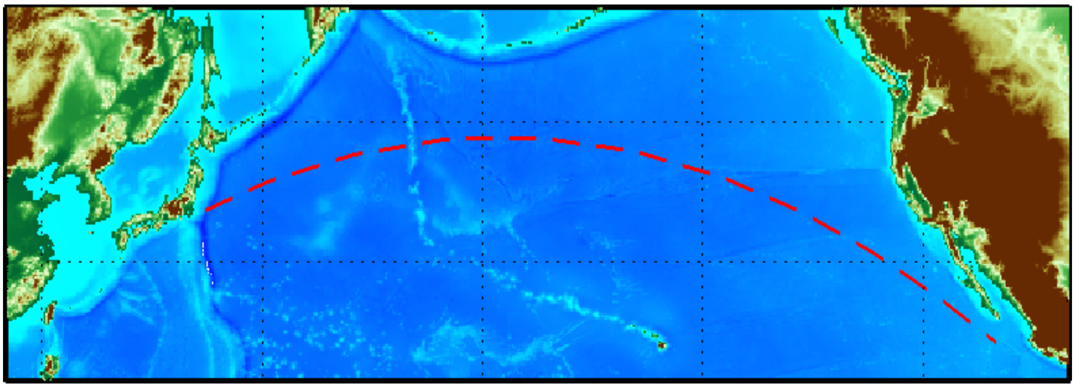

DIRECT Weather Routing

Master's Thesis within the International Master's Programme Naval Architecture and Ocean Engineering & the Nordic Master in Maritime Engineering

Erik Larsson

Martin Hjorth Simonsen

Department of Shipping and Marine Technology
Division of Marine Design, Research Group Marine Structures
CHALMERS UNIVERSITY OF TECHNOLOGY
Gothenburg, Sweden 2014

Master's Thesis X-14/300

MASTER'S THESIS IN THE INTERNATIONAL MASTER'S PROGRAMME IN NAVAL ARCHITECTURE AND OCEAN ENGINEERING & THE NORDIC MASTER IN MARITIME ENGINEERING

DIRECT Weather Routing

Erik Larsson
&
Martin Hjorth Simonsen

Department of Shipping and Marine Technology
Division of Marine Design
Research Group Marine Structures
CHALMERS UNIVERSITY OF TECHNOLOGY
Gothenburg, Sweden 2014

.

DIRECT Weather Routing

Erik Larsson

Martin Hjorth Simonsen

Master's Thesis, X-14/300
Department of Shipping and Marine Technology
Division of Marine Design
Research Group Marine Structures
Chalmers University of Technology
SE-412 96 Gothenburg
Sweden
Telephone: +46 (0)31-772 1000

Cover:
Weather route optimization North Pacific

Reproservice/ Department of Shipping and Marine Technology
Gothenburg, Sweden 2014

DIRECT Weather Routing
Master's Thesis in the International Master's Programme in Naval Architecture and Ocean Engineering & the Nordic Master in Maritime Engineering
Erik Larsson and Martin Hjorth Simonsen
Department of Shipping and Marine Technology
Division of Marine Design, Research Group Marine Structures
Chalmers University of Technology

Abstract

A significant increase in fuel prices over the past decades and focus on global warming have lead to a large interest in fuel efficiency within shipping. Weather routing represents one increasingly recognised method for reducing fuel consumption, hereby lowering emissions. Determining the speeds and route for a ship requires a balancing of multiple objectives including cost, time of arrival, and ship and cargo safety. This calls for advanced multi-objective optimisation algorithms to be implemented.

This thesis presents the theory behind weather routing and a state-of-the-art study on available software and research. It also aims at providing the reader with a basic understanding of what elements make up a weather routing tool and what is expected of a state-of-the-art routing tool.

In this thesis the DIRECT algorithm is used in the construction of a novel weather routing tool. The output from the tool being routes with optimised speed profiles. The constructed routing tool is demonstrated on several simple test cases determining the capabilities of routing around land, avoiding storms by speeding up or slowing down and utilising weather to the advantage of a minimised fuel consumption.

Key words: DIRECT algorithm, Emission reduction, Fuel saving, Ship weather routing

DIRECT Weather Routing
Examensarbete inom:
Naval Architecture and Ocean Engineering & the Nordic Master in Maritime Engineering
Erik Larsson och Martin Hjorth Simonsen
Institutionen för sjöfart och marin teknik
Avdelningen för Marine Design
Forskargruppen Marine Structures
Chalmers tekniska högskola

Sammanfattning

En betydande ökning av bränslepriserna under de senaste årtiondena och ökat fokus på den globala uppvärmningen har lett till ett stort intresse för bränslebesparingar inom sjöfarten. Ruttplanering baserat på väderprognoser representerar en alltmer erkänd metod för att minska bränsleförbrukningen och därmed också utsläppen. När en ruttplanering görs är det en avvägning av många parametrar så som kostnad, ankomsttid, fartygets och lastens säkerhet. Detta ställer krav på att använda tillräckigt avancerade optimeringsalgoritmer.

Denna uppsats presenterar teorin bakom ruttplanering baserat på väderprognoser och en state-of-the-art-studie om tillgängliga program och forskning på området. Den syftar också till att ge läsaren en grundläggande förståelse för vilka parametrar som ska tas hänsyn till i ett ruttplaneringsverktyg och vad som förväntas av ett s.k. state-of-the-art ruttplaneringsverktyg.

I denna uppsats används DIRECT algoritmen för konstruktion av ett nytt ruttplaneringsverktyg. Resultatet från optimeringen är en rutt med optimerade hastighetsprofiler. Ruttplaneringsverktyget demonstreras med hjälp av flertalet enkla testfall som visar på dess förmåga att undvika land, undvika stormar genom att öka hastigheten eller sakta ner och utnyttja vädret till fartygets fördel för en minimerad bränsleförbrukning.

Nyckelord: Bränslebesparing, DIRECT algoritm, Ruttplanering, Utsläppsreduktion

Preface

This thesis is a part of the requirements for the master's degree in Naval Architecture and Ocean Engineering at Chalmers University of Technology, Gothenburg. The work has been carried out at Maersk Maritime Technology in Copenhagen from January to mid April of 2014. After Copenhagen the work has been continued at the Division of Marine Design, Department of Shipping and Marine Technology, Chalmers University of Technology until June of 2014.

The support from our supervisor Associate Professor Wengang Mao, acting head of research group Marine Structures at the Division of Marine Design, Department of Shipping and Marine Technology, Chalmers University of Technology, has been invaluable. We would also like to thank our supervisor from Alto University, Professor, D.Sc. (Tech), Jerzy Matusiak.

The authors would also like to thank our industry supervisor Kim Henriksen, Lead Naval Architect, Hydrodynamics at Maersk Maritime Technology(MMT) for all the help and support he has given us.

We would like to express our gratitude to the employees at MMT for all the feedback and encouragement. Special thanks goes to Dr. Jesper Dietz, Lead Vessel Performance Analyst MMT, for all his support during the project.

The DIRECT algorithm has shown potential for routing and we are thankful to Dr. Daniel E. Finkel for developing and making the algorithm available.

The Authors, Gothenburg 2014-06-05

List of Abbreviations

2DDP	Two Dimensional Dynamic Programming
3DDP	Three Dimensional Dynamic Programming
AWT	Applied Weather Technology
COG	Centre of Gravity
DIRECT	DIviding RECTangles
DMI	Danish Meteorological Institute
DNV	Det Norske Veritas
ECA	Emission Controlled Areas
ECDIS	Electronic Chart Display and Information System
ECMWF	European Centre for Medium-Range Weather Forecasts
ETA	Estimated Time of Arrival
GA	Genetic Algorithm
GFS	Global Forecast System
GL	Germanischer Lloyd
HFO	Heavy Fuel Oil
IMO	International Maritime Organization
Ind.	Indirectly
ITTC	International Towing Tank Committee
LNG	Liquefied Natural Gas
MARPOL	Marine Pollution
MC	Monte Carlo
MGO	Marine Gas Oil
MMT	Maersk Maritime Technology
MSC	Maritime Safety Committee
NAS	Neighborhood Assessment Strategy
NOx	Nitrogen Oxides
PM	Particulate Matter
RAO	Response Amplitude Operator
RDO	Response Drift Operator
RPM	Revolutions Per Minute
RTOFS	Real-Time Ocean Forecast System

SECA	Sulphur Emission Controlled Areas
SMHI	Swedish Meteorological and Hydrological Institute
SOx	Sulphur Oxides
SPOS	Ship Performance Optimization System
TBI	To Be Included
TEU	Twenty-foot Equivalent Unit
UKC	Under Keel Clearance
UNCTAD	United Nations Conference on Trade And Development
VVOS	Vessel and Voyage Optimisation Solution

Notations

Greek Letters

α	Lipschitz constant, K in graphs
α_{corr}	Correction factor for block coefficient
δ	One third side length in hyper-rectangle
ϵ	Small positive number
η_H	Hull efficiency
η_O	Open water efficiency
η_R	Relative rotative efficiency
η_S	Transmission efficiency, the efficiency of shaft line and gearbox
η_T	Total efficiency
$\overline{\Omega}$	Transformed solution space, initial hyper-cube
μ	Penalty parameter
μ_{beam}	Weather direction reduction factor for beam sea
μ_{bow}	Weather direction reduction factor for bow seas
$\mu_{following}$	Weather direction reduction factor for following sea
μ_{red}	Weather direction reduction factor
ϕ	Wave encounter angle
∇	Volume of displacement
ρ_a	Mass density of air

Roman Letters

A_T	Reference area for wind resistance
$\overline{BM}$	Length between centre of buoyancy and metacenter
BN	Beaufort number
c	Centre of hyperrectangle
C_{AA}	Resistance correction to ITTC-57 for container ships
C_B	Block coefficient
C_{air}	Drag coefficient
D	Distance in nautical miles
d_j	Distance from centre of hyper-rectangle to corners

Roman Letters - Continued

$f(c)$	Object function
$fuel(p_i, pi + 1)$	Function for calculating fuel between two points
Fn	Froude number
g	Acceleration due to gravity at the surface of the earth
$g(x)$	Constraint function
$\overline{GM}$	Metacentric height
I	Moment of inertia about the longitudinal axis
k	Form factor describing the viscous resistance of the hull form
k	Radius of gyration about the longitudinal axis through the COG
$\hat{K}$	Lipschitz constant, modified to all values
L	Ship length
M	Solution space
n	Number of intermediate points
N	Number of dimensions
P_D	Delivered power
P_E	Effective power
R_A	Model ship correlation resistance
R_{air}	Wind resistance in the direction of the flow velocity
R_F	Frictional resistance calculated through the ITTC-57 equation
R_{APP}	Appendage resistance
R_W	Wave making and wave breaking resistance
R_B	Pressure resistance due to a bulbous bow near the water surface
R_{total}	Total resistance
R_{TR}	Pressure resistance due to immersed transom
T	Roll period
$totalFuel$	Function for calculating total fuel consumption for a trip
V	Ship speed
V_{limit}	Limiting speed for surf-riding and boraching
V_{loss}	Percentage speed loss
V_S	Ship speed
w_i	Length of dimension i

Contents

1 Introduction

One of the major concerns in the shipping industry is the minimisation of fuel consumption through optimum operation. The driving forces are increased fuel prices, new legislation and increased environmental concern. Optimum operation should also be achieved with regards to cargo- and personnel safety and with focus on expected time of arrival. This master thesis project was set up through Chalmers University of Technology in cooperation with Maersk Maritime Technology, MMT, part of the A.P. Møller group, to understand and investigate weather routing systems for energy efficient shipping.

Ship weather routing is the development of an optimum track and speed for ocean voyages based on forecasted weather, sea conditions, and the individual characteristics of a ship for a particular transit (Bowditch, 2002). Both fuel consumption and expected time of arrival are not only coupled with the optimum track but are also directly connected with speed/power profiles (Notteboom and Carriou, 2009). Therefore most ocean-crossing ships are instrumented with a weather routing system for achieving the least fuel consuming route while arriving on time. The benefits of using these routing systems are well documented in e.g. (Chen et al., 1998).

Concerning ship and cargo safety the use of weather routing is of crucial importance. According to statistical estimations container ships lose between 2,000 and 10,000 containers at sea each year with a total value of approximately $370 million (Podsada, 2001). Further, drifting containers may cause hazards to the shipping environment. Container losses are mainly caused by large ship motions and accelerations due to encountered weather. Proper use of weather routing software and ship monitoring might have prevented accidents such as APL China (Ginsberg, 1999) and the M/V Derbyshire (Derbyshire, 1980).

Increasing available computer power on board ships and onshore leads to a fast development in complexity in both algorithms used and in system modelling. This results in an expanding and fast changing market. A mapping of the state of the market is therefore needed in order to determine what is required of a weather routing system, what is to be required in the near future, and where a greater research effort is needed.

The computation time is a limiting factor as the most current weather information must be used in order to produce routes with proper validity. Thus exceeding the time interval for weather forecasts will make the optimised route obsolete. The use of algorithms and methods capable of producing optimal results in a minimal amount of time is therefore essential.

Chalmers University of Technology has the ambition of contributing to a sustainable future. Eight areas of advance have been defined to achieve this goal. One of these being transport. The UNCTAD Review of Maritime Transport (UNCTAD, 2013) shows a drastic growth in the demand for maritime transport over the past 40 years thus adding to the need for safer and more energy efficient shipping, which this thesis aims at addressing.

1.1 Objective

The main objective of this thesis is the development of a conceptual routing tool with respect to fuel consumption, safety, and ETA. The routing tool should have the capability of considering ship specific operational performance. This including involuntary reductions and increases to resistance from the influence of weather.

The project may be divided into two main aspects to accomplish the main objective:

1. The development of a tool for weather routing.
    - An overview of what makes up a weather routing tool.
    - A state of the art study, mapping market leaders in routing, and analysis of the features of their respective routing tools. Research and development within weather routing should also be included.
    - The construction of a flowchart for the routing tool. Listing of all the inputs and requirements to be considered for the route optimisation, such as weather information, operational profiles of the ship under different weather conditions, engine power and speed relation, and expected time of arrival.
    - A literature survey on mathematical algorithms used in weather routing and identification of the most suitable for the current project. In the maritime industry and research community, the most often used algorithms for ship weather routing are: Dynamic Programming (Chen, 1978; Avgouleas, 2008), Dijkstra Algorithm (Hagiwara, 1982), and the genetic-evolutionary algorithm.
    - The construction of interfaces for all the inputs/requirements and implementation of the chosen algorithm to design/plan the optimal ship route with respect to minimum fuel consumption, ETA, as well as safety. In general it is route optimisation about constant power.

    - The construction of a working program capable of producing an output to be used in weather routing.

2. The demonstration of how the tool performs when routing in different environments.
    - The selection of a case study ship and environments for testing.
    - The demonstration of the developed routing tool to plan the optimum route and compute the corresponding fuel consumption.

1.2 Limitations

The task of developing a new weather routing software bringing in new features and methods is of a magnitude requiring limitations to the scope of the project due to the limited time available. Focus will be on the optimisation algorithm and the implementation of said.

- As fuel consumption models are provided by MMT, these will only be described in basic theory as would be required for the construction of a generic model.
- Ship and cargo safety will only be described briefly in theory and only very simply implemented in the program.
- The routing should be a conceptual program showing the capabilities of the algorithm and possibilities to add on important features. Focus should not be on making the tool ready for commercial use.
- As focus is on a conceptual program, benchmarking against other products will not be carried out and only simple cases with demonstrative focus are to be tested.

1.3 Disposition

This report is divided into six main chapters:

- Chapter 1 introduces the reader to the goals and methodology of the project.
- Chapter 2 describes the theory and requirements of a weather routing tool.
- In Chapter 3 is presented a state of the art study mapping out the market.
- Algorithm selection and a description of the selected algorithm is presented in Chapter 4.
- The produced routing tool is described in Chapter 5.

- The created weather routing tool is demonstrated in Chapter 6.
- Finally conclusions and recommendations for future work are found in Chapter 7.

2 Concept of Weather Routing

Weather routing is the process of producing the most favourable route and possibly speed/power profile for a voyage taking weather into account. The primary use being for ocean transits, but it is also used for coastal navigation. The main objective of weather routing is minimising cost taking into account arrival time and safety. When minimising the cost of a crossing there are many costs to take into account e.g. fuel, crew and capital costs. In this project focus has been on fuel consumption only.

This chapter is intended to give the reader an overview of weather routing. Focus is on the elements needed for routing and on the challenges faced when trying to optimise ship routes. To further the understanding of the structure of a weather routing tool Figure 2.1 is intended to provide a general overview.

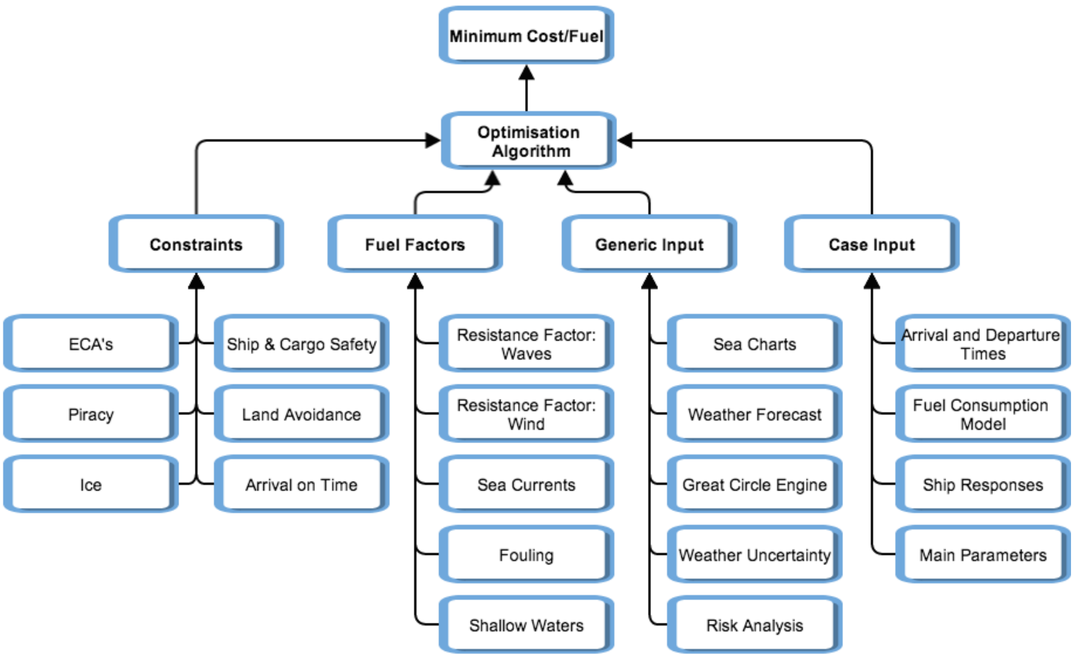

Figure 2.1: The structure of elements making up weather routing

Referring to Figure 2.1 the main objective is a route that requires minimum cost/fuel. To achieve this an, optimisation algorithm is implemented taking into account several factors. The optimisation may be said to be based on four major areas; constraints, fuel factors, generic input, and case specific input. Where constraints include the limitations to a produced route, fuel factors are the environmental factors affecting the fuel consumption, generic input are the input which are not ship specific, and case input are the ship and voyage specific input.

For a fixed arrival time with little margin, the most important cost to be estimated is the fuel consumption, which is most often the parameter optimised. To enable this estimation a ship specific fuel consumption model is needed. Also needed are estimations of internal systems and external forces affecting the fuel consumption. This model will be discussed in further detail in the the following section.

Addressing the more generic elements regarding the navigation of ships the use of nautical charts should be mentioned. These are most often incorporated in the ECDIS (Electronic Chart Display and Information System) system. Having printed charts on board was until recently a flag state requirement. ECDIS systems with back up systems are now certified for on board use without the requirement of printed charts. The ECDIS charts most commonly used are so called vector charts. With the data stored in vector data sets the same chart may be presented in a wide range of scales. The information from the vector charts can be used and interpreted in routing software. Most importantly this means water depth and land can be used as input in the route optimisation in order to avoid land. A produced best route may be presented either in the routing system or in the ECDIS system allowing easier use for captains.

Great circle navigation is the practice of traveling the shortest distance between two points on a sphere. In plane geometry a straight line represents the shortest distance between two points. On a sphere however the great circle is the shortest way between two points. Earth is not perfectly spherical but the approximation is satisfactory for the purpose of routing. When earth is to be represented as a plane map a projection must be used to represent a great circle route. The standard nautical projection used is the Mercator projection. The Mercator projection is a cylindrical projection with the ability to represent lines of constant course, known as rhumb lines or loxodromes, as straight lines with 90 degree angles to the meridians. Figure 2.2 shows the great circle and the rhumb line between two points plotted on mercator and Figure 2.3 shows the two lines plotted on a sphere. In Figure 2.3 the reasoning behind the great circle navigation may be seen.

In order to achieve minimum fuel consumption it is essential that routing is done using great circle navigation. The great circle distance between two points on the globe is calculated according to equation (2.1) below.

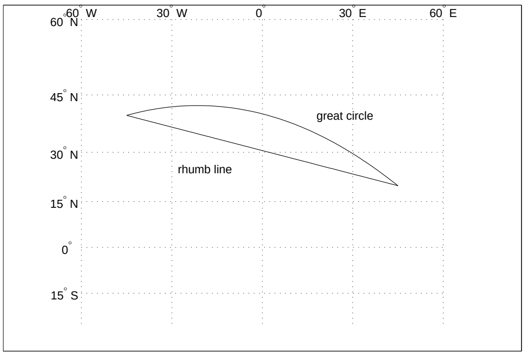

Figure 2.2: Great circle and rhumb line on mercator

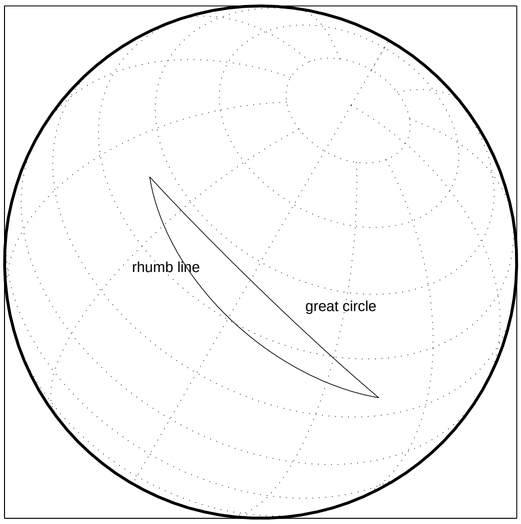

Figure 2.3: Great circle and rhumb line on sphere

$$D = \arccos(\sin(lat_A)\cdot\sin(lat_B)+\cos(lat_A)\cdot\cos(lat_B)\cdot\cos(lon_B-lon_A))3440\text{nM} \quad (2.1)$$

For especially container ships arrival and departure times are most commonly fixed at specific hours with little margin. (Mao et al., 2012) has shown that changing the departure time can reduce fatigue damage significantly. However, in most cases this is not an option. As weather routing is highly applicable to container vessels routing is done taking into account this small margin to arrival and departure times.

2.1 Fuel Consumption Model

A fuel consumption model is the relation between achieved speed and fuel consumption. Several parameters influence this relation. Figure 2.4 illustrates some of these factors.

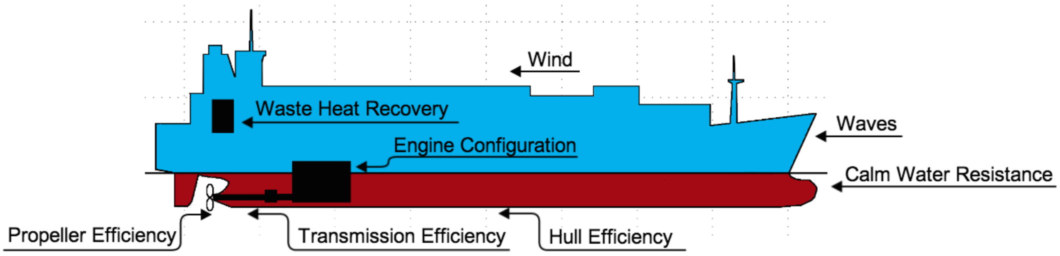

Figure 2.4: Ship fuel factors and components in the fuel consumption model

The construction of a fuel consumption model may be done using generic theory, using complex theories, through measurements and modelling, or as a combination. The level of possible complexity may be determined by the amount of available information on the ship. To achieve a good fuel consumption model the estimations should though be tuned based on measured data.

The fuel consumption is related to the required produced power and the specific fuel oil consumption of the engines. Models relating power output and fuel consumption are most often provided by the engine manufacturer based on extensive testing, but may need adjusting based on full scale on board tests.

2.1.1 Effective Power

The requirement for produced power, or effective power, may be determined based on calculated calm water resistance and added resistances, when no service allowance is added by the following equation. This does not take into account the hotel load, and is only related to resistance:

$$P_E = R_{total} \cdot V \tag{2.2}$$

From effective power the delivered power may then be determined as:

$$P_D = P_E \cdot \eta_T \tag{2.3}$$

where η_T is the total efficiency:

$$\eta_T = \eta_H \cdot \eta_O \cdot \eta_R \cdot \eta_S \tag{2.4}$$

where
η_H is the hull efficiency
η_O is the open water efficiency
η_R is the relative rotative efficiency
η_S is the transmission efficiency, the efficiency of the shaft line and gearbox

The efficiencies and an updated method for their estimations are proposed in (Kristensen and Lützen, 2012), based on earlier works, making it possible to incorporate these into a generic model requiring little knowledge on ship specific data if extended testing has not been carried out.

2.1.2 Ship Resistance

To enable the calculation of the effective power, the resistance must be estimated. The resistance of ships is composed of a large number of elements. Resistance originates from both calm water resistance and added resistance from waves as well as wind. The calm water resistance of a typical container vessel, according to (Larsson and Raven, 2010) , is 20% wave making and wave breaking. The remaining 80% are related to viscous resistance. In seaway the resistance may be larger due to effects of waves and wind. The added resistance is of the magnitude of 15-30%,(Seo et al., 2013).

Calm Water Resistance

Several theoretical methods for calculating calm water resistance exist including ITTC-57 and Holtrop and Mennen. With limited available ship specific information the ITTC-57 is to be preferred possibly through the updated version proposed in (Kristensen and Lützen, 2012) taking into account the effects of bulbous bows, correction for hull form and position of the longitudinal centre of buoyancy. With more available ship specific information Holtrop and Mennen's method is to be preferred, (Holtrop and Mennen, 1978):

$$R_{total} = R_F(1+k) + R_{APP} + R_W + R_B + R_{TR} + R_A \tag{2.5}$$

where:
R_F is the frictional resistance calculated through the ITTC-57 equation.
$1+k$ is the form factor describing the viscous resistance of the hull form.
R_{APP} is appendage resistance.
R_W is wave making and wave breaking resistance.
R_B is pressure resistance due to having a bulbous bow near the water surface.

R_{TR} is pressure resistance due to immersed transom at stern.
R_A is model ship correlation resistance.

From the equation the calm water resistance may be determined and modelled for a range of speeds and drafts.

Resistance from Waves

The added resistance caused by encountered waves may be modelled through extensive model testing or strip theory to determine heave and pitch and from sea spectrums obtain added resistance. A third option does not require knowledge of the hull lines and thus makes for a more generic method of obtaining the added resistance, though accuracy may be limited compared to the mentioned methods due to the difficulty of modelling the nature of waves. Such a method is described in (Kwon, 2008), and is summed up in the following sections. This method does though take into account both waves and wind simultaneously.

The percentage of speed loss is given by, (Kwon, 2008):

$$V_{loss} = \alpha_{corr} \cdot \mu_{red} \cdot \frac{\Delta V}{V} 100\% \tag{2.6}$$

where
ΔV is speed loss due to head weather.
V is design service speed.
$\frac{\Delta V}{V}$ is the speed loss in head weather given by Equations (2.7) through (2.9).
α_{corr} is the correction factor for block coefficient, C_B and Froude number, Fn, seen in Table 2.1.
μ_{red} is the weather direction reduction factor given by Equations (2.10) through (2.12).

The speed loss in head weather condition is given by:

$$\frac{\Delta V}{V} 100\% = 0.5BN + \frac{BN^{6.5}}{2.7\nabla^{\frac{2}{3}}} \tag{2.7}$$

$$\frac{\Delta V}{V} 100\% = 0.7BN + \frac{BN^{6.5}}{2.7\nabla^{\frac{2}{3}}} \tag{2.8}$$

$$\frac{\Delta V}{V} 100\% = 0.5BN + \frac{BN^{6.5}}{22\nabla^{\frac{2}{3}}} \tag{2.9}$$

where
BN is the Beaufort number.
∇ is the displaced volume.

Table 2.1: Correction Factor α_{corr} (Kwon, 2008)

Block Coefficient, C_B	Loading Condition	Correction Factor α
0.55	normal	$1.7 - 1.4Fn - 7(Fn)^2$
0.60	normal	$2.2 - 2.5Fn - 9.7(Fn)^2$
0.65	normal	$2.6 - 3.7Fn - 11.6(Fn)^2$
0.70	normal	$3.1 - 5.3Fn - 12.4(Fn)^2$
0.75	laden or normal	$2.4 - 10.6Fn - 9.5(Fn)^2$
0.80	laden or normal	$2.6 - 13.1Fn - 15.1(Fn)^2$
0.85	laden or normal	$3.1 - 18.7Fn - 28(Fn)^2$
0.75	ballast	$2.6 - 12.5Fn - 13.5(Fn)^2$
0.80	ballast	$3.0 - 16.3Fn - 21.6(Fn)^2$
0.85	ballast	$3.4 - 20.9Fn - 31.8(Fn)^2$

Equation (2.7) is for vessels in laden condition, Equation (2.8) for vessels in ballast condition, both for block coefficients of 0.75, 0.80 and 0.85 with the exception of container ships.

Equation (2.9) is for block coefficients of 0.55, 0.60, 0.65 and 0.70 for vessels in normal condition, for container ships.

Weather direction reduction factors are given as:

$$2\mu_{bow} = 1.7 - 0.03(BN - 4)^2 \tag{2.10}$$
$$2\mu_{beam} = 0.9 - 0.06(BN - 6)^2 \tag{2.11}$$
$$2\mu_{following} = 0.4 - 0.03(BN - 8)^2 \tag{2.12}$$

Equation (2.10) is for $30° - 60°$, Equation (2.11) for $60° - 150°$ and Equation (2.12) for $150° - 180°$.

According to (Kwon, 2008), this approximation method shows good accuracy in comparison to more extensive calculation methods.

As may be seen form the above, the method is very simple and also easy to implement as a solution to getting the added resistance with limited ship specific information.

Resistance from Wind

Wind has a smaller, but still considerable, effect on the resistance than that of waves. The effect is especially considerable for larger container and cruise ships.

A simplest way for calculating the wind resistance, is proposed by (Holtrop, 1988), presented in ITTC-78 as:

$$R_{air} = 1/2 \cdot \rho_a \cdot V_S^2 \cdot A_T \cdot C_{air} \tag{2.13}$$

R_{air} is the drag force, which is by definition the force component in the direction of the flow velocity.
ρ_a is the mass density of air.
V is the velocity of the ship relative to the wind.
A_T is the reference area.
C_{air} is the drag coefficient - a dimensionless coefficient related to the ship geometry and taking into account both skin friction and form drag.

A more sophisticated method is proposed by (Fujiwara et al., 2006). The method is based on wind tunnel and towing tank tests and is a development of the method proposed five years earlier (Fujiwara et al., 2001). In (Kristensen and Lützen, 2012) is proposed a more extensive list of air resistance coefficients, C_{AA}, as a correction to the ITTC-57 method. This method has a specially designed equation for container ships:

$$C_{AA} \cdot 1000 = 0.28 \cdot TEU^{-0.126} \tag{2.14}$$

but never less than 0.09.

In (Andersen, 2013) the directionality and speed of the wind has been studied for container ships. It is here presented that the container stacking used on container ships is of importance to the added air resistance and thus leads to an additional level of complexity in the calculations if taken into account. Some conclusions are drawn in (Andersen, 2013) and are summed up here:

- Changes to the container configuration on the fore deck are of great importance for the magnitude of the longitudinal force.
- A random container configuration can increase the longitudinal force significantly. Large irregularities such as many empty bays can increase the longitudinal force by 70-100% compared to the fully loaded reference ship for relative wind from around 0°.
- Streamlining of the container configuration on the fore deck has little influence on the longitudinal force.

The study presents some equations for calculating the wind resistance for different container configurations, but will not be discussed further here.

As for waves, the influence of wind from different directions and of different speeds may be studied more closely through model tests. In this case through wind tunnel tests. The results can then be extrapolated and mapped out to cover full directionality and relevant speeds.

Effect of Ocean Currents

The effects of currents may in some cases increase or decrease the average speed of the ship e.g. along the the coast of Japan, the Kuroshio current, with up to two knots (Chang et al., 2013). When considering currents, both global circulation currents and tidal currents are of interest for routing purposes. Figure 2.5 shows a general overview of global circulation currents.

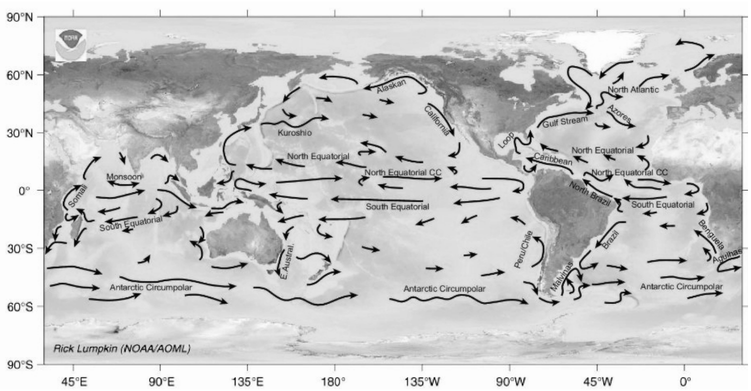

Figure 2.5: Overview of global surface currents courtesy of U.S. National Oceanic and Atmospheric Administration

Tidal currents are caused by gravitational effects from the moon, the sun, and the rotation of the earth. Tidal current are special in the way that they change direction during the day. In some areas there is flood (high tide) twice a day and in some only once. The currents generated by tidal forces differ significantly across the globe. In Pentland Firth, between Scotland and the Orkney island the current may exceed 10 knots in some parts of the straight. This illustrates well the need for implementation of currents in a routing tool.

The effects of currents may be taken into account in the fuel consumption model through a change in the speed needed produced by the engine, e.g. sailing in 2 knots head current will result in the ship having to produce effective equivalent to sailing 2 knots faster to obtain the desired speed over ground. This does however only take into account the longitudinal component of the current and not the effects of drift.

2.1.3 Major Factors Affecting Fuel Consumption

In addition to the factors already mentioned to have an effect on the fuel consumption many more are to be mentioned. The following briefly addresses some.

Both wind, waves, and currents have a direct effect on the added resistance, but an indirect effect is also present when the forces have perpendicular components on the ship. These cause both drift and yaw which in turn leads to added drag and thus resistance on the hull and rudder due to rudder motions required to keep the ship on a desired course.

If a ship travels in shallow water the vessel will squat. Squatting is an effect of interaction between the ship, the bottom and the water in between. The interaction causes a venturi effect which increases the draft of the vessel and slows it down. Due to this effect a voluntary speed reduction is often preferred in confined waters with small UKC, (Under Keel Clearance). According to (Molland, 2008) the speed loss is often calculated with (Schneekluth, 1978) speed loss curves based on the work by H. Lackenby.

Both hull and propeller fouling have great effect on the fuel consumption. The extra fuel consumption is both time dependent and dependent on where and how the ship has sailed. Marine growth causes an added frictional resistance which has lead way to both hull and propeller cleaning regimes as this may after a time period be economically viable. Figure 2.6 shows the effects of fouling on achieved speed.

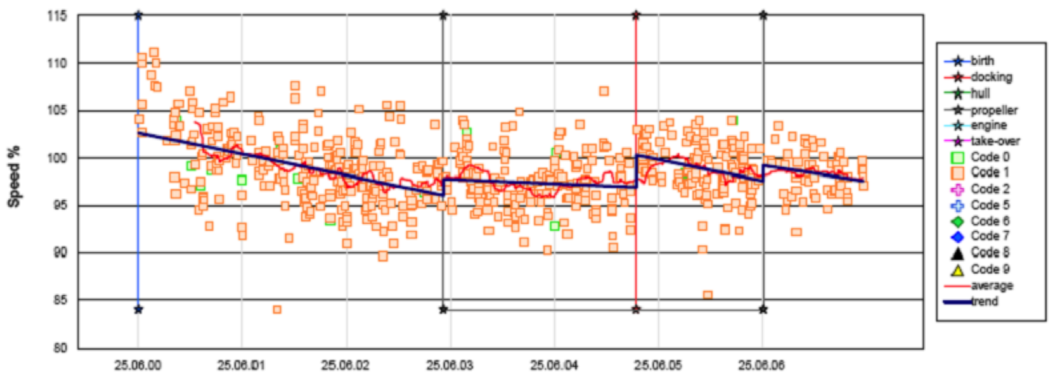

Figure 2.6: Effects of fouling, courtesy of MMT.

The loading condition indirectly has an influence on the fuel consumption. This is due to both draft, trim and the effected air resistance as addressed previously. Especially valid for container ships considering container stacking.

Waste heat recovery systems recover energy from the hot exhaust gases. This energy can be used for electricity production or returned to the propulsion system via an axle engine increasing the power delivered to the shaft. As waste heat recovery is only possible in certain operational areas this may lead to a major modification to the total fuel consumption of the ship for certain effective power demands as auxiliary generators may be relieved.

An axle generator is another component adding to the complexity of the fuel consumption model. When used it reduces the delivered power to the propulsion system and thus increases the required delivered power of the main engine in order to sustain the demand for effective power. The axle generator may only be required in certain situations which may then be incorporated into the decision making for a weather routing system.

Multiple engines increase the complexity of the fuel consumption model. Having two engines of the same type is rather simple but there are many more complex installations with e.g. two pairs of engines in so called father and son arrangement. Having a ship which is able to run in different operational modes an analysis needs to be performed in order to compute the most efficient setup for a specific delivered power. This in turn adds high complexity to fuel consumption curves where the algorithm may have to switch between different curves. Furthermore, dual fuel engines are emerging on the market, utilising different fuels with different efficiencies and costs.

Proper maintenance has a direct influence on the relation between effective power and fuel oil consumption. Different overhauls of machinery, mainly being the main and auxiliary engines have a different effects on this relation.

Summing up, based on simple ship specific data it is possible to make a somewhat extensive model relating ship speed to fuel consumption taking into account effects not included in the open water resistance calculations. When extensive tests have not been carried out or are unavailable such a model may be required even though based on approximative equations. As a result, the calculated fuel consumption for a full journey may not fully correspond to the real consumption.

With added information and modelling all of the above mentioned factors may be added to the fuel consumption model as modules to the basic system model.

2.2 Constraints

The operational range of a ship is governed by a number of constraints. Some are regulated by flag states, port states, and class rules and some are intuitively straight forward such as travelling over land or in water depths less than the draft of the ship. Ship motions are to be modelled closely to obtain limiting states but may be avoided through a crude simplification of maximum wave height. This simplification is though not a viable solution as ship motions and limiting states are related to both magnitude, frequency, and direction for many constraints.

Ship response predictions are needed when weather routing is to be extended from storm avoidance to proper routing. Responses are predicted using transfer functions which relate the encountered wave environment to the ship responses. The transfer functions used are Response Amplitude Operators (RAO) and Response Drift Operators (RDO). RAO's and RDO's may be determined through a variety of methods, but will all require significant ship specific information or modelling. Even with a modelling of transfer functions there remains the difficult task of predicting the detrimental, nonlinear responses of ships.

Accelerations are the main parameters to be considered when looking at dangerous ship motions. Large amplitudes of motions pose a potential danger to ship, cargo and crew. Large motions may lead to serious incidents and even capsize.

IMO has in MSC1. Circ 1228(IMO, 2007) proposed recommendations for avoidance of specific potentially detrimental situations. They are based on ship length and speed. The program developed in this thesis has been prepared for implementation of these rules but upon basic calculation it was seen that only the most adverse weather would result in violations and these calculations were left out. The recommendations have been criticised in, (Schiller, 2011), for not being accurate and should be used with caution.

2.2.1 Motion Based Constraints

Several constraints relating ship and cargo safety to ship motions are described in the following.

Pitch Motion

One problem related to pitching is slamming. Slamming occurs when the relative motion between ship and sea is large. If the sea meets the bow or stern of the ship at high relative velocity the ship may suffer a slamming event. Slamming is characterised by high impact loads much larger than other wave loads. Slamming will cause vibrations of the hull girder and sometimes the vessel suffers local buckling.

Probability of slamming is best calculated using RAO's and visualised using polar plots.

Roll Motion

Roll motions can generate severe acceleration on both ship and cargo. These accelerations may cause crew injuries or discomfort, cargo damage or loss, and even damage to structural members. Large angles of roll can further lead to dangerous situations such as large amounts of green water on deck, submerging of compartments not weather tight and ultimately capsize. One of the worst cases related to roll is parametric roll described further down.

Broaching

Broaching is a wave induced event leading to a sudden large amplitude course deviation. Broaching may occur in following and quartering seas and there are a number of events that fall under the the definition of broaching:

- Single wave broaching or surf-riding.
- Very steep waves overtaking with low ship speed.
- Low frequency and large amplitude yaw motions which build-up as successive waves strike the ship from behind.
- In moderate sea states, at high speed and slowly overtaken by the waves.

According to MSC1. Circ 1228(IMO, 2007) surf-riding and broaching may occur when the angle of encounter is in the range of 135 to 225 degrees for speed exeeding that given in equation 2.15.

$$V_{limit} = \frac{1.8L}{\cos(180\text{-}\phi)} \tag{2.15}$$

where
V_{limit} is the limiting ship speed.
L is the ship length.
ϕ is the encounter angle.

Synchronus Roll

Synchronous roll is according to MSC1. Circ 1228 (IMO, 2007) defined as: Large rolling motions that may be excited when the natural rolling period of a ship, Equation 2.16, coincides with the encounter wave period. In case of navigation in following and quartering seas this may happen when the transverse stability of the ship is marginal and therefore the natural roll period becomes longer.

$$T = \frac{2\pi k}{\sqrt{g\overline{GM}}} \tag{2.16}$$

where:
T is the natural rolling frequency.
k is the radius of gyration.
g is gravitational acceleration.
$\overline{GM}$ is the metacentric height.

Parametric Roll

Parametric rolling is a highly dangerous phenomenon. The driving force is large variations in ship stability. The phenomenon leads to sudden and sometimes very large roll angles. Ship types sensitive to parametric roll are ships with large bow flares and wide transoms like car carriers and container vessels. Submerging the bow flare and transom will lead to a rapid increase of water plane area, increasing the metacentric height.

$$\overline{BM} = I/\nabla \tag{2.17}$$

where:
I is the longitudinal moment of inertia.
∇ is displaced volume.

Having a large metacentric height in turn leads to shorter roll periods, see Equation 2.16. The combination of large roll angles and short roll periods generates extreme acceleration, dangerous for both the ship and its cargo. The phenomenon is now well documented in e.g. (Fossen and Nijmeijer, 2012).

2.2.2 Pirated Waters

In recent years piracy has become a real threat along some of the most important shipping routes. The most well know area is the the waters off the horn of Africa. Even though naval ships operate in the area, pirate attacks are still a large concern to shipping companies. IMO has produced a large amount of publications with operational guidance for ships and shipping companies. Especially relevant for weather routing is the MSC.1/circ.1339 (IMO, a). In this publication IMO recommends ships to keep a speed of at least 18 knots when passing through high risk pirated waters. This means that there is a need for incorporating minimum speed areas in the routing software.

2.2.3 Ice

Ships have two different ice related problems. The first being ice in the water which the ship needs to pass through or go around. The other problem is icing of the ship which raises the centre of gravity and may render the ship unstable. Within ship routing in general only ice in the ocean is considered.

Ice is an important variable for navigation in large parts of the oceans. For some ships ice may need to be represented as impassable zones as their hulls and classifications might not allow ice navigation. For other ships ice navigation is allowed in certain areas according to port state rules. Important to note is that the power speed curves will change radically depending on the ice conditions.

2.2.4 Emission Controlled Areas

IMO has set up rules concerning the allowable emissions from ships. Limitations are both global and regional. Regional emission areas are known as Emission Controlled Areas (ECA). In different areas limits are set up for e.g. NOx, SOx, and PM:

Baltic Sea: SOx.
North Sea: SOx.
North American Sea: SOx, NOx, PM.
United States, Carribean Sea: SOx, NOx, PM.

Highly relevant for weather routing are the so called Sulphur Emission Controlled Areas (SECA). In SECA areas the limit for sulphur content in the fuel is 1.00% until January 1st 2015 when the limit is lowered to 0.10% (IMO, b). Table 2.2 shows specifications on the allowed sulphur emissions until 2020. Emission levels are significantly lower than what they are today. This means that measures need to be taken when sailing in these areas. Some ships will run on other fuels such as MGO or LNG while others will continue using HFO but have scrubbers installed on board. When it comes to weather routing these areas are of interest if the ship normally operates on fuel with higher sulphur content and needs to switch to another fuel when sailing in SECA areas. This is the case for ships normally using HFO and using MGO in SECA areas.

2.3 Weather and Weather Forecasts

The weather forecasts available today are limited in how far into the future they predict weather well enough to be used as a foundation for calculations. Due to the lack of reliable forecasting, some part of the journey may have to be calculated based on weather estimated on the basis of statistical data for that time of year.

Table 2.2: SOx emission limits, according to MARPOL Annex VI

Limits to SOx emissions outside ECA's	Limits to SOx emissions inside ECA's
4.5%m/m prior to January 1st 2012	1.5%m/m prior to July 1st 2010
3.5%m/m after January 1st 2012	1.0%m/m after July 1st 2010
0.5%m/m after January 1st 2020	0.1%m/m after January 1st 2015

Weather forecasts are provided by so called weather providers. Two examples of weather providers are Swedish Meteorological and Hydrological Institute, (SMHI), and Danish Meteorological Institute, (DMI). The most common way of communicating weather data for onboard weather routing systems is via email.

Predicting weather for routing is the single most important variable for obtaining good routing results. Having access to the best available weather forecasts is crucial. Not only the accuracy varies between different providers, the resolution also varies significantly. The resolution differs over the oceans with better resolution close to the coast. Coarser resolutions is also a way to keep the size of data down when sending it to ships via expensive satellite communication systems.

2.3.1 Weather Parameters in Routing

Previous subsections have described factors important to routing, therefore it is possible to list parameters needed in a weather forecast:

- Wind speed and direction
- Wave direction, height and period
- Current speed and direction
- Ice conditions: Thicknesses, presence of ice bergs and drift ice

2.3.2 Uncertainty and Risk

Weather forecasts are commonly considered fairly accurate the first few days. After this the uncertainty increases which is seen by looking at the differences of forecasts. The quality of forecasts vary between areas since the weather models need to be adjusted and tuned for different areas of the globe. As a rough generalisation weather forecasts are more accurate in the northern hemisphere than the southern (Bowditch, 2002).

A problem routing providers are faced with for long voyages is what input to use when there is no valid forecast the last days of the voyage. Forecasts ranging further than the first 14-16 days often do not directly include the parameters needed for routing. A forecast ranging 30 days may only include pressure distributions in the atmosphere and is thus not applicable for routing purposes.

In most cases weather routing generates small diversions from the shortest route. Sometimes the diversion are larger due to e.g a large storm. Making this large diversion can be considered taking a financial risk. If the fallout differ a lot from the predicted forecast, the ship might have taken a significantly longer route than needed consuming a lot of extra fuel. Further, the reliability of the involuntary speed reduction and the predicted storm track are subject to uncertainties. Choosing a route ahead of a storm can be considered more risky than one after it depending on the uncertainties.

Ensemble forecasting is a way of forecasting taking uncertainties into account. By changing the initial conditions slightly between simulations the same model will generate a range of possible future weather situations. Analysing the differences and spreads of these leads to a more complex input increasing the computation effort but also a possibility to make better routing. The accuracy of this method can be further improved by using data from more then one forecast provider.

3 State of the Art Study

A state of the art study has been carried out to give an offset for the project by mapping capabilities to be included in a modern routing software. Two different aspects of state of the art have been considered. The first is to study software already in use and the second is to study methods investigated in research.

The methodology used consists of three main activities. The first is the mapping of the market leaders within weather routing software. This has been done through email communication with the companies behind the software's and a literature study on them. The selected companies were each asked to fill out a questionnaire, Appendix A. The results of this study is presented in the following. The second is a study on current and previous research to map the evolution of weather routing. The third is participation in network groups and workshops on weather routing to gain an insight in the practical use of weather routing and research and development.

The general findings from the study may be seen illustrated in Figure 3.1. Features and functions are divided into three main categories dependent on whether the functionality is mature and included in current software, less mature and in the process of implementation to be included in future releases or only present i research and may be part of an implementation in the future. The components in the figure are further discussed in the following. It should be noticed that some elements are presented in more than one place.

3.1 Current Systems on the Market

The identified most relevant companies were studied and asked to contribute with their knowledge and expertise. They are referred to as market forerunners. The discussions have not been quantified and included in the report but questionnaires may be seen in Appendix A.

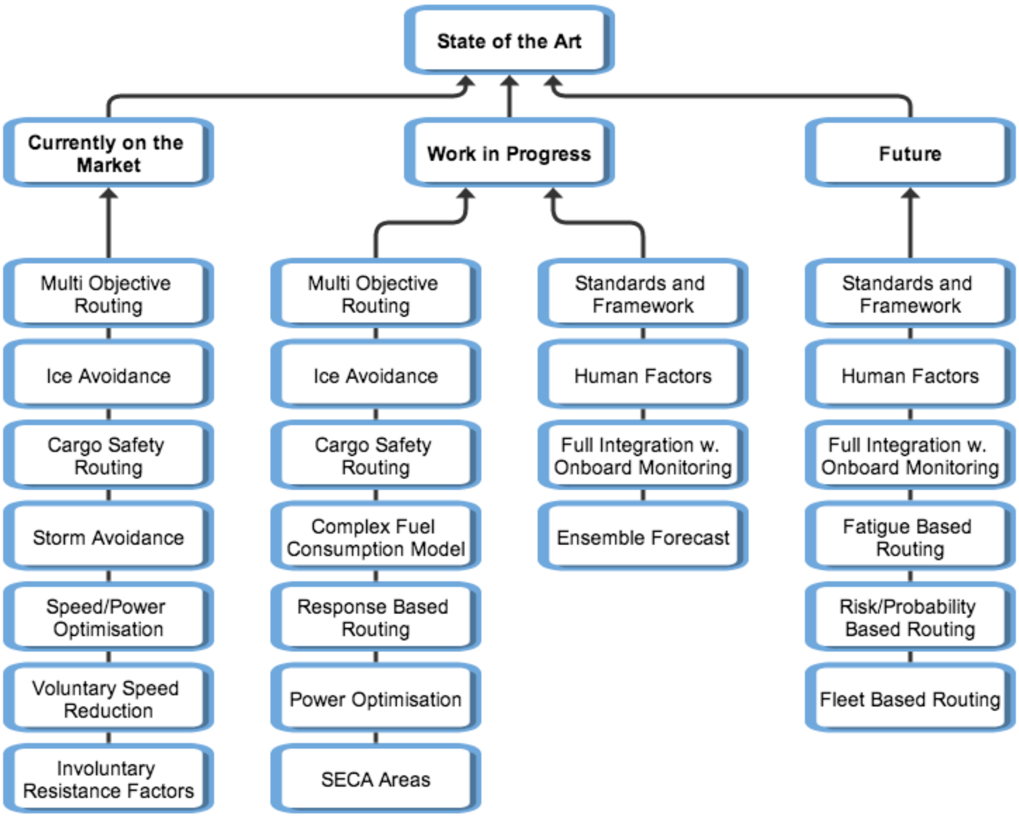

Figure 3.1: State of the art in weather routing

3.1.1 Market Forerunners

The following introduction of the selected companies is based on information found on their respective web pages. An overview of the software capabilities is presented in Table 3.1, mainly based on the questionnaires.

AWT (StormGeo) - BonVoyage

AWT provides both shore based and on board routing based on minimisation of voyage time or fuel consumption. BonVoyage is AWT's on board solution, which provides a graphical user interface taking into account surface pressure, surface winds, significant waves, swell, tropical storms, ice, current and sea surface temperature, all delivered to the system via email. AWT recently became part of the StormGeo group.

Force Technology - SeaPlanner

SeaPlanner is an on board system developed through cooperation between FORCE Technology and DMI Maritime Service. SeaPlanner offers on board routing with the objectives of either minimum voyage time, fixed ETA with constant power, fixed ETA with optimised speed, or fixed ETA with constant RPM.

Jeppesen - VVOS

Jeppesen is part of Boeing and based in the United States. The route optimisation software provided is called Vessel and Voyage Optimisation Solution, (VVOS). The software uses ship specific data to make trade-offs among ETA, fuel consumption, ship motions, hull stresses, weather, and sea conditions.

MeteoGroup - SPOS

MeteoGroup's onboard routing software is called Ship Performance Optimisation System, (SPOS), Onboard. SPOS is designed to take ship specific characteristics into account. The routing is based on both weather forecasts and ocean current data.

SeaWare (StormGeo) - Enroute

SeaWare Enroute is a Swedish on board ship routing software. SeaWare is a StormGeo company. It has support for ship specific power-speed curves as well as seakeeping characteristics. SeaWare is integrated in the DNV-GL software DNV-GL navigator. One of the special characteristics in the software is the model for predicting parametric roll.

Table 3.1: Selected software capabilities, *Ind. denotes "Indirectly", TBI denotes "To Be Included", ∗Simulated annealing (Monte Carlo)*

Feature	StormGeo AWT	Force Technology	Jeppesen	Meteo Group	StormGeo SeaWare
Name of software	BonVoyage	SeaPlanner	VVOS	SPOS	EnRoute
Power opt.	Yes	Yes	Yes	Yes	Yes
Surface currents	Yes	Yes	Yes	Yes	Yes
Tidal currents	Yes	Yes	Yes	TBI	?
Ice avoidance	Yes	Ind.	Yes	Yes	Shown
Ship dynamics	Yes	Yes	Yes	Module	Yes
Graphical weather	Yes	Yes	Yes	Yes	Yes
Length of forecast	16 days	10 days	15 days	9 days	15 days
Optimisation time	15s	ca. 1 min	Minutes	Minutes	15+s
Pirated waters	Yes	Yes	Yes	Ind.	Yes
SECA's	Yes	Yes	Yes	Yes	Yes
Wind on speed	Yes	Yes	Yes	Yes	Yes
Waves on speed	Yes	Yes	Yes	Yes	Yes
Hull fouling	Ind.	Yes	Ind.	Ind.	Ind.
Propeller fouling	Ind.	Yes	Ind.	Ind.	Ind.
Shallow water	No	Yes	Yes	No	TBI
Draught & trim	Yes	Yes	Yes	Yes	Yes
Algorithm	No info	MC*	3DDP.	In house	GA
Weather provider	GFS	DMI	Several	In house	ECMWF/ RTOFS
Coastal routing	Yes	Yes	Yes	Yes	Yes
Forecast res. span	2/0.125	0.5/0.1 deg	*ca*1 deg	?	*var.*/0.125

3.1.2 Functionality

The available softwares' are capable of optimising for different objectives such as: specified ETA, lowest fuel consumption, low ship responses etc. Due to the significant increase in fuel prices there is a clear trend of minimising fuel consumption. Changing the optimisation objective is a possibility in all of the best software. The best weather routing providers have the possibility of routing with dynamic speed or power profiles, enabling ships to speed up or slow down in order to avoid bad weather.

Concerning involuntary speed reduction many components are taken into account when routing with the best software, including: wind, waves, currents, fouling, and engine maintenance. The added resistance in sea-way is of great interest to weather routing. In (Shao et al., 2006) speed reductions are handled through reduction formulas presented by Kwon in (Kwon, 2008) and used by IMO in (IMO, 2009). The formulas are a simplification leading to reduction percentages and may be seen in Section 2.1.2.

Constraints in this context are operation limiting criteria. In (Shao et al., 2006), constraints handled are surf-riding, synchronous roll, parametric roll and reduction of intact stability. This is done following revised IMO guidelines (IMO, 2007) leading to maximum conditions. (Mannarini et al., 2013) also follows the guidelines outlined in (IMO, 2007). These methods have been criticised for being to simple . Therefore the development is heading towards response based routing. (Padhy et al., 2008) handles weather constraints through a sea-keeping analysis of the ship. Vessel dynamics is often treated in connection to added resistance. In (Padhy et al., 2008) the responses to weather is handled through RAO's. The area of ship's response based routing is somewhat immature and we can expect the accuracy of it to increase significantly in the future.

3.1.3 Applied Algorithms

The use of many different optimisation algorithms has been presented in the past decades. The use of a good algorithm may have a large effect on the quality of solution provided. An in depth analysis and presentation of algorithms is done in Section 4. The most commonly seen are 3DDP (3 Dimensional Dynamic Programming), GA (Genetic Algorithm) and the Dijkstra algorithm.

3DDP

(Shao et al., 2006) describe the development of a forward dynamic programming method for weather routing. This method computes a route based on both course and power to allow for the ship to slow down or speed up in order to save fuel. The 3DDP method shows significant fuel saving compared to the more common 2DDP method. The method presented in (Shao et al., 2006) does not take into account uncertainty of weather but only handles weather forecast data directly. The method was in (Shao and Zhou, 2011) measured against three different GA's. The 3DDP method showed improved performance compared to all three GA methods.

Dijkstra

(Mannarini et al., 2013) describes the development of a concept model relying on a modification to the Dijkstra algorithm. The method only takes into account wave

height, peak period, and direction along with safety restrictions based on these parameters.
The method does not at the stage described in the article handle voluntary speed reductions.

(Padhy et al., 2008) gives another method for routing using a modification to the Dijkstra algorithm. In this method voluntary speed reduction is also possible. The method takes into account weather influence on the speed of the ship and may also handle restricted areas, such as land, through the assigned weight of edges.

Genetic Algorithm

Often cited is the work done in (Hinnenthal, 2008). This gives an elaborate suggestion to the use of a multi-objective genetic algorithm. Ship responses and constraints are modelled using ship response operators. In this proposed method the objectives of the optimisation are both fuel consumption and ETA optimising both route and velocity profiles.

It is further noted that at least 9 variables for the velocities and 7 variables for the course are needed to obtain sufficient information on a route crossing the North Atlantic.

3.2 Work in Progress and the Future of Weather Routing

It is of interest to determine how weather routing has evolved, where it is now and where it is moving. Much focus is given to algorithms, ship dynamics and weather. Another discussed topic is fatigue routing. Research has shown that container vessels in the North Atlantic trade may increase their fatigue life by 50% using fatigue routing (Mao et al., 2012), there is though not any direct coupling to immediate costs and it has therefore not been implemented yet. Including fatigue routing is taking multi-objective routing a step further.

As of January 1st 2015 the so called SECA areas will come into force as described in Section 2.2.4. This means higher operational costs for most larger ships operating in these areas. Current weather routing providers handle the SECA areas in different manners. Some only display them without taking their effects into the optimisation, some take into account the effects of possibly switching fuel or requiring scrubbers and some add them as no-go areas. The providers have not implemented the SECA areas yet are to implement them in coming releases.

Piracy is a current and highly prioritised issue within shipping. Piracy services are seldom included in routing software. It is often provided as separate or add-on service. Piracy is also connected to risk based routing which is discussed further down.

To determine limitations to operational ranges is a challenge. The level of precision is dependent on the available ship specific information. The best software uses RAO's and RDO's in predicting ship responses in different sea-states for different loading conditions. Simplified methods are still not very well defined or tested and further work in this area is expected.

There is a general trend within shipping making everything possible shore based. This is also the case with routing. It is possible to make computations onshore using far more computing power than on board. The results can then also be analysed by experts before being sent out to the vessels. A major issue is human factors. Taking away this responsibility from the captain creates a difficult situation as captains are responsible for the ship and without proper background knowledge on the produced routes they may be reluctant in implementing them.

Weather uncertainty handling is an area of research which has great potential to improve routing. Taking into account the uncertainties related to the forecasted weather may lead to very different routes. This either done through a weighting in on available forecasted weather or through ensemble and super ensemble forecasts. Not many software or research projects do though yet use these. Some examples of the use of ensemble forecasts are presented in (Hinnenthal, 2008) and (Skoglund et al., 2012).

Connected to the uncertainty and spread of weather forecasts there is a possibility to do risk based routing. Also weighing into this is the difference between the track of different routes. A route ahead of a storm would most often be considered as a higher risk route compared to a route plotted behind the same storm. Concerning weather, weather forecasts with mean and standard deviations etc. could be used.

To sum up, weather routing is under constant development. The main driving forces are currently fuel minimisation. The new SECA regulations have received great focus recently along with piracy as both are pressing matters for ship owners and operators. Constantly increasing computing power enables response based routing which in turn will increase safety for ship, cargo and onboard personnel. Response based routing will most likely be coupled with on board monitoring systems to achieve even more precise models. New algorithms are emerging as the system modelling becomes better and computing power increases, leading to new possibilities within routing.

4 Optimisation Algorithm

4.1 Algorithm Selection

There exists a vast amount of available optimisation algorithms. Selecting the most appropriate for a specific task can be a challenge. It requires analysis of what information is available for calculations, allowable convergence time, whether the solution space includes local minima, which may need to be handled separately and much more.

Many optimisation algorithms require derivatives of the included functions and thorough knowledge of the underlying calculation structures. For this project such information is not available as several calculations are done through black box functions provided by MMT. This alone limits the number of available algorithms for this specific project.

As the algorithm, through the final program, should be able to run without having to do major adjustments and act in a more generic manner, few obvious choices remain.

(Chen, 2011) mentions that most weather routing systems use a variation of the Dijkstra algorithm and 3D Dynamic Programming. Other possibilities include the Genetic Evolutionary Algorithm and Monte Carlo with simulated annealing, which are used for a wide variety of purposes. Lastly the Direct Algorithm is analysed here as it also can handle black box optimisations in a time effective manner. The algorithms considered are described more thoroughly in the following.

4.1.1 Dijkstra Algorithm

The Dijkstra algorithm was proposed in 1959 by Edsger Dijkstra in a paper called *A Note on Two Problems in Connexion with Graphs*, with the purpose of solving the Shortest Path Problem. It has since gained popularity and is widely used in computer science and operations research, (Sniedovich, 2006).

The algorithm works on two principles:

- The subpath of any shortest path is itself a shortest path.

- Given $d(a,c)$ is the shortest distance between points a and c, then $d(a,b)+d(b,c)$ will always have a larger or equal distance between a and c

From the starting node all possible edges to neighbouring nodes are tested and assigned a cost value. This value may e.g. be a distance or a fuel consumption. The route with the lowest end node value again tests possible edges to neighbouring nodes and values are assigned. When a node already reached is tested, the second principle stated above is used. If the end node value for the newly tested route is smaller the new value is assigned and the original route to the point disregarded as part of an optimum solution. If the value is higher the new route to that node is not created. This way routes evolve towards the final node, and only one optimum solution will remain. An example of such a route optimisation may be seen in Figure 4.1, the bold lines mark the routes with the smallest values, and the one stretching from point **s** to point **t** the optimum route.

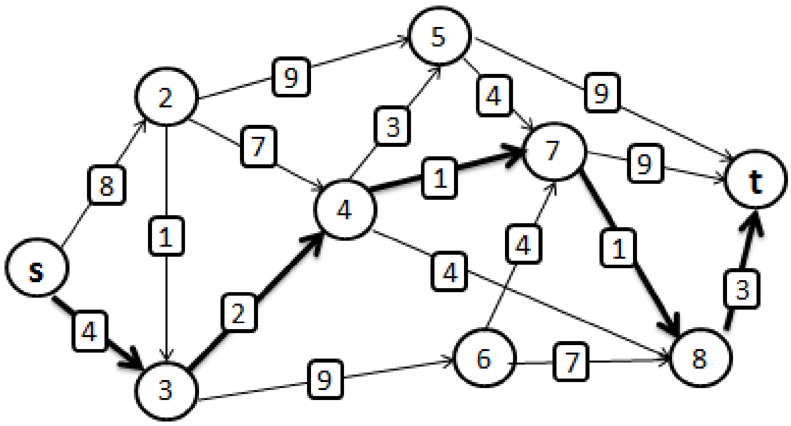

Figure 4.1: Example of Dijkstra shortest path.

The Dijkstra algorithm may further be developed for edge costs to be functions of time to take into account dynamic situations as time varying weather fields for the use in dynamic weather routing. And further to take into account voluntary speed variations, (Mannarini et al., 2013).

Advantages

The Dijkstra algorithm will always find the optimal route. For cases of constant speed and handling of ship motions through reduction curves only, the algorithm is very fast.

Disadvantages

The route is restricted to move along nodes and thus is highly grid dependent, therefore proper gridding is highly important and will for speed be case dependent. There exist many different modifications to the algorithm and selecting the optimal one to use may be difficult. Modifying the algorithm to handle dynamic weather may be complex. Arrival time and minimum fuel consumption must be handled simultaneously.

4.1.2 3D Dynamic Programming

Dynamic programming is based on Bellmans principle of optimality; "an optimal policy has the property that whatever the initial state and initial decision are, the remaining decisions must constitute an optimal policy with regard to the state resulting from the first decision", (Shao and Zhou, 2011).

Where the more conventional 2D Dynamic Program may be used to compute an optimum route, by determining headings, this method is restricted to determining the route using constant speed or power for the entirety of the voyage. The 3D Dynamic Programming model, (3DDP), allows for the optimisation of sailing profile as well, through speed/power control, (Wei and Zhou, 2012).

According to the Bellman principle the problem is broken down into simpler subproblems, i.e. stages. In this method the control variables are determined on the basis of preceding stages. (Shao and Zhou, 2011). Figure 4.2 shows the division into stages and states as used in the example seen in (Shao and Zhou, 2011). Here stages are separated by a distance ΔX and states by ΔY. The states, seen in the figure as 'grid' contain both geographical location and time.

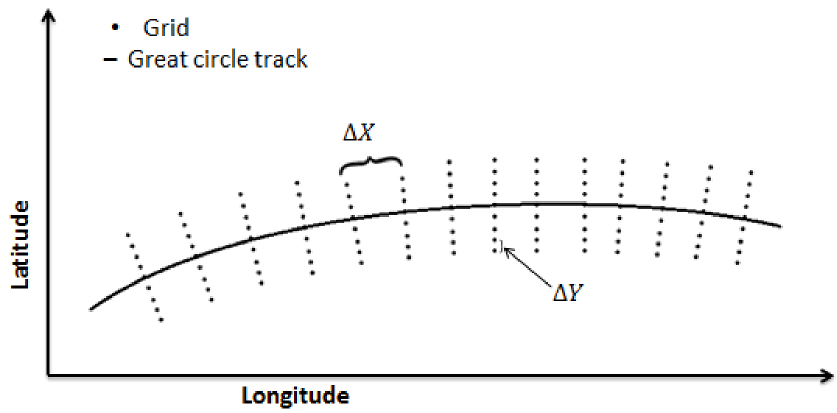

Figure 4.2: Division into stages and states.

Grid points containing the crossing of islands or similar are then deleted. From each grid point the defining parameters, including fuel consumption are calculated for getting to all states in the next stage. Once the whole system has been mapped out a backward calculation procedure is employed to identify the routes with optimal fuel consumption.

The 3DDP method does not use a set arrival time but instead leads to resulting fuel consumptions at different arrival times.

Advantages

Impassable areas may be easily handled by skipping state points containing e.g. land. The 3DDP method allows for the user to select a route from different arrival times depending on fuel consumption. Gridding may be done relatively simply as it may be done by spreading grid points perpendicularly at uniform intervals from at each stage form the great circle route. The calculations may be done using constant power of the main engine and calculating influences of weather through time steps.

Disadvantages

The accuracy of the result using 3DDP is highly dependent on the grid fineness. While this method shows the advantage of easy handling of land constraints, finding the shortest route around or between e.g. islands requires significant grid fineness. The computation time is dependent on the fineness of the grid, which may have to be fine.

4.1.3 Genetic Evolutionary Algorithm

Following is based on (Burke and Kendall, 2005) and (Szlapczynska, 2013).

The genetic algorithm replicates natures evolution with principles of survival of the fittest. It does so through an initial population of solutions, either created through random sampling, routes with advantages in fuel consumption, ETA, or safety, or as a combination. The routes are evaluated and given a fitness value, based on fuel consumption, ETA, safety or through combinations. A selection operator determines the routes containing best fitness values which may then be crossed over, mutated or handled through specialised operators to form a new solution population. This population has then in turn inherited the qualities of the previous solution and evolved from this. The newly created population replaces the old and the process is repeated until a satisfactory solution quality or maximum optimisation time is obtained.

When the population iterations are terminated, the final population forms a solution space which may be handled in several ways. A Pareto front leads to the limiting best solutions in the solution space. An example of such may be seen in Figure 4.3.

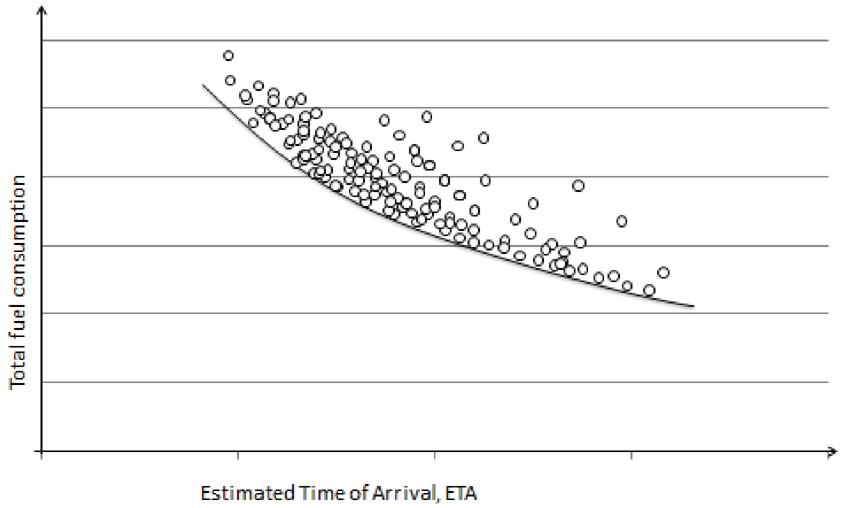

Figure 4.3: Pareto front resulting from multi objective genetic algorithm.

From the final Pareto-set a ranking may be performed, by assigning significance values to the optimisation criteria, e.g. fuel consumption, ETA or safety, to produce a ranking of solutions which reflects the requirements to a best solution.

The search space for the algorithm may be both discrete or continuous. For application in ship routing the use of a continuous search space is preferred. The genetic algorithm has the possibility of multi criterial search which has the advantage of being able to produce a final solution directly, compared to e.g. single criterion modified isochrone methods.

The method does not ensure a globally optimal solution and is highly affected by the number of initial sampling points both in quality of solution and computation time. If too few points are sampled or the computation time is short, there is a possibility that the algorithm may only converge towards a local minimum as these give good fitness values which are inherited, without enough time for mutations to possible direct the solution space towards a global minimum.

Advantages

The genetic algorithm is relatively straight forward to use. The algorithm may be stopped at any time and an optimal solution produced based on a Pareto front and ranking. Through mutation, crossover and other specialised operators the algorithm is not highly likely to become stuck in local minima. From the Pareto fronts the operator has the possibility, through ranking parameters, of selecting a solution suitable with regards to both ETA, fuel consumption and safety.

Disadvantages

Arrival time and minimum fuel consumption must be handled at the same time. The algorithm does not ensure the optimum solution, but will rather give an approximation to it which improves with increasing function iterations. The genetic algorithm is not very fast as it is highly dependent on the size of the initial population which in turn increases computational time. Parameters must be tuned and are likely case specific for optimal computing.

4.1.4 DIRECT Algorithm

The DIRECT algorithm is developed on the basis of Lipschitz optimisation. (Jones et al., 1993). The name of the algorithm comes from the describing words DIviding RECTangles, which illustrate the way the algorithm moves towards an optimum. The method has fast convergence and requires only few evaluations for each iteration. The number of iterations directly determines how precise the optimum solution is calculated.

The algorithm is simple in its nature and only few settings have to be adjusted for it to run properly. The algorithm searches for both local and global solutions simultaneously and the global search it not restricted by a possible intensified local search contrary to many other algorithms.

Advantages

The direct algorithm may be set to stop at a specified amount of function iterations or evaluations and will provide a best solution. The algorithm has fast global convergence.

Disadvantages

The complexity of the optimisation using the DIRECT algorithm increases rapidly with the number of intermediate waypoints between start and end. Each waypoint leads to 3 additional dimensions in the problem as a waypoint contains information on latitude, longitude and speed entering the point. Penalty parameters have to be adjusted and may be case dependent for optimal computing. The computation time increases rapidly with number of search dimensions.

4.1.5 Selection of Algorithm

As thorough work has been carried out on the application of most of the algorithms in weather routing it was found interesting to test an algorithm not seen in this application to see if this might lead to any benefits.

The simplicity of use played a major role in the selection process as it was beyond the scope of the project to perform a larger modelling of a more complex algorithm.

Based on this the DIRECT algorithm was selected. Even with the weaknesses of the algorithm possibly playing an important role in its use in routing it was decided that the probable benefits were of interest to be investigated.

4.2 The DIRECT Algorithm

4.2.1 Lipschitzian Optimisation Using Shubert's Algorithm

Following description is based on (Jones et al., 1993).

The Direct algorithm is a sampling algorithm and does therefore not require specific knowledge of the object- or constraint functions. As the precision of the solution is dependent on the computation time a satisfactory solution may be at the cost of computational time. According to (Jones et al., 1993), the algorithm has though shown to be very competitive with other algorithms in the same class.
The method is based on the ideas behind the Lipschitz optimisation with Shubert's algorithm, but tries to overcome some of the limitations there are connected with this method. The following describes the classical Lipschitz optimisation with Shubert's algorithm.

Lipschitzian optimisation is based on Lipschitz continuity. In R^1, following applies:

For $M \subseteq R^1$ and $f : M \rightarrow R$ the function f is Lipschitz continuous in M with Lipschitz constant α if:

$$|f(x) - f(x')| \leq \alpha|x - x'| \quad \forall x,x' \in M \tag{4.1}$$

If $M = [a,b]$, f must satisfy the inequalities:

$$f(x) \geq f(a) - \alpha(x - a) \tag{4.2}$$

$$f(x) \geq f(b) + \alpha(x - b) \tag{4.3}$$

The point where the line described by the inequalities intersect leads to a first estimate of a minimum, x_1, see Figure 4.4. As the solution must lie above the V-shape created by the two lines, α must be larger than or equal to the numerically largest derivative.

Using Shubert's algorithm, M is split into $[a,x_1]$ and $[x_1,b]$. The functional value of x_1 is evaluated and the same inequalities are applied to the new subsets, divided by $f(x_1)$. This way two new lower bounds are found, see Figure 4.5.

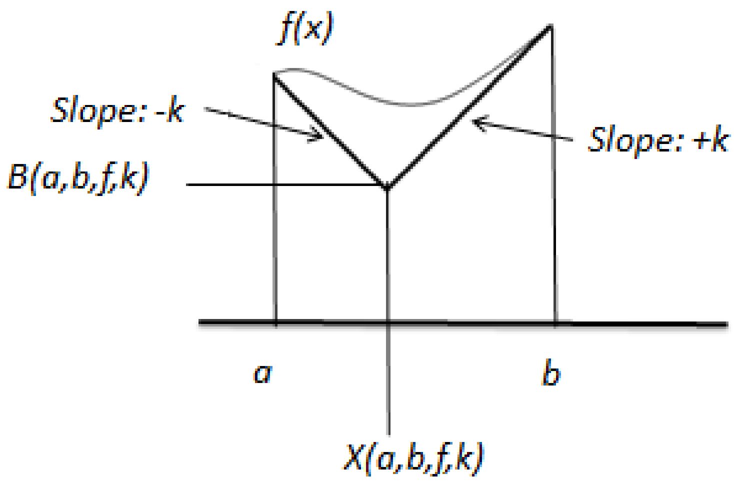

Figure 4.4: The Lipschitzian lower bound for an interval. k is referred to as α in this text

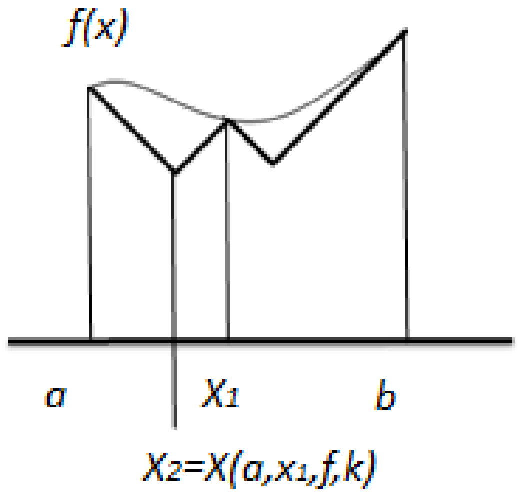

Figure 4.5: The Lipschitzian lower bound and interval for first iteration.

Both lower bounds are found, and the one with the lowest estimated function value is split in the next iteration and the method is repeated. For the following iterations, when evaluating which subset contains the smallest functional value, all lower bounds that have not been used as a boundary for a subset are taken into account. This is how both the local and the global search is ensured.

Shubert's algorithm does not have any direct mulitdimensional analogy. The limits of M are evaluated at initiation and will thus require 2^n evaluations for n dimensions.

Further the Lipschitz constant, α may be difficult to determine at initiation and a poor estimate may result in slow or no convergence.

4.2.2 DIRECT Initialisation

Following is based on (Jones et al., 1993; Finkel, 2003).

The Direct algorithm overcomes the problems concerned with both estimating the Lipschitzian constant and continuity. Instead the algorithm searches using all possible α-values.

Initially the search area is transformed into a hyper-cube, as:

$$\overline{\Omega} = \left\{x \in R^N : 0 \leq x \leq 1\right\} \tag{4.4}$$

Where x is a vector containing the vaiables. The centre of the hyper-cube is c_1 with functional value $f(c_1)$. The points $c_1 \pm \delta e_i$ are evaluated for $i = 1,...,N$, where N is the number of dimensions/variables and δ is one third of a side length of the hyber-cube. For each dimension, the lowest functional value is determined as:

$$w_i = \min(f(c_1 + \delta e_i), f(c_1 - \delta e_i)) \tag{4.5}$$

The dimension with the smallest functional value, w_i, is divided into three, whereafter the runner up is divided into three, continuing to N. The process is illustrated in Figure 4.6.

4.2.3 DIRECT Iterations

After the division into hyper-rectangles, the algorithm checks for potential optimality. This distinction is made if there exists a $\hat{K} > 0$, so that following applies:

$$f(c_j) - \hat{K}d_j \leq f(c_i) - \hat{K}d_i \qquad \forall i \tag{4.6}$$

$$f(c_j) - \hat{K}d_j \leq f_{min} - \epsilon|f_{min}| \tag{4.7}$$

Where f_{min} is the smallest of all registered functional values at the current iteration, c_j is the centre of hyper-rectangle j, similarly for i, d_j is a measure of the hyper-rectangle, the program uses the distance between the centre c_j and the corners, ϵ is a small positive number used to ensure that $f(c_j)$ is better than the current best, the program uses a value of $1 \cdot 10^{-4}$.
If a hyper-rectangle is identified as being potentially optimal it is divided in the dimension with the longest side. If the hyper-rectangle is equally large in all directions it is divided as in the initialisation.

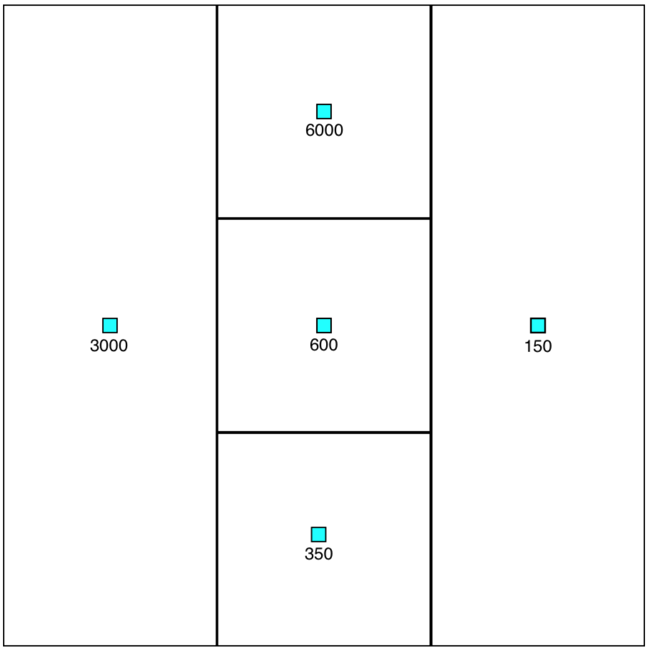

Figure 4.6: Initial subdivision into hyper-rectangles, (Finkel, 2003)

After subdivision, the new centres are evaluated and the next iteration is initiated by checking for potential optimality.

An example of the subdivisions of a search space in two dimensions after 191 evaluations may be seen in Figure 4.7.

4.2.4 DIRECT Constraint Handling

In order to handle variable constraints one of two methods may be used, a penalty function or the neighbourhood assessment strategy (NAS), (Dir, 2004).

When using the penalty function points with a penalty are evaluated as:

$$f(x) + \mu^T \max(g(x),0) \tag{4.8}$$

where μ is a penalty parameter that is individually specified for each constraint and $g(x)$ is formulated generally as:

$$g(x) \leq 0 \tag{4.9}$$

Thus the object function receives a penalty when a constraint is violated, $g(x) \geq 0$. The use of a penalty function requires knowledge of the object- and constraint functions to properly decide the penalty parameters. The use of penalty function would also result in the possibility of using a soft constraint. By setting the penalty value

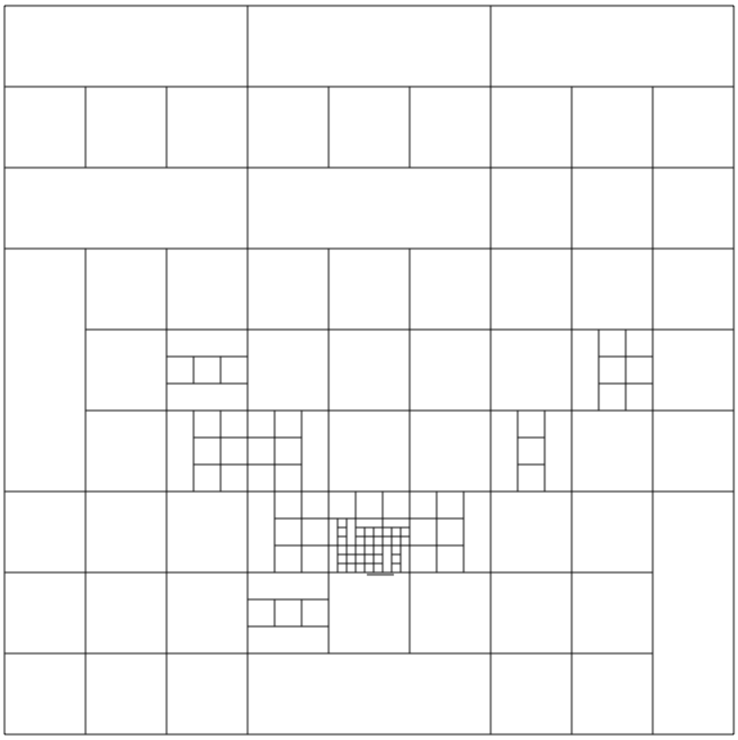

Figure 4.7: Example of search space subdivision in two dimensions after more than 100 iterations, inspired by (Finkel, 2003)

low, violation of the constraint may to some extend be allowed as a tradeoff of e.g. higher seas for smaller fuel consumption. The constraints would have to be hard constrains for land by setting the penalty parameters high.

5 Development of Weather Routing Tool

The program is structured such that through a main script the optimisation is called. Modules may be added through calls in the main script e.g. constraints depending on the complexity of the case and desired output from the user.

The program may be seen illustrated in the form of a flow chart in Figure 5.1.

The user must input the following ship specifics:

- Number of intermediate points
- Starting position and time
- End position and time
- Ship identifier for calculation directory
- Draught
- Trim

As an input for the DIRECT algorithm an object function is specified as the total fuel consumption of the entire journey and thus represents a minimisation problem. The free variables optimised are locations of intermediate points on the route and speed profiles for journey legs. Constraints may be added through functions giving positive values for violations. If a constraint is violated the represented value is multiplied with a penalty parameter leading to an increase in fuel consumption calculated through the object function.

All produced code may be found in Appendix B.

5.1 Basic Program

The ship specific data is not handled directly in the program as it is an integrated part of the calculation software MSPS, provided by MMT. Calls relating speed to

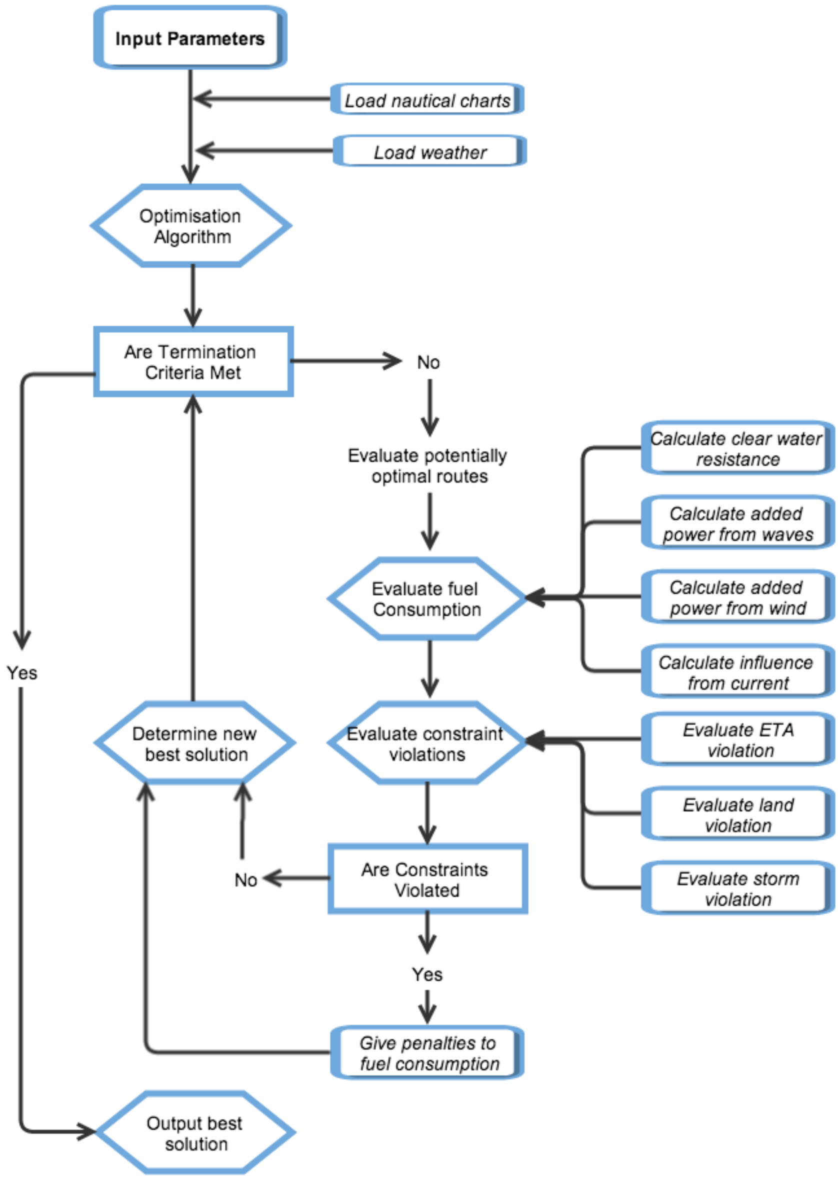

Figure 5.1: Flow chart for routing tool

fuel consumption and added resistance due to wind and waves are through MSPS ship specific as the calls requires an identifier for the vessel in the calculation library.

The provided software relates the ship speed to fuel consumption taking into account waste heat recovery, the use of axle generator and much more and thus gives a very thorough modelling of the ships' capabilities utilising years of collected data from the fleet.

Nautical charts with water depths as well as weather information is loaded into the program as a full environment. The charts are represented as a matrix with a resolution of 1° for both latitude and longitude. As the weather data is dynamic with time it is represented in 3D matrices. A layer in the matrices represents the weather in a time interval of 6 hours as a standard, and with a spatial resolution of 1° as for the nautical charts. The handling of environmental data has been made generic and can thus handle resolutions differing from the preset. The computation time is though highly dependent on both time interval and spatial resolution.

Great circle navigation is handled through functionality ready built in to the MatLab mapping toolbox.

Contrary to most routing algorithms the DIRECT algorithm does not require gridding of the searched space as solutions to the minimisation problem are sought in continuous space for both latitudes, longitudes, and speed profiles. This in turn is one of the strengths of the algorithm as creating a grid with capabilities of handling the dynamics of encountered weather properly may present a challenge. In this program only the possible search area is limited to relevant oceans and land.

As a global minimum is virtually impossible when searching in a continuous space, the optimisation does not terminate when a global minimum is obtained, but rather when a preset number of maximum iterations of function evaluations is achieved.

For each iteration the algorithm performs several function evaluations where centres of hyper rectangles, i.e. sampled routes are tested and their total fuel consumption calculated in order to evaluate possible optimality.

5.2 Free Variables

The free variables optimised in the program are latitudes and longitudes for intermediate waypoints and the entrance speed to each intermediate point as well as the end point. To develop the conceptual model using the DIRECT algorithm speed profiles have been used rather than preferable power profiles in order to avoid having to time step to intermediate way points to take into account arrival times considering influences from weather. This will in turn lead to a much simpler and faster program ideal for proving the concept. Further, the computation time calculating speed to power is less than one hundredth of the time required going from power to speed, and thus adds to the total required computation time.

The free variables are in the program represented in a matrix, *pointIdentifiers*, where latitudes, longitudes, speed, and ETA are represented column wise. Included in the matrix is information on start and end points even though these are fixed. The use of entrance speed to intermediate points has been selected rather than exit speeds as this in turn allows for en route optimisation.

Using n intermediate way points leads to an equal amount of free latitudes and longitudes and $n+1$ free speeds as the entrance speed to the end point is also optimised. This in turn leads to $n \cdot 3+1$ free variables and same amount of searched dimensions, limiting the used amount intermediate way points through resulting computation time.

To speed up optimisation the solution space has been simplified through the assumption that the intermediate points must be "chronological", in the sense that the ship will not travel back and forth longitudinally. This assumption has been made to limit the search space for each space for tested routes traveling mainly through longitudes. The longitudinal search space has thus been divided into equal intervals between start and end points.

5.3 Object Function

For the routing algorithm the object function may be formulated as:

$$\min\ totalFuel \tag{5.1}$$

where *totalFuel* is defined as the sum of fuel used for each route leg. The fuel consumption for a journey leg is defined as the calculated time spent on a journey leg multiplied with the specific fuel consumption calculated with added or reduced power due to influences of weather:

$$totalFuel = fuel(A,p_1) + fuel(p_1,p_2) + ... + fuel(p_n,B) \tag{5.2}$$

As calculating the influence of weather is computationally more demanding than calculating the fuel consumption with no influence of weather, the object function has been constructed such that the influence of weather is only calculated if the calculated ETA is within a pre-specified interval of time from the specified ETA. For a case with a sailing time of 360 hours, this value could e.g. be ± 5 hours.

The calculation is this way divided into two functions depending on the ETA. A simple fuel calculation that does not take into account the influences of weather, *simpleFuelCalculation*, and one that does, *complexFuelCalculation*.

When calculating *complexFuelCalculation*, firstly a 3D matrix is created based on the sampled route, *WP3D*, short for way point 3D. The matrix is based on sampling in each time interval using a preset of 20 points for each interval. The latitude and longitude corresponding to the sampled time in said interval is rounded to full latitude and longitude complying with the grid resolution used throughout the program. In the corresponding time-space field in WP3D a 0 is then replaced with a 1. This way the route is mapped out as 1's in a zeroed 3D matrix.

WP3D has the same dimensions as the imported weather environment and the matrices may thus be element wise multiplied, this both for speeds, directions and frequencies. The result being 3D matrices containing only relevant weather information. As the remainder of the elements are 0's, and mean values are used based on differences from 0, a small correction of 0.01 is added by adding $WP3D * 0.01$ to the computed relevant matrices.

The additional resistance due to the influence of the weather is then determined and recalculated to additional required power to sustain the set speed of each time step and meaned for each journey leg to compute the additional fuel consumption. In pseudo-code looking like:

if $setETA - offTime \leq calculatedETA \leq setETA + offTime$
  $fuel = complexFuel$
    create WP3D
    calculate added power for each time step
    add mean of added power for journey leg to fuel calculation
    calculate total fuel consumption based on speed, additions and time
  else
  $fuel = simpleFuel$
    calculate total fuel consumption based on speed and time
end

All cases have been tested using the fuel consumption models provided by MMT, referred to as complex fuel calculation model. Additional models have been tested to determine the robustness of the conceptual program using; an exaggeration of the MMT provided model, where the required power has been squared to magnify the effects of changes in speed, this model is referred to as exaggerated fuel consumption model. Further, tests have been carried out on fuel consumption models based on regressions to the provided fuel consumption model, using relations of V^3, $V^{3.5}$ and V^4, as the relation to speed is for container ships known to be in this area.

5.4 Constraints

The constraints for the optimisation are handled as functions of the form:

$$g(x) \leq 0 \tag{5.3}$$

When a constraint is violated, $g(x) > 0$, a penalty is applied to the object function value on the basis of the violation:

$$f(x) + \mu \max(g(x),0) \tag{5.4}$$

Thus artificially adding to the total fuel consumption. A solution that violates a constraint if deemed infeasible. The function value is still evaluated as it is used to determine potential optimality in further iteration steps. Thus the exact value of a penalty only needs to adjusted so that the algorithm converges towards a feasible solution. Setting the penalty too small will result in the algorithm finding best solutions in an infeasible space. Setting the penalty too large will result in the algorithm having to spent more function evaluations to find a solution close to boundary, e.g. for arrival exactly on time. This way even though it is intuitively worse for the route to be planned crossing land than arriving on time the penalties only work as a tool for the convergence of the algorithm.

5.4.1 Estimated Time of Arrival

The ETA is calculated through the $pointIdentifiers$, through the distances between points and speeds, thus the arrival time at the end point. As arriving late may to some degree be accepted as a tradeoff on the fuel consumption a variable, *offTime*, is optional in the program. A soft penalty is added to violating this additional time upon arrival. As arrival times exceeding this are without interest as solutions, a hard penalty is added for violation.

5.4.2 Land/Shallow Water

As the search space for each point identifier is designed individually for the spatial variables, x, y, as rectangles, a result will be that the search space may overlap land. The route is sampled and where the water depth limit is violated a 1 is added to a violation sum. The violation is further related to the distance traveled over land. This value is used as the function $g(x)$ in Equation (5.4). The penalty is hard as no tradeoff is accepted for fuel consumption.

5.4.3 Safety

As the simplifications to the safety limits outlined in Section 2.2.1 do not for this specific ship lead to violations except for very extreme weather conditions instead a maximum wave height is introduced. A violation is calculated similarly to the method used when determining added resistance due to the influence of weather.

Optimally safe operation should be determined based on analysis of ship response related to maximum accelerations and roll angles to ensure both cargo safety and crew safety and comfort. Such analysis is beyond the scope of this project.

5.5 Pre-processing

In order to speed up the program a route is optimised without considering speed profiles as this may be done very fast using DIRECT. Based on the total distance the average speed considering only land avoidance may be calculated. The search interval for the speed profiles is then based on this average speed and the maximum speed so that the average speed is the average value of the minimum and maximum speeds. By doing so solutions using this speed are tested early on as the hyper rectangles are divided into three equals and their centre values used for evaluation.

The user may select using a simpler version of the *simpleFuelCalculation* to have the program run faster. If this option is selected a pre-processing is initialised where the speed to fuel consumption is fitted to a simple equation which may then be run inside MatLab. This will not directly affect a final feasible solution as the complex fuel calculation is run for feasible solutions.

6 Demonstration of Weather Routing Tool

The purpose of this chapter is to verify and validate the capabilities of the produced routing tool. Several conceptual cases have been constructed with the purpose of determining the different capabilities: shortest path, land avoidance, storm avoidance, effect of current, wind, and combinations. The results of the tests are described in the following.

All conceptual models have been made to retrieve four intermediate way points between the start and end points of the route and the four speeds entering the intermediate points and the speed entering the end point. This leads to 13 parameters, and thus 13 searched dimensions. Increasing the number of searched intermediate way points increases the requirement for function evaluations and algorithm iterations drastically.

To simplify calculations there has not been distinguished between eastern and western longitudes and rather the degrees range from 0-360. Based on this the routing is done between, $\{latitude, longitude\}$, $\{25, 118\}$ and $\{3, 257\}$, with an arrival time 380 hours after departure.

Figure 6.1 shows an example of how the points are placed to form a route. Start and end points have also been plotted as the most easterly and westerly points. It should be noted that all cases displayed in the following section have met the constraints set up.

The optimisations have been run with termination after 50,000 function evaluations to ensure a proper convergence to a best solution is achieved. The optimisations could though for the most cases have been terminated much earlier. An example of the convergence of and optimisation may be seen in Figure 6.2.

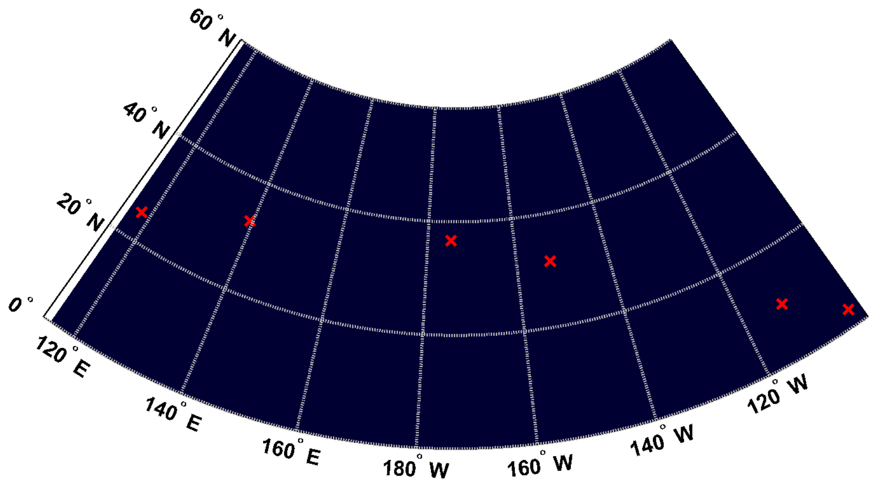

Figure 6.1: Example of placement of points to form a route.

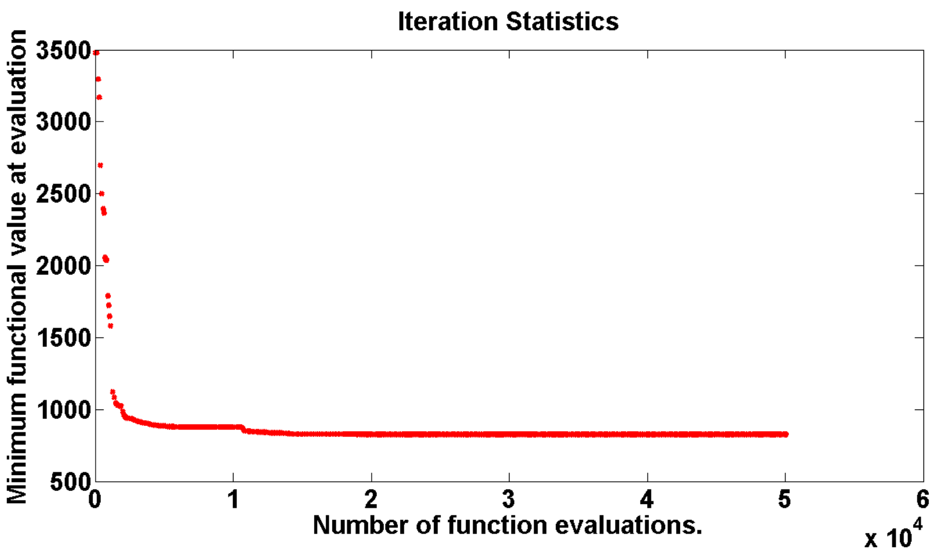

Figure 6.2: Example of convergence of a solution with termination at 50,000 function evaluations.

6.1 Shortest Route

The conceptual model of the shortest route has been tested in order to determine the algorithm's capabilities of finding the shortest great circle route between start and end points. With no interference of weather or land the way points should lie directly on a great circle line between the two points and the five speeds should be of the same magnitude giving a uniform speed profile. Lastly, the ETA should be met.

As mentioned the different fuel consumption models, described in Section 5.3, have been tested for this case in order to determine the baseline of their individual capabilities.

In order for the program to search more intelligently and thus save time for computing a solution, the speed interval searched has been modified to be case specific. This has been done through a preprocessing where the shortest route is computed without evaluating speeds. The optimum speed is then calculated as the total distance of the voyage divided by the designated voyage time. With known maximum speed, the minimum speed is set so that the optimum speed is mean of maximum and minimum. This has been done due to the rectangle divisions the DIRECT algorithm performs, where midpoints of the rectangles are evaluated at the initial stages, thus solutions using the optimum average speed are tested at an early stage.

The results have been benchmarked against the optimum solution calculated as the direct great circle route between start and end points. Table 6.1 shows the results of the testing. The first column shows the fuel consumption model used, the second column shows the distance of the produced best route. The third column shows the difference from this distance to the optimum distance. As the different models compute completely different fuel consumptions they may not be directly compared, instead the last column shows the difference in fuel consumption given the speed had been uniform using the same model, e.g. the best route produced using the exaggerated fuel consumption model has a 0.32% higher fuel consumption than for the same route given a uniform speed profile for the journey.

Table 6.1: Results from shortest route conceptual optimisation

Fuel cons. model	Distance $[nM]$	Difference [%]	Diff. from opt. fuel cons. [%]
Optimal	7882.3	-	-
Complex	7882.4	0.002	0.00
Exaggerated	7882.5	0.003	0.32
V^3-regression	7882.4	0.001	0.01
$V^{3.5}$-regression	7882.4	0.002	0.01
V^4-regression	7882.3	0.000	0.01

For this test case only the ETA penalty has been applied as a constraint. The penalty parameter has only been slightly adjusted for the solution to converge. From Table 6.1 it may be seen that all distances are close to that of the great circle route. The route produced using the V^4-regression fuel consumption model may be seen in Figure 6.3. In the figure the optimum great circle route is plotted as magenta points. It may be seen from the figure that the routing tool is capable of finding the great circle shortest route.

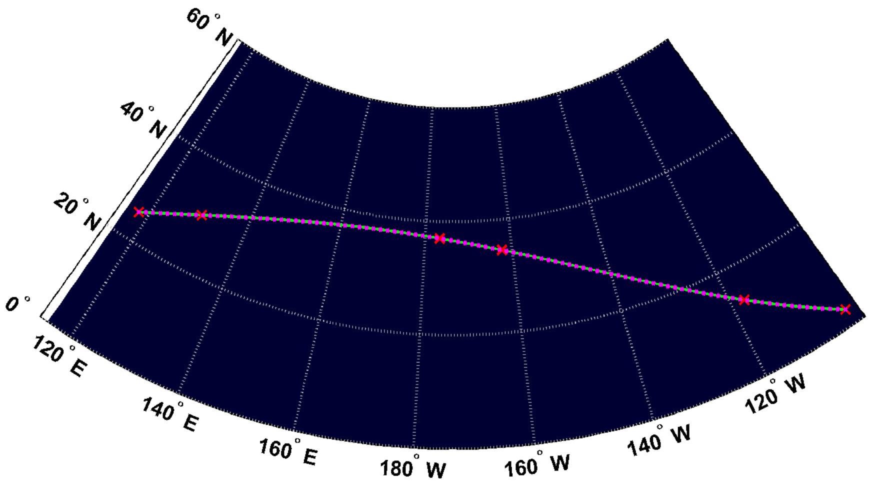

Figure 6.3: The optimised shortest route, using the V^4-regression fuel consumption model. The magenta points display the optimal great circle route.

The produced speed profiles are plotted against distance traveled in Figure 6.4. Optimally the speed profiles should be of same magnitude and as may be seen from Figure 6.4 this is reasonably obtained for the three regression models. The complex and exaggerated fuel consumption models show less capability of producing uniform speeds. Referring to Table 6.1, this has for the complex fuel consumption model no influence on the fuel consumption. This is due to the nature of the model allowing a tradeoff between speeds.

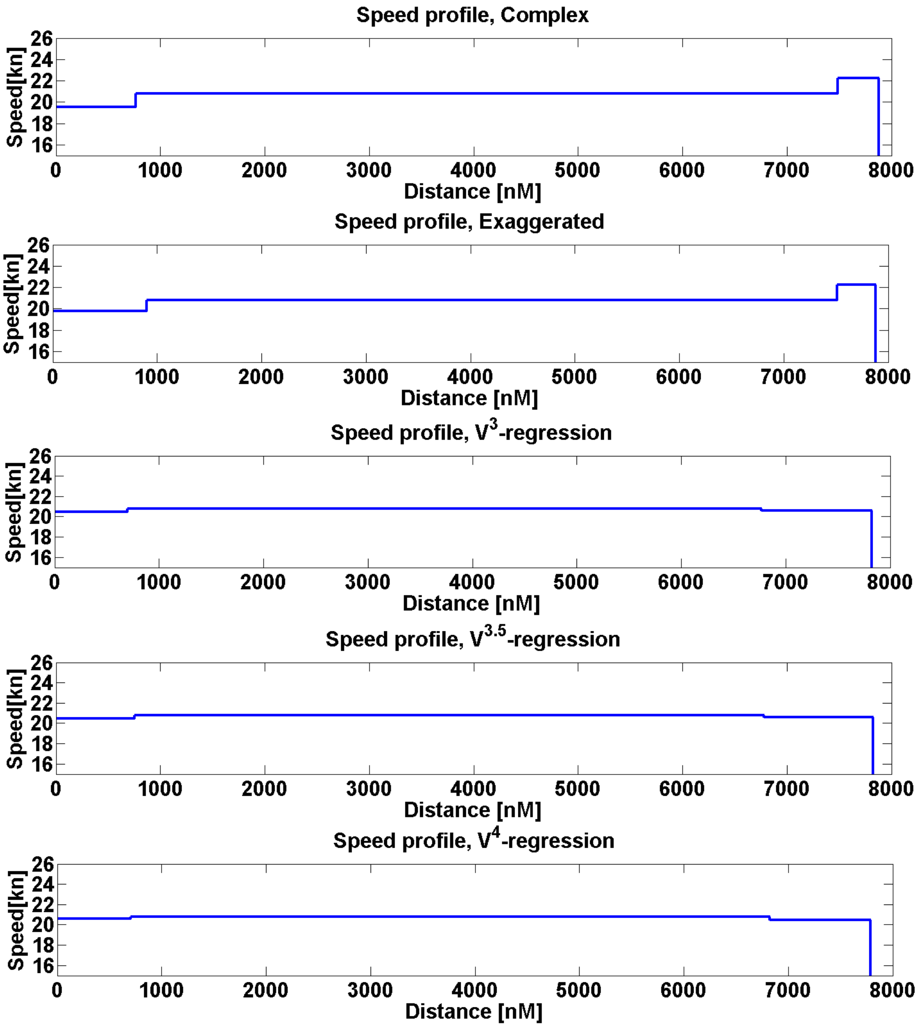

Figure 6.4: The produced speed profiles using different fuel consumption models in a clean test environment.

6.2 Land Avoidance

For the program to route around land an additional penalty is added. Thus two constraints are applied for this test case. The case has been set up so that a patch of land is placed over the great circle route produced in the case for shortest the route, as may be seen in Figure 6.5. The great circle route is plotted in as magenta points. Thus an optimal route will be the combination of the great circle route form the start point to the upper right corner of the island and the great circle route from there to the end point. The speed should optimally be uniform as for the previous case.

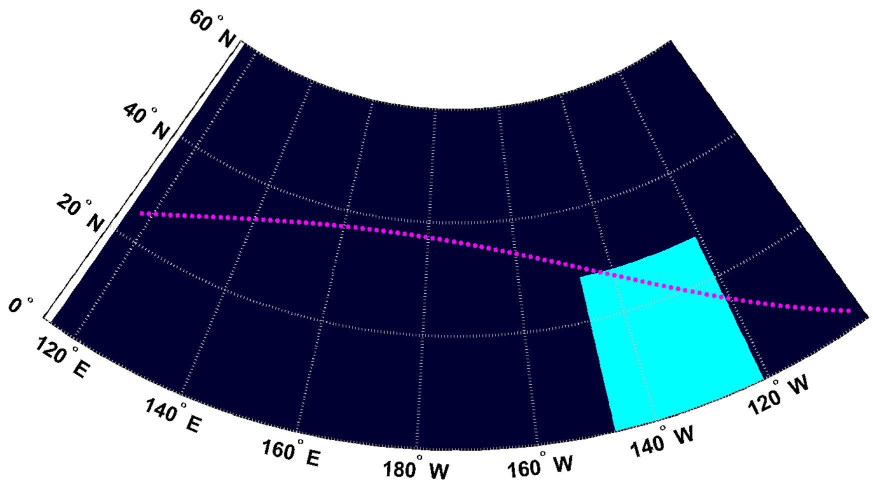

Figure 6.5: The test case with a patch of land placed as an obstacle for the great circle route, including the shortest route when no obstacles are included

As for the shortest route the penalties for ETA and routing over land have only been slightly adjusted to give convergence. No further medium or fine tuning of the penalties have been performed.

The results of the optimisation may be seen in Table 6.2. As for the previous case, the routing tool performs well at finding the shortest route, with the largest deviation being for the $V^{3.5}$-regression fuel consumption model deviating with 0.11%. As for the previous case the fuel consumption has been related to the optimal fuel consumption given a uniform speed profile. The tested cases show good performance in fuel consumption except for the exaggerated fuel consumption model having 1.37% higher fuel consumption than that for uniform speed. Similar to the previous case the complex fuel consumption model leads to a route with optimal fuel consumption, again this is related to the nature of the model allowing for a tradeoff between speeds.

The produced route using the V^3-regression model and the optimal route may be seen in Figure 6.6. As expected from the produced distances, the route follows the optimal great circle route.
Figure 6.7 shows the produces speed profiles using the different fuel consumption models. As for the previous case the complex and exaggerated models have shown

Table 6.2: Results from routing around a patch of land

Fuel cons. model	Distance $[nM]$	Difference [%]	Diff. from opt. fuel cons. [%]
Optimal	7946.9	-	-
Complex	7954.4	0.09	0.00
Exaggerated	7955.1	0.10	1.37
V^3-regression	7949.1	0.03	0.08
$V^{3.5}$-regression	7955.5	0.11	0.01
V^4-regression	7953.6	0.08	0.21

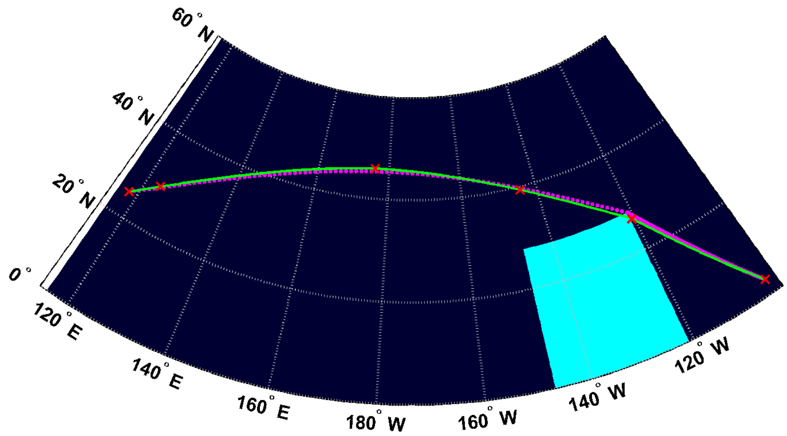

Figure 6.6: Optimised route avoiding land using the V^3-regression fuel consumption model.

to be the least capable of producing a uniform speed profile for the journey and the regression models performing significantly better.

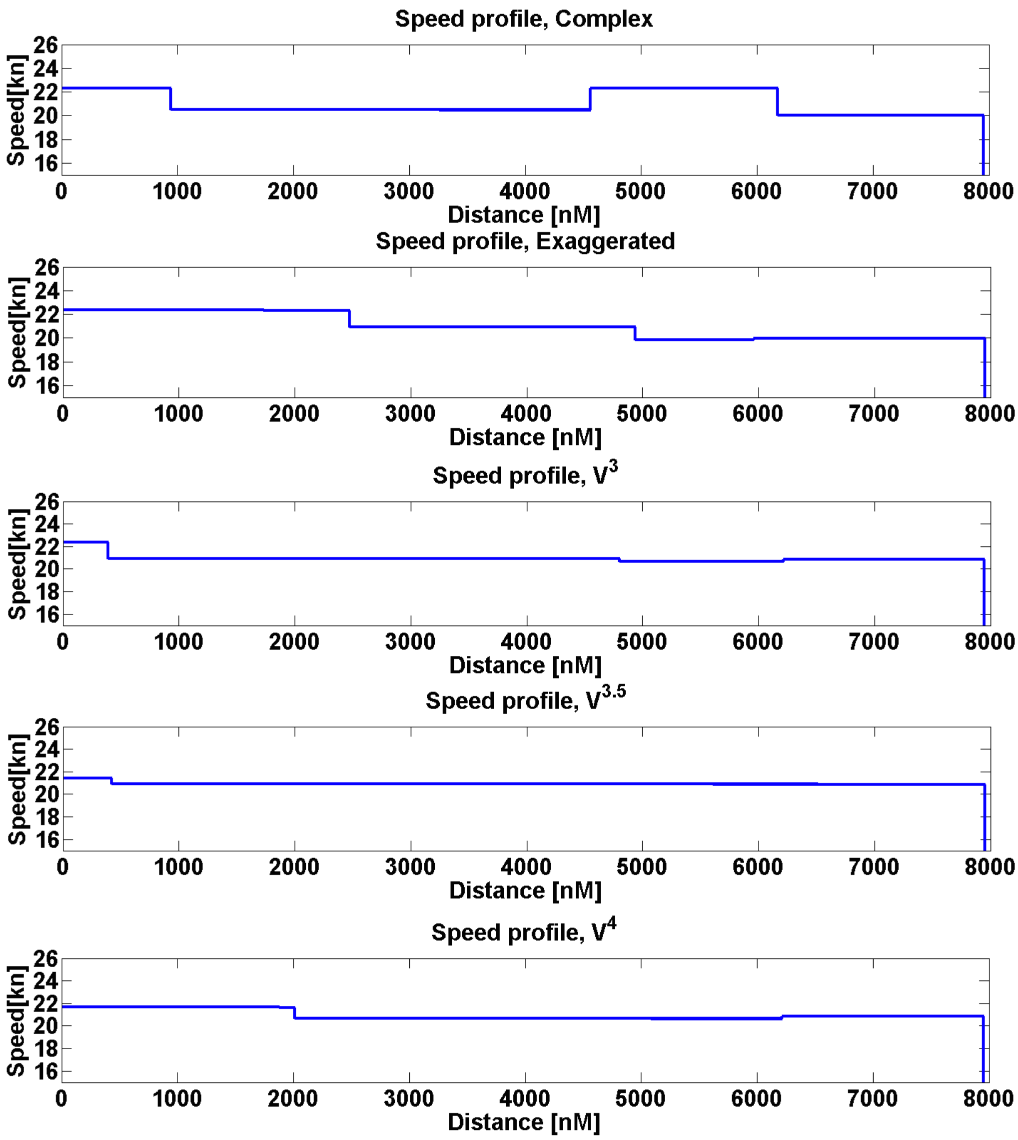

Figure 6.7: The produced speed profiles using different fuel consumption models, avoiding land

6.3 Storm Avoidance

In order to determine the capabilities of speeding up and slowing down as a method of avoiding storms a vertical strip with a wave height exceeding the maximum allowed has been introduced to a clean environment. The storm appears at hour 192 and disappears at hour 270 between longitudes 176W and 151W. With the storm not present the ship will sail through this area during this period, thus in order to avoid the storm it will have to either slow down in the beginning of the voyage entering the stormy area just when the storm has passed and then speed up to arrive on time or speed up exiting the stormy area as the storm hits and then slowing down.

As the environment for this case is dynamic, sampling must be done differently than when avoiding land which is only two dimensional. For this the three dimensional representation of the route as ones in a matrix of zeros is used, with latitude, longitude and time steps of six hour giving the dimensions.

For this case a max wave height constraint is added to the optimisation. Thus two constraints are used including the ETA. As for previous cases the penalties have only been slightly tuned to give convergence but neither medium or fine tuning has been carried out. For this test case it is not possible to determine a generic optimal solution applicable for all the fuel consumption models. This is due to the optimisation weighing off the effects of route length and required speeds. There is in this way a trade off between the two, leading to different routes and therefore speeds and distances. Table 6.3 shows the route distance for the tested fuel consumption models.

Table 6.3: Results from storm avoidance

Fuel cons. model	Distance $[nM]$
Complex	8010.0
Exaggerated	8014.6
V^3-regression	8012.8
$V^{3.5}$-regression	8009.9
V^4-regression	8009.3

It is seen from the table that even with different fuel consumption models the distances are close to being the same, as may also be seen from Figure 6.8 and zoomed in on Figure 6.9.

The produced speed profiles may be seen in Figure 6.10. It is clear that the ship must speed up in the beginning of the route to avoid the stormy area and must slow down after exiting the stormy area. Optimally the speed should be uniform until the ship is out of the storm and uniform at a lower speed after exiting the area. This is

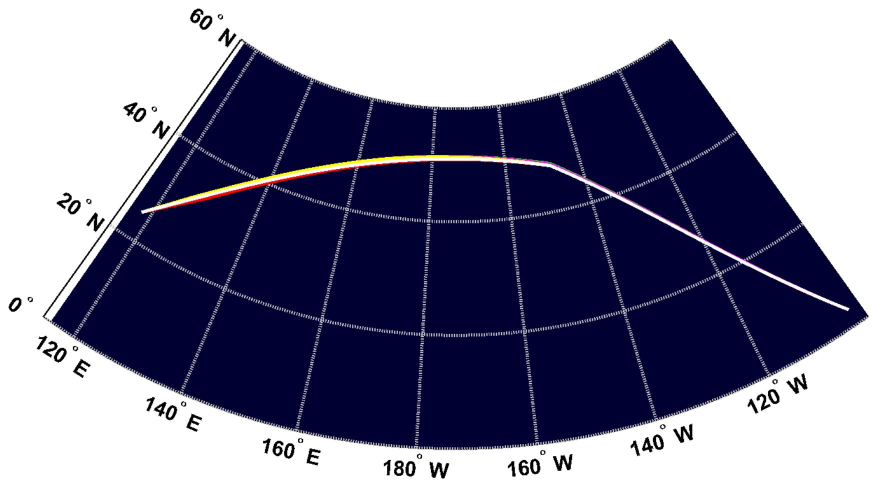

Figure 6.8: Produced routes avoiding storm. Fuel consumption models: complex in red, exaggerated in green, V^3 in magenta, $V^{3.5}$ in yellow, and V^4 in white

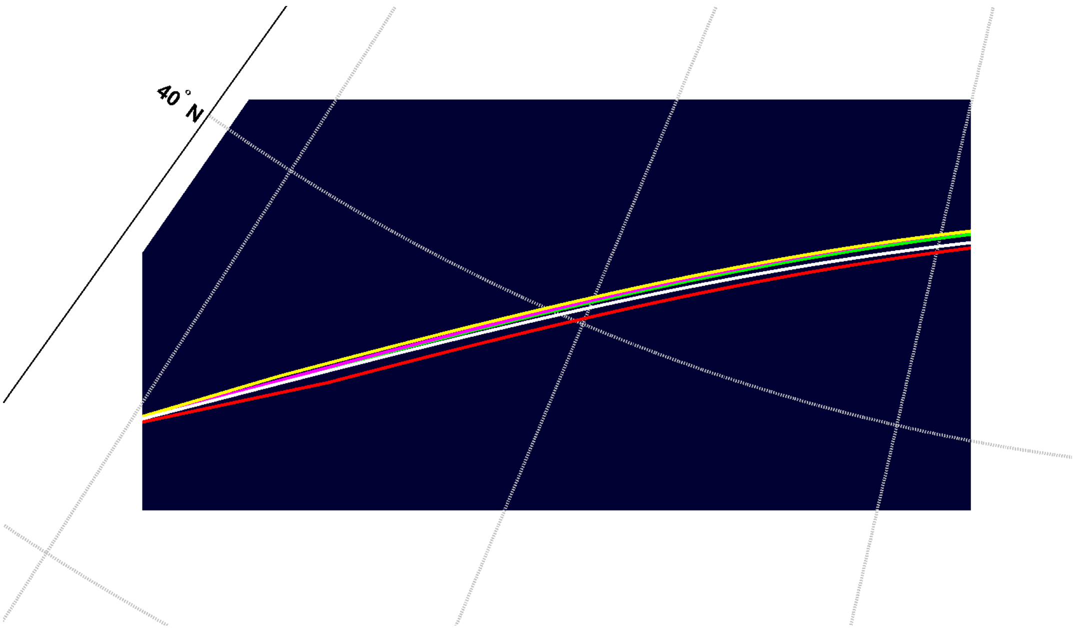

Figure 6.9: Zoom of produced routes avoiding storm. Fuel consumption models: complex in red, exaggerated in green, V^3 in magenta, $V^{3.5}$ in yellow, and V^4 in white

to some degree achieved for all the models. The best results are seen for the $V^{3.5}$- and V^4-regression models.

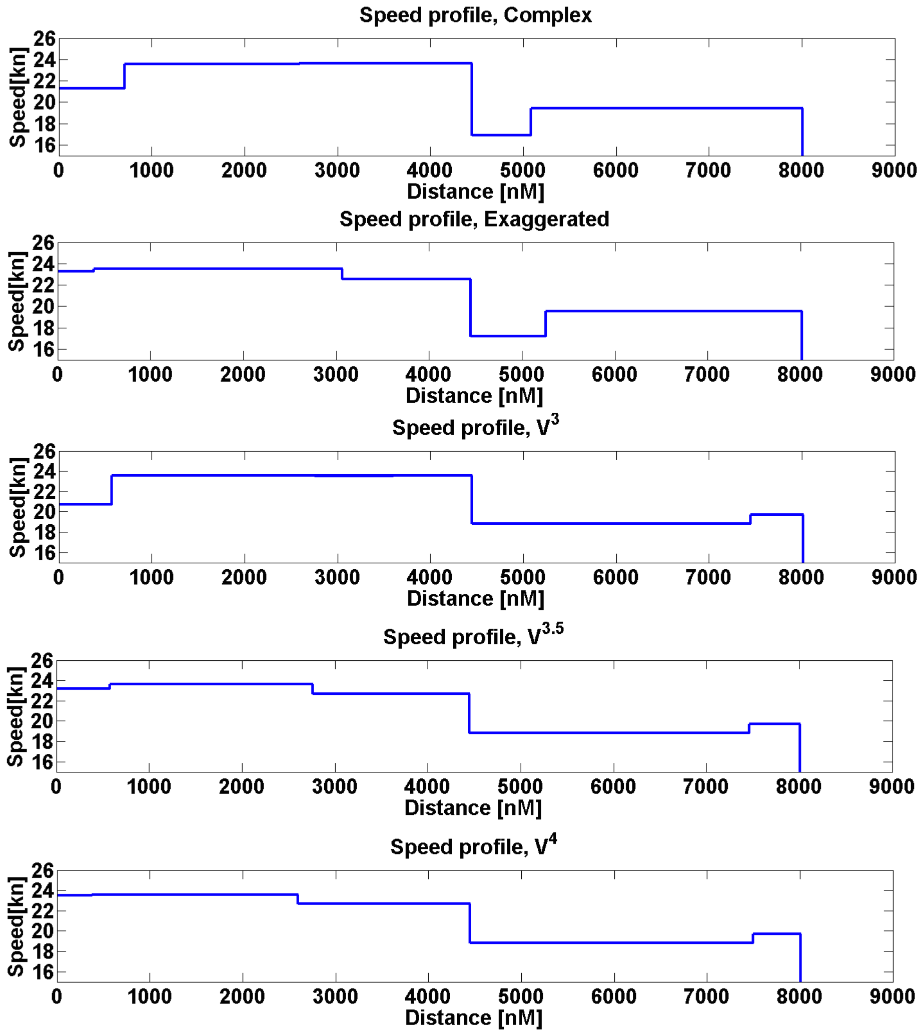

Figure 6.10: Speed profiles avoiding storm

Figure 6.11 shows the route produced using the complex fuel consumption model and Figure 6.12 using the V^4-regression fuel consumption model. In both figures the part of the route shown in blue is the distance traveled at the hour the storm hits and the green part is the remaining part of the voyage. The area shown in brown is the stormy area.

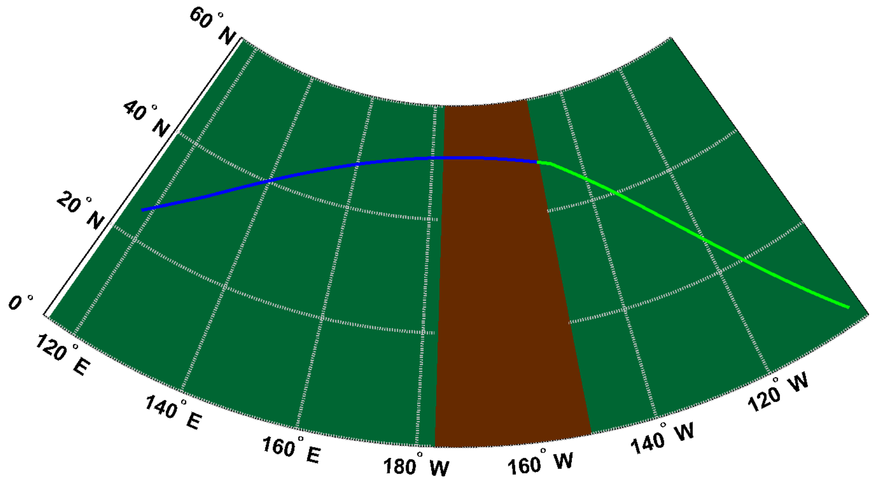

Figure 6.11: The optimised route avoiding storm using the complex fuel consumption model

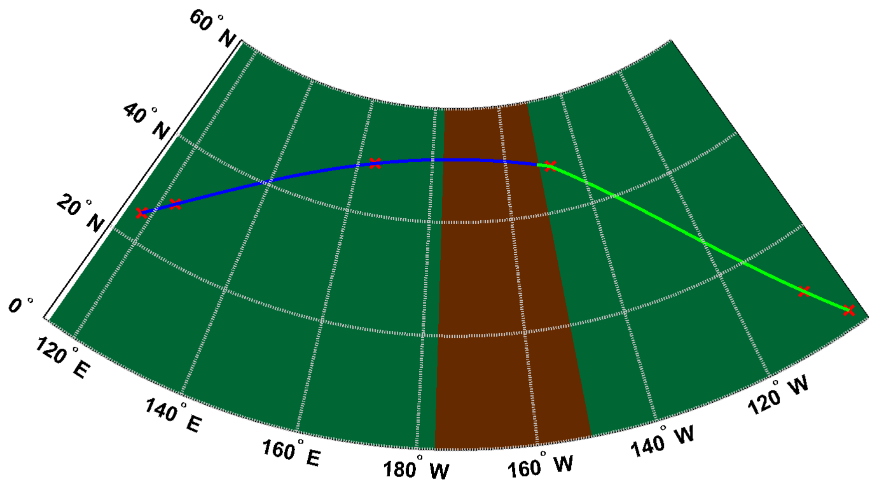

Figure 6.12: The optimised route avoiding storm using the V^4-regression fuel consumption model

6.4 Currents

To test whether the program is able to take into account the effects of currents a test case has been set up with a stripe of current going from a latitude of 18 to 25. The direction of the current is from left to right with a magnitude of 5m/s. This is a strong current but is meant for demonstration purposes. Given the program is able to utilise this current it is expected that the route will differ from the great circle and route closer to the current stripe.

The achieved distances using different fuel consumption models may be seen in Table 6.4. It may be seen that the distances differ very little.

Table 6.4: Results from current influence

Fuel cons. model	Distance $[nM]$
Complex	7972.9
V^3-regression	8005.8
$V^{3.5}$-regression	8005.8
V^4-regression	8032.2

Figure 6.13 shows the route produced using the complex fuel consumption model. It may be seen that as expected the route differs significantly from the great circle route and utilises the stripe of current for the majority of the voyage.

The speed profiles produced using the different fuel consumption models may be seen in Figure 6.14.

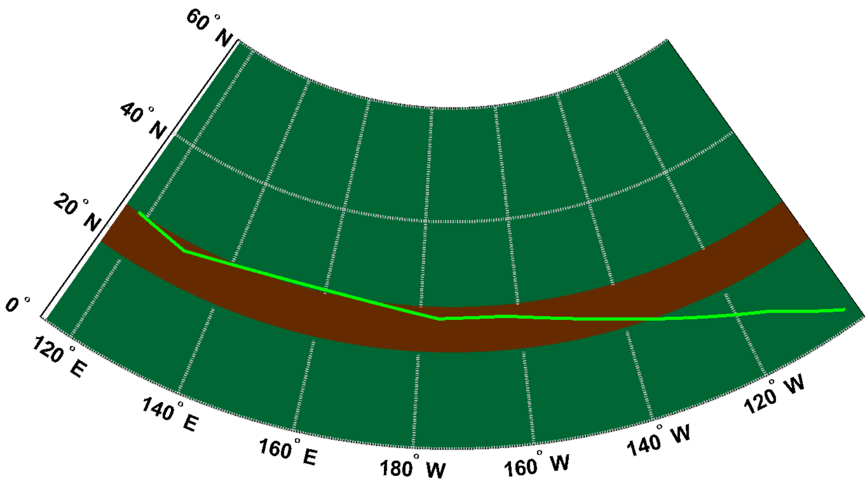

Figure 6.13: Produced route with influence from current, V^4-regression fuel consumption model

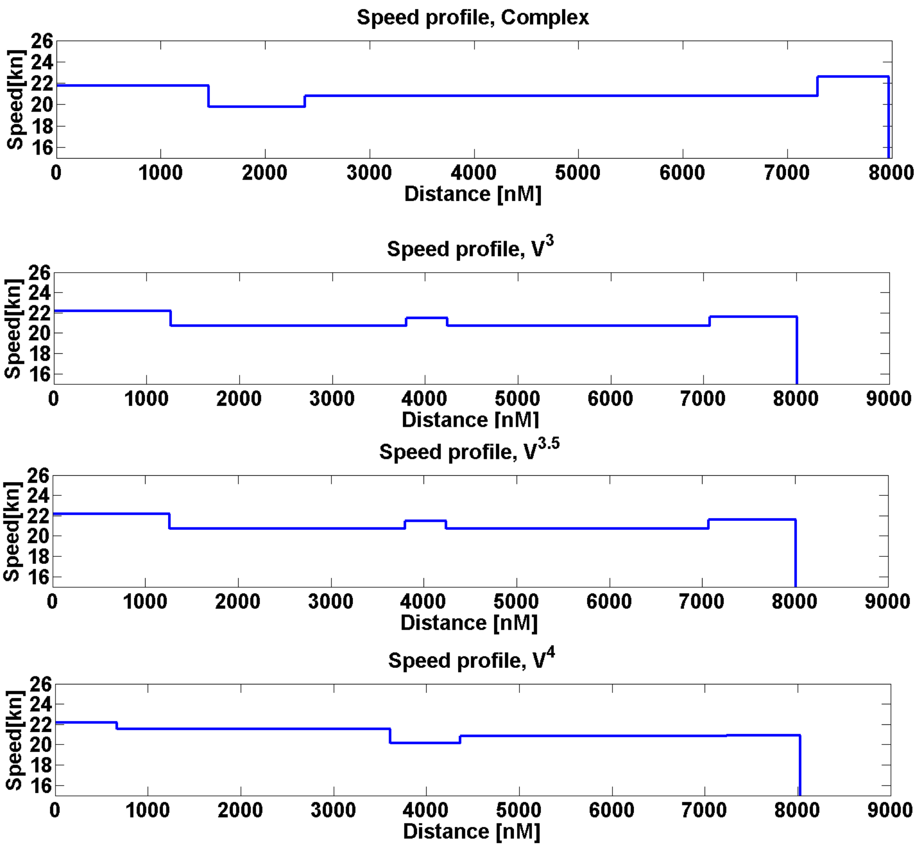

Figure 6.14: Speed profiles with influence from currents

6.5 Wind

To test the influence of wind a similar case to that of current is tested. A wind with a speed of 100m/s has been used with the same directionality as the current though ranging from latitude 18 to 27. The magnitude of the wind speed is due to test showing that a wind of just 50m/s does not have an effect on the produced route.

The distance of the produced routes may be seen in Table 6.5. The route produced using the complex fuel consumption model gives a longer route that those produced using the regression models. Plotting the route, Figure 6.15, does though show a route following the stripe of wind.

Table 6.5: Results from wind influence

Fuel cons. model	Distance $[nM]$
Complex	8002.5
V^3-regression	7955.7
$V^{3.5}$-regression	7955.7
V^4-regression	7955.4

The produced speed profiles using different fuel consumption models may be seen in Figure 6.16.

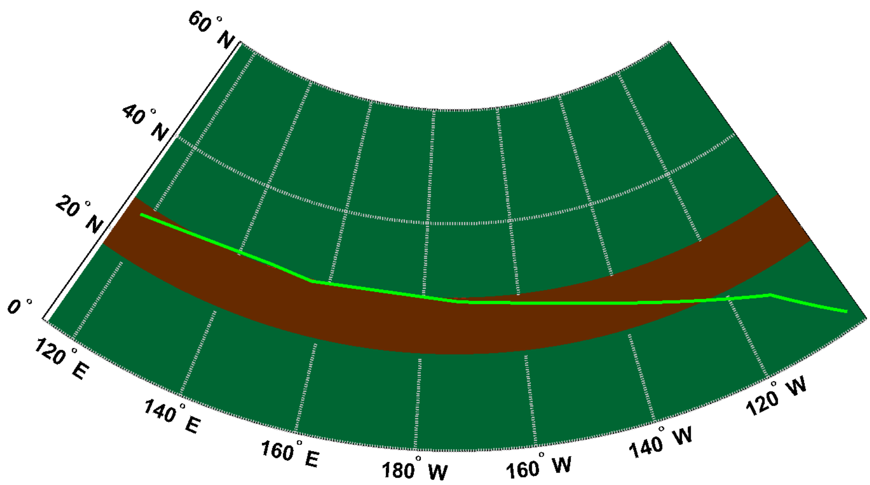

Figure 6.15: Produced route with influence from wind, complex fuel consumption model

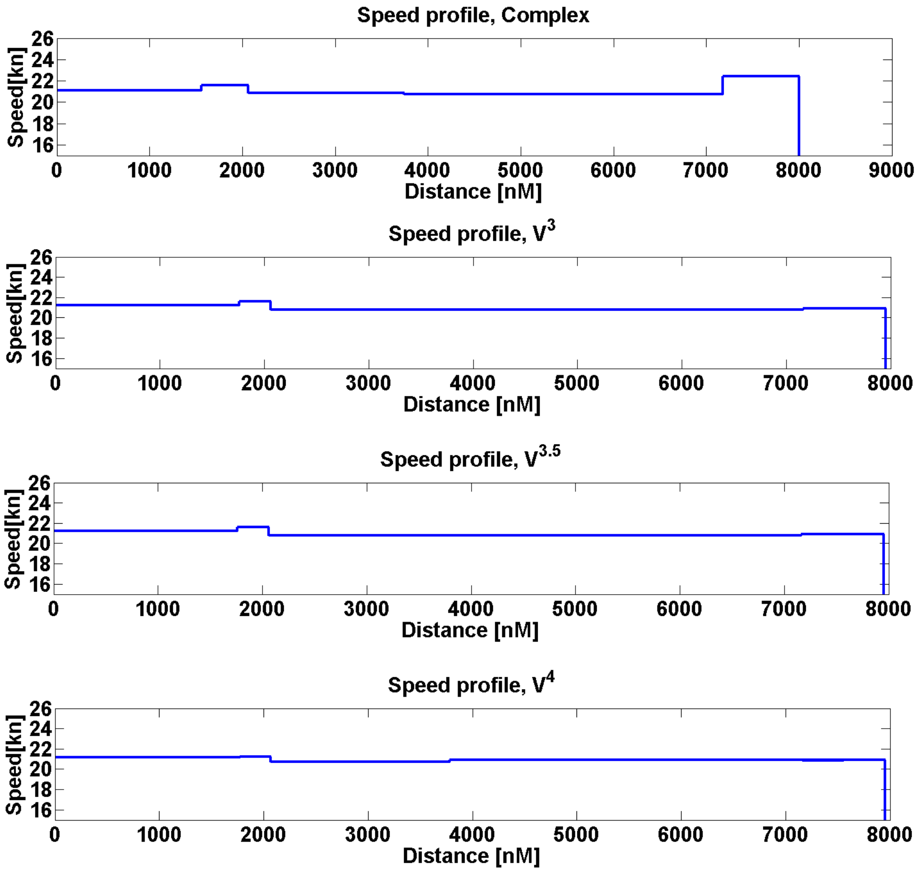

Figure 6.16: Speed profiles with influence from wind

6.6 Waves

The handling of the influence of waves has been set up but and does have an effect on the fuel consumption and the routing of the ship. Setting up a test case to properly illustrate the effects of waves was though not achieved and has therefore been omitted from the result section.

6.7 Land and Storm Avoidance

Having determined the capabilities of the routing tool concerning land and storm avoidance while arriving on time, the three are tested in combination. This is done using an environment that directly combines the environments used in land avoidance and in storm avoidance without modifications to the two obstacles. The resulting distances may be seen in Table 6.6. The same tendencies as for the case of storm avoidance may be seen with the complex fuel consumption model producing a route significantly shorter than the remaining models.

Table 6.6: Results from land and storm avoidance

Fuel cons. model	Distance $[nM]$
Complex	8009.4
Exaggerated	8022.0
V^3-regression	8009.3
$V^{3.5}$-regression	8019.4
V^4-regression	8012.9

The produced routes may be seen in Figure 6.17 and zoomed in Figure 6.18. The same tendencies may be seen here as those for storm avoidance only.

The speed profiles may be seen in Figure 6.19.

Figures 6.20 and 6.21 show the route produced using the V^4-regression model.

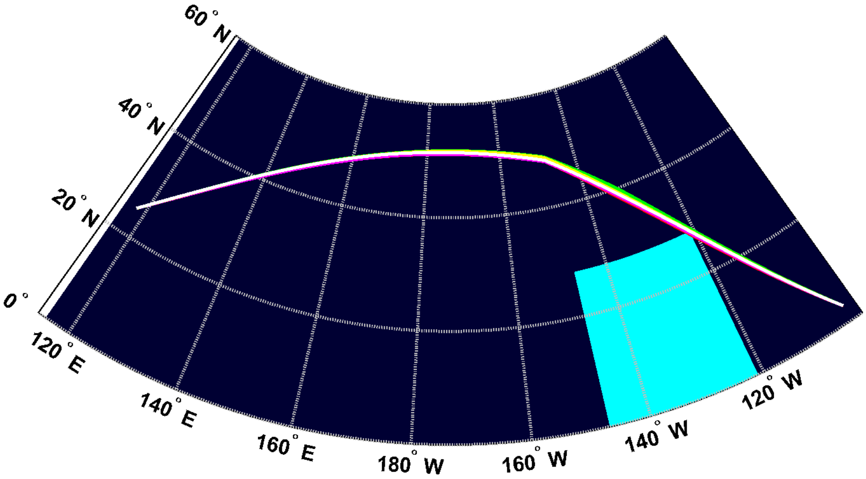

Figure 6.17: Produced routes avoiding land and storm. Fuel consumption models: complex in red, exaggerated in green, V^3 in magenta, $V^{3.5}$ in yellow, and V^4 in white

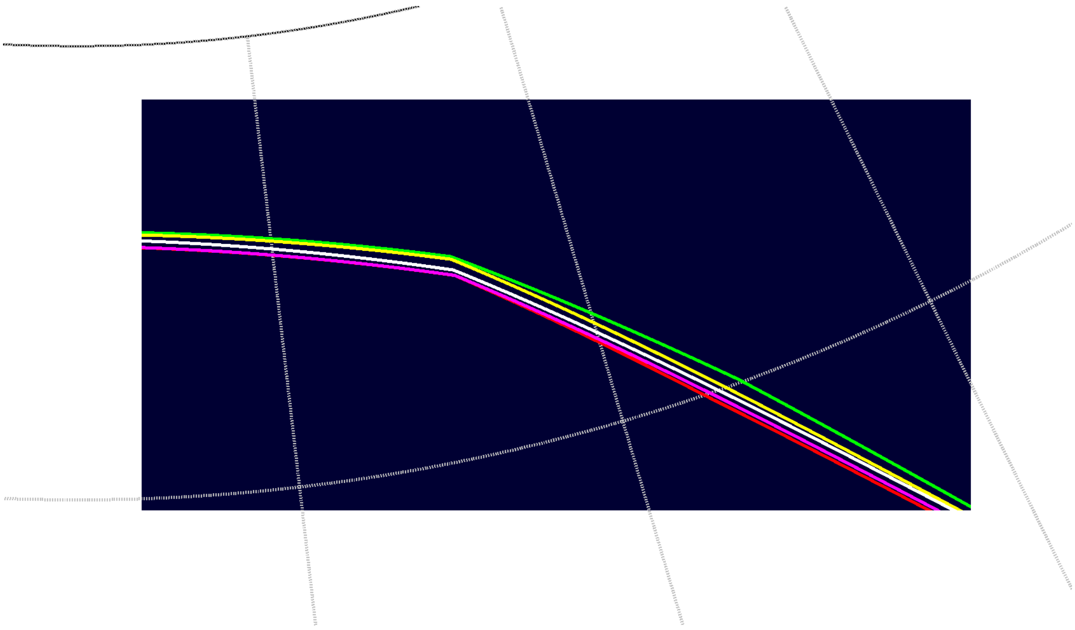

Figure 6.18: Zoom of produced routes avoiding land and storm. Fuel consumption models: complex in red, exaggerated in green, V^3 in magenta, $V^{3.5}$ in yellow, and V^4 in white

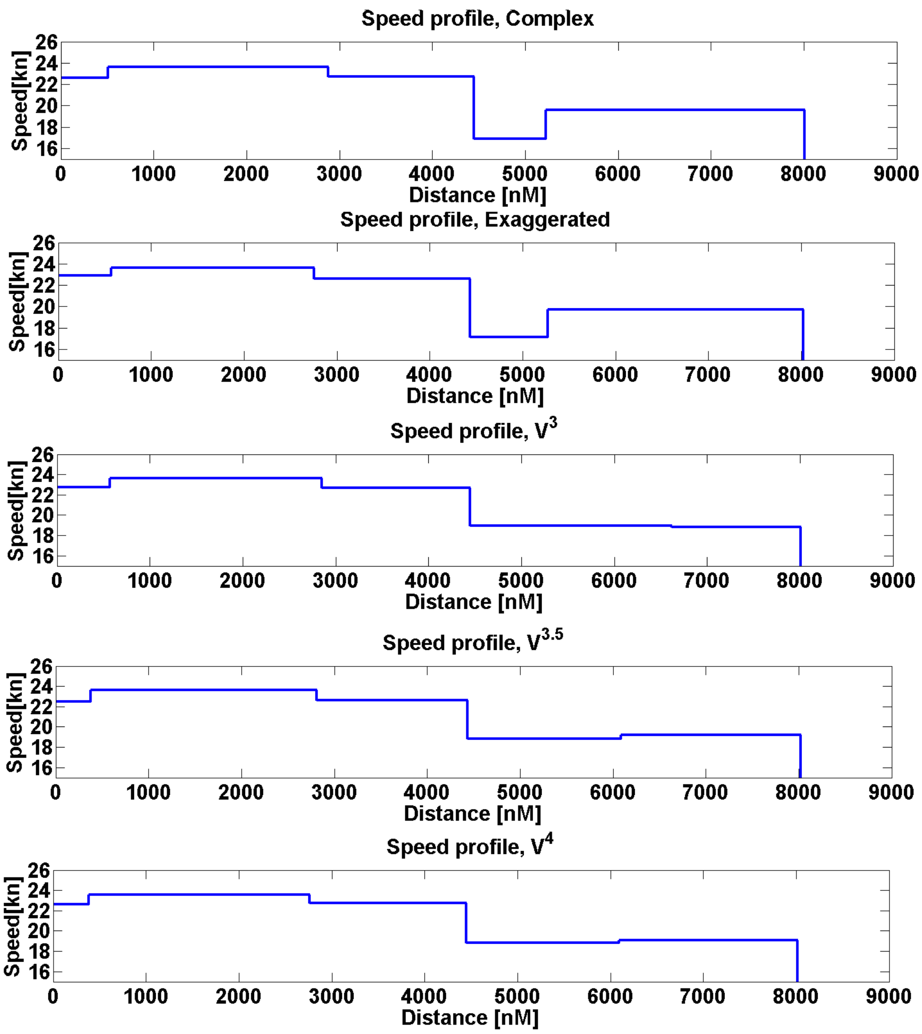

Figure 6.19: Speed profiles avoiding land and storm

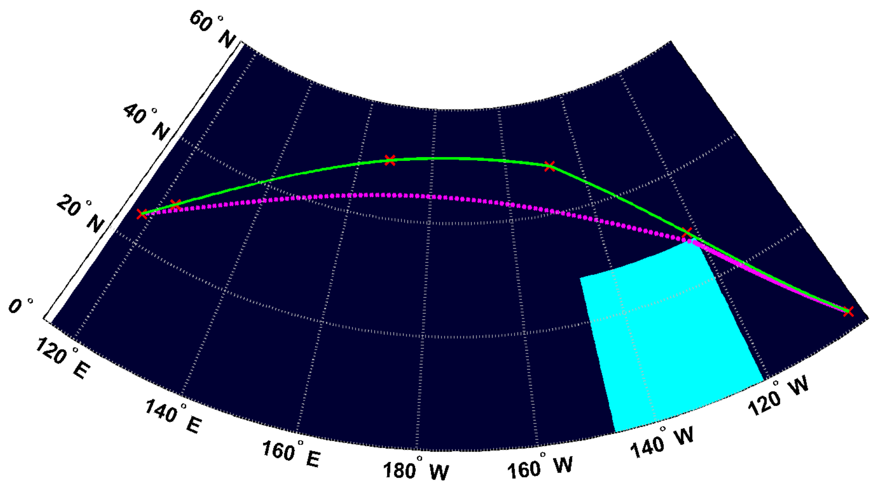

Figure 6.20: Produced route avoiding land and storm using the V^4-regression fuel consumption model, showing land. The magenta points showing the shortest path around land.

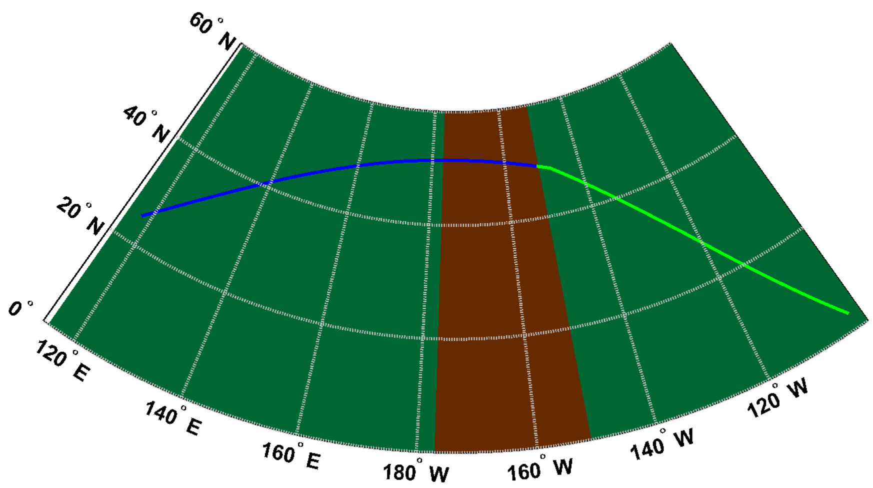

Figure 6.21: Produced route avoiding land and storm using the V^4-regression fuel consumption model, showing the storm. The part of the journey travelled when the storm hits being blue and the remaining part of the journey being green.

7 Conclusion and Future Work

Due to the difference in focus for the presented sections the conclusion and future work has been divided into state of the art, concept of weather routing, and program modelling and performance.

7.1 State of the Art and the Future of Routing

The mapping of the state of the art in weather routing has provided an understanding of the current requirements of a routing program. It has revealed the current trends in the market and where the market is currently moving, including increased focus on weather forecasts, their uncertainties and ship specific dynamics.

Weather routing is an important tool for the shipping industry when increasing fuel efficiency and reducing emissions are to be considered. In this thesis it is concluded that state of the art weather routing has evolved from merely being storm avoidance into multi-objective optimisation. The possibility of optimising speed or power profiles for the journey has become a requirement for the routing tools. It is found that state of the art routing software are able to suggest ship routes with optimum speed profiles while balancing cost, time of arrival, and ship and cargo safety. To become a forerunner many elements must be included but with capabilities of handling cases with sparse available information.

Special attention is given to optimisation algorithms used both in weather routing software and research. The conclusion is that the less advanced algorithms leaning towards brute force are phased out in favour of more advanced algorithms such as versions of 3D Dynamic Programming and Genetic Algorithm. These more advanced algorithms perform better when the optimisations are multi-objective and when computation time is limited. Increase insight to system models and the handling of the routing as a total voyage system will most likely lead way to even more advanced algorithms being implemented.

SECA areas are currently of great interest in weather routing as routing through these areas may directly influence fuel consumption. It was seen that some providers give penalties to routing through the SECA areas and some have them as no-go areas. It may be discussed what the intention of the SECA areas are. If not to keep emissions down in these specific areas alone but rather as an incentive to lower emissions in general the element of corporate social responsibility may be considered when determining how to handle the SECA areas in weather routing.

The handling of weather uncertainties with regards to both forecasts and the time after the validity of a forecast is evidently of great interest to weather routing, as it limits both the routing and the certainty of the produced routes. Further investigation into this matter is for the most part up to the weather providers or through cooperation between the routing and weather companies. A cooperation between the two may also lead way to risk based routing. Research on this matter is likely to surface in coming years.

One question that is still left unanswered is whether weather routing is moving towards shore based optimisations rather than currently being on board. Even though the computation power of laptops and smaller computers are increasing, the complexity and demand of the optimisations are also increasing. This will continue leading to a limitation in the on board routing capabilities, where a shore based solution will increase the possible complexity of the algorithms and optimisation routines. Shore based optimisation does though require proper available data connections to the ships. Further, a shore based solution will take the decision making out of the hands of ship masters and this human factor may be damaging for the use of the routing suggestions proposed from the routing programs. Thus the future of on board versus shore based optimisation is still uncertain.

7.2 Concept of Weather Routing

The concept of weather routing has been outlined through the components making up constraints, fuel consumption models and basic elements needed. An example of the construction of a generic fuel consumption model is included. This can be used as a basis when not having a large amount of ship specific information available.

The proposal of elements making up a weather routing tool has been mainly aimed at simpler, yet generic, modelling. In such a model uncertainties play a large role for the estimated total fuel consumption. Especially considering the influences of wind and waves, where research is still going on in order for the development of simpler models. The reason for the focus on the simpler models is that many shipping companies have old or chartered vessels without the availability of hull lines or extensive test results including response operators and wind tunnel test. Thus focusing on a more academic and more precise model might not be currently industrially applicable.

Handling of constraints are of great importance as these are limiting factors of the routing and if modelled incorrectly pose a danger to ship, cargo, crew, and environmental safety. The handling of both ETA and land is when using the DIRECT algorithm very straight forward. Safety has though not been handled thoroughly in this project.

The determination of broaching and synchronous roll have been proposed in IMO, MSC1. Circ 1228 (IMO, 2007), and has been tested for the vessel used in the project, showing that only the most severe weather conditions lead to limitations on the routing. The quality of these formula is further questionable as they have been revised recently and not tested extensively.

Cargo safety is of great concern as this in turn might indirectly limit the need for looking at roll and pitch motion considering crew and ship safety. Further, as weather routing is mainly applicable to container ships and containers are still commonly lost at sea the interest of this limit is large.

In the program safety/storm avoidance has only been implemented through a maximum wave height, which will not be sufficient for proper routing. Evaluating the limiting states for cargo safety is complicated to model in the simple, generic manner that has been sought for the fuel consumption model. Proper modelling may to some extend require proper knowledge of response amplitude operators in order to determine maximum roll angles and accelerations. Further analysis of maximum accelerations and roll angles must also be determined in order to map the full limiting state for a ship at specified draft, trim and loading condition. For the applicability of the program, such analysis must be carried out.

7.3 Program Modelling and Performance

A conceptual weather routing program has been constructed based on the DIRECT algorithm, showing capabilities of finding great circle route, routing around obstacles such as islands, slowing down/speeding up in order to avoid a time dependent storm, utilising currents and wind for saving fuel, and the combination of the mentioned. The program only handles safety of crew, ship, and cargo through maximum wave height and would have to be modified to properly take these matters into account for use in real routing. The code is however prepared for further functions such as improved ship and cargo safety add-ons.

The program is penalty dependent and for an extended use of it, the penalties would have to be mapped out in order for tuning not to be a requirement for an optimisation. In the current form the program has a fast initial convergence to an optimal solution, but takes more than acceptable time for finding a local optimum. The program is limited in complexity as it may only utilise few intermediate way points between start and end points due to the dimensional dependency of the algorithm coupled with computational time. The performance of the developed software is however good considering it is on a conceptual level.

The program has been modelled such that elements are easily added to the optimisation. This is most relevant concerning constraints such as SECA's, ice, pirating, or a proper safety model. The program is geared such that given a proper identification of limiting states for the ship, this may easily implemented through the use of 3D matrices containing all relevant weather information through the WP3D matrix.

The fuel consumption model has been fully provided by MMT, so the validity of the models have not ben questioned or deeply investigated as access has not been available. In turn this means the program is geared for working with the ships included in the Maersk database only as it fully relies on these models. The same is valid for the added resistance models of wind and waves used in the program. A modelling of these elements will though enable compatibility as the modules may easily switched out.

The conceptual program is for most purposes functioning well. The shortest route is found very fast, but producing a uniform speed profile where one would be expected is slower. This is to some degree due to an unforeseen limitation to the search space as the object algorithm receives a direct penalty in fuel consumption for arriving later than the desired ETA. In turn this means that when the shortest route has been found, arriving on time, a potentially better route only exists by adjusting multiple speed profiles simultaneously, thus speeding up on one stretch while slowing down on another. Minimising the ETA penalty was tested as a means to solve this problem, but bringing the penalty down too low results in the ETA only increasing as it is more fuel efficient.

It is seen from the tested cases that the fuel consumption models of higher order perform better, as expected. This is due to the algorithm properly recognising the effects of high and low speeds. Producing results using only high order fuel consumption models was tested as an alternative. For the cases tested in this project it is seen that this is a possible result as the routes are very close to being the same and the difference with some probability due to the algorithm iterations. Thus it may be a possibility to disregard the actual fuel consumption and instead use a model that exaggerates the effects of the model. This should though be studied further before applied and may lead to complications when fuel consumption curves have humps and hollows.

The optimisation is to some extend sensitive to penalty tuning. It will run and find a good solution when just adding large enough penalties for the algorithm to converge, but will require more computation time. Thus for a more extensive use of the program a full penalty dependancy study should be carried out, mapping the penalties for the program to be able to run automatically in a minimum amount of time.

With a proper safety model added to the program and access to other routing services, the program should be benchmarked in order to properly determine the capabilities. The focus in this project has been to prove the concept of applying the DIRECT algorithm to the weather routing problem rather than reaching a fully functional program to be benchmarked against other available solutions. From this it may be seen that there may exist a potential in using the DIRECT algorithm for this purpose, though most probably for a shore based solution.

The program has been constructed to compute speed profiles rather than power profiles. For a commercial use instead power profiles should be computed as maintaining a constant speed for a journey leg will only be possible in a calm environment. The program should otherwise be coupled with a voyage optimisation method to achieve the power profiles to be used on board. Further developing the code would require a major re-write using time stepping, but the base of the program and the methods used would not need to be changed significantly.

Bibliography

Release-notes to direct.m, June 2004. URL http://www4.ncsu.edu/~ctk/Finkel_Direct/userguide_addendum.pdf.

I. M. V. Andersen. Wind loads on post-panamax container ship. *Ocean Engineering*, 58:115–134, 2013.

K. Avgouleas. Optimal ship routing. Master's thesis, MIT, US, 2008.

N. Bowditch. *The American Practical Navigator*. National Imagery and Mapping Angency, 2002.

E. K. Burke and G. Kendall. *Search Methodologies*. Springer, 2005.

Y.-C. Chang, R.-S. Tseng, G.-Y. Chen, P. C. Chu, and Y.-T. Shen. Ship routing utilizing strong ocean currents. *The Journal Of Navigation*, 66:825–835, 2013.

H. Chen. *A dynamic program for minimum cost ship routing under uncertainty*. PhD thesis, MIT, US, 1978.

H. Chen. Voyage optimization supersedes weather routing. Technical report, 2011.

H. Chen, V. Cardone, and P. Lacey. Use of operation support information technology to increase ship safety and efficiency. *SNAME transactions*, 106:105–127, 1998.

Derbyshire. The sinking of mv derbyshire, 1980. URL http://www.liverpoolmuseums.org.uk/maritime/exhibitions/derbyshire/. information availiable 2014-03-06.

D. E. Finkel. *DIRECT Optimization Algorithm User Guide*. Center for Research in Scientific Computation, North Carolina State University, March 2003.

T. I. Fossen and H. Nijmeijer, editors. *Parametric Resonance in Dynamical Systems*. Springer, 2012.

T. Fujiwara, M. Ueno, and Y. Ikeda. An estimation method of wind forces and moments acting on ships. In *In Procedings of the Mini Symposium on Prediction of Ship Manoeuvring Performance.*, 2001.

T. Fujiwara, M. Ueno, and Y. Ikeda. Cruising performance of a large passenger ship in heavy sea. In *In Proceedings of the Sixteenth (2006) International Offshore and Polar Engineering Conference*, 2006.

Steve Ginsberg. Ship disaster leaves apl lost in sea of legal woes. *San Fransisco Buisness Times*, March 21, 1999.

H. Hagiwara. *Weather routing of (sail-assisted) motor vessel.* PhD thesis, University of Tokyo, Japan, 1982.

J. Hinnenthal. *Robust Pareto Optimum Routing of Ships utilizing Deterministic and Ensemble Weather Forecasts.* PhD thesis, Technischen Universitat Berlin, 2008.

J. Holtrop. A statistical resistance prediction method with a speed dependent form factor. *Scientific and Methodological Seminar on Ship Hydrodynamics*, 1988.

J Holtrop and G. G. J. Mennen. An aproximate power prediction method. *Journal of International Ship Building Progress*, 25 revised 29 (1982), 1978.

IMO. Msc.1/circ.1339 - best management practices for protection against somalia based piracy. Technical report, IMO - International Maritime Organization, a.

IMO. Marpol 73/78 annex vi regulation 14. Technical report, IMO - International Maritme Organization, b.

IMO. Msc.1/circ.1228 - revised guidance to the master for avoiding dangerous situations in adveres weather and sea conditions. Technical report, IMO - International Maritme Organization, 2007.

IMO. Mepc 59/inf.10 - prevention of air pollution from ships. Technical report, IMO - International Maritime Organization, 2009.

D. R. Jones, C. D. Perttunen, and B. E. Stuckman. Lipschitzian optimization without the lipschitz constant. *Journal of Optimization Theory and Application*, 79(1):157–181, October 1993.

H. O. Kristensen and M. Lützen. Prediction of resistance and propulsion power of ships. Project no. 2010-56, emissionsbeslutningsstøttesystem work package 2, report no. 04, Technical University of Denmark and University of Southern Denmark, October 2012.

Y. J. Kwon. Speed loss due to added resistance in wind and waves. *The Naval Architect - RINA*, March 2008.

L. Larsson and H. C. Raven. *Principles of Naval Architecture Series - Ship Resistance and Flow*. Society of Naval Architects and Marine Engineers (SNAME), 2010.

G. Mannarini, G. Coppini, P. Oddo, and N. Pinardi. A prototype of ship routing decision support system for an operational oceanographic service. *The International Journal on Marine Navigation and Safety of Sea Transportation*, 7(1):53–59, March 2013.

W. Mao, J. W. Ringsberg, and I. Rychlik. What is the potential of using ship fatigue routing in terms of fatigue life extension? In *Proceedings of the Twenty-second (2012) International Offshore and Polar Engineering Conference Rhodes, Greece, June 17th to 22ht, 2012*, 2012.

A. F. Molland, editor. *The Maritime Engineering Reference Book*. Elsiver, 2008.

T. Notteboom and P. Carriou. Proceeding of the 2009 international association of maritime echonomists (iame). In *Fuel Surcharge Practices of Container Shipping Lines: Is it About Cost Recovery or Revenue Making*, 2009.

C. P. Padhy, D. Sen, and P. K. Bhaskaran. Application of wave model for weather routing of ships in the north indian ocean. *Natural Hazards*, 44:373–385, 2008.

J. Podsada. Lost sea cargo: Beach bounty or junk? *National Geographic News*, 2001.

P. Schiller. Analysis of the possibility to avoid maritime accidents by the application of msc.1/circ. 1228 and msc/circ. 707. Master's thesis, Hamburg University of Technology - Institute of Ship Design and Ship Safety, 2011.

H. Schneekluth. *Ship Design for Efficiency and Economy*. Elsevier, 1978.

M.-G. Seo, D.-M. Park, K.-K. Yang, and Y. Kim. Comparative study on computation of ship added resistance in waves. *Ocean Engineering*, 73:1–15, 2013.

W. Shao and P. Zhou. Development of a dynamic programming method for low fuel consumption and low carbon emission from shipping. In *Low Carbon Shipping 2011 Conference Papers*, 2011.

W. Shao, P. Zhou, and S. K. Thong. Development of a novel forward dynamic programming method for weather routing. *Journal of Marine Science and Technology*, 11(1), 2006.

L. Skoglund, J. Kuttenkeuler, and A. Rosen. A new method for robust route optimization in ensemble weather forecasts. 2012.

M. Sniedovich. Dijkstra's algorithm revisited: the dynamic programming connexion. *Control and Cybernetics*, 35(3):600–620, 2006.

J. Szlapczynska. Multicriteria evolutionary weather routing algorithm in practice. *the International Journal on Marine Navigation*, 7(1):61–65, March 2013.

UNCTAD. Review of maritime transport 2013. Technical report, UNCTAD - United Nations Conference on Trade and Development, 2013.

S. Wei and P. Zhou. Development of a 3d dynamic programming method for weather routing. *TransNav*, 6(1):79–85, March 2012.

A - State of the Art

In this appendix you will find the replies to the State of the Art questioner.

A.1 AWT - BonVoyage

General

- How do you expect routing to be performed in 5 and 10 years? Onshore? Onboard?

Both for the following reasons;

1. At present there is no mandated requirement for all vessels to be provided with some form of routing assistance but this will come in the next 5 to 10 years. When it does it will be through the owners side of the business, and will have a minimum requirement to install something onboard similar to BVS.
2. Charterers will always want to check the performance of their vessel versus the C/P, and this is where the shore based service comes in to play. Unless you can change the Owner / Charterer relationship there will always be two ways to deliver the service.

- Do you optimize speed profiles along the route?
  - Yes
- Is the program ECDIS compatible?
  - Yes
- Is the optimization done onboard?
  - Yes if BVS is fitted. No if the routing service provided is from shore.
- Is the program both capable of oceanic and coastal routing?
  - Yes. Coastal routing is enhanced by the recent addition of HYCOM currents married to tidal current data.

- How do you expect routing to be performed in 5 and 10 years? Onshore? Onboard?
  - This question has been asked previously but there are combinations which have not been addressed, such as incorporating routing services into charting systems/chart correction systems etc, with seamless swapping between the systems, incorporating the necessary navigation and routing decisions.
- How do you handle ships with sparse specific information available, like chartered vessels?
  - Shore based services coupled to a vast database of ship response information allowing us to accurately analyse vessel performance without the need to revert to specific ship line drawings etc.

Weather forecast

- How does the program handle weather data for time steps beyond the forecasted weather?
  - In the next release this will be provided in the form of a ship specific speed down calculation rather than a generic speed loss calculation which most systems are using at this time.
- Who is your weather forecast provider?
  - At this time GFS
- What is the resolution of the forecast grid?
  - This is dependent upon what forecast dataset is requested. It varies between $1/8^{th}$ of a degree and 2 degrees.
- How often is weather forecast information received?
  - Weather forecast data is received and processed every 6 hours. Processing is done and available for use operationally at around 6 hours from the base time.

- How do you handle the uncertainty in weather forecasts? Do you route with super ensemble forecasts?
    - yes

Ship specific factors

- Is it possible for the routing tool to adapt to vessels current loading condition?
    - Yes
- Can the tool predict responses with ship specific RAOs?
    - Yes

Speed factors

- Do you optimize the route taking surface currents into account?
    - Yes
- Do you optimize the route taking tidal currents into account?
    - Yes
- Do you optimize the route taking wind drag/resistance into account?
    - Yes
- Are shallow water effects taken into account when estimating speed?
    - No.
- Are propeller and hull fouling taken into account?
    - No and Yes. No, in that there is no specific modeling built into the software for this. We did look at the option at the design stage but there are so many variables we decided we could never make the software accurate enough so instead gave the master the option to adjust the speed loss criteria for such outside influences.

Precaution areas

- Are pirated waters included in the routing?
    - Yes
- Are areas with ice included?
    - Yes. They are displayed and can be partially or completely avoided either manually or automatically.
- Are ECA/SECA areas included?
    - Yes and as with piracy and ice area can be automatically avoided.
- Can you assign different fuel to be used in SECA areas (with the related higher cost)?
    - Yes

If allowed to share

- What is a standard onboard optimization time for a trans-atlantic or trans-pacific crossing?
    - Approx 15 seconds
- Which optimization algorithm does the program use?
    - No comment.

A.2 Force Technology - SeaPlanner

Questionnaire route optimization

SeaPlanner 2.4 from Force Technology / DMI

General

- Do you optimize speed profiles along the route?
  Yes, the program iterates several speed profiles in search for the most fuel economic speed. The optimization criteria depend on the selected optimization mode.
- Is the program ECDIS compatible?
  Force Technology does not have a partnership with any ECDIS providers, and as such we do not have been able to obtain access to specifications of the available exchange formats that the ECDIS providers use. However, SP 2.4 contains an import/export function aimed at sharing routes with the ECDIS system. Currently three formats for exchange to ECDIS are offered (Furuno, Maris, Navi-Sailor) plus the open standard format GPX.
- Is the optimization done onboard?
  Yes
- Is the program both capable of oceanic and coastal routing?
  Yes

Weather forecast

- How does the program handle weather data for time steps beyond the forecasted weather?
  The last time step of the weather forecast is repeated for time steps beyond the forecasted weather.
- Who is your weather forecast provider?
  Dansk Meteorologisk Institut (DMI)
- What is the resolution of the forecast grid?
  The user orders the resolution for his forecasts within the programme.
  For large oceanic areas the best resolution is 0.5 minutes, 12 hour intervals for 10 ten days.
  For local areas like the North Sea the best resolution is 0.1 minutes, 1 hour intervals for 54 hours.
- How often is weather forecast information received?
  For oceanic areas every 12 hours. For local areas every 3 hours.

Ship specific factors

- Is it possible for the routing tool to adapt to vessels current loading condition?
  Yes, the user will provide draught fore and draught aft and if motions are to be calculated then also GM.
- Can the tool predict responses with ship specific RAOs?
  Yes

Speed factors

- Do you optimize the route taking surface currents into account?
  Yes.

- Do you optimize the route taking tidal currents into account?
  Yes.
- Do you optimize the route taking wind drag/resistance into account?
  Yes.
- Are shallow water effects taken into account when estimating speed?
  Yes.
- Are propeller and hull fouling taken into account?
  Yes, as well as deterioration of engine performance (increased SFOC).

Precaution areas

- Are pirated waters included in the routing?
  Yes
- Are areas with ice included?
  Yes
- Are ECA/SECA areas included?
  Yes
- Can you assign different fuel to be used in SECA areas (with the related higher cost)?
  No. At present the aim is a saving fuel for environmental reasons rather than saving money. That may change in a future release.

If allowed to share

- What is a standard onboard optimization time?
  Depending on the requested optimization mode and complexity of the route the optimization time varies from a few seconds to several minutes. A standard optimization would be less than a minute.
- Which optimization algorithm does the program use?
  Simulated annealing (Monte Carlo).
- How often is an onboard optimization run?
  In tracking mode the route is reoptimized several times per hour and whenever a new weather forecast is available.

A.3 Jeppesen - VVOS

How do you expect routing to be performed in 5 and 10 years? Onshore? Onboard?

The line between onboard or onshore started to blur when broadband connectivity between ship and shore becoming increasingly common. Perhaps it will all be in the "Cloud"?

- How does the program handle weather data for time steps beyond the forecasted weather? Climatology
- Who is your weather forecast provider? Several sources including www.oceanweather.com
- What is the resolution of the forecast grid? Is it different for coastal navigation/oceanic navigation Several depending on the need for hi-resolution
- Do you optimize the route taking tidal currents into account? Only in selected areas where data is available
- Are shallow water effects taken into account when estimating speed? We have access to depth sounding from C-Map charts. Resistance will increase in shallow water
- Are propeller and hull fouling taken into account? Yes in an indirect way
- Are pirated waters included in the routing? Yes as NOGO zone
- Are areas with ice included? Yes
- Are ECA/SECA areas included? yes
- Can you assign different fuel to be used in SECA areas (with the related higher cost)? Coming soon
- What is a standard onboard optimization time for a typical trans-atlantic or trans-pacific route? Depends on the grid size, roughly a few minutes
- Which optimization algorithm does the program use? 3D dynamic programming
- How do you handle ships with sparse specific information available, like chartered vessels? We have cumulated a library of hull forms over the past 20 years we can modify using reported data
- How do you handle the uncertainty in weather forecasts? Do you route with super ensemble forecasts? Ensemble forecast is available to our routers. Stochastic Dynamic Programming requires extensive computing power. This is an area that need more research.

A.4 Meteo Group - SPOS

Questionnaire route optimization

General

- How do you expect routing to be performed in 5 and 10 years? Onshore? Onboard?
    - Combination, initial route planning onshore, shift towards ECDIS integration onboard. Responsibility will stay 'at the Master'
- Do you optimize speed profiles along the route?
    - Yes (fixed ETA calculations, but outcome advice is to set one speed in calm seas)
- Is the program ECDIS compatible?
    - What is ECDIS compatible…SPOS integrates with several ECDIS systems and route planning tools like Voyager and E-Navigator.
- Is the optimization done onboard?
    - Yes for SPOS, no for RouteGuard
- Is the program both capable of oceanic and coastal routing?
    - Yes
- How do you expect routing to be performed in 5 and 10 years? Onshore? Onboard?
    - Same question as above…
- How do you handle ships with sparse specific information available, like chartered vessels?
    - What means 'handle' in this context? The required ship specific input for SPOS needs to be given by the Master. He/she will know these characteristics.

Weather forecast

- How does the program handle weather data for time steps beyond the forecasted weather?
    - Climatic averages
- Who is your weather forecast provider?
    - We buy and get data from all well-known global models, additionally we add regional models with nautical focus. From all this input we derive our own Nautical MeteoBase which is proven to be the best forecast model for wind and waves at sea.
- What is the resolution of the forecast grid?
    - Depends on the product. Not limited. But for SPOS we use 1 degree. Mainly because file sizes still need to be kept small.
- How often is weather forecast information received?
    - Max. 4 times a day
- How do you handle the uncertainty in weather forecasts? Do you route with super ensemble forecasts?
    - In our Nautical MeteoBase we also use the ECWMF ensemble for wind and waves. For timesteps further in the future we increase the weight/impact of this ensemble.

Ship specific factors

- Is it possible for the routing tool to adapt to vessels current loading condition?
    - Yes, in the normal SPOS version users can define loading-specific ship profiles considering speed reduction tables and fuel curves.
    - In the Seakeeping SPOS version users can dynamically change the loading conditions. The routing part will take the changes into account when calculation the vessels motions.
- Can the tool predict responses with ship specific RAOs?
    - The Seakeeping module in SPOS can.

Speed factors

- Do you optimize the route taking surface currents into account?
    - Yes

- Do you optimize the route taking tidal currents into account?
  - Not yet. Planned this year.
- Do you optimize the route taking wind drag/resistance into account?
  - Yes
- Are shallow water effects taken into account when estimating speed?
  - No
- Are propeller and hull fouling taken into account?
  - Not specific. Depends on the users' input.

Precaution areas

- Are pirated waters included in the routing?
  - For SPOS not, for RouteGuard yes
- Are areas with ice included?
  - Yes
- Are ECA/SECA areas included?
  - Yes
- Can you assign different fuel to be used in SECA areas (with the related higher cost)?
  - No

If allowed to share

- What is a standard onboard optimization time for a trans-atlantic or trans-pacific crossing?
  - minutes
- Which optimization algorithm does the program use?
  - Combined methods with additional parameters and restriction settings. Developed in-house.

A.5 SeaWare - EnRoute

General

- Do you optimize speed profiles along the route?
    - Yes
- Is the program ECDIS compatible?
    - We can export and import ECDIS-files but the files needs to be transferred by USB-stick to the ECDIS.
- Is the optimization done onboard?
    - Yes. The same type of optimization is also done by our route analysts/forecasters if the vessel doesn't have onboard software or if the master requests this.
- Is the program both capable of oceanic and coastal routing?
    - Yes

Weather forecast

- How does the program handle weather data for time steps beyond the forecasted weather?
    - At the moment we assume calm sea. In a future version we will use statistical averages of the weather instead.
- Who is your weather forecast provider?
    - ECMWF for atmospheric and ocean data and RTOFS for ocean current
- What is the resolution of the forecast grid?
    - 0.125° and 3 hour time steps are the highest available but lower resolution is in general used onboard to vessel to decrease the size of the files transferred to the vessel.
- How often is weather forecast information received?
    - By automatic e-mails to the vessel. Multiple options exist. See the manual section 2.3.3 in the manual.

Ship specific factors

- Is it possible for the routing tool to adapt to vessels current loading condition?
    - Yes
- Can the tool predict responses with ship specific RAOs?
    - Yes, Seaware EnRoute and EnRoute Live can do this

Speed factors

- Do you optimize the route taking surface currents into account?
    - Yes
- Do you optimize the route taking tidal currents into account?
    - No
- Do you optimize the route taking wind drag/resistance into account?
    - Yes
- Are shallow water effects taken into account when estimating speed?
    - No, planned for later releases
- Are propeller and hull fouling taken into account?
    - Hull fouling is entered when setting up the vessel particulars

Precaution areas

- Are pirated waters included in the routing?
  - They are shown on map but not included in the optimization
- Are areas with ice included?
  - They are shown on map but not included in the optimization
- Are ECA/SECA areas included?
  - They are shown on map but not included in the optimization. Planned to be included in the optimization in the next release.
- Can you assign different fuel to be used in SECA areas (with the related higher cost)?
  - Planned for next release but does not exist at the moment

If allowed to share

- What is a standard onboard optimization time?
  - Highly dependent of the route length, the forecast resolution and if any ship motions are calculated. From 15 seconds for an Atlantic crossing and upwards.
- Which optimization algorithm does the program use?
  - A custom genetic algorithm of the evolutionary type
- How often is an onboard optimization run?
  - When initiated by the crew. We recommend running it when a new forecast is received.

B - Program Code

B.1 Main Script

```matlab
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%                                                                         %
%   WEATHER ROUTING - USING DIRECT ALGORITHM                              %
%                                                                         %
%   CREATED BY MARTIN HJORTH SIMONSEN AND ERIK LARSSON                    %
%   20 JAN 2014 - ALL RIGHTS RESERVED                                     %
%                                                                         %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
clear all
tic
global n A B shipID WP3DStandard FasterFasterYES WaterDepth
global undercoverPenalty2 SeaTemp earliestArrival additionalOffTime
global draught trim pointIdentifiers ETA p q maplegend DepthLimit
global avgSpeed latestArrival undercoverPenalty WaveHeightLimit f1 f2
global plotThatShit TimeResolution npoints Resolution stormTime
global WaveDir CurrentSpeed CurrentDir WindSpeed WindDir WaveHeight

% REDACTED LOAD OF MAERSK MODULE; %

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% PRIMARY VARIABLES
n = 4; % Number of searched points.
A = [24.48, 118.09, 0]; % Start coordinates and hour.
B = [17.96, 257.81, 360]; % End coordinates and hour.
shipID = 71; % Ship id.
draught = 8; % Draught.
trim = 0; % Trim.
load environmentFinalLandAndStorm.mat % Tested weather environment.

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% PENALTIES
ETAPenalty = 5;
WavePenalty = 2;
LandPenalty = 0.1;
undercoverPenalty = ETAPenalty; % Additional penalty for faster convergence
undercoverPenalty2 = 0; % Additional penalty for faster convergence

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

FasterFasterYES = 1; % Insert 1 if you want it Faster Faster. This utilises
% fitted speed - fuel consumption curve.

% PASSING OPTIONS FOR THE MAIN DIRECT OPTIMIZATION
options.maxits = 5000; % Maximum number of iterations
options.maxevals = 20000; % Maximum number of function evaluations
options.maxdeep = 10000; % Maximum rectangle divisions.
options.ep = 10^-12; % Jones factor.

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

```
% SECONDARY VARIABLES
WaveHeightLimit = 9; % Limiting wave height.
DepthLimit = -20; % Minimum sailing water depth.
maxSpeed = 25; % Max speed of vessel [kn].
SeaTemp = 15; % Sea temperature.

% COMPUTATIONAL VARIABLES
offTime = 0; % How many hours late may the ship be.
additionalOffTime = 5; % Used as interval for complex fuel calc.
Nlat = 56; % Northern latitude for map.
Wlong = 114; % Western longitude for map.
Resolution = 1; % Map resolution [deg].
maplegend = [Resolution, Nlat, Wlong];
TimeResolution = 6; % Resolution of timeintervals for weather in hours.
npoints = 20; % Number of sampling points during TimeStep.

latBounds = [16, 55];% Latitude bounds [Lower, Upper]
longDivide = (B(2)-A(2))/n; % Creating longitudinal intervals
latestArrival = B(3) + offTime;
earliestArrival = B(3) - offTime;
Sizing = size(WaveHeight);
WP3DStandard = zeros(Sizing(1),Sizing(2),Sizing(3));
f1=[16,55,Sizing(1)];
f2=[116,260,Sizing(2)];

% OPTIONS FOR PREPROCESSED DIRECT OPTIMIZATION
optionsSimple.maxits = 2500;
optionsSimple.maxevals = 6000;

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%


% PREPROCESSING TO DETERMINE UNDISTURBED AVERAGE SAILING SPEED

plotThatShit = 0; % determines if solutions are plotted in Direct

% DEFINING BOUNDS FOR SEARCHED VARIABLES FOR NO WEATHER OR SPEED ROUTE
for i = 1:n
    boundsSimple((i-1)*2+1,1) = latBounds(1);
    boundsSimple((i-1)*2+1,2) = latBounds(2);
    boundsSimple((i-1)*2+2,1) = A(2)+(i-1)*longDivide;
    boundsSimple((i-1)*2+2,2) = A(2)+(i)*longDivide;
end

% PASSING MINIMIZATION PROBLEM, CONSTRAINTS, BOUNDS AND OPTIONS TO DIRECT
ProblemSimple.f = 'MinimumDistance';
ProblemSimple.numconstraints = 1; % Number of constraint functions
ProblemSimple.constraint(1).func = 'LandConstraintSimple';
ProblemSimple.constraint(1).penalty = LandPenalty;
[fminSimple, xminSimple, histSimple] = Direct(ProblemSimple,boundsSimple...
    ,optionsSimple);

avgSpeed = fminSimple/B(3);
```

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% PREPROCESSING THE SHIP SPECIFIC CURVES TO IMPROVE OPTIMIZATION SPEED

if FasterFasterYES == 1
    powertune = zeros(1,101);
    powertune(1)=0;
    for i = 1:100
        powertune(i+1) = [redacted]
    end
    x = linspace(0.25,25,100);
    x = [0 , x];

    p = polyfit(x(37:end),powertune(37:end),3);

    % GETTING THE POWER TO FUEL CONSUMPTION CURVE FOR FASTER CALCULATIONS
    fueltune = zeros(1,101);
    fueltune(1)=0;
    for i = 1:100
        fueltune(i+1) = [redacted]
    end

    q = polyfit(powertune,fueltune,1);
end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% MAIN OPTIMIZATOIN CALL

% Ensures the average speed for an undisturbed route is centered in the
% speed interval.
avgSpeed = 20.3972;
minSpeed = 2*avgSpeed - maxSpeed; % Min Speed [kn]

plotThatShit = 1; % 1 for iteration plot in DIRECT.

% DEFINING BOUNDS FOR SEARCHED VARIABLES FOR MAIN PROBLEM
bounds = zeros(n*3+1,2);
bounds(n*3+1,1) = minSpeed;
bounds(n*3+1,2) = maxSpeed;
for i = 1:n
    bounds((i-1)*3+1,1) = latBounds(1);
    bounds((i-1)*3+1,2) = latBounds(2);
    bounds((i-1)*3+2,1) = A(2)+(i-1)*longDivide;
    bounds((i-1)*3+2,2) = A(2)+(i)*longDivide;
    bounds((i-1)*3+3,1) = minSpeed;
    bounds((i-1)*3+3,2) = maxSpeed;
end

% PASSING MINIMIZATION PROBLEM, CONSTRAINTS, BOUNDS AND OPTIONS TO DIRECT
Problem.f = 'TotalFuelConsumption';
Problem.numconstraints = 3; % Number of constraint functions
```

```
Problem.constraint(1).func = 'ETAConstraint'; % Constraint function 1
Problem.constraint(1).penalty = ETAPenalty; % Penalty 1 .. 4.5
Problem.constraint(2).func = 'LandConstraint';
Problem.constraint(2).penalty = LandPenalty; % 0.5
Problem.constraint(3).func = 'MaxWaveHeightConstraint';
Problem.constraint(3).penalty = WavePenalty;
[fmin, xmin, hist] = Direct(Problem,bounds,options);

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% VISUALISATION OF THE SOLUTION
toc

% SOLUTION PLOTTING
pointIdentifiers = zeros(n+2,4);
pointIdentifiers(1,1) = A(1);
pointIdentifiers(1,2) = A(2);
pointIdentifiers(1,3) = 0;
pointIdentifiers(1,4) = 0;
ETA = 0;
for i = 2:n+1
    pointIdentifiers(i,1) = xmin((i-2)*3+1);
    pointIdentifiers(i,2) = xmin((i-2)*3+2);
    pointIdentifiers(i,3) = xmin((i-2)*3+3);
    ETA = ETA + 60*distance(pointIdentifiers(i-1,1),...
        pointIdentifiers(i-1,2),pointIdentifiers(i,1),...
        pointIdentifiers(i,2))/pointIdentifiers(i,3);
    pointIdentifiers(i,4) = ETA;
end
pointIdentifiers(n+2,1) = B(1);
pointIdentifiers(n+2,2) = B(2);
pointIdentifiers(n+2,3) = xmin(n*3+1);
pointIdentifiers(n+2,4) = pointIdentifiers(n+1,4)...
    + 60*distance(pointIdentifiers(n+1,1), pointIdentifiers(n+1,2)...
    ,pointIdentifiers(n+2,1), pointIdentifiers(n+2,2))/...
    pointIdentifiers(n+2,3);
disp(pointIdentifiers)

figure(1)
[latlim, longlim] = limitm(WaterDepth, maplegend);
worldmap(latlim, longlim)
meshm(WaterDepth,maplegend,size(WaterDepth),WaterDepth)
demcmap(WaterDepth)
hold on
plotm(pointIdentifiers(:,1),pointIdentifiers(:,2),'b','LineWidth',2)
plotm(pointIdentifiers(:,1),pointIdentifiers(:,2),'xr','MarkerSize',...
    15,'LineWidth',2)

figure(2)
plot(hist(:,2),hist(:,3),'xr')
xlabel('Number of function evaluations.');
ylabel('Minimum functional value at evaluation');
title('Iteration Statistics');
```

Published with MATLAB® R2013a

B.2 Total Fuel Consumption Script

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%                                                                         %
%   OBJECT FUNCTION USED IN DIRECT                                        %
%                                                                         %
%   INITIALISES THE FUEL CALCULATION USING EITHER SIMPLE OR COMPLEX FUEL  %
%   CALCULATIONS.                                                         %
%                                                                         %
%   CREATED BY MARTIN HJORTH SIMONSEN AND ERIK LARSSON                    %
%   30 APR 2014 - ALL RIGHTS RESERVED                                     %
%                                                                         %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
function totalFuel = TotalFuelConsumption(x)

% REDACTED LOAD OF MAERSK MODULE; %

global n A B earliestArrival undercoverPenalty listing hitViolation
global ETAViolation pointIdentifiers ETA latestArrival undercoverPenalty2
global additionalOffTime

% The point identifiers are created based on the solution sampled. The
% point identifiers are saved in the matrix pointIdentifiers, with
% latitude, longitude, entrance speed to point and time row wise
% respectively.
pointIdentifiers = zeros(n+2,4);
pointIdentifiers(1,1) = A(1);
pointIdentifiers(1,2) = A(2);
pointIdentifiers(1,3) = 0;
pointIdentifiers(1,4) = 0;
for i = 2:n+1
    pointIdentifiers(i,1) = x((i-2)*3+1);
    pointIdentifiers(i,2) = x((i-2)*3+2);
    pointIdentifiers(i,3) = x((i-2)*3+3);
    pointIdentifiers(i,4) = pointIdentifiers(i-1,4) + ...
        60*distance(pointIdentifiers(i-1,1), pointIdentifiers(i-1,2)...
        ,pointIdentifiers(i,1), pointIdentifiers(i,2))/...
        pointIdentifiers(i,3);
end
pointIdentifiers(n+2,1) = B(1);
pointIdentifiers(n+2,2) = B(2);
pointIdentifiers(n+2,3) = x(n*3+1);
pointIdentifiers(n+2,4) = pointIdentifiers(n+1,4) + ...
    60*distance(pointIdentifiers(n+1,1), pointIdentifiers(n+1,2),...
    pointIdentifiers(n+2,1), pointIdentifiers(n+2,2))/...
    pointIdentifiers(n+2,3);

ETA = pointIdentifiers(n+2,4); % ETA for sampled route.
listing = ceil(pointIdentifiers(:,4));

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% CREATES THE WAY POINT 3D MATRIX USED IN COMPLEX CALCULATIONS
WayPoint3D()
```

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% DETERMINES THE TOAL FUEL BY CALLS TO SIMPLE OR COMPLEX
if ETA > latestArrival+additionalOffTime || ETA < ...
        earliestArrival-additionalOffTime
    totalFuel = SimpleFuelCalculation()/1000; % tons of fuel
else
    totalFuel = ComplexFuelCalculation()/1000; %tons of fuel
end

% GIVES PENALTIES TO FUEL CONSUMPTION
overETA = ETA - B(3);
if overETA > 0
    totalFuel = totalFuel + overETA*undercoverPenalty*2;
end
if hitViolation > 0
    totalFuel = totalFuel + hitViolation*undercoverPenalty2*2;
end
```

Published with MATLAB® R2013a

B.3 Simple Fuel Calculation Script

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%                                                                          %
%   SIMPLE FUEL CALCULATION USED IN TOTALFUELCONSUMPTION                   %
%                                                                          %
%                                                                          %
%   CREATED BY MARTIN HJORTH SIMONSEN AND ERIK LARSSON                     %
%   30 APR 2014 - ALL RIGHTS RESERVED                                      %
%                                                                          %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

function totalFuel = SimpleFuelCalculation()

% REDACTED LOAD OF MAERSK MODULE; %

global n shipID
global draught trim pointIdentifiers FasterFasterYES p q

if FasterFasterYES > 0
    totalFuel = 0;
    for i = 2:n+2
        totalFuel = totalFuel + (pointIdentifiers(i,4)...
            -pointIdentifiers(i-1,4))...
            *polyval(q,polyval(p,pointIdentifiers(i,3)));
    end
else
    totalFuel = 0;
    for i = 2:n+2
        totalFuel = totalFuel + (pointIdentifiers(i,4)-...
            pointIdentifiers(i-1,4))*%REDACTED MAERSK CALL
    end
end
```

Published with MATLAB® R2013a

B.4 Complex Fuel Calculation Script

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%                                                                          %
%   COMPLEX FUEL CALCULATION FUNCTION, TAKING INTO ACCOUNT INFLUENCES      %
%   FROM WIND WAVES AND CURRENT                                            %
%                                                                          %
%   CREATED BY MARTIN HJORTH SIMONSEN AND ERIK LARSSON                     %
%   30 APR 2014 - ALL RIGHTS RESERVED                                      %
%                                                                          %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

function totalFuel = ComplexFuelCalculation()

% REDACTED LOAD OF MAERSK MODULE; %

global n shipID COG
global WaveHeight WaveDir CurrentSpeed CurrentDir WindSpeed WindDir
global draught trim pointIdentifiers WP3D SeaTemp

% The correction is there for the 0 values not to be overlooked in mean.
Correction = WP3D*0.01;
% The last sailing hour for tested route
upperTime = ceil(pointIdentifiers(n+2,4));

% Preallocating the logged avg. directions and magnitudes for each sailing
% hour
WavePow = zeros(upperTime,1);
WindPow = zeros(upperTime,1);
CurrentEffectiveSpeed = zeros(upperTime,1);

% Assign SOG for each sailing hour
for k = 2:n+2
    SOG((ceil(pointIdentifiers(k-1,4))+1):...
        ceil(pointIdentifiers(k,4)))= pointIdentifiers(k,3);
end

% Get average relative encountered weather and added power for each
% sailing hour
for i = 1:length(COG)

    % Producing encountered magnitude time layer matrix
    WindSpeedEncounter = WP3D(:,:,i).*WindSpeed(:,:,i) ...
        + Correction(:,:,i);
    WaveHeightEncounter = WP3D(:,:,i).*WaveHeight(:,:,i)...
        + Correction(:,:,i);
    CurrentSpeedEncounter = WP3D(:,:,i).*CurrentSpeed(:,:,i)...
        + Correction(:,:,i);
    WindDirEncounter = WP3D(:,:,i).*WindDir(:,:,i) + Correction(:,:,i);
    WaveDirEncounter = WP3D(:,:,i).*WaveDir(:,:,i) + Correction(:,:,i);
    CurrentDirEncounter = WP3D(:,:,i).*CurrentDir(:,:,i)...
        + Correction(:,:,i);

    % Mean Wave Height, Wind Speed and Current Speed for time layer
```

B.5 Way Point 3D Script

```matlab
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%                                                                            %
%   WAYPOINT3D CREATES THE 3D MATRIX CONTAINING THE TIMESPACE ROUTE          %
%   REPRESENTED AS ONES IN A ZEROED MATRIX                                   %
%                                                                            %
%   CREATED BY MARTIN HJORTH SIMONSEN AND ERIK LARSSON                       %
%   30 APR 2014 - ALL RIGHTS RESERVED                                        %
%                                                                            %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

function WayPoint3D()

global n WP3D COG npoints Resolution
global WP3DStandard TimeResolution
global pointIdentifiers f1 f2

WP3D = WP3DStandard; % Restores WP3D to a zeroed matrix

% Following loop creates a list of latitudes and longitudes for each
% timestep
WayPoint = [];
for k = 1:n+1
    pos=gcwaypts(pointIdentifiers(k,1),pointIdentifiers(k,2),...
        pointIdentifiers(k+1,1),pointIdentifiers(k+1,2),...
        ceil(pointIdentifiers(k+1,4)/TimeResolution)-...
        ceil(pointIdentifiers(k,4)/TimeResolution));
    WayPoint=[WayPoint,[pos(2:end,1)'; pos(2:end,2)']];
end

% Adds 360 degrees to the positions to bring to a continuous coordinate
% system for ease of computation.
indices=find(WayPoint(2,:)<0);
WayPoint(2,indices)=WayPoint(2,indices)+360;

% Following loop creates a listing og course over ground and modifies the
% WP3D matrix so the sampled route in time space is created.
COG=zeros(1,length(WayPoint(1,:)));
for j=1:length(WayPoint(1,:))-1
    dx=WayPoint(1,j+1)-WayPoint(1,j);
    dy=WayPoint(2,j+1)-WayPoint(2,j);
    if dx == 0
        if dy == 0
        elseif dy > 0
               COG(j)= 90;
        else
            COG(j) = 270;
        end
    elseif dx > 0
        if dy >= 0
            COG(j) = atand(dy/dx);
        else
            COG(j) = atand(dy/dx) + 360;
```

```
        end
    elseif dx < 0
        if dy == 0
            COG(j) = 180;
        else
            COG(j) = atand(dy/dx) + 180;
        end
    end

    latitudes=(linspace(WayPoint(1,j),WayPoint(1,j+1),npoints));
    longitudes=(linspace(WayPoint(2,j),WayPoint(2,j+1),npoints));
    for h=1:npoints
        latnow = round(((round(latitudes(h)*Resolution)/Resolution)-...
            f1(1))*Resolution);
        lonnow = round(((round(longitudes(h)*Resolution)/Resolution)-...
            f2(1))*Resolution);
        WP3D(latnow,lonnow,j) = 1;
    end
end
```

Published with MATLAB® R2013a

B.6 Land Constraint Script

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%                                                                          %
%   LANDCONSTRAINT IS THE CONSTRAINT ON HITTING LAND USED IN DIRECT        %
%                                                                          %
%   CREATED BY MARTIN HJORTH SIMONSEN AND ERIK LARSSON                     %
%   30 APR 2014 - ALL RIGHTS RESERVED                                      %
%                                                                          %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

function value = LandConstraint(x)

global n pointIdentifiers
global maplegend WaterDepth DepthLimit

value = 0;
for i = 1:n+1
    depth = mapprofile(WaterDepth,maplegend,[pointIdentifiers(i,1),...
        pointIdentifiers(i+1,1)],[pointIdentifiers(i,2),...
        pointIdentifiers(i+1,2)]);
    HitLand = length(depth(depth>DepthLimit))/length(depth)*60....
    *distance(pointIdentifiers(i,1),pointIdentifiers(i,2),...
    pointIdentifiers(i+1,1),pointIdentifiers(i+1,2));
    value = value + HitLand;
end
```

Published with MATLAB® R2013a

B.7 ETA Constraint Script

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%                                                                          %
%   ETACONSTRAINT IS THE CONSTRAINT ON ARRIVAL TIME USED IN DIRECT         %
%                                                                          %
%   CREATED BY MARTIN HJORTH SIMONSEN AND ERIK LARSSON                     %
%   30 APR 2014 - ALL RIGHTS RESERVED                                      %
%                                                                          %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

function value = ETAConstraint(x)

global ETA latestArrival

value = ETA - latestArrival;
```

Published with MATLAB® R2013a

B.8 Max Wave Height Constraint Script

```matlab
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%                                                                           %
%   MAXWAVEHEIGHTCONSTRAINT IS THE CONSTRAINT RELATED ON MAXIMUM WAVE       %
%   HEIGHT USED IN DIRECT                                                   %
%                                                                           %
%   CREATED BY MARTIN HJORTH SIMONSEN AND ERIK LARSSON                      %
%   30 APR 2014 - ALL RIGHTS RESERVED                                       %
%                                                                           %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

function value = MaxWaveHeightConstraint(x)

global n pointIdentifiers hitViolation WaveHeightLimit WP3D WP3DStandard
global WaveHeight TimeResolution

HitStorm = WP3DStandard;
HitStorm = WP3D.*WaveHeight; % A matrix of wave heights encountered.
p = size(HitStorm,3);
inter2 = zeros(p);

dist1 = [];
for i = 2:n+2
    dist1 = [dist1, repmat(pointIdentifiers(i,3),1,ceil(...
        pointIdentifiers(i,4)/TimeResolution)-ceil(...
        pointIdentifiers(i-1,4)/TimeResolution))];
end

for i = 1:length(dist1)
    inter = HitStorm(:,:,i);
    inter2(i) = dist1(i)*length(inter(inter > WaveHeightLimit))...
        /length(inter(inter > 0));
end

value = sum(nansum(inter2));
hitViolation = value;
```

Published with MATLAB® R2013a

B.9 Case Tester Script

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%                                                                         %
%   CASETESTER TESTS A ROUTE SOLUTION FOR HITTING LAND AND STORM AND      %
%   CALCULATES THE ETA. AS AN OUTPUT THE FUNCTION CREATES A PLOT OF THE   %
%   TESTED ROUTE AND SPEEDS.
%                                                                         %
%   CREATED BY MARTIN HJORTH SIMONSEN AND ERIK LARSSON                    %
%   30 APR 2014 - ALL RIGHTS RESERVED                                     %
%                                                                         %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

function [ETA, speeds, landHit, stormHit] = casetester(x)

global A B n WaterDepth maplegend DepthLimit
global  WaveHeightLimit WP3D WaveHeight WP3DStandard TimeResolution

pointIdentifiers = zeros(n+2,4);
pointIdentifiers(1,1) = A(1);
pointIdentifiers(1,2) = A(2);
pointIdentifiers(1,3) = 0;
pointIdentifiers(1,4) = 0;
for i = 2:n+1
    pointIdentifiers(i,1) = x((i-2)*3+1);
    pointIdentifiers(i,2) = x((i-2)*3+2);
    pointIdentifiers(i,3) = x((i-2)*3+3);
    pointIdentifiers(i,4) = pointIdentifiers(i-1,4)...
        + 60*distance(pointIdentifiers(i-1,1), pointIdentifiers(i-1,2),...
    pointIdentifiers(i,1), pointIdentifiers(i,2))/pointIdentifiers(i,3);
end
pointIdentifiers(n+2,1) = B(1);
pointIdentifiers(n+2,2) = B(2);
pointIdentifiers(n+2,3) = x(n*3+1);
pointIdentifiers(n+2,4) = pointIdentifiers(n+1,4)...
    + 60*distance(pointIdentifiers(n+1,1), pointIdentifiers(n+1,2)...
    ,pointIdentifiers(n+2,1), pointIdentifiers(n+2,2))/...
    pointIdentifiers(n+2,3);

ETA = pointIdentifiers(n+2,4);
speeds = pointIdentifiers(:,3)';
WayPoint3D()

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

HitStorm = WP3DStandard;
HitStorm = WP3D.*WaveHeight; % A matrix of wave heights encountered.
p = size(HitStorm,3);
inter2 = zeros(p);

dist1 = [];
for i = 2:n+2
    dist1 = [dist1, repmat(pointIdentifiers(i,3),1,ceil(...
        pointIdentifiers(i,4)/TimeResolution)...
```