the cheapest one.

If a vertex is to be removed, there is an unlimited possible ways to manipulate neighbouring vertices in order to preserve coverage of obstacles. For this method only a selection of cases are considered, as outlines in Figure 3.3. Only the two closest neighbours are allowed to change, and only in a very strict manner as shown in the figure. The cost or violation of removing a vertex is based on shape-preservation. Two metrics are combined to make up the total cost. The first one is the area the polygon needs to extend with to remove the given vertex safely. The second one is the total distance vertices need to move in order to remove the given vertex safely. The former is convenient for preserving the overall shape, while the latter is convenient for conserving corners and avoiding sharp corners. A weight parameter a is used to choose were the details should be preserved. The formal definition of the cost is dependent on the case and is therefore shown as c_k in Algorithm 1. A high value of a yield high preservation of corners, and a low value of a yield high overall preservation. This is a tuning variable which needs to be chosen to achieve the desired balance in preservation.

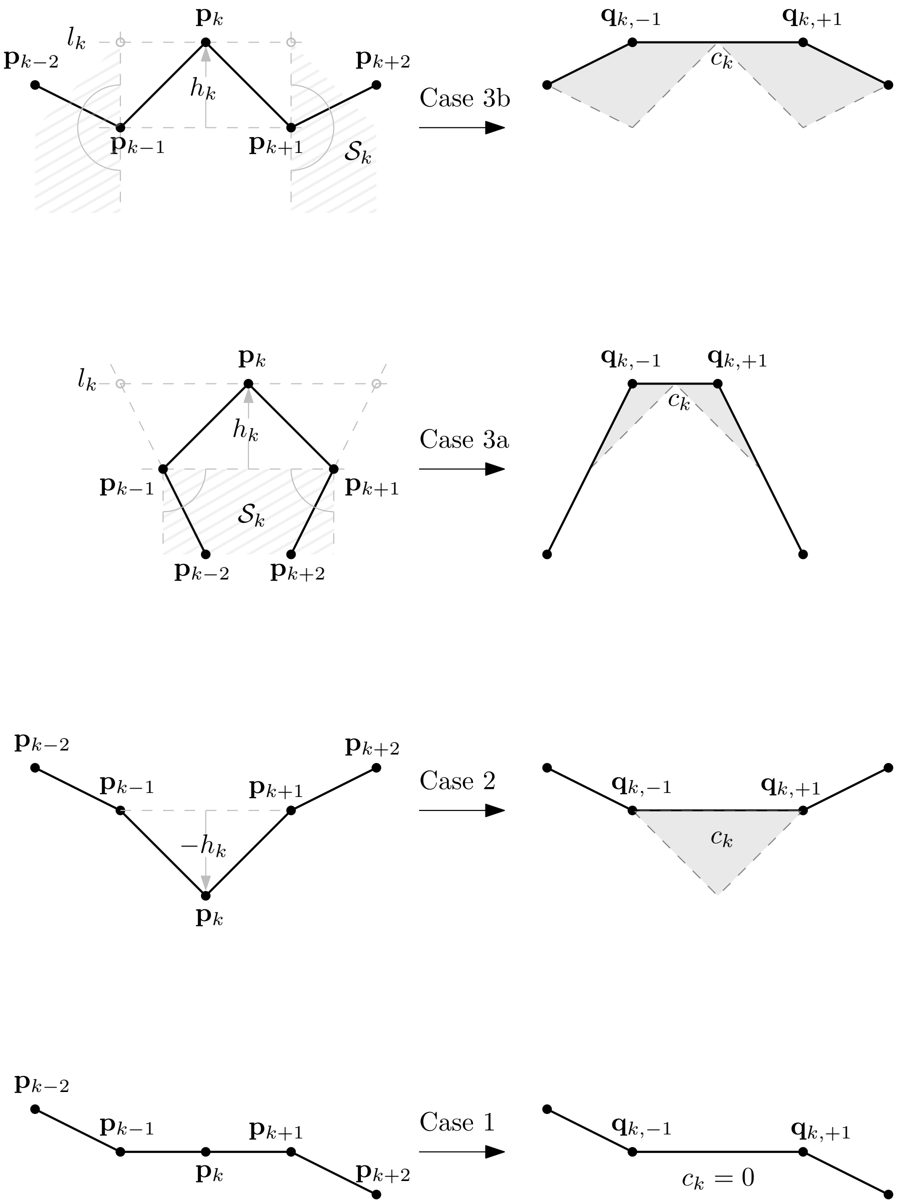


Figure 3.3: Evaluation of removing vertex k for the considered cases.

procedure ReducePolygon($\mathcal{P}$, c_{max})
  for all $k \in \mathcal{P}$ **do**
    $\mathbf{q}_{k,-1}, \mathbf{q}_{k,+1}, c_k \leftarrow$ EvaluateRemoveVertex(k)
  loop
    $k \leftarrow \min_k c_k$
    if $c_k > c_{max}$ **then**
      break
    $\mathbf{p}_{k-1} \leftarrow \mathbf{q}_{k,-1}$
    $\mathbf{p}_{k+1} \leftarrow \mathbf{q}_{k,+1}$
    $\mathcal{P} \leftarrow \mathcal{P}/k$
    $\mathbf{q}_{k-1,-1}, \mathbf{q}_{k-1,+1}, c_{k-1} \leftarrow$ EvaluateRemoveVertex($k-1$)
    $\mathbf{q}_{k-2,-1}, \mathbf{q}_{k-2,+1}, c_{k-2} \leftarrow$ EvaluateRemoveVertex($k-2$)
    $\mathbf{q}_{k+1,-1}, \mathbf{q}_{k+1,+1}, c_{k+1} \leftarrow$ EvaluateRemoveVertex($k+1$)
    $\mathbf{q}_{k+2,-1}, \mathbf{q}_{k+2,+1}, c_{k+2} \leftarrow$ EvaluateRemoveVertex($k+2$)
procedure EvaluateRemoveVertex(k) ▷ See Figure 3.3
  $\mathbf{q}_{k,-1} \leftarrow \mathbf{p}_{k-1}$ ▷ Default case
  $\mathbf{q}_{k,+1} \leftarrow \mathbf{p}_{k+1}$
  $c_k \leftarrow \infty$
  if $h_k > 0$ and $\mathbf{p}_{k-2} \in \mathcal{S}_k$ and $\mathbf{p}_{k+2} \in \mathcal{S}_k$ **then** ▷ Case 1
    $\mathbf{q}_{k,-1} \leftarrow \text{intersection}(l_k, \text{line}(\mathbf{p}_{k-2}, \mathbf{p}_{k-1}))$
    $\mathbf{q}_{k,+1} \leftarrow \text{intersection}(l_k, \text{line}(\mathbf{p}_{k+2}, \mathbf{p}_{k+1}))$
    $c_k \leftarrow \text{area}(\Delta(\mathbf{p}_{k-1}\mathbf{p}_k\mathbf{q}_{k,-1})) + \text{area}(\Delta(\mathbf{p}_{k+1}\mathbf{p}_k\mathbf{q}_{k,+1})) + a(dist(\mathbf{p}_{k-1}, \mathbf{q}_{k,-1}) + dist(\mathbf{p}_{k+1}, \mathbf{q}_{k,+1}))$
  else if $h_k < 0$ **then** ▷ Case 2
    $c_k \leftarrow \text{area}(\Delta(\mathbf{p}_{k-1}\mathbf{p}_k\mathbf{p}_{k+1}))$
  else if $h_k == 0$ **then** ▷ Case 3
    $c_k \leftarrow 0$
  return $\mathbf{q}_{k,-1}$, $\mathbf{q}_{k,+1}$, c_k

Algorithm 1: The implemented polygon reduction algorithm

The algorithm is heavily based on simple geometry which is described using the following notations. $\text{line}(\mathbf{p}_1, \mathbf{p}_2)$ denotes the line passing through $\mathbf{p}_1$ and $\mathbf{p}_2$. $\text{intersection}(l_1, l_2)$ denotes the point where the lines l_1 and l_2 intersect. $\Delta(ABC)$ denotes the triangle made up of the corners A, B and C. $\text{area}(s)$ denotes the area of the shape s. $\text{dist}(\mathbf{p}_1, \mathbf{p}_2)$ denotes the distance between $\mathbf{p}_1$ and $\mathbf{p}_2$.

l_k is the line perpendicular to $\text{line}(\mathbf{p}_{k-1}, \mathbf{p}_{k+1})$ passing through $\mathbf{p_k}$. $\mathcal{S}_k$ is the area bounded, as illustrated in Figure 3.3, by $\text{line}(\mathbf{p}_k, \mathbf{p}_{k-1})$, $\text{line}(\mathbf{p}_k, \mathbf{p}_{k+1})$ and $\text{line}(\mathbf{p}_{k-1}, \mathbf{p}_{k+1})$. One way to check if a point $\mathbf{p}$ is in $\mathcal{S}_k$ is to check if the angle of $\text{line}(\mathbf{p}_{k-1}, \mathbf{p})$ and the angle of $\text{line}(\mathbf{p}_{k+1}, \mathbf{p})$ is in the sector as illustrated in Figure 3.3.

An great example of the method is shown in Figure 3.4. For this example and all further usages, the parameters $c_{max} = 6e4$ and $a = 1e3$ are used. The example illustrates the effectiveness of the method. The number of vertices are drastically reduced by 98.4 % while always covering the obstacle and maintaining detail in the important outer corners.

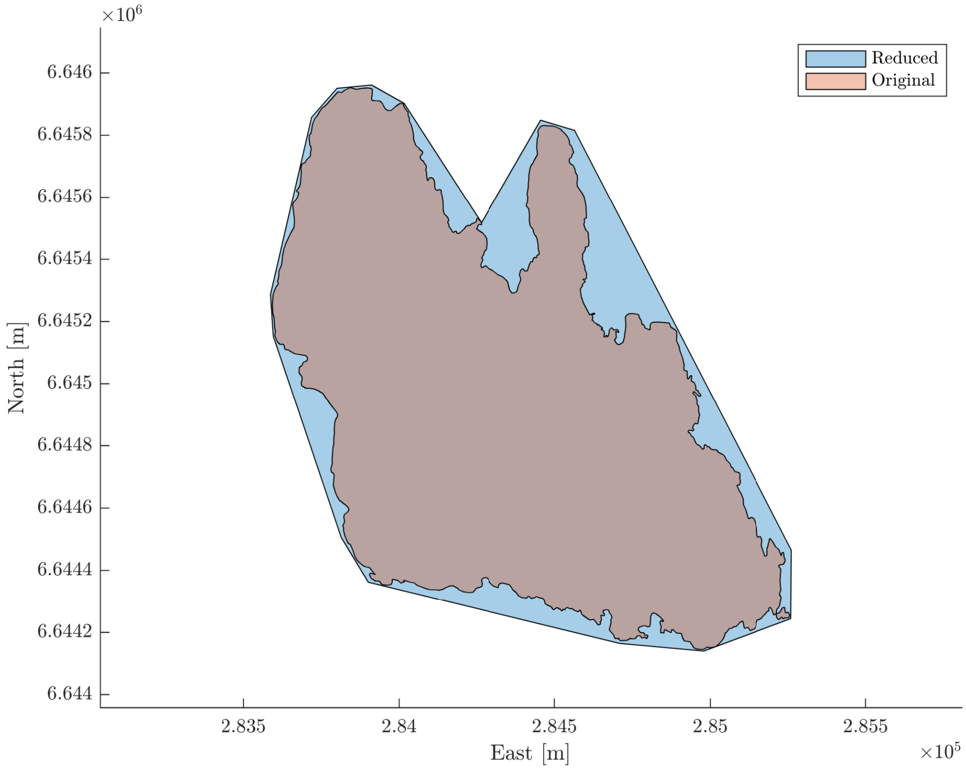


Figure 3.4: ENC shore line after 98.4 % vertex reduction. $c_{max} = 6e4$ and $a = 1e3$. The number of vertices are drastically reduced while always covering the obstacle and maintaining detail in the important outer corners.

Figure 3.6 shows the worse performance if a is set to zero and reducing the same amount as in the previous example. The shape is not preserved well. Also note the development of spikes. These are not desirable in the route planning because a spike might drastically influence the choise of path even though the area it extends the obstacle with may be small. In contrast, choosing a large smoothing parameter a leads to a large loss in area which the reduction method extends the polygon with. These two examples illustrates the important of choosing a well balanced a for the given scenario.

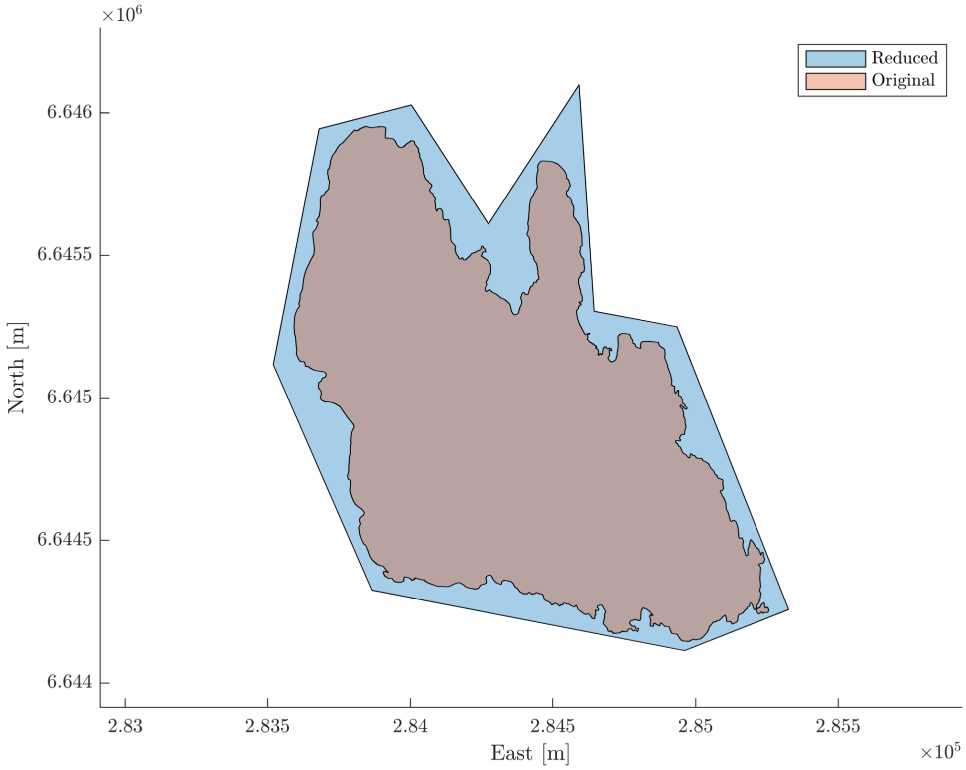


Figure 3.5: ENC shore line after 98.4 % vertex reduction. $c_{max} = 1e5$ and $a = 0$. The number of vertices are drastically reduced, but the shape in important areas is not maintained well. Spikes are undesirable.

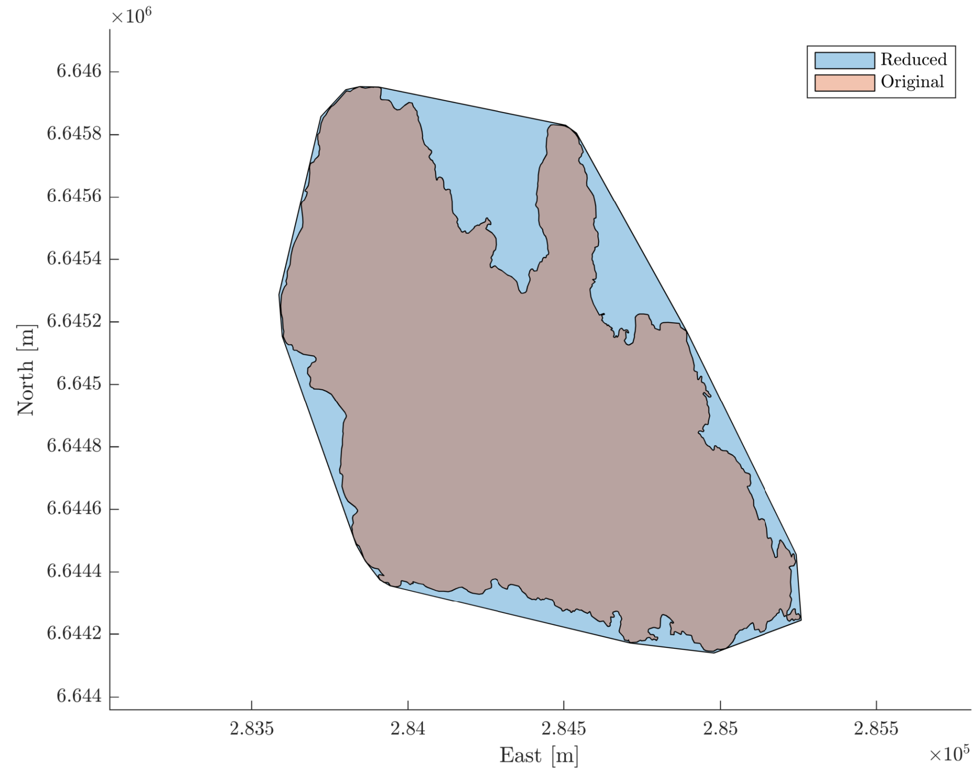


Figure 3.6: ENC shore line after 98.4 % vertex reduction. $c_{max} = 1e10$ and $a = 1e9$. The number of vertices are drastically reduced to ensure preservation in corners, but the reduction also leads to a large loss of area.

3.4 Generate configuration space

Until now the workspace $\mathcal{W} \in \mathbb{R}^2$ has been considered. Which is the space where the land areas from the original charts makes up the obstacles and the free space is all reachable waters. The configuration space $\mathcal{C} \in \mathbb{R}^2 \times \mathbb{S}$ also considers depth, so that all land areas and all waters where the depth may be shallower than the safety depth makes up the obstacle space. All other reachable waters make up the free space. The purpose of defining the configuration space in such a manner, is to ensure ECDIS compliance.

The ENC charts have been reduced, but are still scattered. This is also the step where all ENC charts are merged into one. The configuration space is generated by investigating all depth areas in all reduced charts. If the minimum depth in an area is shallower than the safety depth of the ship, the area is dangerous. All dangerous areas are merged together with all land areas into one area, namely the obstacle space $\mathcal{C}_{obs}$. The free space is not the inverse of the obstacle space. The free space also excludes unreachable lakes or other unreachable areas. Figure 3.7 shows the generated configuration space.

Figure 3.7: Obstacle spaces in a portion of the considered scenario.

3.5 Generate safe configuration space

The safe configuration space is the configuration space where the ship can navigate safely based on optical aids. The safe space is an extension made to the traditional configuration space to ensure that the requirement of optical route navigation is satisfied. This step can be skipped if optical route navigation is not a requirement.

A safe configuration space is well defined in areas with many lights. However, making up a safe configuration space is not as simple as requiring to always see a light. Seeing a light is not a precondition for safety. The safe configuration space implemented for this method is based on four navigation principles separated to four zones. The four navigation principles are light sector navigation, approaching optical aids, bearing optical aids and dead-reckoning. They are separated to the safe zone, retraction zone, bearing zone and the dead-reckoning zone, respectively.Figure 3.8 shows the four zones.

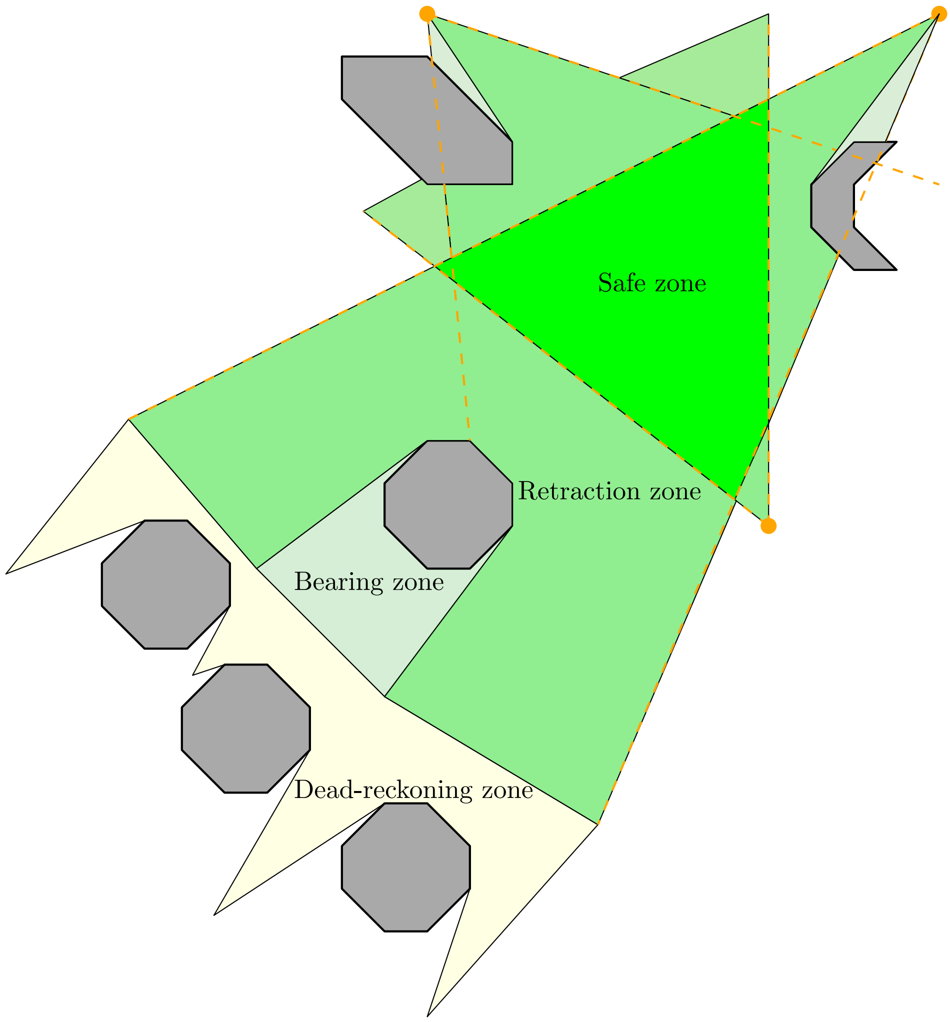


Figure 3.8: The safe configuration space consists a safe zone, a restractable zone, a bearing zone and a dead-reckoning zone. Orange dots illustrates lights with light sectors marked with dotted orange lines.

The safe zone is an area at sea where the ship may not know the position exactly, but is guaranteed to be inside the configuration space and thus safe from collision with the sea bed. This is ensured by constructing the safe zone such that it is a subset of the configuration space and all edges of the polygon describing the area follows an edge of a light sector. This is shown in Figure 3.8 where the light sectors make up the sides of the polygon defining the safe zone. This means the the navigator knows exactly when leaving the safe zone by observing the lights around and performing the required action to keep inside the safe zone. The properties of the safe zone ensures that if the ship loses GPS signal or other navigational equipement in the middle of the night with no visual sight of the land, the navigator can remain safe by only watching the surrounding lights. A problem with the safe zone is that they are only appearing in well lit and open oceans. Thus it is impossible to plan a route entirely in safe zones, even though this would make the routes optically navigable. The solution for this implementation is to extend the safe zone with other zones to extend the coverage. With all zones included in the safe configuration space, a route can possibly be made between most locations because the coverage is significantly improved from using a safe zone alone.

The retraction zone is an area at sea where it is possible to reach a safe zone by approaching or distancing lights, i.e. sailing in the direction of or from visible light sources. This mean that if all navigational aids are lost in the middle of the night, a safe zone can be reached by sailing in the direction of or from a light. The position can also be estimated by using compass measurements, but is not necessary to remain safe. This is a natural way to extend the safe zone, because safety can be ensured if the safe zone can be reached by such an easy action.

The bearing zone is an area at sea where at least one light source is visible and a retraction or safe zone can be reached by bearing. In a bearing zone it is not safe to approach the light, but by knowing the heading of the ship from compass and using the light as reference, the position can be estimated good enough to reach a retraction zone or safe zone safely. The safe configuration space is further expanded by the dead-reckoning zone.

The dead-reckoning zone is an area at sea where there are no visual lights, but a bearing, retraction or safe zone can be reached by estimating the current position from previous measurements. Dead-reckoning is naturally not a precise way to estaimte the position, and drifting is expected. The position error is expected to increase with time such that there is a maximum distance that can be travelled with no measurements. The error is also expected to increase more during turning. For the dead-reckoning zone, turning is not considered, which is a limitation to the method.

Because of the complexity of the safe configuration space, and time-limitations, the complete safe configuration space was not implemented. A simplified safe configuration space consisting only of the safe and retraction zone was implemented instead. There are other arguments for not implementing the complete space as well. The distance between Stavanger and Bergen is highly lit, meaning dead-reckoning may hardly be needed because at least one light is probably visible at all times. The lights are also placed strategically where most bigger ships would sail. The considered ship is simply not allowed into shallower waters where dead-reckoning might have been usable, because of the limitation of the ECDIS and size of the ship.

The implementation of finding the safe zones is highly based on geometry. First, all sector lines shown in orange in Figure 3.8 are found from the light sources. Then it is mapped which lines each line intersects with and at which positions. Then a search is utilized to find all n-dimensional polygons made up from the lines. Only the polygons completely in the free configuration space are merged togheter to make upthe safe zone. The implementation of finding the retraction zone is approximated. The light sectors are split into multiple smaller sectors. Each sector is either limited by the range of the light or the intersection with an obstacle.

3.6 Simplify space

The configuration space or safe configuration space is originally three dimensional, consisting of the posision and heading of the ship. Planning in three dimensions is not desirable or may not be feasible because many methods do no scale well in high dimensions. A standard technique is simplifying the space by padding by a safety distance such that rotation of the ship is neglectable. If position is measured from the centre of the ship and no drifting is expected, half the length of the ship is the minimum. The principle is shown in Figure 3.9.

For the given scenario, the ship length is about 70 meters such that safety distance is chosen to be 35 meters. In practice this is a small margin, but is feasible under good conditions. The safety distance is chosen so small to increase maneuverability and then also the potential to choose from more routes.

a) Original space b) Simplified space

Figure 3.9: Spaces can be simplified from three to two dimension by padding all obstacles.

3.7 Generate grid

A grid is made up by a collection of cells.The grid is generated by initializing a cell containing the whole area. If the cell is in collision with an obstacle, it is decomposed into four equally sized cells. The procedure is repeated for all cells until it has reached a minimum size. The minimum size is set to 15 meters, which is small enough to sail through all narrow passages in the given scenario. Figure 3.10 shows the generated variable sized grid for a portion of the chart.

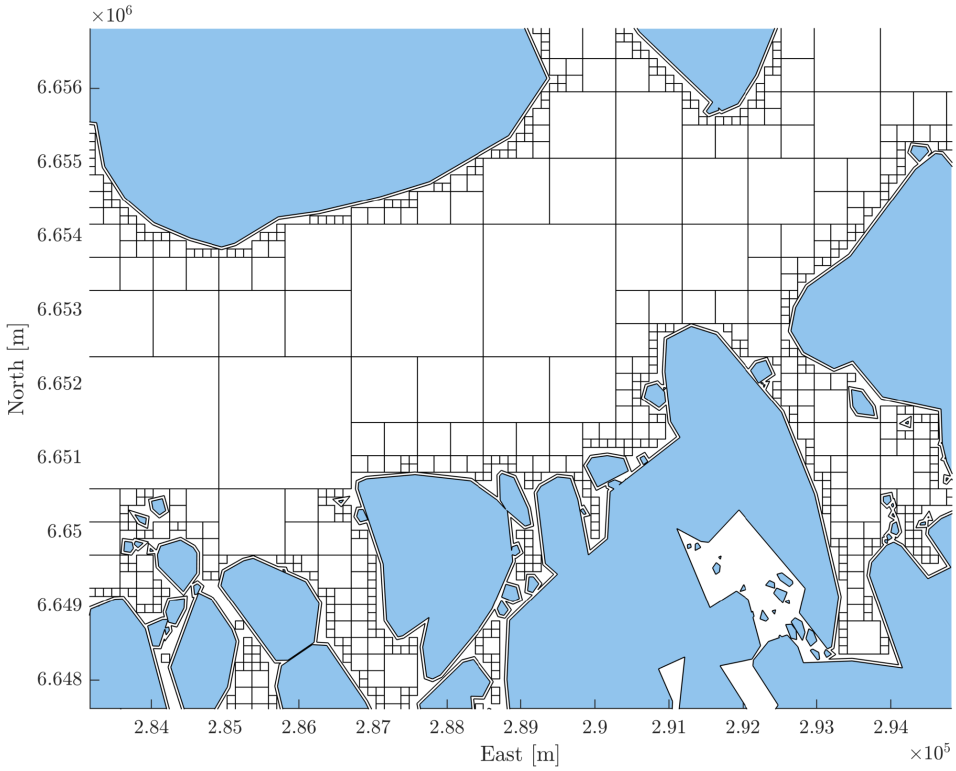


Figure 3.10: The generated variable sized grid. Only cells down to 100 meters are shown.

3.8 Generate roadmap

Two roadmaps are generated. One is generated from the previously generated grid and one from the TSS.

The roadmap from the grid is generated by connecting the center of all cells to the center of all neighbour cells. A neighbour cell is a cell that intersects the cell, i.e. all cells directly to the left, right, top, down and all four corners. The roadmap generated on top of the grid is shown in Figure 3.11.

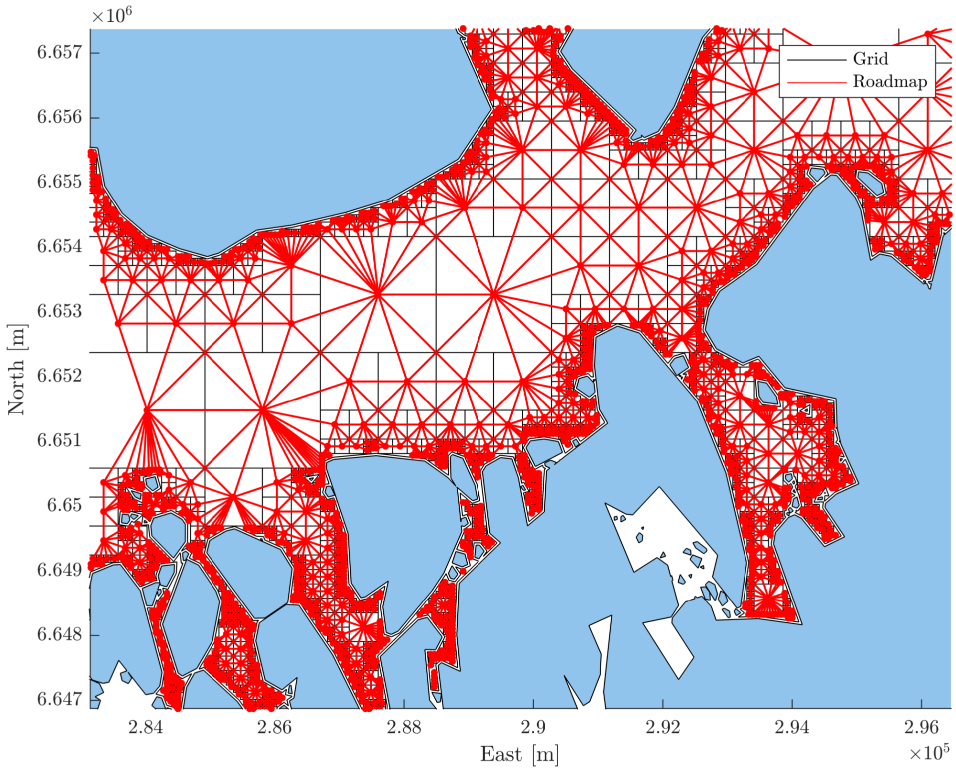


Figure 3.11: The generated roadmap from the grid. Only cells down to 100 meters are shown.

The roadmap from the TSS is generated by strictly obeing the traffic lane directions and recommended directions. The generated roadmap is shown in Figure 3.12. The roadmap is made by extending a line from each lane direction definition (arrow) until the traffic direction terminates. The traffic direction terminates when leaving the TSS zone or crossing a separator. The complete TSS zone is extended by 1500 meters to incorporate safe entering and leaving of the TSS. The lines are then combined to make up a roadmap.

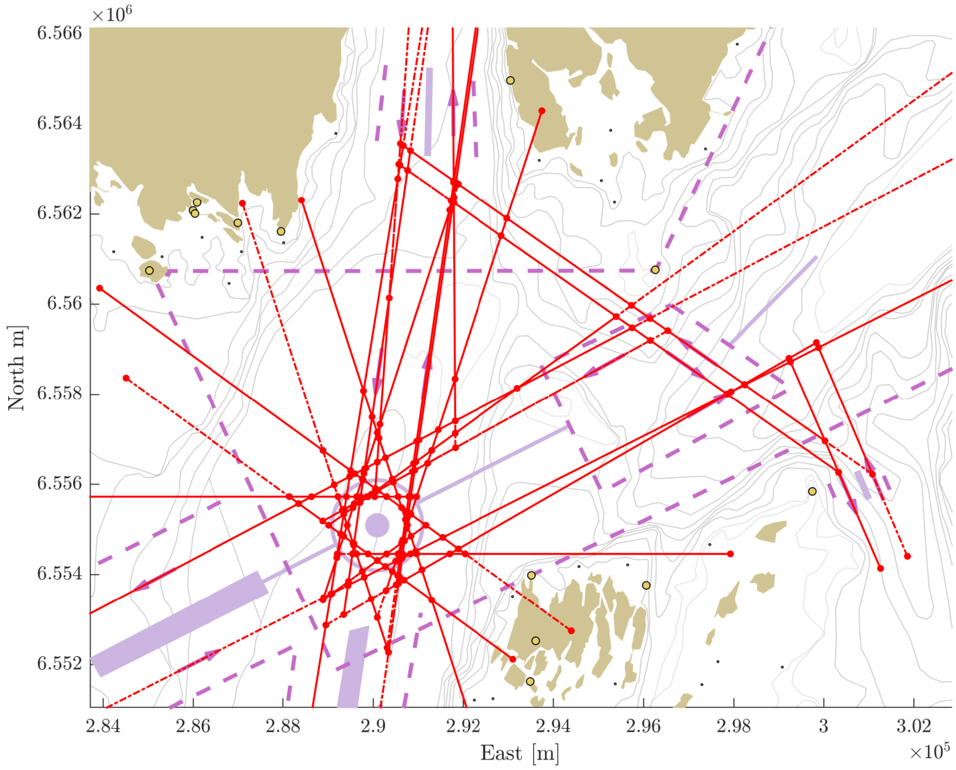


Figure 3.12: The generated roadmap from TSS. Note that the roadmap is not bi-directional, directions are just not indicated.

3.9 Combine roadmap

The roadmap generated from the TSS and the roadmap generated from the grid are combined. This is done by first removing the part of the grid roadmap where the TSS is defined. They are then combined by isolating the terminating vertices of the TSS roadmap, i.e. the roadmap ends that enters and leaves the TSS area. Each terminating vertex is then connected to the closest vertex of the grid roadmap.

3.10 Generate fast-query maps

The route planning search needs the functionality to check the forecasts for a given position. All forecasts are originally defined by a uniform grid in LL coordinates. That means it is fast to look up the wave height, current direction etc for a given LL position. The exact cell to check is trivially calculated. The problem arises when it is converted to UTM and the grid is no longer uniform. It is then necessary to convert the query position to LL coordinates in order to get the exact data. This is a time-consuming operation if executed many times, which it will.

The implemented solution is to approximate a new uniform grid in UTM coordinates which will be used as a look-up table. This is done by making a new uniform grid in UTM coordinates. The center of each UTM cell is converted to LL coordinates and used to look-up the original data point, which is used in the given UTM cell. There might exist better approximations, but for this purpose it is good enough because the distortion in converting to UTM is not that significant.

3.11 Turnable

If a route segment, i.e. a waypoint and two connecting legs, is turnable is determined from the ship turn radius. The radius of acceptance R_{acc} which is calculated from the turn radius should be smaller than a maximum value $R_{acc,max}$ as illustrated in Figure 3.13. $R_{acc,max}$ is defined as half the length of the shortest leg connected to the considered waypoint. In theory the accepted radius may be larger depending on the previous or next turn. But because the routes are built piece by piece, this law ensures that turnability can be calculated without knowing the previous or next turn.

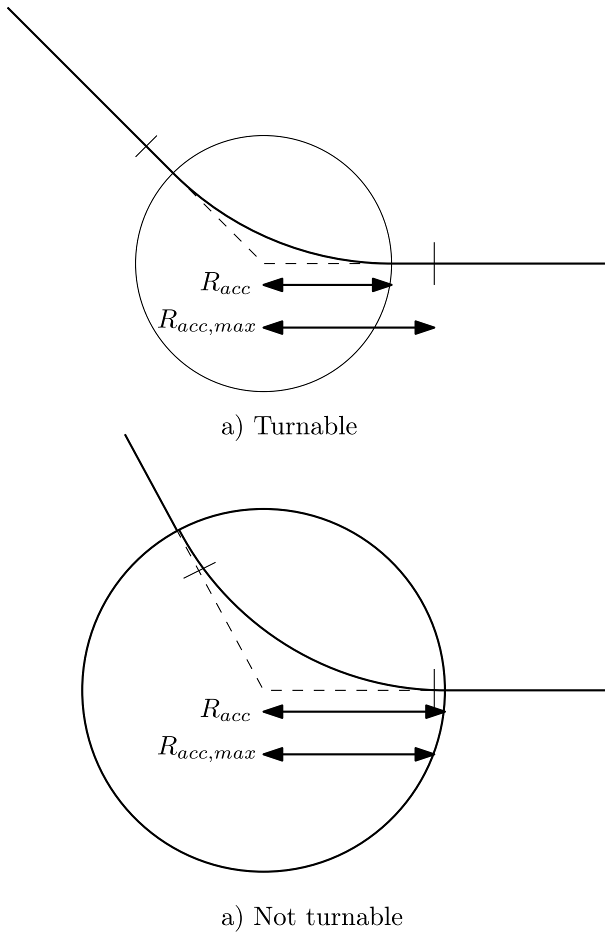


Figure 3.13: Examples of one turnable and one not turnable route segment.

3.12 Add termination

Up until this point, the mission itself has not been considered. The roadmaps are completely generated without even knowing where we are going from and to. This is by design in order to push as much work as possible to the pre-processing stage. At this point the combined roadmap can be extended by the start and end positions. This is done by connecting the start and end positions to their closest roadmap vertex.

3.13 Performance

The search needs a way to determine how well it is performing in order to find the best solution. This is traditionally done by performance metrics. The search is performed piecewise, so a piecewise performance metric is required. That means that the performance of each turn t_i and leg l_i needs to be found, and thus also defined. The choice of performance metric strongly affect the solution, arguably even more than the search method itself. Therefore the choice of metric should not be left to chance. The following metrics are the ones used for the results.

A performance metric with the objective of minimizing the number of turns is defined as

$$f_{turns}(t_i, l_i) \triangleq 1. \tag{3.1}$$

The motivation of reducing the number of turns is to reduce turning and rudder wear and tear.

A performance metric with the objective of minimizing the travel distance is defined as

$$f_{distance}(t_i, l_i) \triangleq |t_i| + |l_i| \tag{3.2}$$

where $|\cdot|$ indicates the length of the turn or leg. Travel distance is not necessarily the fastest, but may be a good indicator if weather is not significant.

A performance metric with the objective of minimizing the sharpness of the turns is defined as

$$f_{turning}(t_i, l_i) \triangleq \angle(t_i) \tag{3.3}$$

where $\angle(t_i)$ is the angle of the turn, i.e. the angle of the next leg relative to the angle of the current leg. Minimizing sharpness instead of number of turns may be a better metric for reducing turning or rudder wear and tear.

A performance metric with the objective of minimizing the travel time is also dependent on the average current strength c_i on l_i and in the direction of l_i and is defined as

$$f_{time}(t_i, l_i, c_i) \triangleq \frac{|t_i| + |l_i|}{c_i + V_d} \tag{3.4}$$

where V_d is the planned speed for the journey. Minimizing travel time may yield better results than minimizing distance. Weather such as current may affect some routes to become slower than longer routes.

A performance metric with the objective of avoiding waves is defined as

$$f_{time}(t_i, l_i, w_i) \triangleq w_i \tag{3.5}$$

where w_i is the average wave height in l_i. This metric is tested because it is believed that avoiding waves may yield lower actuator wear and tear.

The metrics can also be combined, but there is no trivial way to determine which metric more important. For this thesis, they are analysed separately.

3.14 Search for route

A roadmap is already made, and the mission endpoints are included in the roadmap. The route is found by utilizing an A* search on the roadmap while maintaining turnability and optimizing on one of the performance metrics. A general A* search is based on knowing the start node, cost of going from one node to another node, the heuristic cost from a node to the goal, termination for when the goal is reached, which are possible neighbour nodes and if the transition from one node to a neighbour node is feasible. As shown in Figure 3.14, the start node and termination for when the goal is reached is determined by the mission endpoints. The cost is determined from the performance metrics. Possible neighbour nodes are determined from the roadmap, and feasibility is determined by turnability. The heuristic cost Note that collision is not a criteria for feasibility, because this criteria is already baked into the roadmap. The A* search returns a network of parents which is used to construct the route.

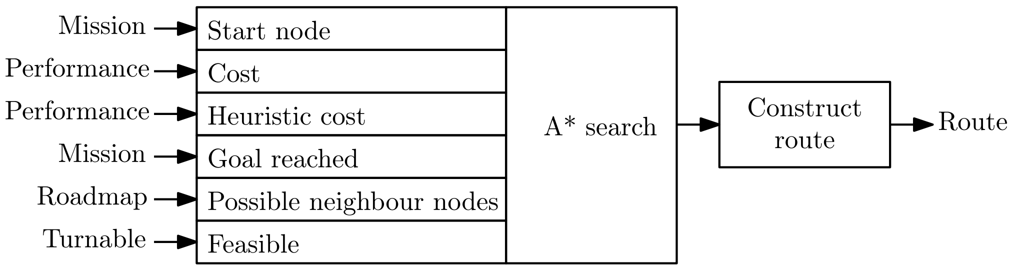


Figure 3.14: The implemented search based on A*.

3.15 Smooth route

The produced route is highly dependent on the roadmap. The problem of the roadmap is that the density in open ocean is low, and the quality of the generated routes in these areas are similarly low. Because the problem is in open waters, it is fairly easy to apply smoothing to improve these areas. Figure 3.15 shows the raw route from the A* solver compared to the smoothed route.

The implemented smoother starts at the first waypoint in the route. It then checks if it is possible to remove the next waypoint by considering both collision and turning feasibility. If it is feasible to remove the next waypoint, it stores the change in performance of removing that waypoint and continues to check if it is possible to also remove the second next waypoint. The method continues until removing a waypoint is no longer feasible. The method then chooses to reduce to the waypoint that yielded best performance. This procedure is repeated from the next waypoint after reduction, until the end is reached.

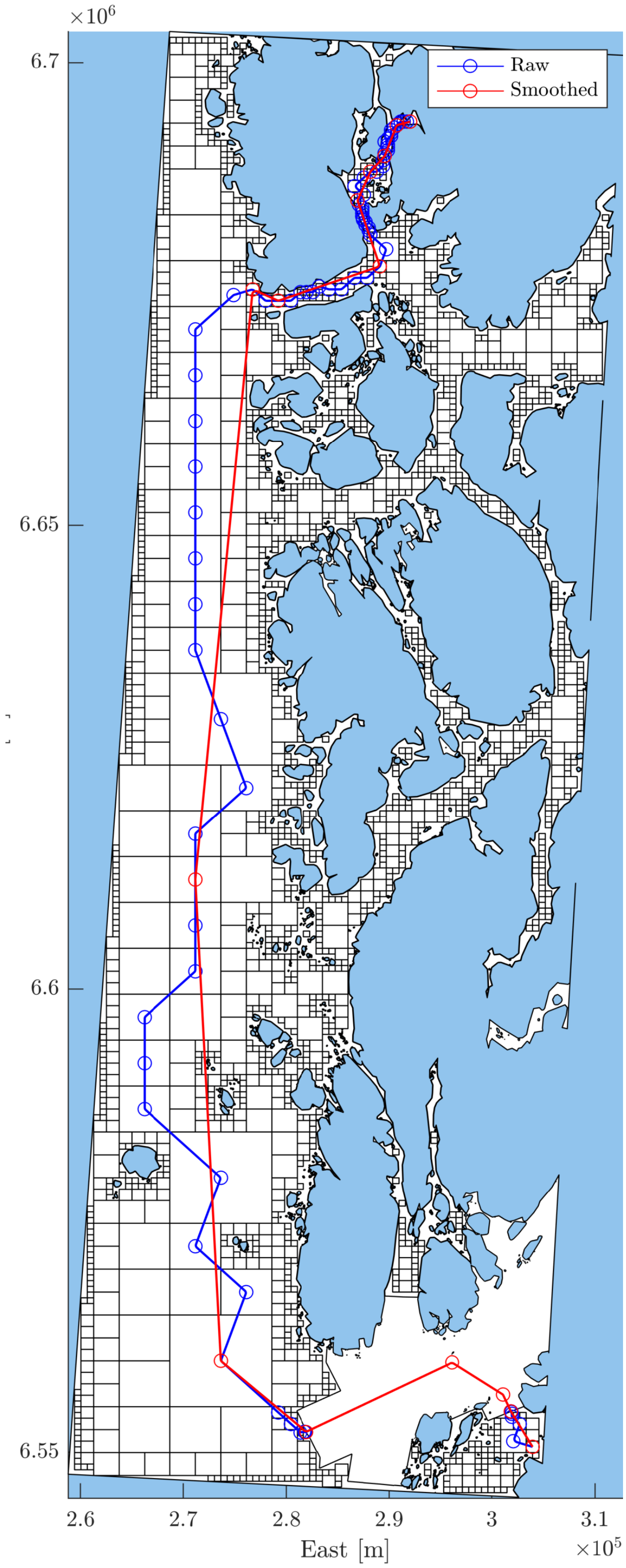


Figure 3.15: Example of smoothing of the raw path using the turning performance metric.

Chapter 4

Results and discussion

This chapter contains the validation and benchmarking of the automatically generated routes from the previous part, as well validation and benchmarking of existing route planning methods for comparison. The validation and benchmarking details are presented, and then the main results are presented. Other results are presented last.

4.1 Validation

Validation is performed by an experienced navigator and an ECDIS. As shown in Figure 4.1, the ECDIS automatically checks if a route is feasible, meets a deadline and is compliant to other ECDIS requirements. Feasible means that the route is geometrically correct, i.e. all turns have long enough legs to satisfy the sharpness of the turn and the turn radius of the ship. The ECDIS also estimates the ETA to check if it is possible to reach the destination before a deadline. The other ECDIS requirement is related to never cross an area where it is a chance to hit the seabed.

The requirements to comply with Traffic Separation Scheme (TSS) and be able to navigate using optical aids is validated manually by an experienced navigator. The navigator browses the route waypoint by waypoint and determines from experience if the route violates TSS rules or if it is safe to navigate using optical aids.

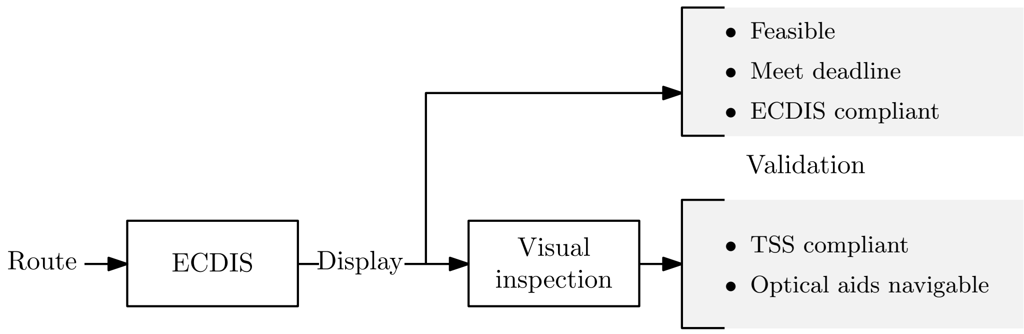


Figure 4.1: Route validation process

4.2 Benchmarking

If a route is validated, it can be benchmarked such that performance of the route can be compared to the performances of other routes. To achieve a trustworthy and objective benchmarking, well-defined performance metrics and a trust-worthy simulation environment is used.

Benchmarking is performed using the high-fidelity simulator K-Sim as shown in Figure 4.2 which is an approved maritime industrial simulator. The motivation for using a high-fidelity simulator instead of experimental data is repeatability. Getting acceptance for optimality with the use of empiric data has proven hard. The complexity of empiric data and lack of knowing all disturbances gives questionable results.

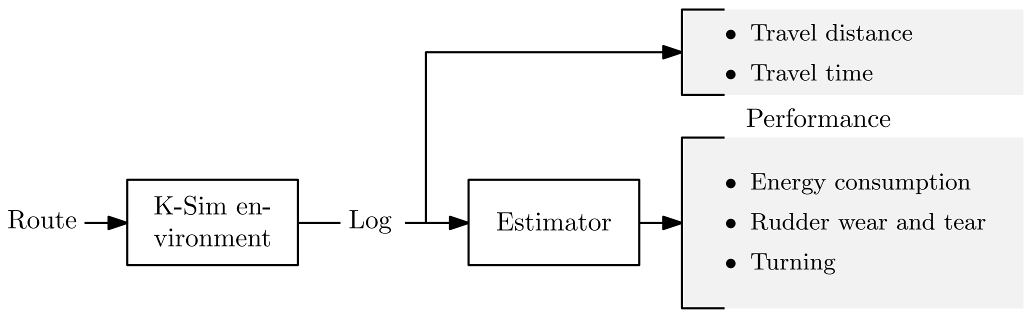


Figure 4.2: Route benchmarking process

The performance metrics are the previously defined desirables from the problem formulation. Travel distance and travel time can be directly gathered from the log data of the simulation, but the other metrics must be estimated from the log data. Because the total engine power $p(t)$ can be gathered from the simulation, the total energy consumption is trivially estimated by

$$c_e(t) \triangleq \int_0^t p(\tau)d\tau. \tag{4.1}$$

Rudder wear and tear happens because of change in the rudder angle and is therefore estimated with

$$c_a(t) \triangleq \int_0^t \dot{r}(\tau)d\tau. \tag{4.2}$$

where r is the rudder angle.

Turning is similarly estimated by

$$c_t(t) \triangleq \int_0^t \dot{\psi}(\tau)d\tau \tag{4.3}$$

where ψ is the ship heading.

The Kongsberg Simulator (K-Sim) is not a single program or service, but an environment where a collection of services are connected together to make up the simulator. For the purpose of this simulation, only ECDIS and the operator station is the services of interest. The rest of the environment is considered a black-box as shown in Figure 4.3. The route is given to an ECDIS which based on virtual measurements from the rest of the K-Sim environment utilizes tracking, i.e. route following, by controlling the ships rudder and propeller. An operator station is connected to the same environment which continuously log data. The operator station is also used to configure ships, set weather etc, but is not important for the workflow.

Figure 4.3: Kongsberg simulator environment

Because of limitations in the K-Sim environment, currents, wind and waves can not be defined by arbitrary positions. The currents and wind are simulated in different areas as shown in Figure 4.4, which are constructed to mimic the forecasts for currents and wind. Waves are simulated from wind. This means that the forecasts and simulated values are different. This is realistic in practice.

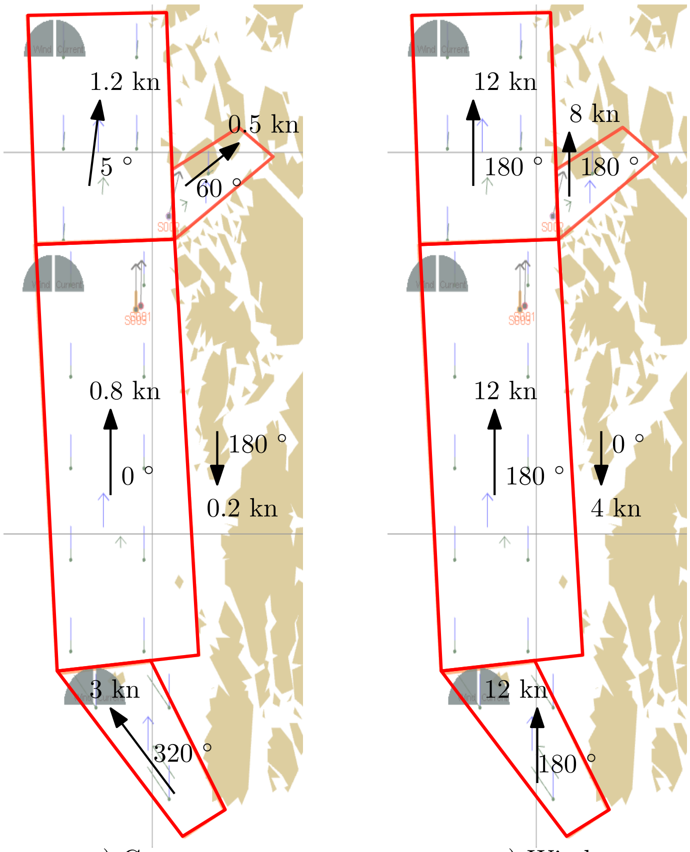


Figure 4.4: Simplified weather in Kongsberg Simulator environment

4.3 Simulation setup

Simulations are done in a similar manner to increase repeatability and comparability. All routes are create from $\mathbf{p}_0 = [5.58, 59.05]^T$ to $\mathbf{p}_f = [5.235, 60.325]$. Simulations start from state $\nu_0 = [\mathbf{p}_0^T, \pi/6]^T$ to a circle of radius 500 m at $\mathbf{p}_f$ The deadline is set to 6 hours. The nautical charts are simplified by bounding latitude and longitude by 59-60.4 and 4.8-5.6, respectively. The considered wind is visualized in Figure 4.5, waves are visualized in Figure 4.6 and currents are visualized in Figure 4.7.

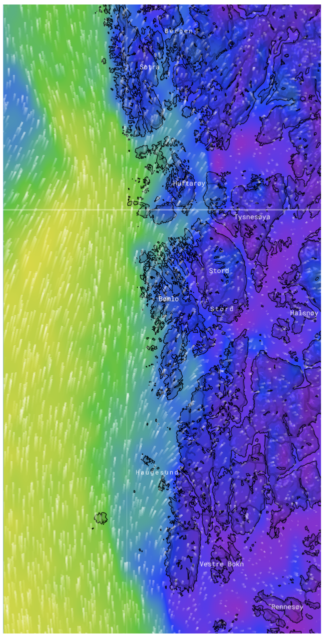


Figure 4.5: Wind of scenario visualized.

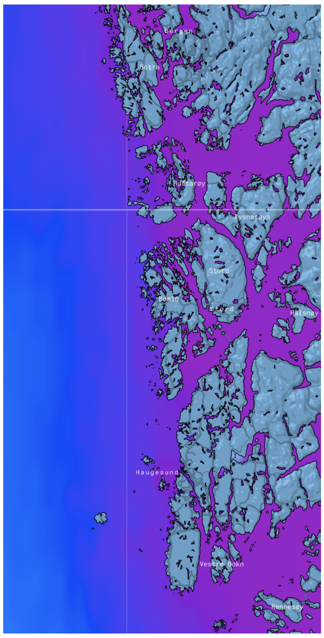


Figure 4.6: Waves of scenario visualized.

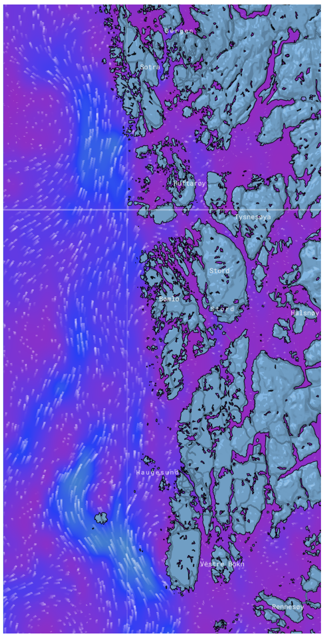


Figure 4.7: Sea currents of scenario visualized.

4.4 Overview

The list of all 16 generated routes and 19 executed simulations are summarized in Table 4.1. Each simulation is given a unique id for reference in comming results. For each simulation a route is defined by the method and objective used to generate it, whether it utilizes TSS or optical aids and whether weather is simulated.

7 of the routes were created and tested in 10 simulations to establish a base performance based on existing solutions. The base routes were manually created by a navigator trying to find an economical, safe or navigable route. An economical route is a route with low energy consumption and travel time. A safe route is a route which is robust to changes in environment, i.e. not performing significantly worse if conditions such as sea or weather should change. A navigable route is a route which is easy to follow, i.e. has low turning. A navigator created two sets of these routes, one where all requirements were satisfied, and one where all but optically navigable were satisfied. The purpose was to determine the performance difference in weather optical navigability is satisfied or not. Furthermore, the non-optically aids navigable routes were simulated both with and without weather. The motivation was to better understanding the affect of the performance metrics in a less constrained environment. Another method based on a pre-existing planner was also included as a base test. The planner implemented in ECDIS of unknown implementation, was used to generate an initial route. In order to satisfy TSS

compliance and optical aids navigation, the routes were modified by a navigator to incorporate all requirements. The base simulations 1 to 10 are summarized in Table 4.1.

The tailing 9 routes were generated automatically by the implemented route planner for different performance metrics, and by respecting TSS regulations and optical aids or not. The purpose of testing multiple performance metrics was to figure out which metrics are best for beating the base tests, and the purpose of testing the same metrics with and without respecting TSS and optical aids was to figure out the influence of TSS and optical aids on the performance. The automatically generated routes are summarized in Table 4.1 as simulations 11 to 19.

#	Route: Method	Route: Objective	Route: TSS	Route: Optical aids	Simulator: Weather
1	Manual	Economical	✓		
2	Manual	Safe	✓		
3	Manual	Navigable	✓		
4	Manual	Economical	✓		✓
5	Manual	Safe	✓		✓
6	Manual	Navigable	✓		✓
7	Manual	Economical	✓	✓	✓
8	Manual	Safe	✓	✓	✓
9	Manual	Navigable	✓	✓	✓
10	Existing auto/Manual	Inherit from auto	✓	✓	✓
11	Auto	Few turns			✓
12	Auto	Low distance			✓
13	Auto	Low turning			✓
14	Auto	Low travel time			✓
15	Auto	Avoid waves			✓
16	Auto	Few turns	✓	✓	✓
17	Auto	Low distance	✓	✓	✓
18	Auto	Low turning	✓	✓	✓
19	Auto	Low travel time	✓	✓	✓

Table 4.1: Summary and reference table for all executed simulations.

Each experiment was simulated utilizing track control on each of the routes under the specified simulator conditions. The complete list of all results are summarized in Table 4.2.

#	Feasible	Meet deadline	ECDIS compliant	TSS compliant	Optical aids navigable	Travel distance [nm]	Travel time [hh:mm:ss]	Energy consumption [kWh]	Rudder wear and tear [°]	Turning [°]
	Validation					**Benchmark**				
1	✓	✓	✓	✓		88.2	5:12:20	63141	1623	1426
2	✓	✓	✓	✓		89.6	5:16:30	63836	958	978
3	✓	✓	✓	✓		93.5	5:30:19	66515	537	586
4	✓	✓	✓	✓		88.2	5:14:30	63572	1638	1436
5	✓	✓	✓	✓		89.6	5:19:20	64414	913	981
6	✓	✓	✓	✓		93.6	5:14:10	63253	621	714
7	✓	✓	✓	✓	✓	88.26	5:14:40	63546	1525	1389
8	✓	✓	✓	✓	✓	89.23	5:18:00	64085	915	926
9	✓	✓	✓	✓	✓	93.76	5:14:50	63347	667	737
10	✓	✓	✓	✓	✓	89.68	5:20:50	61768	1045	1126
11	✓	✓	✓			97.21	5:28:30	66183	872	916
12	✓	✓	✓			83.15	4:53:00	59084	1200	1090
13	✓	✓	✓			96.36	5:26:20	64120	1007	943
14	✓	✓	✓			83.50	4:54:00	59312	1275	1171
15	✓	✓	✓			92.51	5:30:20	66688	1324	1237
16	✓	✓	✓	✓	✓	100.38	5:41:50	67158	857	899
17	✓	✓	✓	✓	✓	83.93	4:55:50	59676	1285	1162
18	✓	✓	✓	✓	✓	99.25	5:35:50	65961	870	811
19	✓	✓	✓	✓	✓	84.17	4:56:30	59833	1262	1205

Table 4.2: Performance of all specified route planning methods. Id references to Table 4.1

Based on the results, the following topics are discussed

- Main result: The performance of new and existing methods are compared in order to determine which approach is better.

- Weather influence. The performance of manually created routes are tested with and without weather.
- Performance metrics. Some performance metrics are tested on the implemented method to understand which metrics to use.

4.5 Main result

The main result is the comparison of the derived method for generating routes and existing, manual route planning methods. The existing methods are three manually created routes designed to be economical, safe and navigable, and a manually adjusted route that is automatically created by an existing route planner and post-processed by human. The relevant results are extracted from the overview and shown in Table 4.3. The simulated trajectories are shown in Figure 4.8. It can be observed from the trajectory that the implemented route planner finds a significantly different route than the manually created routes. An ECDIS validated that the routes were feasible, met the deadline and were ECDIS compliant. A visual inspection by an experienced navigator validated the routes to be TSS compliant and optical aids navigable. It can also be observed from Table 4.3 that the implemented method generally performs better on travel distance, time and energy consumption, and a little worse in rudder wear and tear and turning. This can be observed better in Figure 4.9.

#	Validation					Benchmark				
	Feasible	**Meet deadline**	**ECDIS compliant**	**TSS compliant**	**Optical aids navigable**	**Travel distance [nm]**	**Travel time [hh:mm:ss]**	**Energy consumption [kWh]**	**Rudder wear and tear [°]**	**Turning [°]**
7	✓	✓	✓	✓	✓	88.26	5:14:40	63546	1525	1389
8	✓	✓	✓	✓	✓	89.23	5:18:00	64085	915	926
9	✓	✓	✓	✓	✓	93.76	5:14:50	63347	667	737
10	✓	✓	✓	✓	✓	89.68	5:20:50	61768	1045	1126
17	✓	✓	✓	✓	✓	83.93	4:55:50	59676	1285	1162

Table 4.3: Performance of implemented method compared to manually created routes.

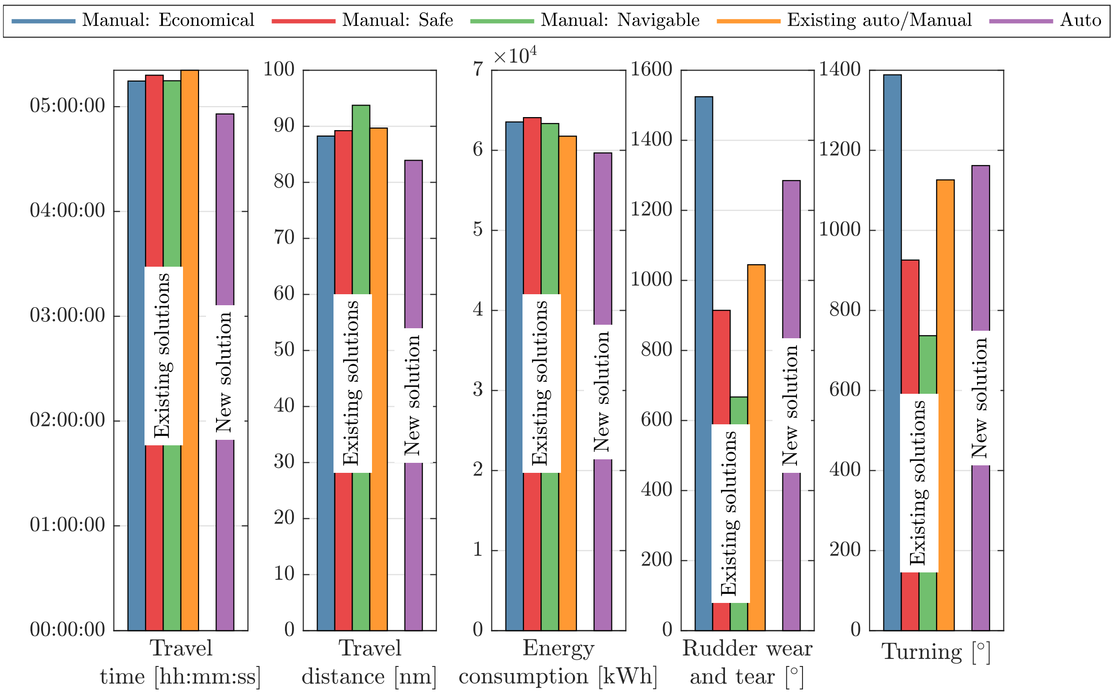


Figure 4.9: Performance of the automatically created route compared to other routes.

How much the automatically generated route is performing better than the base tests are shown in Table 4.4.

#	Travel distance [%]	Travel time [%]	Energy consumption [%]	Rudder wear and tear [%]	Turning [%]
	Benchmark improvements				
7	-4.91	-6.00	-6.09	-15.70	-16.33
8	-5.94	-6.97	-6.88	+40.51	+25.49
9	-10.48	-6.04	-5.79	+92.61	+57.67
10	-6.42	-7.79	-3.39	+23.00	+3.17

Table 4.4: Improvement of new solution to existing solutions

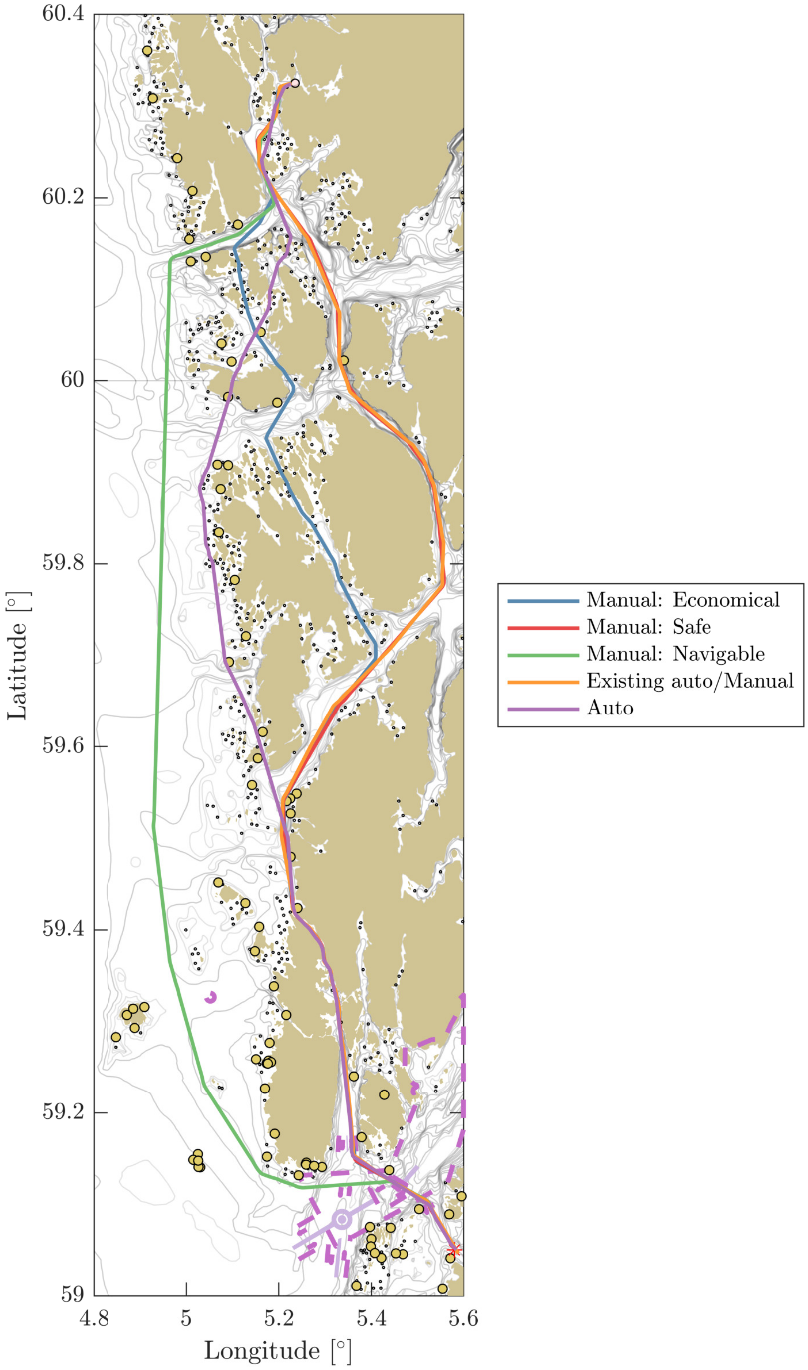


Figure 4.8: Simulated trajectories of TSS and optical compliant routes. ©Kartverket 2019.

The automatically generated route is outperforming the manually created routes on all travel distance, travel time and energey consumption. It also performed better than the most economical manual route in terms of rudder wear and tear and turning. This shows that the implemented method performed better than the manually created routes in terms of economy. However, in general the implemented method did not perform better than the manual routes in terms of rudder wear and tear and turning. This means that the implemented method outperformed the manually created routes if the increase in rudder wear and tear and turning is acceptable compared to manual routes. For the performane of manual routes shown in Table 4.3, it was shown that an decrease in travel time of around 4 %, resulted in an increase in rudder wear and tear and turning by around 50 %. Thus, the increase in rudder wear and tear is acceptable and the implemented method is shown to perform better than the manually created routes under the same constraints. Computation time is also an argument for the same result. Manually creating a route is expected to take some hours, while the automatic approach only takes a few minutes.

4.6 Weather influence

The economical, navigable and safe routes that are manually created without reguarding optical aids are simulated with and without weather influence. The simulated trajectory is shown in Figure 4.10. The safe route is created inshore away from the open sea such that it may be navigable even in tough weather. The navigable route is created offshore far away from obstacles using few turns. The navigable route is crossing narrow waters to save time and energy.

The validation and performance of the routes are shown in Table 4.5, and compared in Figure 4.11. It can be observed that weather plays a significant role in the performance of the routes, especially for the navigable route which is most vulnerable to weather. The navigable route is clearly the slowest and least energy-efficient without weather, but becomes the fastest and most energy efficient route with weather, even though the travelled distance is unchanged. In contrast, the economical route which is the fastest and most energy-efficient without weather is beat with weather. This means that the shortest route is not necessarily the fastest or most energy-efficient, and weather is an important factor which should be considered in automatic planning.

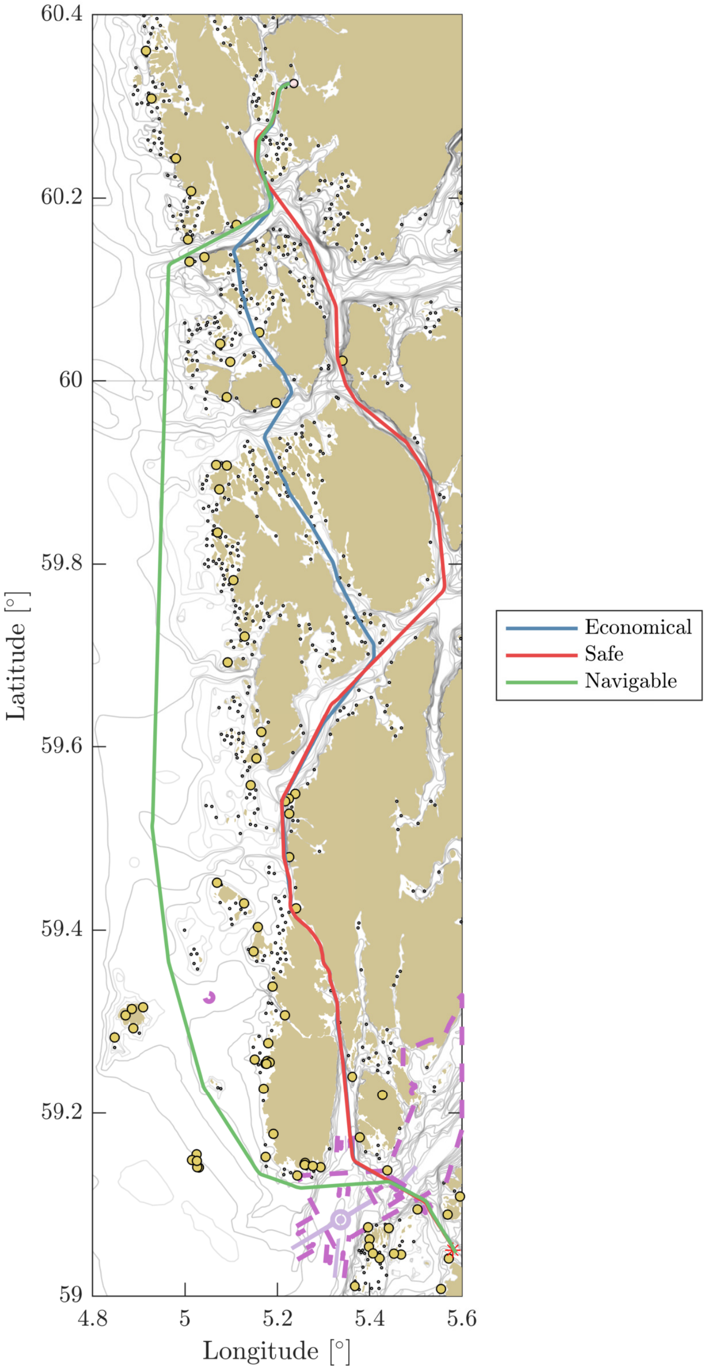


Figure 4.10: Simulated trajectory of manual routes that do not consider optical aids. ©Kartverket 2019.

#	Feasible	Meet deadline	ECDIS compliant	TSS compliant	Optical aids navigable	Travel distance [nm]	Travel time [hh:mm:ss]	Energy consumption [kWh]	Rudder wear and tear [°]	Turning [°]
	Validation					**Benchmark**				
1	✓	✓	✓	✓		88.2	5:12:20	63141	1623	1426
2	✓	✓	✓	✓		89.6	5:16:30	63836	958	978
3	✓	✓	✓	✓		93.5	5:30:19	66515	537	586
4	✓	✓	✓	✓		88.2	5:14:30	63572	1638	1436
5	✓	✓	✓	✓		89.6	5:19:20	64414	913	981
6	✓	✓	✓	✓		93.6	5:14:10	63253	621	714

Table 4.5: Performance of manual route planning that do not consider optical aids.

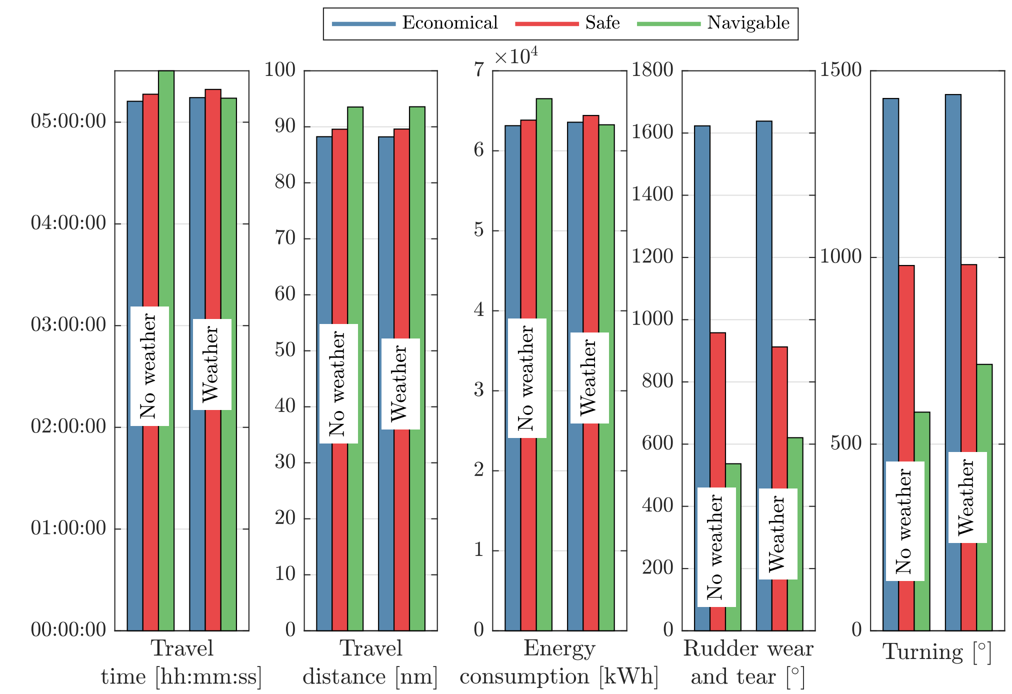


Figure 4.11: Performance of the manually created economical, safe and navigable routes with and without simulated weather.

Figure 4.12 shows the development of the performance metrics over time. It can be observed that energy consumption is highly dependent on time. This means that exess energy consumed in turns or in worse weather and sea conditions are neglectable. It can also be observed that turning and rudder wear and tear is correlated. This means that rudder wear and tear is highly dependent on how many turns are utizlied and less on sea or weather conditions.

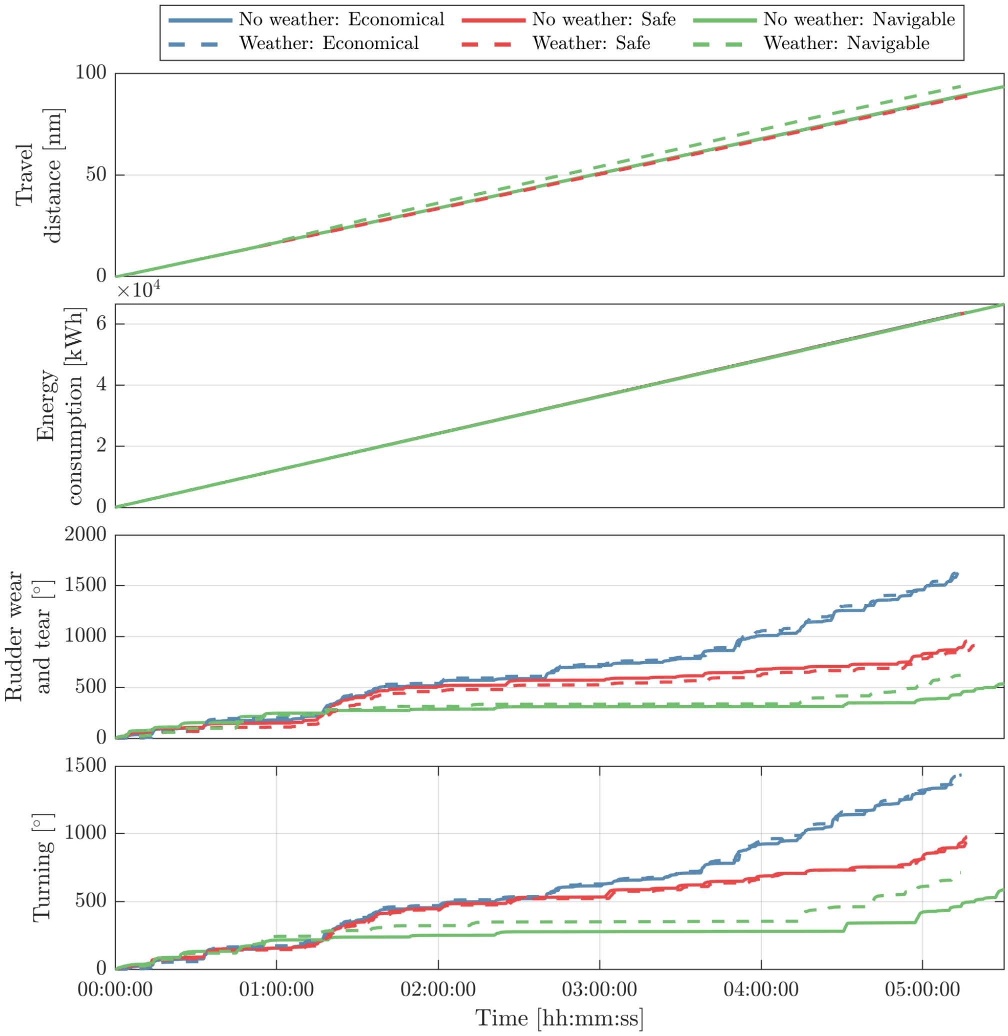


Figure 4.12: Historical performance of the manually created economical, safe and navigable routes with and without simulated weather.

4.7 Performance metrics

A collection of performance metrics are tested for the derived, automatic route planner. For this test TSS and optical aid navigation are not considered in order to limit constraints. The simulated trajectories for all tests are shown in Figure 4.13 and the results shown in Table 4.6. The performance results are visualized in Figure 4.14, and also the development of the performance metrics over time in Figure 4.15.

#	Validation					Benchmark				
	Feasible	Meet deadline	ECDIS compliant	TSS compliant	Optical aids navigable	Travel distance [nm]	Travel time [hh:mm:ss]	Energy consumption [kWh]	Rudder wear and tear [°]	Turning [°]
11	✓	✓	✓			97.21	5:28:30	66183	872	916
12	✓	✓	✓			83.15	4:53:00	59084	1200	1090
13	✓	✓	✓			96.36	5:26:20	64120	1007	943
14	✓	✓	✓			83.50	4:54:00	59312	1275	1171
15	✓	✓	✓			92.51	5:30:20	66688	1324	1237

Table 4.6: Results of automatically generated routes without considering TSS or optical aids.

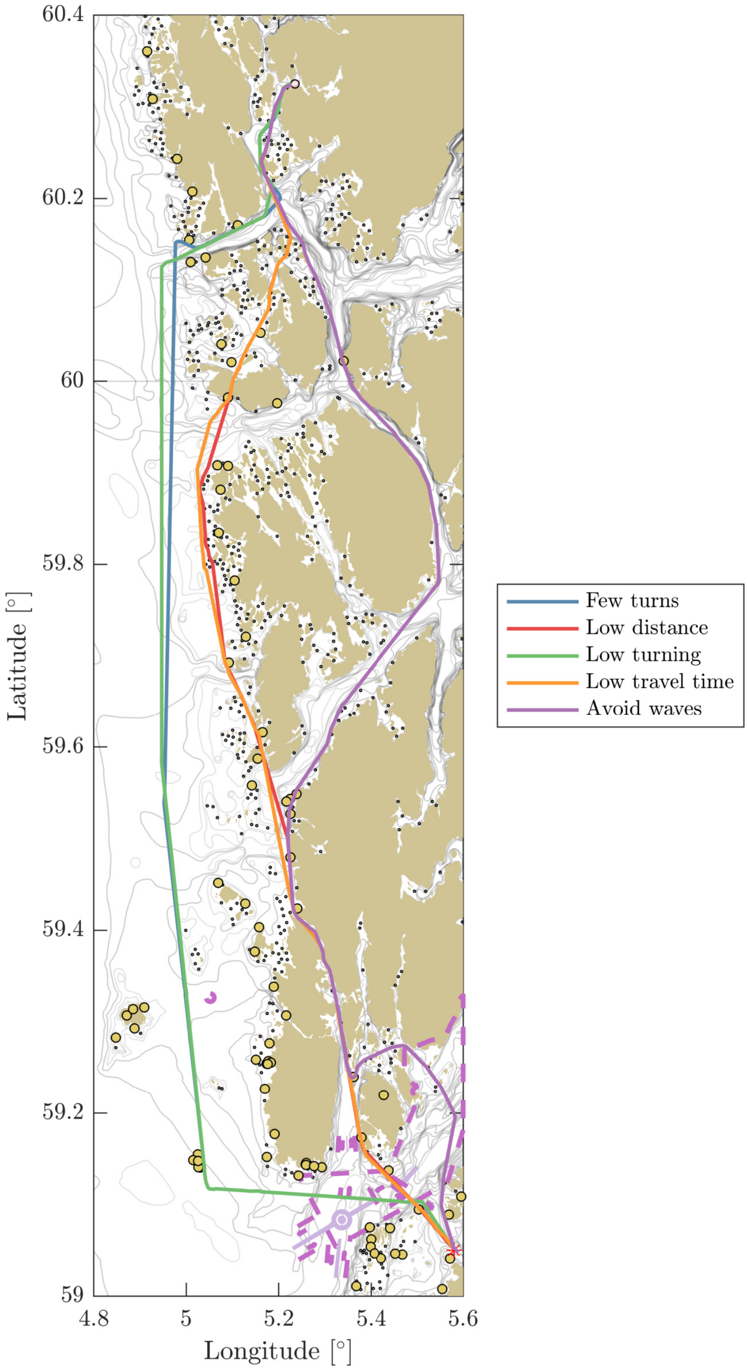


Figure 4.13: Simulated trajectory of automatically generated routes based on a variety of performance metrics. ©Kartverket 2019.

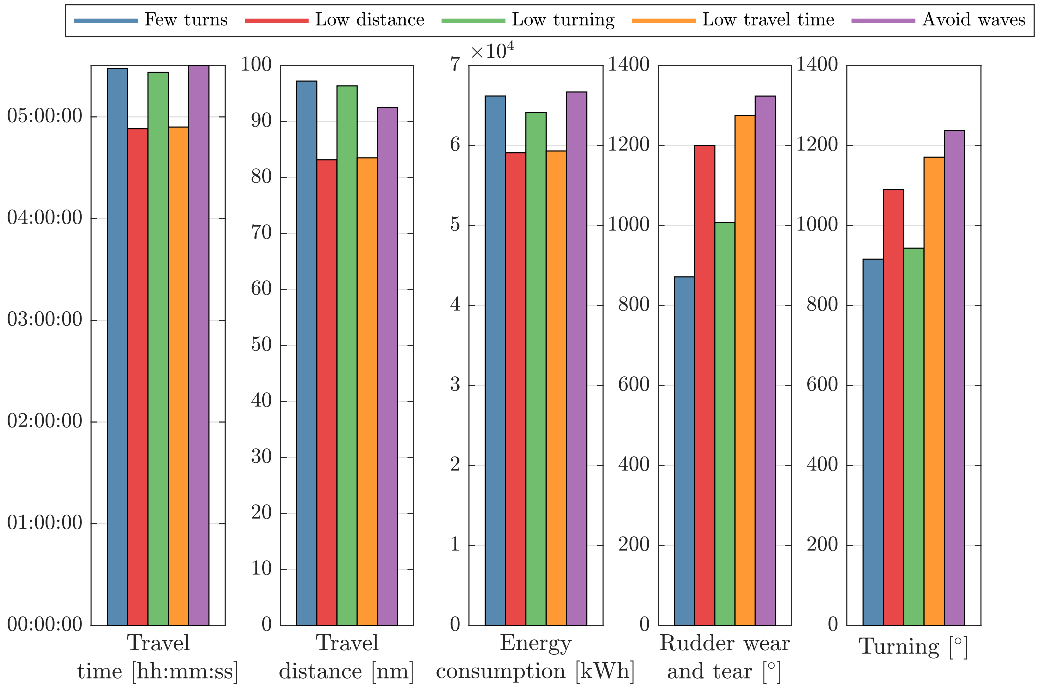


Figure 4.14: Performance of automatically generated routes based on a variety of performance metrics.

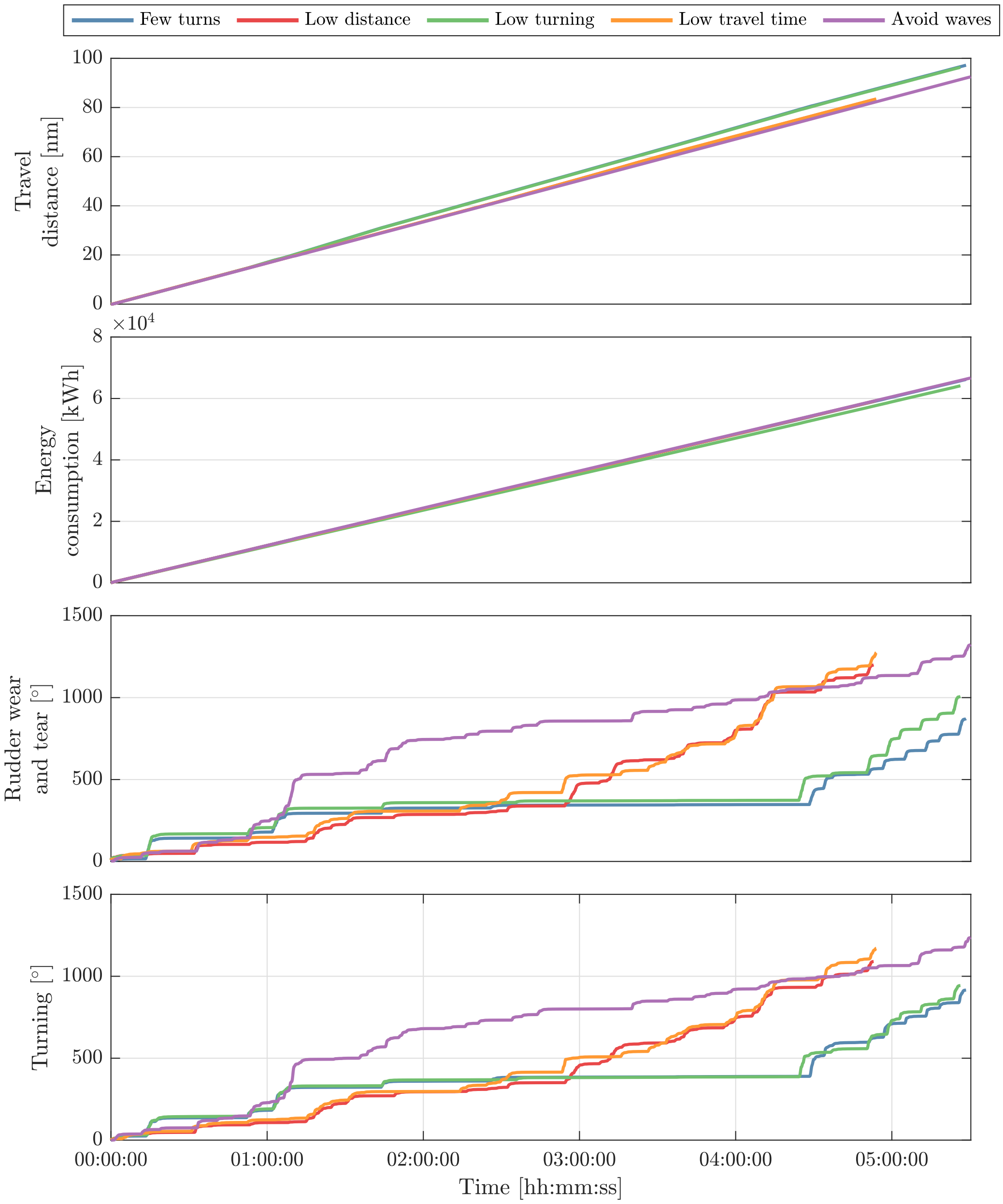


Figure 4.15: Historical performance of the automatically generated routes based on a variety of performance metrics.

It can be observed from Figure 4.14 that the performance metric of low travel time and low distance yielded similar results. They had the best travel times, distances and energy consumptions. Their simulated trajectories were also similar except for some parts of the route. Sometimes the low travel time route is taking advantage of the currents to attempt get to the goal faster by traveling further into the open ocean. Even though the shortest time route is taking this action, the route based on a metric of minimizing distance performed slightly better overall. This does not necessarily mean that a low distance metric is better. It can also be observed that the metric of low distance resulted in lower travel time than the route generated from a metric of low distance. Thus, the route based on low distance resultet in a faster route. This is most likely because of the deviations in the weather forecast and the simulated weather. As it was observed from weather influence, the shortest route is not the best route under deviations in weather and sea. That means that the route based on low travel time is more robust than the route based on low distance. Thus, a performance metric of low travel time as implemented yielded great performance in terms of travel time, travel distance and energy consumption. Argumentation for the same result can be collected from the historical performance shown in Figure 4.15. Energy consumption is highly dependent on time. The ship is hardly using any excess energy in the turns. However, reducing turning may still be a good idea in terms of maintaining speed to travel to the goal faster and thus more energy-efficient. Travel distance is also highly dependent on time. This means that the fastest route is probably also a route of low travel distance. This motivates using a performance metric based on travel time.

In terms of rudder wear and tear and turning, the routes based on few turns and low turning had best results. Few turns had its largest impact on rudder wear and tear, and low turning had naturally its largest impact on turning. However, few turns had best performance for both rudder wear and tear and turning, even outperforming low turning in terms of turning. Thus, a performance metric of few turns yielded great performance in terms of low rudder wear and tear and low turning. Argumentations for the same results can be gathered from Figure 4.15. Rudder wear and tear and turning are kept close to non-changing in periods of sailing straight, but for every turn all rudder wear and tear and turning are changing significantly. Also, reducing the number of turns, the travel time and consequently also energy consumption and travel distance, can be reduced even further.

Avoiding waves yielded the worst performance overall. This means that following the safest routes by avoiding all harsh waters is not the the best performing solution. This might result in more robust solutions, but this aspect is not considered in this thesis.

Chapter 5

Conclusion and further work

5.1 Conclusion

Throughout this thesis, an automatic method for finding a well-performing route which is usable in practice was designed and implemented. The method was validated to satisfy the requirements of being feasible, meeting the deadline, being ECDIS compliant, optical aids navigable and TSS compliant as defined in the introduction. The method was then benchmarked and compared to three manually created routes and one manually corrected route which was originally created by an automatic planner implemented in Kongsberg ECDIS.

It was validated that the automatic planner generated routes according to the minimum requirements. It could also be concluded that the implemented method performed better than the manually created routes. The generated route and manually created routes were validated according to the requirements. The implemented method resulted in around 5 % lower travel distance, 6 % lower travel time and 6 % lower energy consumption, 15 % lower rudder wear and tear and 16 % lower turning than the shortest and fastest manual route. It also took a couple of minutes to generate automatically, instead of hours to create manually.

It was also found that weather and sea state has a significant influence on the performance of the route, and so there is an advantage to consider this when applying automatic planning. With completely calm weather and sea, the shortest route is probably the best route in terms of the considered metrics. With normal weather and sea, the shortest route is probably no longer the best route.

It was identified that fast and energy-efficent routes are similar, and that routes of low actuator wear and tear and little turning are similar. This means for example that a fast route is also energy-efficient. It was also identified a tradeoff between a fast and energy-efficient route, and a route of little turning and low actuator wear and tear. This means that in order to generate a fast and energy-efficent route, higher turning and higher actuator wear and tear must be tolerated.

Several performance metrics were tested and benchmarked. It was found that a good performance metric for finding fast and energy-efficent routes should focus on travel time and consider weather and sea state. A good performance metric for finding routes of little turning and low actuator wear and tear was found to focus on number of turns.

5.2 Further work

Naturally, there may exist better methods for generating even better routes based on the same scenario, requirements and desirables. However, the real value of optimizing further to find better routes is limited. The reason is that the route is based on theoretical values which may not yield the same results in practice. The weather and sea state are just forecasts and in practice they might be very different and yield very different performances. The ship may also keep another speed than anticipated during planning. Also, the route is just a plan and in practice the ship will deviate from the route in order to comply with traffic and other dynamics in the environment. This means that there might not be a strong enough motivation to optimize planning further based on the same problem.

An issue with the validation process is that validation of TSS compliance and optical aids navigable is done manually by visual inspection. This is a process which has motivation to be automated. A motivation is that a validation routine could be used by ECDIS systems in order to aid manual planning which is not expected to be replaced by automatic planning anytime soon. This problem was not solved in this thesis. The implemented method is managing to generate TSS compliant and optical aids navigable routes because it simplifies the requirements. This means that it is not possible to use the theory of the implemented method directly to validate an arbitrary route. Therefore, automatic TSS and optical aids validation is suggested as further work.

Bibliography

Aurenhammer, Franz (1991). "Voronoi diagrams-a survey of a fundamental geometric data structure". eng. In: *ACM Computing Surveys (CSUR)* 23.3, pp. 345–405. ISSN: 1557-7341.

Bhattacharjee, Shilavadra (2019). *What is Electronic Chart Display and Information System (ECDIS)?* URL: https://www.marineinsight.com/marine-navigation/what-is-electronic-chart-display-and-information-system-ecdis/.

Bitar, Glenn (2017). "Towards the Development of Autonomous Ferries". MA thesis. Trondheim, Norway: Norwegian University of Science and Technology.

Fossen, Thor I. (2011). *Handbook of marine craft hydrodynamics and motion control*. eng. Chichester, England: John Wiley & Sons, Ltd. ISBN: 1-283-40556-3.

Junghwan Lee, Heechan Shin and Sung-eui Yoon (2017). "Data-driven kinodynamic RRT". In: *2017 18th International Conference on Advanced Robotics (ICAR)*, pp. 91–98. ISSN: 978-1-5386-3157-7.

Karaman, Sertac and Emilio Frazzoli (2011). "Sampling-based algorithms for optimal motion planning". eng. In: *The International Journal of Robotics Research* 30.7, pp. 846–894. ISSN: 0278-3649.

Kartverket (2019). *S-57 electronic nautical charts between Stavanger and Bergen (License number: 19/801)*. URL: kartverket.no.

Kuwata, Y. et al. (2009). "Real-Time Motion Planning With Applications to Autonomous Urban Driving". eng. In: *Control Systems Technology, IEEE Transactions on* 17.5, pp. 1105–1118. ISSN: 1063-6536.

Latombe, Jean-Claude (1991). *Robot motion planning*. eng. New York.

LaValle, Steven M. (2006). *Planning Algorithms*. Ed. by University of Illinois. Cambridge University Press. ISBN: 9780521862059.

Pedersen, Benjamin Pjedsted and Jan Larsen (2009). "Modeling of ship propulsion performance". In: *World Maritime Technology Conference WMTC2009, Jan. 21*. Vol. 24.

Schack, Christian (2019). *Voyage Planning*. SeaPlanner. URL: http://messe.no/ExhibitorDocuments/191291/6888/3155-6-en%20Seaplanner_web.pdf?Exhibiti.

Webb, Dustin J. and Jur van Den Berg (2013). "Kinodynamic RRT*: Asymptotically optimal motion planning for robots with linear dynamics". eng. In: IEEE, pp. 5054–5061. ISBN: 9781467356411.

NTNU
Norwegian University of
Science and Technology