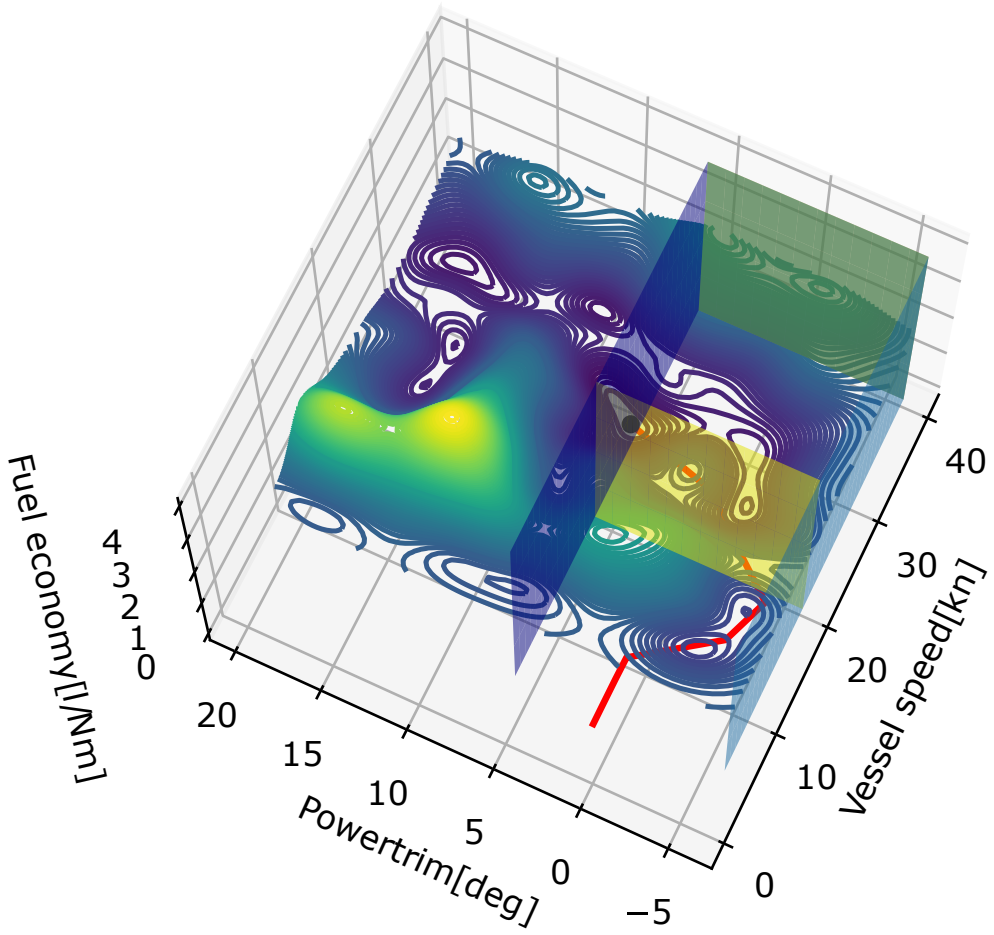


Marine guidance using AI

Development of a optimization system for marine vessels

Master's thesis in System, control and mechatronics

Lukas Ljungquist
Axel Måneskiöld

DEPARTMENT OF MECHANICS AND MARITIME SCIENCES
CHALMERS UNIVERSITY OF TECHNOLOGY
Gothenburg, Sweden 2022
www.chalmers.se

Master's thesis 2022:25

Marine guidance using AI

Development of a optimization system for marine vessels

Lukas Ljungquist
Axel Måneskiöld

CHALMERS
UNIVERSITY OF TECHNOLOGY

Department of Mechanics and Maritime Sciences
Applied Artificial Intelligence Research Group
Chalmers University of Technology
Gothenburg, Sweden 2022

Marine guidance using AI
Development of a optimization system for marine vessels
Lukas Ljungquist
Axel Måneskiöld

Thesis advisor: Mikael Ahlstedt, CPAC Systems AB
Thesis examiner: Peter Forsberg, Department of Mechanics and Maritime Sciences

Master's Thesis 2022:25
Department of Mechanics and Maritime Sciences
Division of Vehicle Engineering and Autonomous Systems
Applied Artificial Intelligence Research Group
Chalmers University of Technology
SE-412 96 Gothenburg
Telephone +46 31 772 1000

Cover: Illustration of a statistical model for a vessel, built using support vector regression, utilized by the speed profile. The red line corresponds to the minimum energy path found by the genetic algorithm. Moreover, the black dot is the most fuel-efficient motor configuration yielded by the particle swarm optimization. The four different hyperplanes correspond to the constraints of the system.

Typeset in LaTeX
Printed by Chalmers Reproservice
Gothenburg, Sweden 2022

Marine guidance using AI
Development of an optimization system for marine vessels
Lukas Ljungquist
Axel Måneskiöld
Department of Mechanics and Maritime Sciences
Chalmers University of Technology

Abstract

Navigating and handling a vessel optimally is by no means an easy task. Not only does one have to find the appropriate path to a destination while avoiding obstacles but also consider external forces such as weather conditions. Further, traveling by water is often very energy-consuming. A guidance system to facilitate such tasks should have great potential, not the least in the form of environmental effects. This thesis has constructed a marine guidance system that utilizes modern optimization methods to enable a more efficient way of voyage planning and handling of a vessel concerning fuel economy.

Stochastic optimization and black-box modeling have enabled a system to handle and optimize complex environments using minimal information. The results show that it is possible to create accurate fuel consumption models regardless of the configuration of vessel model and engines.

Further, a robust path planning system has been derived that manages to optimize several objectives and help drivers achieve more efficient voyage planning. The system has been shown to outscore the voyage planning capabilities of everyday drivers of the sea. These results prove the capabilities of a field that still has much to explore.

Keywords: Artificial Intelligence; Marine Environment; Image Processing; Path Planning; Evolutionary Algorithms; Clustering; Optimization; Support Vector Regression

Acknowledgements

We want to thank our supervisor Mikael Ahlsedt and examiner Peter Forsberg for their support and rewarding discussions throughout the work of this thesis. We also take this opportunity to thank all personnel at CPAC system AB for their help. An extra thanks to CPAC systems AB and AB Volvo Penta for letting us spend time at their facilities and allowing us access to their equipment. Everything from unlimited energizing coffee to boat rides aboard luxurious yachts in the glazing sun. We would also like to thank our opponents David Näslund and Martin Asplund for the valuable feedback!

A special thanks are also sent to Mikael Adielsson, Andreas Nyman, and Mattias Johansson at Volvo Penta for their support and help with data management.

Lukas Ljungquist and Axel Måneskiöld, Gothenburg, June 2022.

Thesis advisor: Mikael Ahlstedt, CPAC Systems AB
Thesis examiner: Peter Forsberg, Department of Mechanics and Maritime Sciences

Contents

List of Figures

List of Tables

List of Algorithms

Acronyms

$\mathcal{GP}$ Gaussian Process. xi, 21, 22

AIS Automatic Identification System. 3, 64
API Application Programming Interface. 30

C-Space Configuration Space. xi, 2, 3, 5, 6, 9, 40, 65
COLREG International Regulations for Preventing Collisions at Sea. 1, 5, 6, 63

EA Evolutionary Algorithm. 2, 4, 13, 27
ECD Electronic Chart Display. 1, 29
ECMWF European Centre for Medium-Range Weather Forecasts. 3
EI Expected Improvement. 22
ENC Electronic Navigational Chart. 1, 3, 29, 30, 63

FC-Space Free Configuration Space. 8, 9, 30, 60
Fr Froude number. 19

GA Genetic Algortihm. xi, xiii, 2, 3, 13, 15, 32, 33, 37, 39, 44–48, 50, 51, 54, 55, 59, 60, 62, 63
GNSS Global Navigational Satellite System. 1
GRIB General Regularly-Distributed Information in Binary Form. 30

KKT Karush-Kuhn-Tucker. 26

MAD Median Absolute Deviation. 41, 52, 53, 56
MAE Mean Absolute Error. 41, 52, 53, 56
MEP Minimum Energy Path. xii, xiii, 39, 55, 62–64
MLS Multi Level Selection. 6, 15
MLSGA Multi Level Selection Genetic Algorithm. xiii, 15, 29, 32, 34, 36, 40, 59, 60, 65
MOO Multi Objective Optimization. 16, 17, 29, 34

PCA Principal Component Analysis. 10, 32, 44, 65
PRM Probabilistic Roadmap. 2, 8, 9, 30, 65
PSO Particle Swarm Optimization. xi, 3, 27, 37, 39, 53, 54, 61

RMSE Root Mean Square Error. 41, 52, 53, 56
RPM Revolutions Per Minute. 4, 37, 40, 41, 52, 55, 61

RRT Rapidly-Exploring Random Tree. 2

SOG Speed Over Ground. 37, 38
SVM Support Vector Machine. 23
SVR Support Vector Regression. xi–xiv, 3, 4, 23, 24, 26, 37–39, 52–57, 62, 65

USV Unmanned Surface Vehicle. 5

1
Introduction

Chapter one gives an introduction to the area of maritime travel and vessels. Relevant research is presented to give an understanding of the work conducted in the area of the thesis. The thesis's problem formulation and scope are formulated with specified research questions. Lastly, the general outline of the thesis is presented.

1.1 Background

The usage of maritime vessels for transporting people and goods has been a well-established way of transportation throughout history. Today there are several different types of watercraft with vastly different usage areas. A vital part of maritime transportation is to navigate from the initial position to the desired endpoint. The commander of the vessel uses, with advantage, a nautical chart for navigation during the route. For larger and more advanced vessels, the usage of radar can often also be found. The nautical chart incorporates essential cartographic references to enable navigation concerning sea depth. Modern charts are stored and used through Electronic Chart Display (ECD), which integrates Electronic Navigational Chart (ENC) and Global Navigational Satellite System (GNSS).

The ENC contains either vector graphics, consisting of layers in different levels, or Raster Navigational Charts, which has the different layers in bitmap format. Defining a route beforehand can be performed by manually laying out waypoints on the ENC by starting on the present position until the final destination is reached. During the time of traveling, the waypoints will be used as guidelines for the navigation while closely monitoring the outcome of the route, according to the guidelines on voyage planning by International Maritime Organization Resolution [1]. However, one also has to consider several different other parameters. These include the International Regulations for Preventing Collisions at Sea (COLREG), the velocity of the vessel, and the configuration of the motor.

Regarding the resulting fuel economy, the vessel's performance is heavily influenced by the route, selected velocity, and the setting of the motor configuration, e.g., angle of trim or drives. If the vessel has an electronic engine control platform, the resulting fuel consumption at the current velocity can be estimated and displayed. Currently, Eco-driving for boat driving is part of the education for Swedish regulations of mastering yachts, such as the Coastal Yachtmaster Diploma[2]. Despite this, it does not imply the driver has learned how to drive the specific boat as fuel-efficiently as

possible.

The hull shape will heavily affect the optimal speed, considering the best fuel economy of the vessel. In total, there exist three different main categories of hulls. Displacement hulls are supported almost exclusively by buoyancy. Another is the planning hull that instead relies on the hydrodynamic lift, which reduces the wetted area in the sea and reduces the drag. The last one is a semi-displacement hull, a combination of the two.

1.2 Related work

In the history of development economics, voyage planning has been a key factor in the economics of maritime transportation. As a consequence of this, planning of routes for merchant ships is a well-researched topic. Several different approaches have been implemented on ships for long-distance routes[3][4]. In 1957 the isochrone method was proposed for finding the shortest routes assuming constant motor power[5]. This deterministic method is composed of starting in the ship's position and drawing lines corresponding to routes in different directions. This operation is iteratively done using a fixed step length until the goal destination is reached. Since 1957, much advancement has been made, resulting in methods that utilize today's computational power.

One of the most well-researched areas regarding navigation is for applications on mobile robots. It has gained interest in recent years due to the increasing growth of automated mobile systems [6]. The field consists of different methods for modeling the environment and finding paths. The environment is represented as a Configuration Space (C-Space). There exist several methods for creating graphs out of a C-Space. An example of algorithms for this is Probabilistic Roadmap (PRM), Rapidly-Exploring Random Tree (RRT) and Visibility graph. A special case of graphs is grid-based maps, which are also commonly used.

Moreover, as of today, the field of path planning, also known as graph searching, mainly consists of using algorithms that are variants of Dijkstra, A*, Ant colony, Genetic Algortihm (GA) or firefly [6]. Prevalent applications consist of finding the optimal path, with consideration to an objective, in the graph. The objective is usually the length of the path. It is also possible to optimize against multiple objectives.

In the case of multiple objective optimization, successful results have been achieved with the usage of Evolutionary Algorithm (EA). Recent studies have yielded successful results using different variants of EA. In a recent study, Genetic algorithms were applied for path planning considering multiple objectives [7]. The result outperformed path planning algorithms such as A*. Besides this, Grey Wolf Optimizer, which is an EA, has also been used in the field of path planning [8]. The study showed that such usage was applicable in very complex environments and showed promising results regarding feasibility.

In the area of voyage planning, nautical charts are used for representing the C-Space. Previous research has created the environment by using a variant of visibility graphs [9]. Studies have also yielded successful results by converting sea charts into a grid-based map[3][4]. In addition to this, there was successful research done where they used raw ENC data to locate hazardous objects beforehand when navigating an autonomous vessel. The idea in such an application is to use prior information regarding objects to avoid relying on obstacle detection systems [10].

Moreover, several studies have been investigating the use of the Fast marching method for planning routes in maritime environments[3][4]. There have also been successful studies using variants of Particle Swarm Optimization (PSO) and dynamic programming [11][12] to solve routing optimization problems with constraints. In addition to this, GA in combination with dynamic programming has been researched to optimize ships engine power to reduce emissions [13]. Lastly, research has also been conducted on the usage of historical trajectories from Automatic Identification System (AIS) together with Dijkstra and ant colony for planning paths[14]. However, a recent study observed that using a GA for path planning routes for maritime vessels has large benefits compared to other well-known methods. These include trivial methods for path optimization such as dynamic programming, grid search, and isochrone [15].

Research has also shown promising results considering forecast when simulating path planned route for maritime transport over long distances [3] [11]. It has also been investigated how the visualization of simulated weather forecast can be used as support for decisions regarding the route choice for marine vessels [16]. Recent research on decision support for route planning adopted the usage of numerical weather predictions from the European Centre for Medium-Range Weather Forecasts (ECMWF) [17]. Research has likewise been conducted on how wind forecasts can be utilized in order to avoid rough seas when planning a long-distance route [3].

As previously mentioned, fuel consumption is, in addition to the path traveled, affected by the speed profile during the path. Recent studies show that significant energy savings are possible for large vessels using decision support systems [18]. Further, have there been studies focusing on implementing trim optimization on container ship[19]. In addition to this, studies have shown that black box models remarkably improve the current state of art white-box modeling when forecasting the fuel consumption and optimizing the trim on the vessels in real operational conditions[20].

Research focusing on the prediction of the speed and fuel consumption has been performed by training a neural network on external conditions[21]. It has also been investigated how neural networks can be used for speed and trim optimization on larger vessels by using historical voyage reports [22]. Previous research has also established that Support Vector Regression (SVR) yields promising resulting when used for forecasting [23] [24]. It has also been observed that SVR is efficient when used for black-box modeling in system identification [25]. Recent research con-

cluded that SVR yields better results than other black-box models such as stacking-ensemble learning, autoregressive integrated moving average, ridge regression, and random forest models for predicting COVID-19 cases in Brazil [26].

Furthermore, the performance of a machine learning model, such as SVR, will be heavily influenced by the selected hyperparameters. A wide variety of algorithms exist to optimize the hyperparameters of machine learning models. Examples of algorithms are grid search, random search, Bayesian optimization, gradient-based optimization, and EAs [27]. A recent study has shown promising results with the use of Bayesian optimization for tuning a SVR model in order to estimate shear capacity in reinforced concrete members [28].

1.3 Motivation

Every year a total of 90% of the world's goods are transported over the sea. In addition, the recreational boating industry, which produces boats used for leisure, is estimated to have a revenue of 20 billion US$ each year and is expected to grow to over 60 billion US$ by 2027 [29]. Both the sea-bearing transportation and the recreational boating industry are facing several challenges[30]. A contemporary challenge is the environmental one that arises due to engine emissions.

The emissions are largely caused by the fuel that gets depleted during traveling. At the same time, the fuel for when driving the vessel is costly. Therefore a reduction in fuel consumption would result in lower costs and emissions. The fuel consumption of travel from one location to another will vary depending on the route taken. Each different route will be affected by the weather conditions in the traversed area. Depending on the size and direction of these forces, the resulting energy need will vary, and the fuel consumption will follow. Hence, there exists a possibility of optimizing the routes.

Furthermore, the non-optimal configuration of the vessel will also lead to unnecessary emissions. The prominent manufacturer of complete marine propulsion systems Volvo Penta currently optimizes the angle of the drives. Even so, the angle is only considering the Revolutions Per Minute (RPM) of the boat for the control policy [31]. Where the control policy is universal and not optimized against the boat, this creates the possibility of further optimizing fuel consumption by adapting a model against the boat itself.

There are also ongoing trends regarding the implementation of electromobility in vessels, which has several barriers to becoming a substitute for traditional engines[32]. In order to be able to move a vessel across the sea, an extensive amount of energy is needed [33]. One of the significant barriers to the usage of electric vessels are their limited range[32]. The extensive energy demand creates a need to optimize their limited electric power usage.

Moreover, Unmanned Surface Vehicle (USV) is seen as an area with potential in the future[34]. One concern for taking the product further is regarding the navigation. For the vessel to be fully autonomous, it needs to be able to find a path to the desired location. The operated way of the USV must satisfy COLREG rules. Hence, a demand for having a safe, distinct, and efficient planned path exists in this area.

Previous research has investigated how wind forecasts can be utilized to avoid rough seas when planning a route, mainly for merchant ships. However, research to date has not yet determined an efficient way of voyage planning in complex C-Spaces. Nevertheless, it has not been investigated how the weather conditions can be utilized to save fuel while ensuring a comfortable trip. Further, there has been no research concerning automatic optimization of recreational boats regarding fuel consumption.

1.4 Problem formulation

This thesis shall investigate the area of marine guidance. The aim is to construct a guidance system that generates and optimizes routes concerning fuel consumption while maintaining passenger comfort. The path generated by the system has to both consider obstacles and weather conditions. The guidance system also consists of a speed profile that is integrated. The speed profile provides information regarding the optimal speed and motor configuration, given specific time constraints, to minimize fuel consumption. The system will be evaluated against traditional voyage planning, i.e., planned routes of experienced boat users. Besides this, the performance of the speed profile will be validated against previously recorded performance tests for different vessels.

1.5 Research questions

The objective of the thesis is to examine the following research questions:

- What is the performance of a path planning system, that optimizes with respect to given objectives, in comparison to traditional voyage planning?
- What is the performance when implementing an adaptive black-box model to construct a marine guidance system with regard to engine configuration?

1.6 Scope

The following factors will limit the scope of the project:

- Speed profiling is limited to vessels that can provide sufficient data on fuel consumption and relative parameters

- Only static obstacles will be considered for the obstacle avoidance system. Thereby object such as other traveling maritime vessels will be excluded.
- No consideration will be given to following the COLREG rules.
- Path planning considering weather impact can only be done in areas where such data is available.
- No consideration will be taken to the curvature of the earth.
- Only an estimated model of a vessel will be used to compute the forces acting while traveling through water.

Relative parameters are said to be those included in the speed profile. The estimated model will be based on mathematical models that have been utilized in previous research papers.

1.7 Thesis contribution

This thesis propose a way for solving large-scale path planning, in complex C-Spaces, by dividing the problem into two sequences. Firstly by clustering distinct ways of travel towards the desired end destination. Subsequently, a Multi Level Selection (MLS) approach is applied to perceive the optimal path for multiple objectives. Moreover, the thesis also presents a robust method for optimizing vessels' fuel economy by establishing a speed profile, where constraints are utilized to retrieve the most optimal configuration.

1.8 Thesis outline

The thesis is divided into six different chapters. Chapter number one introduce the subject of maritime travel and the challenges the industry faces at this time. Furthermore, chapter two gives a theoretical background of the areas topical to the research of this thesis.

Moving on to chapter three, the methodology used in the thesis is further explored. This chapter is divided into two different main parts. The first dealing with the application used in the path planning system, and the other goes into more detail regarding the development of the speed profile.

Moreover, chapter four presents the results this thesis has produced. Discussion regarding the findings and potential future work in the area is done in chapter five. Lastly, chapter six present the conclusions that have followed this work.

2
Theory

The chapter explain the theoretical background of areas relevant to the work. The work presented in the chapter will help to understand the application areas of the thesis.

2.1 Image processing

The field of image processing consists of several different algorithms to process images. The usage typically exists to enhance an effect on an image or extract features from an image.

2.1.1 Erosion

Erosion is a morphological operation. It can be performed on both binary and grayscale images [35]. By defining an image as a function f and an operator function as b, defined in a space B, the erosion operation can be expressed as Equation 2.1.

$$(f \ominus b)(x) = \text{Infimum}_{y \in B}[f(x+y) - b(y)] \tag{2.1}$$

The resulting output from the operation is the image function modified by the operator, which computes the maximum in the given space B.

2.1.2 Dilation

Dilation is an operation that shares the same properties as erosion, i.e., it is morphological and can be performed both on binary and grayscale images[35]. In contrast to erosion its function is instead defined as in Equation 2.2

$$(f \oplus b)(x) = \text{Supremum}_{y \in E}[f(y) + b(x-y)] \tag{2.2}$$

The output image hence becomes the image modified by the operator function, which computes the local minimum in its defined space E.

2.1.3 Gaussian blur

For images, a Gaussian function, given for two dimensions as in Equation 2.3, can be applied through convolution to filter the image. In the frequency domain, the operation results in attenuating the high-frequency components. The specified cut-off

frequency is determined by the parameter of the Gaussian function, which corresponds to the magnitude of sigma. In other words, the sigma influence how large an effect the center pixel will have on the neighboring pixels.

$$G(x,y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}} \tag{2.3}$$

A Gaussian function utilized on an image is often termed Gaussian blur.

2.2 Graph theory

In its simplest form, a graph can be described as a set of points connecting in between. A formal definition of a graph is [36]:

Definition 1 (Graph) *A graph is a pair (V, E), where V is a set of objects called vertices and E is a set of two element subsets of V called edges.*

Graphs can be divided into two groups, directed and undirected. The difference lies in the direction of travel over an edge. Using undirected graphs allows travel in both directions over all edges. Beyond this, there are different definitions for routes in a graph. The most relevant for this thesis is *path* and *cycle*. These are both previously defined concepts [37].

Definition 2 (Path) *A "trip" in a graph G between vertices along edges is called a path if no vertex is visited more than once.*

Definition 3 (Cycles) *A "trip" in a graph G is called a cycle if its ending and starting vertex are the same, and no other vertex is used more than once.*

A cycle is a loop within a graph, while a path is a non-repeating sequence of vertices from a start position to an end position. To illustrate, Figure 2.1 is a smaller graph with a vertex and edge pointed out.

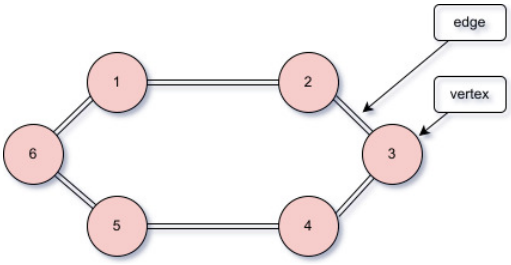


Figure 2.1: An example graph with an edge and vertex clarified

2.3 Probabilistic roadmap

PRM is a method of constructing a network of connecting edges between vertices placed in the Free Configuration Space (FC-Space). The FC-Space is defined as all

positions not placed inside of any obstacles, i.e., all feasible positions on the map. The resulting roadmap is a graph denominated as $R = (N, E)$ [38]. The vertices, N, are randomly generated from the FC-Space and placed in the graph. For each vertex, edges, E, are constructed to other previously placed vertices in the network, if and only if, the interconnecting edge strictly lies in the FC-Space. The resulting PRM will contain a predetermined, N, number of nodes coupled with all of the interconnecting edges, E. When constructing a PRM, the minimum input to the algorithm is the C-Space and the number of nodes to be placed. It is also possible to restrict the number of connections for each new node and the greatest length of an edge.

To show what an PRM can look like, an example is shown in Figure 2.2 [39].

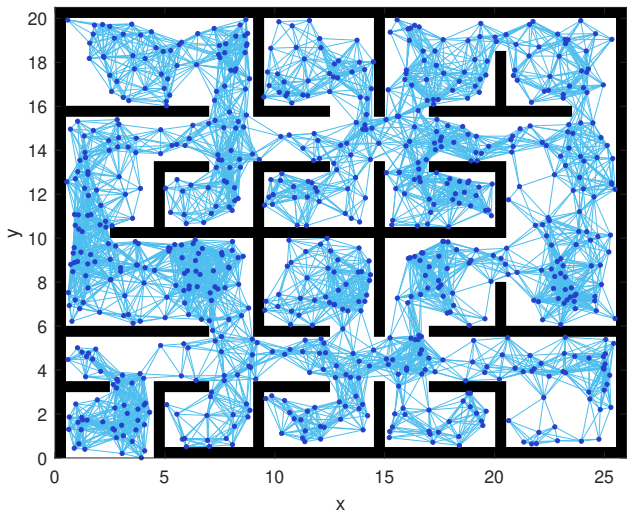


Figure 2.2: Example of probabilistic roadmap in a complex C-Space.

The algorithm for generating a PRM is given in Algorithm 1.

Algorithm 1: PRM algorithm

Data: number of nodes n, number of closest neighbors to examine k,
maximum number of neighbours for each vertex M
Result: G = (V,E)
initialization: $V \leftarrow \emptyset$;
$E \leftarrow \emptyset$;
Generate random points from feasible set ;
for $q \in V$ **do**
  $N_q \leftarrow$ the k closest neighbors of q chosen from V according to dist ;
  for $q^{'} \in N_q$ **do**
    if $q, q^{'}$ *and* $\delta(q, q^{'}) \neq NIL$ **then**
      E $\leftarrow E \cup (q, q^{'})$
    end
  end
end

2.4 Normalization

In order to find trends in data by its features, normalization plays a key roll. For the features to have equal importance the dimension of every data points has to be normalized. Two common ways of normalizing is by the usage of min-max or z-score normalization.

The min-max normalization for a data point x in a dimension that has the largest value $x_{\max}$ and smallest value $x_{\min}$, is calculated according to Equation 2.4.

$$x_{norm} = \frac{x - x_{\min}}{x_{\max} - x_{\min}} \tag{2.4}$$

The above is different from normalizing by z-score, as given in Equation 2.5, where the mean of the dimension is μ and the standard deviation is σ.

$$x_{norm} = \frac{x - \mu}{\sigma} \tag{2.5}$$

2.5 Principal component analysis

The *curse of dimensionality* can occur when dealing with several dimensions, since the demand for data expands with increasing dimension space. This demand consequently escalates to counteract the increasing sparseness in the data. For data containing a high dimensional space, the technique Principal Component Analysis (PCA) can be applied in order to reduce the number of dimensions of the data while minimizing the information loss.

The process extracts the principal components of the data and changes the basis. The aim is to find a projection of the data where the dimensions of the data are uncorrelated. In order to reduce the dimensions, some principal components can be elected, preferably by order of the variance explained. Besides reducing the number of dimensions, and therefore also the amount of data storage, PCA also has the benefit of serving as feature extraction.

PCA functions according to the following methodology. Given a data set with p number of data points with n dimensions.

$$\mathrm{X} = \left\{x_j^i\right\}_{j=1,\ldots n}^{i=1\ldots p} \text{ and } \mathrm{x}^i \in \mathbb{R}^n, \quad i = 1, \ldots, p \tag{2.6}$$

PCA finds a matrix A:$\mathbb{R}^n \xrightarrow{A} \mathbb{R}^q$, with q $\leq n$. That linearly projects the data set X according to Equation 2.7.

$$Y = AX, \quad \text{with } Y = \left\{y^1, \ldots, y^p\right\} \tag{2.7}$$

The projection matrix A is optimized for minimum loss of the reconstruction. In other words, the loss function to be minimized for is given as Equation 2.8, where the projection matrix A is given as Equation 2.9

$$\left\|A^{-1}y^* - x\right\| \tag{2.8}$$

$$A = \begin{bmatrix} (e^1)^\top \\ (e^2)^\top \\ . \\ . \\ (e^n)^\top \end{bmatrix} \text{ and } \begin{cases} \|e^i\| = 1, \forall i \\ (e^i)^\top e^j = 0, \forall i \neq j \end{cases} \tag{2.9}$$

Thus, the rows of the matrix A are orthonormal vectors that form a basis of $\mathbb{R}^q$. Moreover, given Equation 2.8 a single data point can hence be described in the projected basis as Equation 2.10.

$$y = \sum_{i=1}^{q} \left(\left(e^i \right)^\top x \right) e^i \tag{2.10}$$

The covariance matrix of the data set X, with p data points, is given as Equation 2.11 in the case of zero mean.

$$C = \frac{1}{p} \sum_{j=1}^{p} x^j \left(x^j \right)^\top = \frac{1}{p} X X^\top \tag{2.11}$$

The function from Equation 2.8 can hence be reformulated according to Equation 2.12. Thus, finding the linear combination with maximum variance is identical to find the matrix e^j that minimizes $(e^j)^\top C e^j$ [40].

$$\begin{aligned} &\min_{e^j} \left(e^j \right)^\top C e^j \\ &\text{With the constraints:} \\ &\left\| e^i \right\| = 1, \forall i \\ &\left(e^i \right)^\top e_j = 0, \forall i \neq j \end{aligned} \tag{2.12}$$

This optimization problem can be solved by rewriting the optimization problem with the Lagrange method. The resulting Lagrange function, which is a function of e^i and the Lagrange multiplier λ, are given in Equation 2.13.

$$L(e^i, \lambda) = \left(e^i \right)^\top C e^i - \lambda \left(\left(e^i \right)^\top e^i - 1 \right) \tag{2.13}$$

In order to obtain the solution one has to take the partial derivative with respect to e^i and set it to zero, which is expressed in Equation 2.14.

$$\frac{\partial L(e^i, \lambda)}{\partial e^i} = C e^i - \lambda e^i = 0 \tag{2.14}$$

Hence, the solution is given by the eigenvector of the covariance matrix. The condition regarding the orthogonality, given in Equation 2.12, is thus satisfied since the eigenvalue decomposition of the covariance matrix are orthogonal vectors [41]. The stages for computation can be seen in Algorithm 2.

Algorithm 2: Principal components analysis algorithm

Data: Data matrix X = $\{x_1, ..., x_n\} \subseteq \mathbb{R}$
Result: Eigenvectors P $\subseteq \mathbb{R}^{KxM}$ and Data matrix X^*
initialization $\vec{\mu} = \mathbb{E}(X)$;
$X' = X - \mu$;
C = $X'(X')^\top$;
eigendecomposition $PDP^{-1} = C$;
sort P and P^{-1} by eigenvalues D ;
calculate new features $X^* = XP$

2.6 k-means clustering

In order to group several points into a predefined number of clusters, that is, minimizing a specified metric, k-means clustering algorithm can be used. K-means are non-deterministic, unsupervised, and iterative. The number of clusters, k, is set as a parameter. The clustering method minimize the sum-of-squares criterion given in Equation 2.15 [42],

$$J\left(\mu_1, \ldots \mu_K\right) = \sum_{k=1}^{K} \sum_{i \in C^k} d^p(x, \mu_k) \tag{2.15}$$

where x_i represent the data point i, C^k is the data points attached to cluster k, μ_k the geometric centroid for the corresponding cluster C^k and the distance measure $d^p(x, \mu_k)$ between two N-dimensional points are given in Equation 2.16.

$$d^p(x, y) = \sqrt[p]{\sum_{i=1}^{N} |x_i - \mu_k|^p} \tag{2.16}$$

Where p is referred to as the norm used. The algorithm divides the data into Voronoi cells, as shown in Figure 2.3.

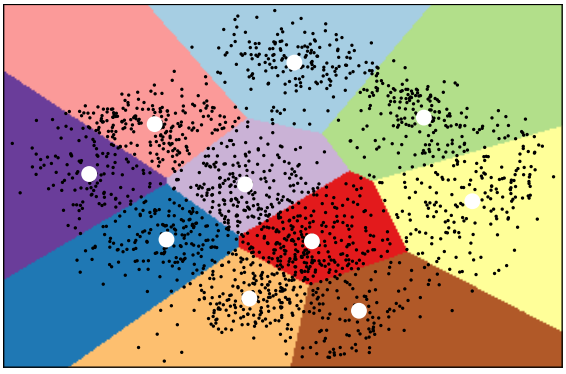


Figure 2.3: Example of Voronoi cells created by applying k-means clustering.

A pseudo code for k-means is given in Algorithm 3.

Algorithm 3: k-means clustering algorithm

Data: Data points X $= \{x_1, ..., x_n\} \subseteq \mathbb{R}^d$, Number of clusters k $\subseteq \mathbb{R}$
Result: A set of k centroids: $C = \{c_1, ..., c_k\} \subseteq \mathbb{R}^d$
initialization $C = \{c_1, ..., c_k\} \subseteq \mathbb{R}^d$ at random;
while *C has not converged* **do**
  $S_i \leftarrow \emptyset, \forall i \in [k]$;
  for $x_i \in X$ **do**
    $j^* = \arg\min_j \|x_i - c_j\|$;
    $S_{j^*} \leftarrow S_{j^*} \cup \{x_i\}$
  end
  $c_j \leftarrow \frac{1}{|S_j|} \sum_{x \in S_j} x, \forall j \in [k]$
end

2.7 Genetic algorithm

GA is a type of EA that is commonly used for function optimization. For a n dimensional function $f(x_1, x_2, ..., x_n)$ the search space is referred to as $\mathbf{x} = (x_1, x_2, ..., x_n)^\top$. In a standard GA, a population consist of a number of individuals. These individuals are all represented by a chromosome that contains all the information about the specified individual. The chromosomes consists of a series of numbers referred to as genes. To retrieve the variable values, the chromosomes need to be decoded in order to evaluate the individual with the use of an objective function, $f(\mathbf{x})$.

The idea behind GA is to apply operations similar to those found in nature to allow a population to evolve (often referred to as generations) towards newer and better solutions. Some of the most common evolutionary operators are selection, crossover, mutation, and elitism [43]. A measurement of the performance of each individual is needed to compare different individuals against each other. This is done using a so-called fitness function. The fitness function has the task of giving numerical measurements on the performance of an individual and is to either be minimized or maximized. Figure 2.4 shows a basic GA with elitism.

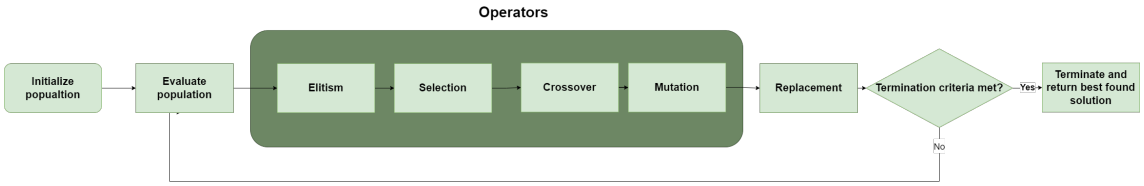


Figure 2.4: Flowchart of a GA utilizing the operators elitism, crossover and mutation.

2.7.1 Operators

The selection process is possible to conduct using several different methods. Two of the most common ones are roulette wheel and tournament select [43]. Roulette wheel builds on the idea that the probability of selecting an individual is proportional to its fitness value compared to the other candidates. While tournament selection relies on a preset probability of selecting the individual with the higher fitness value. Both versions are applicable to select an individual from a larger number of candidates. However, tournament selection will need to do this iteratively, as it can only compare two individuals simultaneously.

Moreover, the crossover is an operator used to combine two parent chromosomes to form a new individual. Crossover is usually done by breaking up the chromosomes at randomly selected indices and then reassembling the split-up parts to form new individuals, see Figure 2.5. If wanted, fixed crossover points are an alternative if one wishes to preserve the chromosome's lengths. The last operator that modifies the structure of a chromosome is mutation. The mutation alters the value of specific genes in the chromosomes. A mutation is applied with a previously defined probability for each gene. Depending on the form of the gene, the mutation is possible to do in multiple ways. However, the common divider is to alter the value of a single gene. Creep mutation can be employed to restrain the magnitude of a mutation when using non-binary chromosomes. The constrain of the magnitude is denoted by creep rate. Figure 2.5 presents an example of a two-point crossover without fixed points, where both the structure and length of the chromosome are modified in this case.

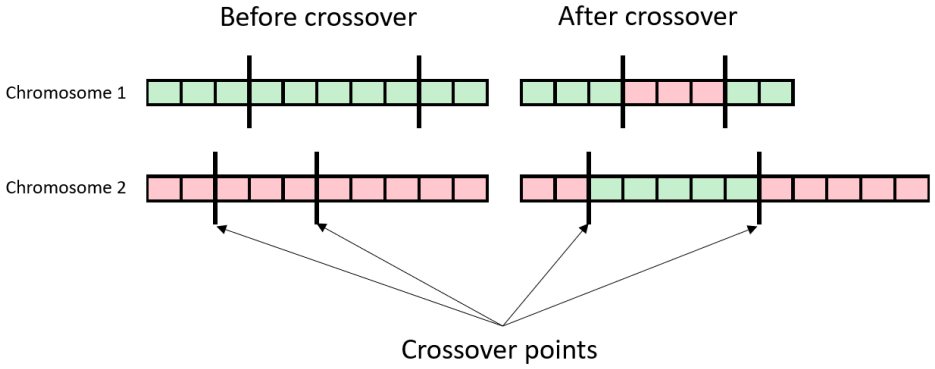


Figure 2.5: Example of two point crossover.

In order to ensure that the fitness value of the best individual in a population does not decrease over time, elitism can be applied. Elitism preserves at least one copy of the individual with the highest fitness value before applying any modifying operators such as selection, crossover, or mutation.

After completing a generation, i.e., the previously mentioned operators have been applied, the existing population is replaced. The fraction of the population that is replaced in each generation is $\frac{k}{N}$. N is the number of individuals in the population,

and k how many individuals are replaced in each generation. One way of doing so is by generational replacement, where the whole population is updated at once, i.e., $k = N$. An alternative is to do it via steady-state replacement, where a fraction of the population is replaced for each generation, i.e., $k = [0, N)$.

Lastly, the GA will run until a termination criterion is met. The criterion is possible to formulate in numerous ways. A few options are a specific number of generations, an objective function value, or a variation of the best objective function value over a specific period.

2.7.2 Multi level selection

The theory of MLS states that selection can be based on multiple levels. That is, an individual's chance of survival is not only based on their ability but also that of the population it is a part of [44]. In Multi Level Selection Genetic Algorithm (MLSGA) the population is divided into several smaller populations, here referred to as subpopulations. In MLSGA the same operations/algorithm is applied to all subpopulations. The contrast is that the different subpopulations are evaluated and compared, hence the multi-level approach. The comparison and elimination are only made in specific instances. This methodology enables the evaluation of different groups of solutions without risking the elimination of non-converged solutions at an early stage of the algorithm.

2.8 Anti aliasing

Anti-aliasing is a technique used to reconstruct and smooth out samples of low resolution. In images, it can be used to avoid so-called "jaggies" that can appear in low-resolution images. See Figure 2.6 as an example. These typically appear when high-resolution images are represented on a screen with a lower resolution. In Figure 2.6, the intensity of the black line is smoothed out. The line still follows the same direction but appears wider to include areas left as white beforehand.

When drawing a line between two coordinates with a finite resolution of the intermediate area, there will always be some jagged edges due to this. By applying anti-aliasing, this behavior can be avoided as best can. The intensity seen in Figure 2.6 could represent to what degree the pixel would have been passed through if the resolution were to be infinite. Using anti-aliasing makes it possible to extract what pixels the drawn line would have passed through in such a case.

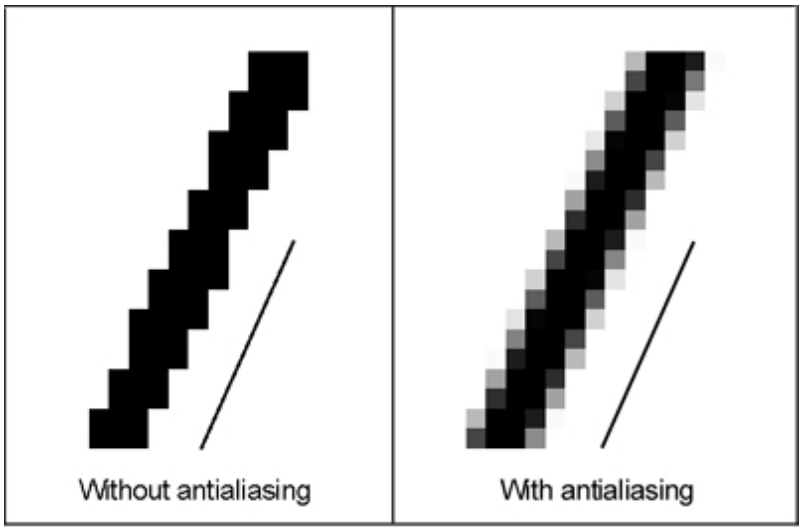


Figure 2.6: To the left is a line drawn in a low resolution image with jaggies. The right is the same line, but where anti-aliasing has been applied and the jaggies are therefore blurred out.

Anti-aliasing can be used to set an intensity between zero and one depending on the distance between the crossed pixel and the parametric line, i.e. a straight line connecting the two points in this case. Using the definition made by Alois Zingl in [45], Equation 2.17 is that of a straight line between two points $P_0(x_0, y_0)$ and $P_1(x_1, y_1)$. Further defining $dx = x_1 - x_0$ and $dy = y_1 - y_0$. The error, e, is defined as Equation 2.17 in a range of $e = [0, 1]$. The previously described intensity can then be assigned according to Equation 2.17 for each crossed pixel as well.

$$\begin{aligned} 0 &= (x_1 - x_0)(y - y_0) - (x - x_0)(y_0 - y_1) \\ e &= (y - y_0)dx - (x - x_0)dy \\ intensity &= 1 - e \end{aligned} \tag{2.17}$$

2.9 Multi objective optimization

Multi Objective Optimization (MOO) can be defined as the optimization of multiple objective functions simultaneously [46]. When solving such an optimization, a trade will appear when no solution can improve an objective function without the cost of impairing another. The set of such solutions will form what is called a Pareto front. Figure 2.7 illustrates this phenomenon in the case of two conflicting objective functions that are to be minimized.

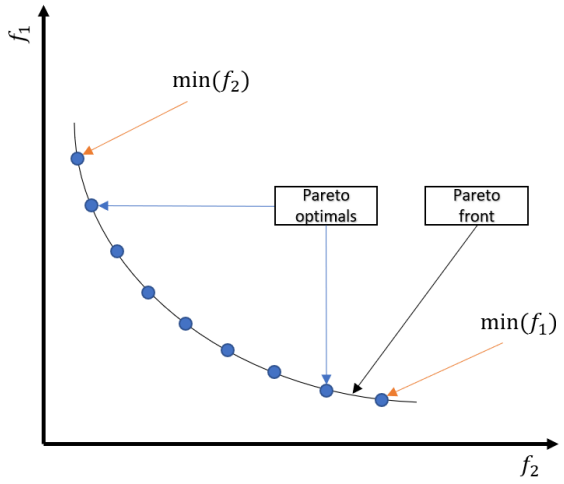


Figure 2.7: A Pareto front for two objectives f_1 and f_2.

The objectives in MOO can be conflicting. Hence, it does not exist a solution that minimizes all objectives simultaneously. However, a solution x can be said to dominate x' if the solution is better in at least one objective but not worse in any other objective denoted $x \prec x'$ Further, a solution y can be said to strictly dominate solution y' if the solution y dominates y' in all objectives, denoted $y \prec\prec y'$

Moreover, all the solutions on the Pareto front can be dominated by another solution. However, the solution on the frontier can not be strictly dominated by another solution. The remaining solution behind the Pareto front is said to be dominated by the Pareto solutions, which are termed non-dominated solutions.

2.10 Vessel kinematics

In total, a vessel has six degrees of freedom. These are the body coordinates(x,y,z) and the rotational coordinates(p,q,r). The most common way is to have the body coordinates centered along the principal axes of inertia [47]. The x-axis corresponds to the longitude axis, the y-axis to the transverse axis, and the z-axis is the normal axis. The rotational coordinates are defined as the rotation around the different axis. More precisely, the rotation around the x-axis is defined as p(roll). Secondly, the rotation around the y-axis is defined as q(pitch). Lastly, the rotation around the z-axis is defined as r(yaw).

Moreover, the pitch of a vessel can be controlled through the trim. The angle of the pitch affects both the performance and the fuel economy. For larger ships, the trim is adjusted before and during the voyage. The loading is placed at determined locations to achieve the desired trim level. Apart from this, ballast water, i.e., water held in tanks in the ship, can also be used to change the trim level [48]. In the case of a smaller vessels, such as a recreational craft, the trim can be adjusted by applying a hydrodynamic force. One way of applying this force is by using trim tabs or interceptors. Both are ways of changing the position of a control surface at

the end of the stern. Due to the systems having control surfaces on the starboard and portside, the vessel's roll can also be controlled. Lastly, if the vessel has power trim, it is also able to alter the pitch angle. This consists of tilting the drive and hence changing the angle of the force generated by the motor.

2.11 Vessel dynamics

There are several factors affecting the dynamics of a vessel traveling at sea. Multiple forces will act on the vessel and affect its motion and the power needed to drive on water. When traveling through and on top of fluids, the dynamic relations are highly complex, that usually can only be solved by numerical approximations. As a result of this, design choices of the hull and engine configuration will highly impact the vessel's behavior and must be accounted for.

2.11.1 Forces acting on a vessel

Two of the main forces acting on the vessel are the vicious and wave-making forces [49]. The magnitude of the viscous force, also known as drag force, is dependent on the relative velocity of the object and the fluid it is passing through. Further, the wave-making is due to the vessel needing to move water to create space for its hull. This operation requires a force to be applied to the water. The result can be seen in waves appearing around and behind the vessel. The estimation of the drag and wave forces are given in Equation 2.18,

$$\begin{aligned} F_D &= \frac{1}{2}\rho v^2 C_D A \quad \textit{(Drag)} \\ F_W &= \frac{1}{2}\rho v^2 C_w L^2 \quad \textit{(Wave)} \end{aligned} \tag{2.18}$$

,where ρ is the density of the fluid, v is the speed relative to the fluid, C_D the drag coefficient, C_w the wave resistance coefficient, A is the cross sectional area and L is the length of the waterline for the vessel.

The general formulas are the same independently of the fluid. However, the characteristics of fluids differ. When a vessel moves through the water, it will move through both water and air. The drag of these two needs to be computed separately. That is because the constant area, drag coefficient, and density will differ. Regarding the wind resistance, a cross-section of the vessel can be considered the area the wind is acting on.

Another great contributor to the characteristics of the forces acting on a vessel is its hull design. A displacement hull will not break the barrier for traveling above the wave in front of the bow. In contrast, both semi-displacement and planning hull will do so. Both of them will generate a compelling amount of lifting force. By applying a large amount of power, the vessel will travel faster than the wave propagation speed, i.e., the speed of the wave. Hence, at this speed, the wave-making

forces will be vastly reduced. The planning hull will go even further. Since the hull will primarily be supported by dynamic pressure, the wetted area will be reduced. The reduced area results in lower drag.

The Froude number (Fr), based on the speed–length ratio, is coupled to the drag coefficient for a given speed of a vessel. Fr is defined in Equation 2.19,

$$\mathrm{Fr} = \frac{u}{\sqrt{gL}} \tag{2.19}$$

where u is the relative velocity between the vessel and the fluid, g is the gravitation and L is the length of the waterline of the vessel. Further, is the Fr of a planning vessel given in Equation 2.20.

$$\mathrm{Fr} = \frac{u}{\sqrt{g\sqrt[3]{V}}} \tag{2.20}$$

Here V is the volumetric displacement of the hull. Measures are often utilized in order to transverse the Fr to the corresponding wave resistance coefficient.

Ordinarily a hull is in displacement when the Fr < 0.4, semi-displacement if $0.4 <$ Fr < 1.0 and planning if $1.0 <$ Fr [50]. Hence, a vessel can be in different modes. The difference lies in the supporting forces, as described in Section 1.1. However, a vessel designed for reaching planning mode is denoted as having a planning hull. Hence, all vessels cannot generate enough speed to reach the semi-planing or the planning mode.

2.11.2 Models for estimating total resistance of a vessel

There are models for estimating the total resistance of vessels traveling through water. Two commonly used ones are Holtrop & Mennen and Savitzky's [51][52]. The usage areas of the two differ in that Holtrop & Mennen is used when traveling in displacing speeds, i.e., up to a Fr of approximately 0.4. Savitzky is instead used for planning hulls and thereby also planning speeds, i.e., Fr of above 1.0. Holtrop & Mennen states that the total resistance of a vessel can be described using equation 2.21. Table 2.1 briefly explain the different variables.

$$R_{total} = R_f(1 + k_1) + R_{APP} + R_W + R_B + R_{TR} + R_A \tag{2.21}$$

Table 2.1: Variables of Holtrop & Mennen

Variable	Description
R_f	Frictional resistance according to the ITTC-1957 friction formula
$1 + k_1$	Form factor describing the viscous resistance of the hull form in relation to R_f
R_{APP}	Resistance of appendages
R_W	Wave-making and wave-breaking resistance
R_b	Additional pressure resistance of bulbous bow near the water surface
R_{TR}	Additional pressure resistance of immersed transom stern
R_a	Model-ship correlation resistance

Further on, Savitzky's method applied as Equation 2.22 to determine the resistance of planning hulls,

$$D = \Delta \tan \tau + \frac{\varphi U^2 C_f \Lambda B^2}{2 \cos \beta \cos \tau} \tag{2.22}$$

where the total drag, D, is computed. In Table 2.2 a description of the variables in the equation is given.

Table 2.2: Variables of Savitzky's method

Variable	Description
Δ	Displacement mass
τ	Trim angle
U	Hull speed
C_f	Frictional coefficient
B	Beam length
β	Deadrise angle
Λ	Mean wetted length-beam ratio
λ	Specific weight of water
φ	λ/g

2.12 Bayesian optimization

Bayesian optimization aims to find the global optimum in a function by sampling intelligently from the parameter space. Hence, one wants to find the minimum for a function $f : \mathcal{X} \to \mathbb{R}$ in some domain $\mathbf{x} \subseteq \mathcal{X}$, as shown in Equation 2.23.

$$\arg\min_{\mathbf{x} \in X} f(\mathbf{x}) \tag{2.23}$$

The optimization method is well suited for optimizing expensive or non-convex functions. Besides this, it is also effective regarding functions with derivatives that are hard to evaluate or ones that do not have a closed-form [53].

The Bayesian optimization relies on the Bayesian theorem[54], given in Equation 2.24.

$$P(A \mid B) = \frac{P(B \mid A) \cdot P(A)}{P(B)} \tag{2.24}$$

Here, $P(A)$ and $P(B)$ are the prior probabilities of observing events A and B. Consequently, $P(A \mid B)$ is the posterior distribution and $P(B \mid A)$ is the probability of B occurring given A. The theorem can be simplified to Equation 2.25. Hence, the conditional probability will only be described as a proportional quantity.

$$P(A \mid B) \propto P(B \mid A) \cdot P(A) \tag{2.25}$$

In Bayesian optimization, A represents the estimated model, and B the observed points from the proper function. Thus, $P(B \mid A)$ corresponds to the probability of observing the data points given the model A. Likewise, will the posterior $P(A \mid B)$ be a scaled version of the probability of $P(B \mid A)$ multiplied by the likelihood of the model $P(A)$. The essence of Bayesian optimization relies on combining the prior model with sampled points from the actual function..

The design of the process for the optimization can be broken down into four steps. The first is initializing a surrogate model, which is a estimate of the true function based on the prior $P(A)$. Next is to find new points to evaluate through an acquisition function, taking the posterior to account. The function is used to maximize the expected utility. It considers both exploration and exploitation in areas with high uncertainty and samples with high values, respectively. A Gaussian Process ($\mathcal{GP}$) is applied to all the evaluated data points in order to build or update the surrogate model. Everything from the second step is repeated until a specified termination criterion is met. A general pseudo-code for Bayesian optimization is given in Algorithm 4.

2.12.1 Gaussian process

$\mathcal{GP}$ is a stochastic process constructing a joint probability distribution, assuming multivariate Gaussian distributions, over the included random variables. In the case of Bayesian optimization $\mathcal{GP}$ is applied for regression. This consist of a function $m(\mathbf{x}) = \mathbb{E}[f(\mathbf{x})]$, where $\mathbf{x} \to \mathbb{R}$. Further, it makes use of a kernel, which is commonly denoted as the covariance function, $k(x, x')$ where $\mathbf{x} \times \mathbf{x} \to \mathbb{R}$. The $\mathcal{GP}$ is formulated in Equation 2.26.

$$y \sim \mathcal{GP}\left(m(\mathbf{x}), k\left(\mathbf{x}, \mathbf{x}'\right)\right) \tag{2.26}$$

The kernel utilized can be adjusted based on assumptions of the data. A $\mathcal{GP}$ regression is illustrated in Figure 2.8.

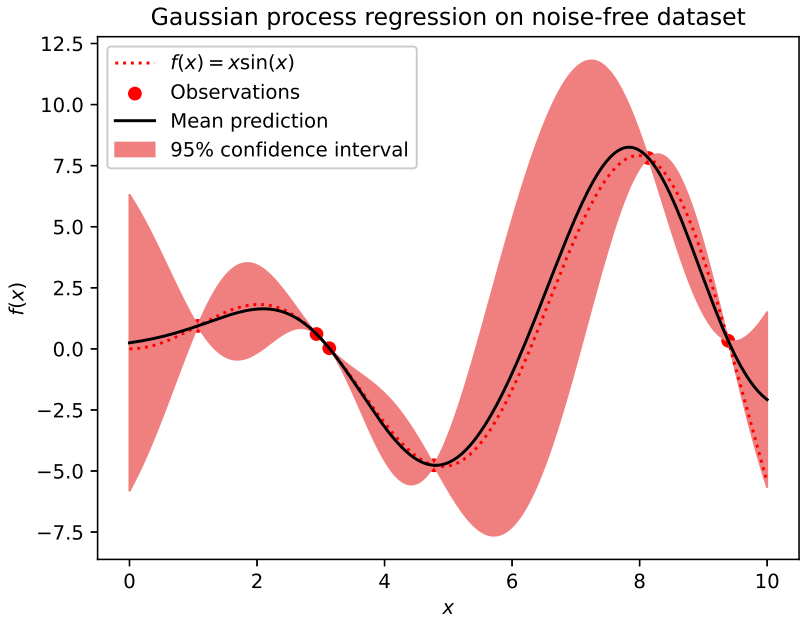


Figure 2.8: Example of $\mathcal{GP}$ regression utilized a function $f(x) = x\sin(x)$. The three dots corresponds to the evaluations, the red area to the prediction within $\pm 2\sigma$ and the black line to the mean prediction $\mu(x)$.

2.12.2 Acquisition function

The purpose of the acquisition function is to derive the next point to evaluate. Hence, any function can be enforced. However, a cleverly chosen function yields the minimization of $f(\mathbf{x})$ when minimizing the acquisition function. Thus, the Expected Improvement (EI) is frequently used. In terms of words, the algorithm calculates the expected improvement for the optimal point. If the real value is less than expected, the point is seen as a local optimum. The algorithm can be formulated as Equation 2.27.

$$\mathrm{EI}(\mathbf{x}) = \mathbb{E}\ \max\left\{f\left(\mathbf{x}^*\right) - f\left(\mathbf{x}^+\right), 0\right\} \tag{2.27}$$

Where $\mathbf{x}^*$ is the parameter searched for and $\mathbf{x}^+$ is the parameter corresponding to the current highest evaluation. The next point $\mathbf{x}^*$ is found according to Equation 2.28.

$$\mathbf{x}^* = \arg\max\ \left(\mathbb{E}\ \max\left\{f\left(\mathbf{x}^*\right) - f\left(\mathbf{x}^+\right), 0\right\}\right) \tag{2.28}$$

The expected improvement of point $\mathbf{x}^*$ in the function can be derived to the one shown in Equation 2.29 [55].

$$\begin{aligned} \mathrm{EI}(\mathbf{x}^*) &= \begin{cases} \left(\mu(\mathbf{x}) - f\left(\mathbf{x}^+\right)\right)\Phi(Z) + \sigma(\mathbf{x})\phi(Z) & \text{if } \sigma(\mathbf{x}) > 0 \\ 0 & \text{if } \sigma(\mathbf{x}) = 0 \end{cases} \\ Z &= \begin{cases} \frac{\mu(\mathbf{x}) - f\left(\mathbf{x}^+\right)}{\sigma(\mathbf{x})} & \text{if } \sigma(\mathbf{x}) > 0 \\ 0 & \text{if } \sigma(\mathbf{x}) = 0 \end{cases} \end{aligned} \tag{2.29}$$

Where μ is the expected value of $\mathbf{x}$, σ is the standard deviation of $\mathbf{x}$, Φ is the cumulative distribution function and ϕ the probability density function for an unit Gaussian distribution.

Algorithm 4: Bayesian optimization algorithm

Data: Objective function f and initial point x_0
Result: Minimum value $min(Y_k)$ and minimizer corresponding to the minimizers x value
initialization: Evaluation of $y_0 = f(x_0)$;
while *Termination criteria not meet* **do**
  fit $\mathcal{GP}_k$ to data X_k and Y_k ;
  get new observation x_{k+1} by solving the acquisition function ;
  Evaluate function $f(x_{k+1})$ to get y_{k+1} ;
  Update the data $X_{k+1} = X_k \cup \{x_{k+1}\}$ and $Y_{k+1} = Y_k \cup \{y_{k+1}\}$
end

2.13 Support vector regression

In order to map an input data X of arbitrary dimensions, into a valued output, SVR can be utilized. The method is linked with the more known Support Vector Machine (SVM). Both are based on statistical learning theory and were proposed by Vapnik [56]. The regression model is non-parametric and instead based on the kernel method. The kernel ϕ transforms the points into a feature space to build a model. In the case of a non-linear kernel, the feature space is projected into a higher dimensional space. Listed in table 2.3 are the different kernels that are commonly used for SVR.

Table 2.3: Common kernels used for support vector regression

Linear	$\phi = x_j' x_k$
Polynomial	$\phi = x_j' x_k^q$
Gaussian	$\phi = \frac{1}{e^{\|x_j - x_k\|^2}}$

The data points X are projected into a feature space, H, as in Equation 2.30.

$$X \to \phi(X) \in H \tag{2.30}$$

In the feature space a linear mapping is then optimized according to Equation 2.31,

$$f(x) = \langle w, \phi(x) \rangle + b, \quad w \in H, b \in \mathbb{R} \tag{2.31}$$

where w is defined as the normal direction of the plane and b as the threshold from the origin in the feature space. An illustration of non-linear SVR can be seen in Figure 2.9.

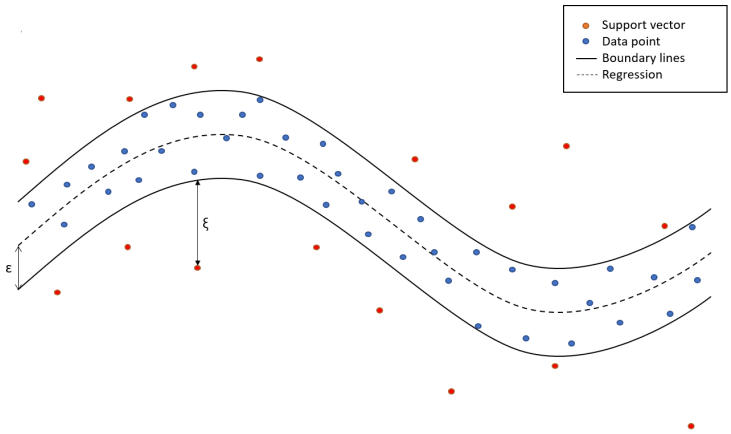


Figure 2.9: An illustration of a non-linear SVR model fitted on a set of data points. The red points are outside the epsilon margin and are hence support vectors.

SVR refers to minimizing the sum of norms for all support vectors, given a parameter $\varepsilon > 0$ that allows for deviations from the function. A support vector is defined as the points that are located at a greater distance than the parameter ε from the function. Hence, the loss function in the feature space is defined as Equation 2.32. The parameter ε is utilized since a model without deviation is inevitable. Thus ε aims at keeping the error within a specific range.

$$|y - f(x)|_\varepsilon = \max\{0, |y - f(x)| - \varepsilon\}, \tag{2.32}$$

Finding the optimal estimate of the function f that minimize the distance is given by the optimization problem in Equation 2.33,

$$\min_w \left(\frac{1}{2}\|w\|^2 + C \cdot \frac{1}{M} \sum_{i=1}^{M} \left| y^i - f\left(x^i\right) \right|_\varepsilon \right) \tag{2.33}$$

where M is the number of data points, and C is a constant parameter that decides the model's regularization. In other words, it is the trade-off between the flatness of the function $f(x)$ and to what amount deviations, more significant than ε, are tolerated.

Moreover, by minimize $\|w\|^2$, it is ensured that the optimization problem is convex. Thus one is guaranteed to find a global optimum. The optimization problem from Equation 2.32 can be formalized as Equation 2.34.

$$\begin{aligned} &\text{minimize} && \tfrac{1}{2}\|w\|^2 \\ &\text{subject to} && \begin{cases} \langle w, \phi(x^i) \rangle + b - y^i \leq \varepsilon \\ y^i - \langle w, \phi(x^i) \rangle - b \leq \varepsilon \end{cases} \quad \forall i = 1, \dots M \end{aligned} \tag{2.34}$$

However, one cannot find a solution to the problem if it is infeasible. Thus, so-called slack variables, ξ_i and $\xi_i^*(i = 1 \dots M)$, are introduced to ensure that there is a feasible solution. Of course, these variables come with a cost since the slack

variables correspond to the cost of a support vector fitted, i.e., the distance from the ε margin. The optimization problem with slack variables can now be written as Equation 2.35.

$$\begin{aligned} &\text{minimize} && \tfrac{1}{2}\|w\|^2 + \tfrac{\mathrm{C}}{M}\textstyle\sum_{i=1}^{M}(\xi_i + \xi_i^*) \\ &\text{subject to} && \begin{cases} \langle w, \phi(x^i)\rangle + b - y^i \leq \varepsilon + \xi_i \\ y^i - \langle w, \phi(x^i)\rangle - b \leq \varepsilon + \xi_i^* \\ \xi_i \geq 0, \quad \xi_i^* \geq 0 \end{cases} \end{aligned} \tag{2.35}$$

The optimization problem in Equation 2.35 can be solved by the use of Lagrange multipliers when setting $\alpha_i, \eta_i \geq 0$ for every inequality [57]. The result of this can be seen in Equation 2.36.

$$\begin{aligned} &\mathrm{L}(w, b, \xi, \xi^*, \alpha_i, \alpha_i^*, \eta_i, \eta_i^*) = \frac{1}{2}\|w\|^2 + \frac{\mathrm{C}}{M}\sum_{i=1}^{M}(\xi_i + \xi_i^*) - \frac{\mathrm{C}}{M}\sum_{i=1}^{M}(\eta_i\xi_i + \eta_i^*\xi_i^*) \\ &- \sum_{i=1}^{M}\alpha_i\left(\varepsilon + \xi_i + y^i - \left\langle w, \phi\left(x^i\right)\right\rangle - b\right) \\ &- \sum_{i=1}^{M}\alpha_i^*\left(\varepsilon + \xi_i^* - y^i + \left\langle w, \phi\left(x^i\right)\right\rangle + b\right) \end{aligned} \tag{2.36}$$

Next part consist of calculating the partial derivatives and setting them to zero. The result of this can be seen in Equation 2.37

$$\begin{aligned} \frac{\partial \mathrm{L}}{\partial b} &= \sum_{i=1}^{M}(\alpha_i - \alpha_i^*) = 0; \\ \frac{\partial \mathrm{L}}{\partial w} &= w - \sum_{i=1}^{M}(\alpha_i^* - \alpha_i)\,\phi\left(x^i\right) = 0 \\ \frac{\partial \mathrm{L}}{\partial \xi_i^*} &= \frac{C}{M} - \alpha_i^{(*)} - \eta_i^{(*)} = 0 \\ \frac{\partial \mathrm{L}}{\partial \xi_i} &= \frac{C}{M} - \alpha_i - \eta_i = 0 \end{aligned} \tag{2.37}$$

By combining Equation 2.36 and 2.37, the dual optimization problem can be formulated, such as Equation 2.38,

$$\begin{aligned} &\max_{\alpha,\alpha^*} && \begin{cases} -\frac{1}{2}\sum_{i,j=1}^{M}(\alpha_i^* - \alpha_i)\left(\alpha_j^* - \alpha_j\right)\cdot k(x^i, x^j) \\ -\varepsilon\sum_{i=1}^{M}(\alpha_i^* + \alpha_i) + \sum_{i=1}^{M} y^i(\alpha_i^* + \alpha_i) \end{cases} \\ &\text{subject to} && \begin{cases} \sum_{i=1}^{M}(\alpha_i^* - \alpha_i) = 0 \\ \alpha_i^*, \alpha_i^i \in \left[0, \frac{C}{M}\right] \end{cases} \end{aligned} \tag{2.38}$$

where $k(x^i, x^j) = \phi(x^i)\phi(x^j)$. Solving the optimization problem yields the Lagrange multipliers to be able to solve the estimate of the optimal projection vector, given in Equation 2.39.

$$w = \sum_{i=1}^{M}(\alpha_i^* - \alpha_i)\,\phi\left(x^i\right) \tag{2.39}$$

The offset b, in Equation 2.31, is obtained by solving for the Karush-Kuhn-Tucker (KKT) conditions given in Equation 2.40.

$$\begin{aligned} &\alpha_i \left(\varepsilon + \xi_i + y^i - \left\langle w, \phi\left(x^i\right)\right\rangle - b\right) = 0 \\ &\alpha_i^* \left(\varepsilon + \xi_i^* - y^i + \left\langle w, \phi\left(x^i\right)\right\rangle + b\right) = 0 \\ &\text{and} \\ &\left(\frac{C}{M} - \alpha_i\right)\xi_i = 0 \\ &\left(\frac{C}{M} - \alpha_i^*\right)\xi_i^* = 0. \end{aligned} \tag{2.40}$$

When the Lagrange multiplier α_i takes the value $\frac{C}{M}$, the corresponding slack variable ξ_i can take any value. Hence, the point can be anywhere outside the ε-margin. Moreover, the computational cost of the regression function grows linearly with the number of data points. An advantage of SVR is that the time complexity does not depend on the dimensionality of the input space[58]. Instead the complexity solely depends on the number of support vectors, i.e. when $(\alpha_i^* - \alpha_i) = 0$ [57].

2.14 Cross-validation

In terms of measuring the generalized performance of a model, cross-validation is a viable option. In other words, the technique serves to validate the statistical generalizability of a model to independent data sets. The outcome is interpreted to detect if a model suffers from overfitting or bias. Overfitting is when a model has accommodated patterns or other unwanted biases in the given data set too closely. The model will perform much worse on new data sets than the previous training results. Several different variants exist in the area of cross-validation. Overall, the methods are divided into two subareas of exhaustive and non-exhaustive cross-validation.

Exhaustive cross-validation consists of methods that divide the training and validation data into different partitions. An example of this is the leave-p-out cross-validation. It consists of using p observations as the training dataset and the rest as the validation dataset. This operation is repeated until all datasets have been utilized for training.

Non-exhaustive cross-validation, on the other hand, does not compute all possible ways of splitting the data set into training and validation. This type of validation includes the widely used k-fold cross-validation. The method divides the samples into n different partitions of equal size, where k is used to train the model, and the rest are used to validate the model. This process is repeated for the following k partitions until all $\frac{n}{k}$ models have been utilized for training.

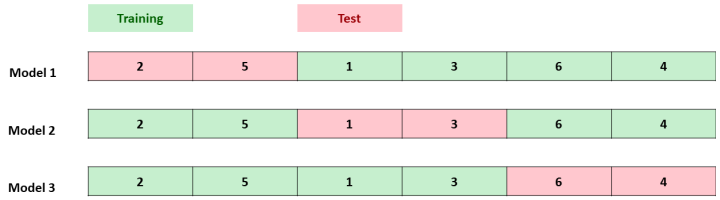


Figure 2.10: K-fold cross-validation with k=6 and n=2. In total three different models will be trained and tested.

2.15 Particle swarm optimization

PSO is an iterative EA used to find optimal solutions to an objective function. The algorithm is based on the traits of swarms and their ability to utilize each other [43]. The swarm, also known as the population in PSO, consist of several particles that all have their specific position(x) and velocity(v). These attributes are described using the same number of dimensions as the optimization problem the algorithm is being used to solve. The position is seen as the proposed solution, while the velocity describes how the position will change over time.

During PSO, both the position and velocity of all particles will be updated in each iteration. The update step will be influenced by the particle's current position relative to its previous best solution and the best solution of the swarm. Different settings can be specified for the relative weights between these two and their influence in the update step of each particle. The most common configuration is to set them equal [43]. Furthermore, the velocity of all particles is restrained to hinder the swarm from diverging. The restrain is to sustain the swarm behavior of the algorithm.

Both the position and velocity of the particle is initialised by uniform sampling in a given range, $[\mathbf{x}_{min}, \mathbf{x}_{max}]$. This type of initialization is to initiate and explore a greater area in the earlier phase of the algorithm. Since the velocity of all particles will be affected by the swarm's best solution, those initially better solutions will attract the rest of the particles in the swarm. In Figure 2.11 a basic PSO algorithm is displayed and in pseudo-code as well in Algorithm 5.

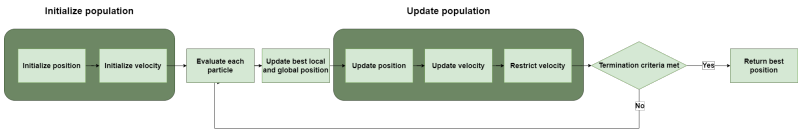


Figure 2.11: Flowchart of a PSO algorithm

Algorithm 5: Particle swarm optimization

Data: Swarm size(N), $f(x)$, dimensions, Δt
Result: Solution of optimization problem
for $i \leftarrow$ *to swarm size* **do**
    $x_{ij} \leftarrow$ uniform sample, i=1,..,N, j=1,...,n;
    $v_{ij} \leftarrow$ uniform sample, i=1,..,N, j=1,...,n;
end
while *Termination criteria not met* **do**
    Evaluate objective function and update best solutions;
    for $i \leftarrow$ *to swarm size* **do**
        if $f(x_i) > f(x_i^{pb})$ **then**
            $x_i^{pb} \leftarrow x_i$;
            if $f(x_i) > f(x^{sb})$ **then**
                $x^{pb} \leftarrow x_i$;
            end
        end
    end
    for $i \leftarrow$ *to swarm size* **do**
        $v_{ij} \leftarrow$ updated velocity, i=1,..,N, j=1,...,n;
        if *Velocity to large* **then**
            Set velocity to limit;
        end
        $x_{ij} \leftarrow x_{ij} + v_{ij}\Delta t$, i=1,..,N, j=1,...,n;
    end
end

3
Methods

Chapter three explains the methodology used for the work in the thesis. The chapter is divided into two parts, one describing the path planning system and the other the speed profile. Further, is the evaluation process used to produce the result of the thesis described.

3.1 Path planning

The proposed method for generating a path in this project was to apply MOO on a set of generated paths via the use of a MLSGA. In the upcoming text, the methodology is explained in more detail. Figure 3.1 provides an overview of the different areas of the system as well as the flow in between.

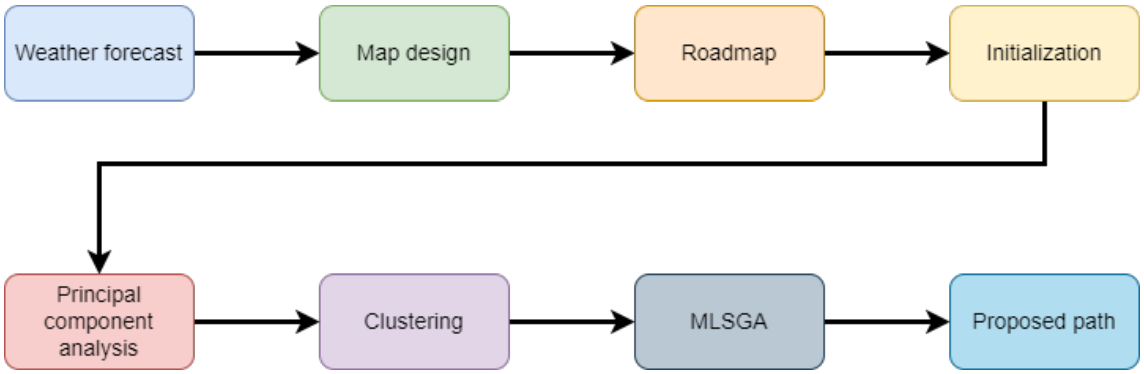


Figure 3.1: Flowchart giving a brief overview of the path planning system.

3.1.1 Import image and Identify land

The first step of the system consisted of creating the environment utilized in the path planning. The ENCs are accessible to two different entities, producers and utilizers. For the Swedish coast, the major company that distributes nautical charts with high-level precision are Hydrographica and sjöfartsverket [59][60]. However, these nautical charts are only sold to distributors of ECDs. From these, one can buy an ECD containing the ENC. However, one cannot access the data of the ENC itself, only view the ENC on display. One viable option that was adapted in this thesis consists of extracting the information using image processing algorithms.

Consequently, the area of interest, i.e., the area between and surrounding the start

and goal position, was loaded into the system as an image. The depth levels in a ENC are in discrete scale. Hence, masking was utilized to obtain the different areas of sea depth and land in the image. Following this, erosion and later dilation, with different kernel sizes and iterations, were applied to remove artifacts in the image. These operations were needed because the masking leaves out less important areas, e.g., sea marks, sea lanes, and borders. These operations resulted in coordinates that would make up what was denominated as the area of interest. Given its specified depth, this area could be described as all positions that the vessel could travel over.

Lastly, the ENC are only approximations of the natural environment and contain measurements with differing amounts of inaccuracies. Consequently, common knowledge for experienced operators on the sea is to hold out from the land area and dangerous rocks in the water when traveling outside of sea lanes. Thus Gaussian blur was applied to the infeasible area. The operation added decaying values, corresponding to a Gaussian function, for pixels close to infeasible areas.

3.1.2 Weather forecast

The weather forecast was loaded from a relevant weather service depending on the area of interest. The forecasts are numerical simulations that were given as General Regularly-Distributed Information in Binary Form (GRIB) files. In the initial phase of the program, the latest forecast was retrieved. An API from the weather services was utilized for retrieving the latest forecast.

The wind was composed of two separate components, U and V. Where U is the wind component parallel to the x-axis, i.e., longitude, and V is the component parallel to the y-axis, i.e., the latitude. The wave forecast contained the height of the waves, in meters, for the specific area. The two wind components were converted into angle and magnitude to have a uniform coordinate system.

Once the weather forecast had been retrieved and transformed, only the area of interest was stored. Since the resolution and scale of the area of interest in ENC can be different, the weather forecast was interpolated using the second-degree spline interpolation.

3.1.3 Initialization

A roadmap inspired by PRM was constructed, given the surrounding environment and the start and end position to enable the initialization of paths. The same methodology as PRM was applied in that each vertex would be coupled with several neighbors. However, instead of randomly placing the vertices, they were sequentially and uniformly placed throughout the map in the FC-Space. Moreover, the number of edges an initialized vertex could make was restricted to control the complexity of the roadmap.

Each path consisted of a sequence of vertices on a map, where every vertex was being represented with an x and y coordinate as the map was in the form of a 2D-coordinate system. The full paths were constructed by connecting the vertices via straight lines throughout the sequence. The intermediate coordinates needed to be recovered to compute the cost of traveling between the vertices. The recovery was made using anti-aliasing. The corresponding objective values could be computed using these coordinates. Using anti-aliasing made it possible to also retrieve a weight of all crossed coordinates given the error in Equation 2.17. These weights were used when evaluating the different objectives, as they would represent to what extent the coordinate in question was traversed. A higher value indicated a greater weight and would contribute more to the evaluation.

The creation of the initial paths consisted of a form of random walk described in Algorithm 6. Initialization of each walk began in the defined starting position. After that, a vertex was uniformly sampled from its neighboring vertices in the graph. The random walk continued until the route reached the goal position or was deemed too long and would then be terminated. To classify a path as too long, the length of a path would need to exceed $\frac{1}{10}th$ of the total number of vertices in the graph. This ratio was according to trial and error, considering the computational time versus the resulting performance. This process was repeated until the size of the population had reached a preset number of routes.

Algorithm 6: Generate route

Data: Start vertex, end vertex
Result: New route
route ← start vertex;
current vertex ← start position;
while *current vertex* ≠ *end vertex* **do**
    Available vertices ← neighbouring vertices;
    next vertex ← uniform sample of Available vertices;
    Append *next vertex* to route;
    current vertex ← next vertex
end

To avoid the unwanted behavior of internal cycles, i.e., loops within the routes. Their occurrence needed to be addressed as they would likely appear and only add complexity to the routes. The process of handling this situation was to identify all vertices that occurred more than once in the route and remove the intermediate vertices between those indexes. All cycles in the original route were eliminated this way. The remaining route could be ensured that it would not contain any cycles. Hence, it fulfills the definition of a path. The process is shown in Algorithm 7.

Algorithm 7: Generate population

Data: Population size, start position, end position, probabilistic roadmap
Result: A set of paths that form a population
for $i \leftarrow$ *to population size* **do**
  route $\leftarrow$ random walk;
  while *route has not reached goal node* **do**
    route $\leftarrow$ random walk
  end
  Identify multiple occurring nodes;
  for $j \leftarrow$ *to number of multiple occurring nodes* **do**
    Remove intermediate nodes from route
  end
  Add path to population
end

3.1.4 Clustering of paths

The next step consisted of applying a MLSGA. Therefore, the population needed to be divided into several smaller populations, now referred to as subpopulations. Several properties were investigated to give measurements of their behavior to enable the possibility of clustering the paths. These were the length, sum of directional changes, and a distribution measurement concerning where the path had been active within the map. The last property was achieved by dividing the map into N number of equally large areas and measuring the relative sojourn in each area.

PCA was used to find the most significant properties for the current environment. Before PCA was applied, the dimensions were normalized in order to retrieve unit variance and hence weigh each dimension equally. By the use of PCA, the number of dimensions was reduced, which was beneficial since k-means can suffer from the curse of dimensionality. The new basis was selected only to contain the three dimensions that described the most variance. After extracting the properties, k-means clustering was carried out to cluster the population in a predefined number of clusters.

3.1.5 Optimisation of paths

A GA was utilized to evolve the different subpopulations. The GA used operators in a usual manner except for the mutation. The selection stage consisted of using tournament selection with a selection size of two. Furthermore, a non length preserving two-point crossover was used between different paths to enable the exchange of partial solutions. In the following stage, elitism was applied. Finally was, the termination criteria set such that a predefined number of generations should have passed.

Moreover, a mutation was defined as altering the coordinates of a vertex, given a mutation probability. Both a varying mutation rate and creep mutation using

an altering creep rate was applied for this project. The variation meant that the possibility of a mutation happening and the distance that a vertex could move via mutation varied over time.

The mutation probability was defined as $\frac{N}{L} \cdot R$ where N was a predefined integer, L the length of the chromosome, i.e., the number of vertices the path consisted of and R was a predefined reduction factor of the mutation probability. Using this method would allow for more significant alterations of the paths in cycles, such that as time moves on, the average number of altered vertices would drop from N down to 1 before being reset to N.

Using a varying creep rate allowed the algorithm to perform more considerable alterations in the initial run portion and then diminish over time. The maximum distance one node could be altered in either direction was initially set to $\frac{1}{5}th$ of the resolution of the image in that direction. During the initialization, most parts of the total search space were expected to have been explored. The different clustered populations should then have focused on exploring different local minima. As more generations passed, more focus was placed on making finer adjustments, i.e., exploitation, to evaluate these different minima.

The creep rate were separate for the possible alterations in the x and y direction. The current rate was determined according to Equation 3.1. The terms res_x and res_y are the resolutions in each dimension of the image.

$$\begin{aligned} span_x &= \Big[0, \quad \Big(1 - \frac{\text{generations passed}}{\text{total generations}}\Big) \cdot \frac{res_x}{5}\Big] \\ span_y &= \Big[0, \quad \Big(1 - \frac{\text{generations passed}}{\text{total generations}}\Big) \cdot \frac{res_y}{5}\Big] \end{aligned} \tag{3.1}$$

In order to minimize the runtime of the program, only a slice of each subpopulation was sampled. The sampling evaluated all paths and picked the best candidates for each subpopulation. In doing so, the characteristics of each subpopulation were sampled.

Each of these subpopulations was passed through the GA to evolve into improved solutions. After all subpopulations had passed through, they were all evaluated as groups. Thus, they were given a fitness value based on the average fitness of the top slice of the subpopulation and the best individual solution. This was done to exclude paths that had been greatly distorted via crossover or mutation and would give a misleading result.

The subpopulation with the lowest score would be eliminated from there. This procedure was repeated until only one subpopulation remained. The best-found solution for the remaining population was presented as the proposed path. In Figure 3.2 an overview of the procedure is presented.

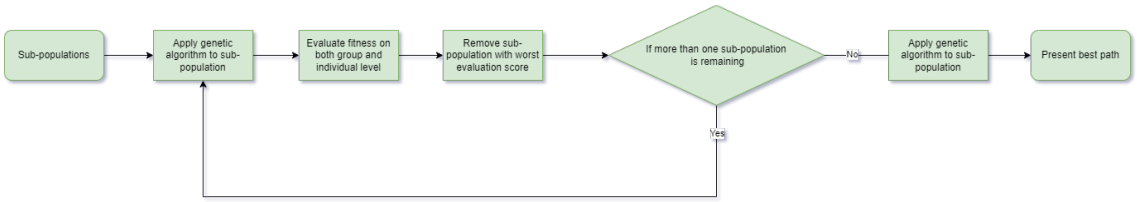


Figure 3.2: An overview of the selection process utilized in by the path planning system.

3.1.6 Optimization objective

The MLSGA was set to optimize against an objective function. The system had three objectives that were optimized to obtain the best solution. Hence, the problem was described as a MOO problem. The sections below explain the different objectives and how they affected the route.

3.1.6.1 Drag from water

One of the largest fuel consumption factors on vessels is the drag from the water. To include the influence of drag as an objective, the estimated drag force was computed given the vessel model and its speed.

Previous studies have been done to estimate the drag of vessels given their speed [61]. They implemented both Holtrop & Mennen and Savitsky method for their areas of interest regarding speed. Combining these results and interpolating the intermediate values, it was possible to derive a simplified model over the drag as a function of its speed. Table 3.1 and 3.2 show the resulting drag of the two methods and Figure 3.3 show the combined results and interpolated function. Using data from previous research enables the building of a sufficient but not perfect representation of the drag.

Table 3.1: Estimated drag, during displacement speed, using Holtrop & Mennen method, for

Speed(Kn)	**Drag(N)**
4	358
5	729
6	1628

Table 3.2: Estimated drag, during planing speed, using Savitsky method

Speed(Kn)	Drag(N)
12	2604
16	3114
20	3379
24	3690
32	4641

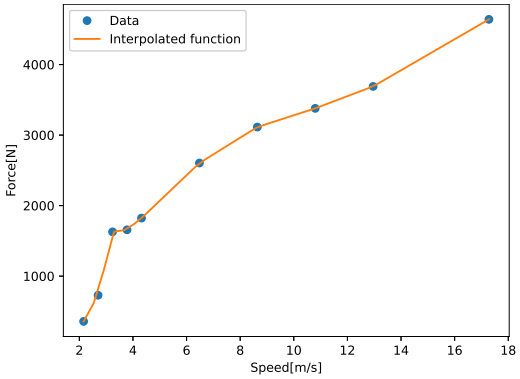


Figure 3.3: Estimated drag for an X-Shore boat. The first three data points correspond to the Holtrop & Mennen method, and the rest to the Savitsky method. Between the data points, a line is interpolated.

For each pixel that the vessel traversed, the value of the drag was multiplied with the corresponding intensity described in section 2.8. The sum of all such values where the contribution to the evaluation of each path.

3.1.6.2 Drag from wind

The drag of the wind was computed using Equation 2.18. The area used was based on measurements of the vessel model, the same one used to estimate the drag from the water. The wind was divided into two components, one along the vessel's heading and the other perpendicular to that. Two cross-sectional areas were used, one for each component. Depending on the relative speed between the vessel and the wind, the wind working along the vessel's heading could have a positive or negative impact. However, the perpendicular wind could only impact negatively.

Throughout the path sequence, the vessel's heading was required to compute the relative speed between the vessel and the wind. The vessel's heading was computed using the given coordinates of the sequence. The angle of each segment was computed and later used to derive the relative force acting on the vessel for each pixel that it passed through. Given a path of length N and with x_i and y_i as its coordinates in each dimensions. Equation 3.2 shows the resulting relative angles.

$$\alpha_i = \arctan\left(\frac{y_{i+1} - y_i}{x_{i+1} - x_i}\right), \quad i = 1, ...N - 1 \tag{3.2}$$

3.1.6.3 Wave height

The evaluation of waves was conducted using a similar methodology as for the drag forces. The wave height was extracted from the weather forecast. The corresponding values were summed up given the pixels the path traversed through. The total sum corresponded to the objective value.

3.1.7 Internal Parameters

Both the roadmap and MLSGA have several parameters that affected both the performance as well as runtime. The MLSGA is also dependent on another set of parameters, shown in Table 3.3.

Table 3.3: Parameters used by the MLSGA in the path planning system.

Parameter	**Description**
Generations	Number of iterations
Original population size	Number of individuals initially generated
Mutation probability	Probability of mutation
Crossover probability	Probability of crossover
Reduction	Reduction of mutation probability for each generation
Size of sampled population	Size of each sub population in MLSGA
Number of sub populations	Number of sub populations in MLSGA

The tuning was centered around finding a setup that gave the algorithm the best basis for finding an as optimal solution as possible, without resulting in an excessively long runtime. By trial and error, the parameters were tuned to give the best performance in the largest amount of environments.

3.2 Speed profile

The speed profile aims to minimize the energy consumption by choosing the most optimal configuration used for traveling from the start position to the desired end location. This is executed by collecting data and building a model over the system using SVR. The hyperparameters are automatically optimized through the usage of Bayesian optimization. Lastly, the configuration that yields the lowest consumption was estimated. This is executed through the usage of PSO. If a vessel had more than one configuration variable, e.g. speed and trim, a GA is also applied to get to the point as energy-efficient as possible. In Figure 3.4 an overview of the system can be viewed.

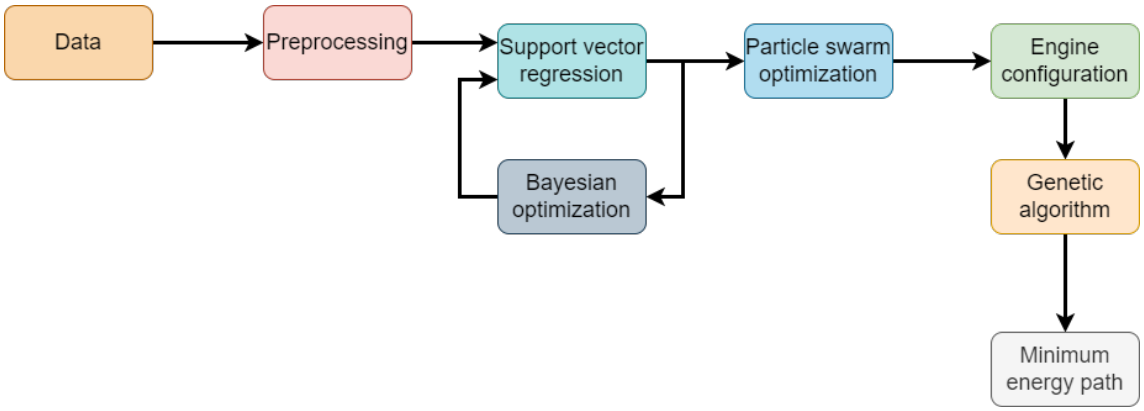


Figure 3.4: Overview of the speed profile system.

3.2.1 Data

In the actual application of the system, data was expected to be gathered and stored during the vessel's travel. For the development phase of the speed profile, already collected data was used due to simplicity. This data originated from three different vessels, shown in Table 3.4. The first was a water taxi, and the second and third were test boats.

Table 3.4: Vessels used to build the speed profile during the development.

Vessel	Driveline	Hull
Vessel 1	Twin motor installation IPS	Planning
Vessel 2	Twin motor installation Outboard	Planning
Vessel 3	Triple motor installation IPS	Planning

The stored properties of the collected data, for the vessels in Table 3.4, was timestamp, fuel rate, RPM and SOG. All Null values in the table were dropped. Further, the vessels Speed Over Ground (SOG) was converted from m/s to knots. Then the fuel economy, corresponding to liter consumed per nautical mile, was estimated. In order to obtain relevant values on the fuel economy, all data points having a velocity

less than 5 knots were excluded. Beyond this, every datapoint has an acceleration as well. The acceleration was estimated by calculating the sampling time from the previous measure. The estimated acceleration is given as the change in velocity between the two measures divided by their sampling time. If the sampling time is larger than a predefined number of seconds, the acceleration is set to NaN as Equation 3.3. This is because too large sampling implied that the vessel had stopped between the measure. Thus the yielded result would not be valid.

$$a_i = \begin{cases} \frac{Vesselspeed(SOG)_{i+1} - Vesselspeed(SOG)_i}{t_{i+1} - t_i} \text{ if } t_{i+1} - t_i \leq \Delta t_{max} \\ \text{NaN if } t_{i+1} - t_i > \Delta t_{max} \end{cases} \tag{3.3}$$

In the case of real-time collection, the number of data points stored can be extensive. Hence, a smaller number of data points was sampled to build the model. Due to SVR finding the solution that minimizes the total error, an equal amount of samples over the area is desired. Thus, the next step was to discretize the values into different bins. The bins consisted of the value $v_{i=5,6,...,N}$, corresponding to a speed of 5 knots up to the maximum speed N recorded for the vessel. Every data point was affiliated to the bin equal to its speed rounded downwards. From these bins, a predefined number of samples was sampled.

3.2.2 Hyperparameter tuning

In order to optimize the performance of a machine learning model the hyperparameters had to be tuned. Several different methods exist to search the parameter space efficiently in this area. For the system to adaptively decide which hyperparameters yields the best performance, this needed to be accomplished without manual interaction. The model evaluation of SVR can be time consuming. Hence Bayesian optimization was utilized for searching the parameter space. One of the algorithm's foremost use cases is for functions that are time consuming to evaluate, as described in Section 2.12. The kernel was specified as Gaussian, and ε was set to 0.001. Hence, the parameters optimized for were C and the kernel size gamma.

Subsequently, cross-validation was implemented to ensure that the model would be more robust to overfitting and bias. The cross-validation consisted of implementing k-fold cross-validation, where k was set to 10. Once the hyperparameter had been optimized, a final model was built with the hyperparameter that yielded the highest score during the tuning. The SVR model was built utilizing the dimensions SOG, acceleration, fuel economy. In the case of trim, this dimension was included as well.

3.2.3 Optimal configuration estimation

The model could now be utilized to serve its purpose. That is to find the optimal velocity concerning fuel consumption. However, it also had to consider other aspects. Firstly, the amount of energy consumed to reach the optimal state. Moreover, it could also be subject to time constraints, i.e., if one wants to reach the final destination within a given time. Hence, the lowest consumption per nautical mile did not necessarily have to be the optimal state.

To find the optimal velocity, given these condition, PSO was utilized. It was applied to detect the estimated lowest fuel economy of the vessel, considering the given bounds. The search space consisted of the dimensions included in the SVR model. In three dimensions, i.e., speed, acceleration, and fuel economy, the PSO objective function was set up according to Equation 3.4.

$$Fuel = Fuel_{steady} + Fuel_{acc} \tag{3.4}$$

$Fuel_{stedy}$ corresponds to the steady state fuel consumption, described in Equation 3.5 and $Fuel_{acc}$, corresponds to the fuel consumption for accelerating, shown in Equation 3.6,

$$Fuel_{steady} = f\left(x, 0\right) \cdot \left(\text{Total distance} - \frac{x^2}{\dot{x}}\right) \tag{3.5}$$

$$Fuel_{acc} = \frac{1}{5}\sum_{k=0}^{4} f\left(5 + \frac{(x-5)k}{4}, \dot{x}\right) \cdot \left(\left(5 + \frac{(x-5)k}{4}\right) \cdot \left(\frac{x-5}{\dot{x}}\right) + \left(\frac{(x-5)^2}{10\dot{x}}\right)\right) \tag{3.6}$$

where $f(x, \dot{x})$ is the support vector function predicting the fuel economy at the velocity x and acceleration $\dot{x}$. For simplicity the acceleration was approximated to be constant.

In the case of more than two dimensions, e.g., with power trim angle or interceptor angle, a GA was applied in order to find the Minimum Energy Path (MEP). The starting point was set to $(0, 0)$, and the desired endpoint was set to the minimum found by PSO. The initialization of the paths was performed through straight lines from the start to the end position. Constraints, corresponding to the physical limits of the system was set. The GA then optimized against the estimated fuel consumption. Hence, for each point in the graph, the fuel consumption was calculated by inserting the coordinates of the path. The resulting path corresponds to the configuration estimated to yield the least energy to reach the optimal state.

3.3 Evaluation

Validation of the system was needed in order to evaluate performance. The validation was executed in two different main categories. The first one was for the path planning algorithm and the second one for the speed profile.

3.3.1 Evaluation of Path planning algorithm

A planned path for voyage traveling can be evaluated under several diverse sets of criteria. The one of interest for this project was the resulting fuel consumption. Suggested routes from the path planning system were compared to some predefined routes to evaluate performance. The routes were performed in two different settings. First, two routes were planned and utilized in real-world conditions. Then a different route was planned in a more complex environment to evaluate the algorithm's feasibility.

Two different candidates drew the routes. Both had been educated about the Swedish regulations of mastering yachts, where voyage planning is an element. However, these routes were created under three consistent guidelines. As previously mentioned, voyage planning mainly relies on traveling by the sea lines. Hence, waypoints for the route were set within the closest sea lines, where available, until the desired destination. Lastly, a new waypoint was only added when there was a need to change the heading to reach the goal destination. That is, no additional waypoints were introduced to create smoother directional changes.

The methodology of this operation consisted of traveling the planned routes of the system and the predefined routes drawn by the candidates while acquiring the resulting fuel consumption. The traveling was done with the boat model Sessa Key Largo 30. The destination of the predefined routes was given in Table 3.5. The routes were additionally evaluated in the form of the score yielded by the predefined paths compared to the route suggested by the algorithm. That is, the same evaluation function used by the MLSGA.

Table 3.5: Locations of the routes examined, with their corresponding start and end coordinates.

Name of location	Start coordinate	End coordinate
Björkö	57.741633, 11.661600	57.738967, 11.708750
Grötö	57.681783, 11.692200	57.718233, 11.676400
Hvaler(Complex C-Space)	58.975703, 11.054759	59.128089, 11.057619

3.3.2 Evaluation of Speed profile

The performance of the speed profile needs to be measured to quantify its performance and robustness. To measure the boat's performance, so-called sea trials were done. This consisted of logging the performance of the boat in the sea. The tests were carried out by driving the boat up to evenly spaced intervals of RPM and logging its performance. The generated model was then evaluated against the sea trial results for the case of the vessels in table 3.6.

Additionally, previously collected data from the different vessels were used to build speed profiles. The three boats used for this are given in Table 3.6. The acquired data proceed according to the methodology in Section 3.2. However, the stored

parameters of the vessels differed. Only one of the three vessels, given in Table 3.6, had collected Speed. Hence, the other was evaluated according to RPM against fuel rate. The reason for this was to obtain the result of building models for other vessel properties. While evaluation of different properties might not yield fully transferable result, it still had the purpose of demonstrating the function of building models for a vessel.

The models were built by randomly sampling 8000 data points from all stored data. Given this data, 200 data points from each bin were randomly sampled. Bayesian optimization was applied in the last step to find the optimal hyperparameters. In total, ten models were built with different random samples. The actual evaluation consisted of calculating the Mean Absolute Error (MAE), Root Mean Square Error (RMSE) and Median Absolute Deviation (MAD) for the models in comparison to the measured sea trial. These measurement will, from here on, be referred to as performance indicators. Due to the fact that the sea trial only was recorded at evenly spaced intervals, only these points were evaluated against. For simplicity, all evaluations were performed on data points yielded by the starboard engine.

Table 3.6: The vessels used to build two-dimensional speed profiles, evaluated against the corresponding sea trial.

Vessel	**Driveline**	**Variables**
Vessel 1	Twin motor installation IPS	Fuel rate and Speed
Vessel 3	Twin motor installation Straight shaft	Fuel rate and RPM
Vessel 4	Twin motor installation IPS	Fuel rate and RPM

The properties of powertrim and interceptors were also evaluated. Since no previous collected data was available all data for the model had to be gathered. The vessels utilized are given in Table 3.7. These operations consisted of driving the vessel at various, evenly spaced speeds. During this, the properties were varied, i.e., the angle of the interceptor or power trim. Likewise, the speed profile in two dimensions, the model was then built according to the methodology stated in Section 3.2. However, in the case of powertrim and interceptor, sea trials were available to validate the performance. Thus, only the functionality could be demonstrated. Hence, it was not possible to quantify the result in terms of performance, possible improvement, or deterioration.

Table 3.7: The vessels used to build three-dimensional speed profiles.

Vessel	**Driveline**	**Variable**
Vessel 2	Twin motor installation Outboard	Fuel rate, speed and power trim
Vessel 5	Triple motor installation IPS	Fuel rate, speed and interceptor

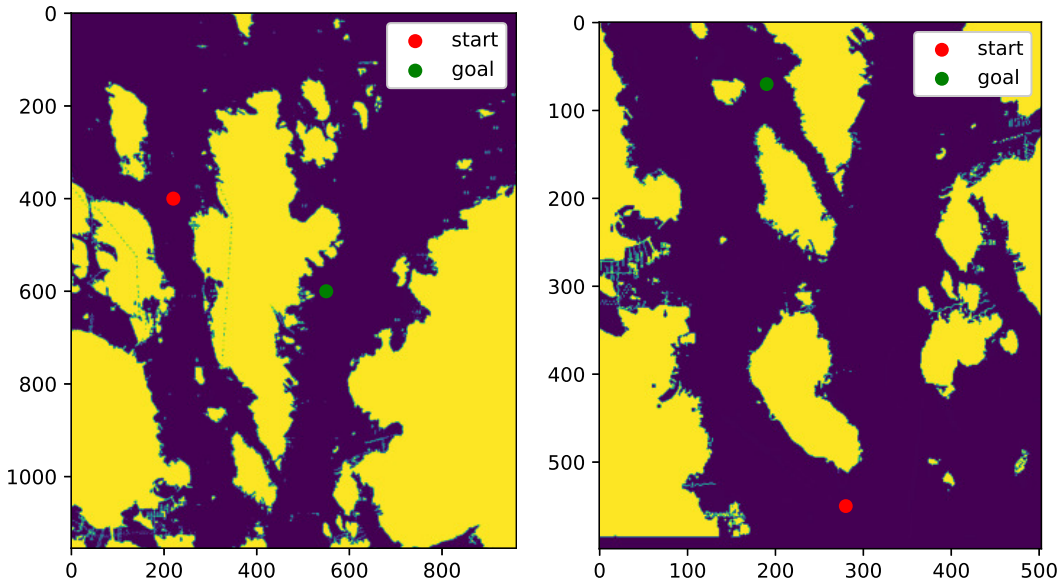
4

Results

This section contains the result for the path planning and speed profile. The first one has been tested in real conditions and evaluated to traditional voyage planning. The second comprises generated models, where some have been validated against steady-state measures.

4.1 Path planning

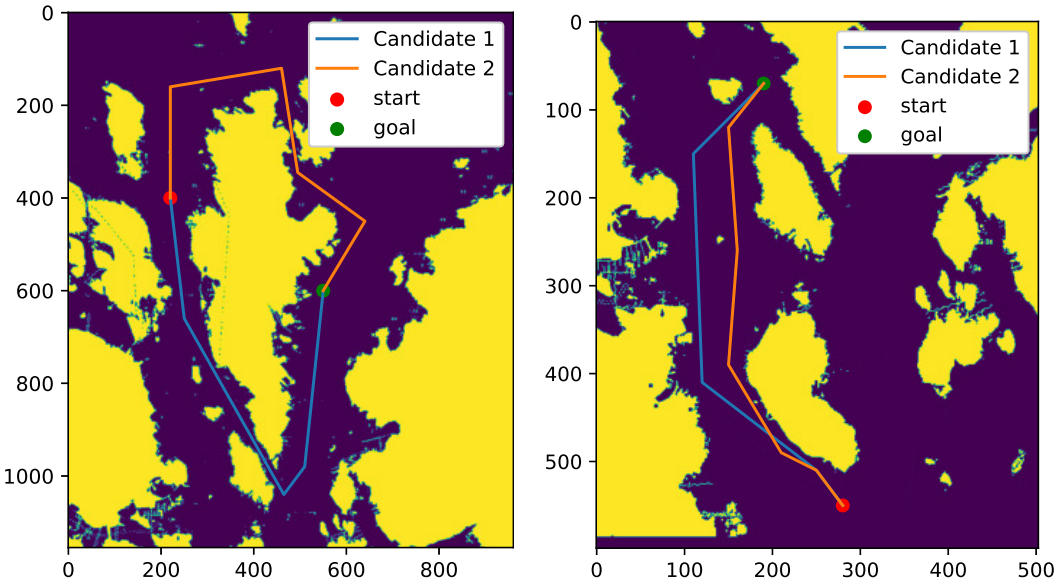
Figures 4.1a and 4.1b show the two environments in the archipelago outside of Gothenburg. The start and end positions are marked as well. The figures are presented with the previously described image processing applied.



(a) Fig. Case 1 - Björkö **(b)** Fig. Case 2 - Grötö

Figure 4.1: The initial and end position of the test routes used for evaluation. The dark blue and yellow colors correspond to the feasible respectively infeasible area.

The resulting planned paths by the two candidates are presented in Figures 4.2a and 4.2b.



(a) Fig. Björkö - with routes **(b)** Fig. Grötö - with routes

Figure 4.2: The routes, drawn manually, by the two candidates

For all three cases, the roadmap and PCA are presented. After each iteration through the algorithm, the resulting paths from the system are displayed. Lastly, the proposed path is presented with the optimal paths for the objective shortest path, least wind resistance, and minimum waves from all routes generated through the algorithm's run. Additionally was, a third and bit more complex environment tested. The candidates did not draw any routes for this case.

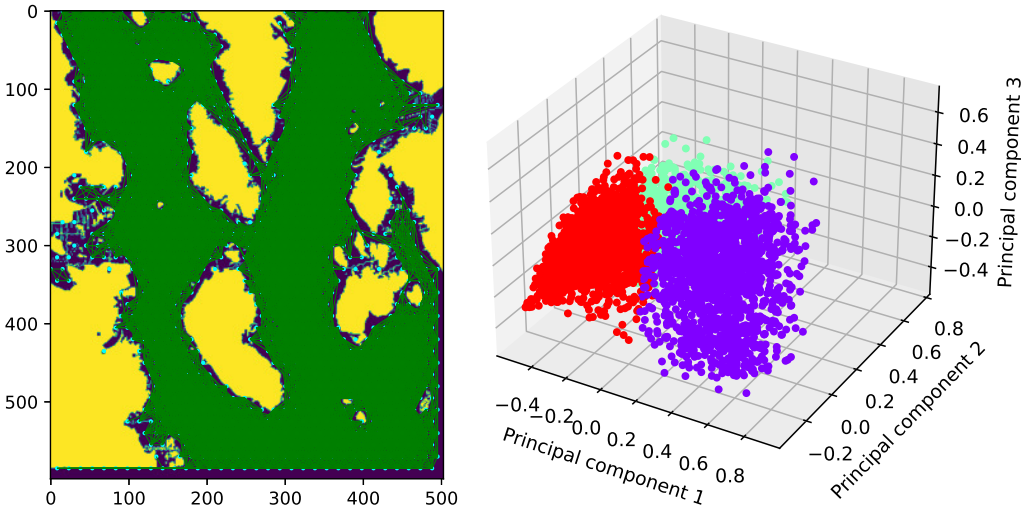
The parameters used in the GA was the same for all runs and they are shown in Table 4.1.

Table 4.1: Parameters used in the GA for the path planning system.

Parameter	Description
Generations	400
Original population size	5000
Mutation probability	$\frac{2}{L}$
Crossover probability	0.2
Reduction	0.995
Size of sampled population	100
Number of sub populations	3

4.1.1 Case 1: Grötö

The roadmap used by the algorithm is seen in Figure 4.3a and the partition of the initialized paths in Figure 4.3b.

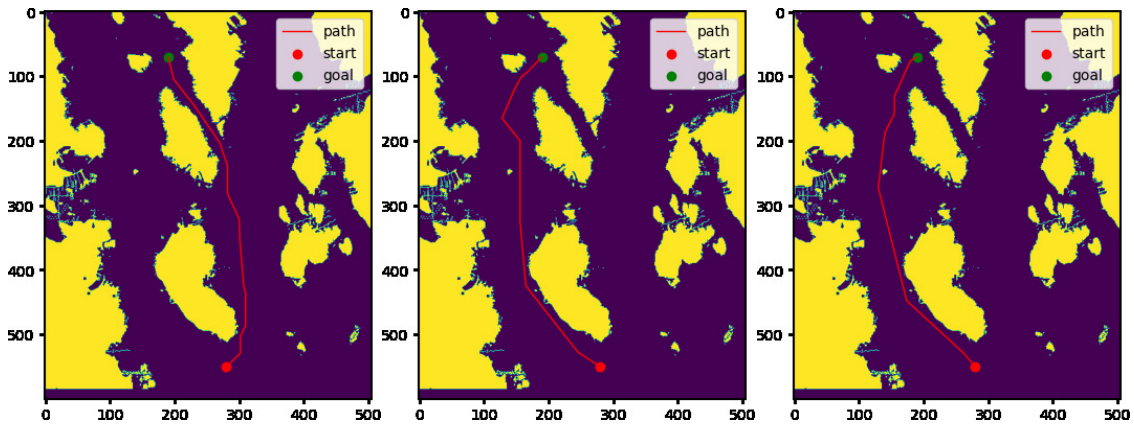


(a) Fig. Roadmap - Grötö. The ver-tices and the green lines to the edges between the vertices.

(b) Fig. K-means partition - Grötö. Every cyan dots correspond to the point corresponds to a path, and the color of the point corresponds to the assigned cluster.

Figure 4.3: Initialization of path planning - Grötö

Figure 4.4 show the resulting best path of each subpopulation after they each have passed through the GA. Table 4.2 contain the resulting fitness score of each subpopulation. These are the scores used to compare the subpopulation between each other.



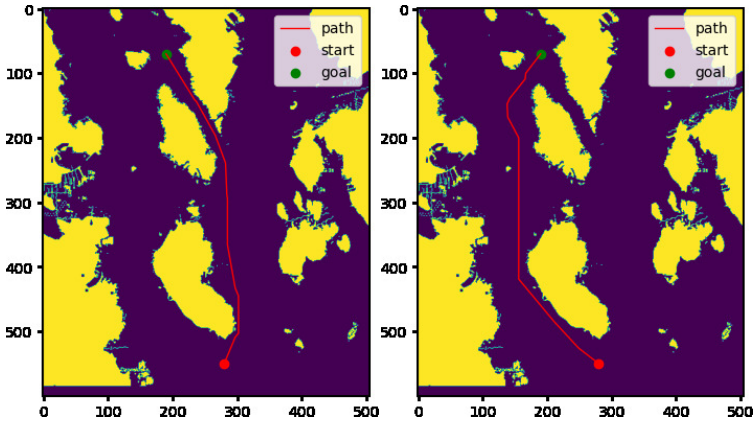
(a) Fig. Subpopulation 1 **(b)** Fig. Subpopulation 2 **(c)** Fig. Subpopulation 3

Figure 4.4: Iteration 1 - Grötö

Table 4.2: Fitness score - Iteration 1 - Grötö

	Subpopulation 1	**Subpopulation 2**	**Subpopulation 3**
Score:	1.12e-5	1.13e-5	1.06e-5

Figure 4.5 show the resulting best path of each subpopulation after they each have passed through the GA the second time. Table 4.3 contain the resulting fitness score of each subpopulation. These are still used to compare the subpopulation between each other.



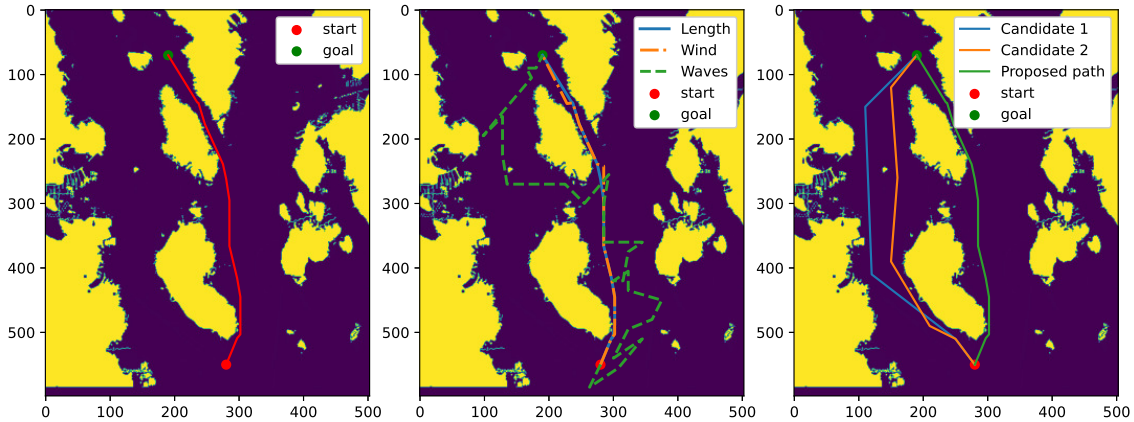
(a) Fig. Subpopulation 1 **(b)** Fig. Subpopulation 2

Figure 4.5: Iteration 2 - Grötö

Table 4.3: Fitness score - Iteration 2 - Grötö

	Subpopulation 1	**Subpopulation 2**
Score:	1.17e-5	1.08e.5

Figures 4.6a and 4.6b are the proposed path by the algorithm and the optimal paths regarding the different objectives. Further is Figure 4.6c showing an overview of the three paths beside each other.



(a) Fig. Proposed path - Grötö **(b)** Fig. Pareto optimal - Grötö **(c)** Fig. Comparison - Grötö

Figure 4.6: Final results - Grötö

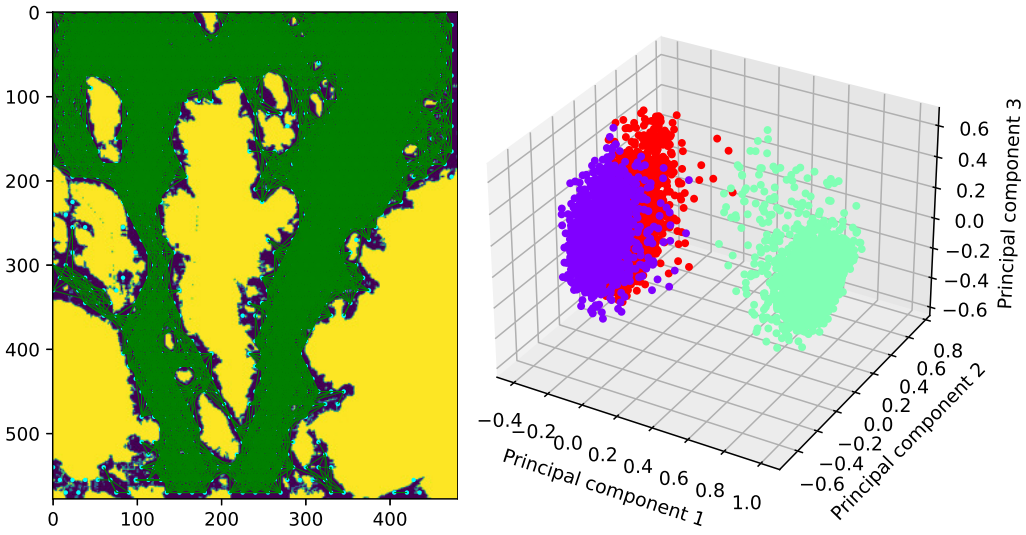
Firstly, Table 4.4 contains the evaluated fitness score of the proposed path and the corresponding scores of the paths drawn by the candidates given the same weather forecast. Secondly, it also presents the resulting fuel consumption when running real-world testing of the paths and their relative consumption.

Table 4.4: Fitness score - Proposed path - Grötö

	Proposed path	**Candidate 1**	**Candidate 2**
Score:	2.02e.5	1.73e-5	1.84e-5
Real consumption[Liter]:	9.36	13.06	11.99
Relative consumption:	1.00	1.39	1.28

4.1.2 Case 2: Björkö

The roadmap used by the algorithm is seen in Figure 4.7a and the partition of the initialized paths in Figure 4.7b.

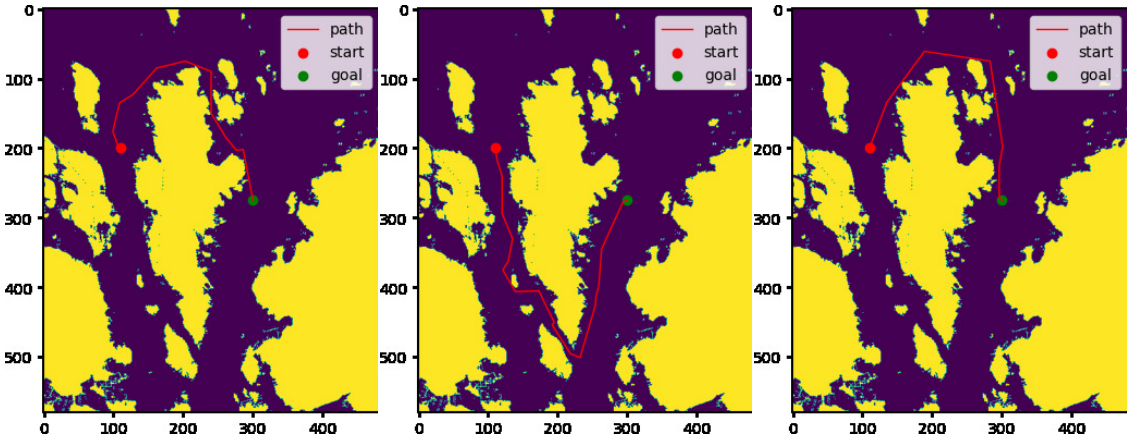


(a) Fig. Roadmap - Björkö. The cyan dots correspond to the vertices and the green lines to the edges between the vertices.

(b) Fig. K-means partition - Björko. Every point corresponds to a path, and the color of the point corresponds to the assigned cluster.

Figure 4.7: Initialization of path planning - Björko

Figure 4.8 show the resulting best path of each subpopulation after they each have passed through the GA. Table 4.5 contain the resulting fitness score of each subpopulation. These are the scores used to compare the subpopulation between each other.



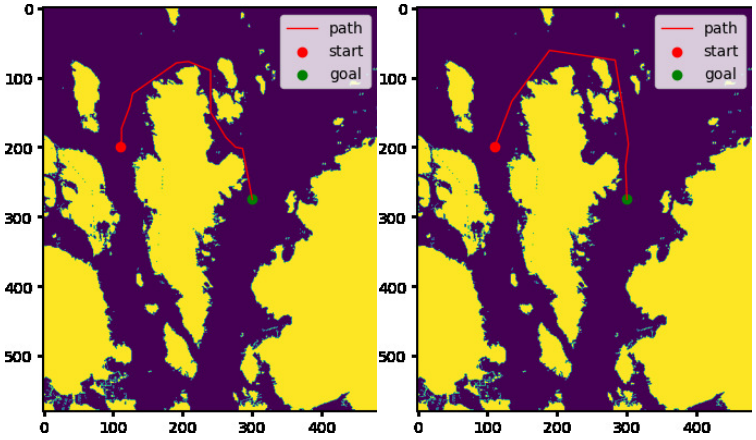
(a) Fig. Subpopulation 1 (b) Fig. Subpopulation 2 (c) Fig. Subpopulation 3

Figure 4.8: Iteration 1 - Björkö

Table 4.5: Fitness score - Iteration 1 - Björkö

	Subpopulation 1	**Subpopulation 2**	**Subpopulation 3**
Score:	1.26 e-5	8.56 e-6	1.47e-5

Figure 4.9 shows the resulting best path of each subpopulation after they each have passed through the GA the second time. Table 4.6 contain the resulting fitness score of each subpopulation. These are still used to compare the subpopulation between each other.



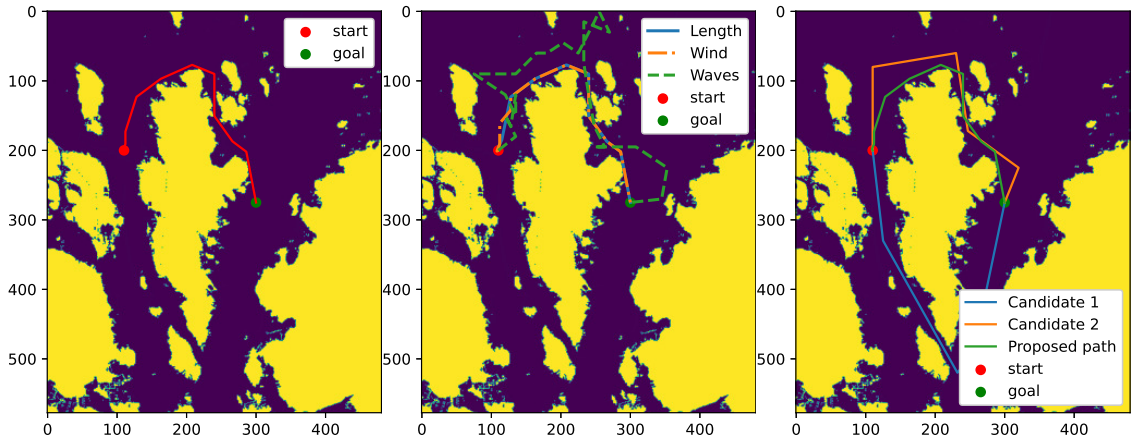
(a) Fig. Subpopulation 1 (b) Fig. Subpopulation 3

Figure 4.9: Iteration 2 - Björkö

Table 4.6: Fitness score - Iteration 2 - Björkö

	Subpopulation 1	**Subpopulation 3**
Score:	1.31e-5	1.26e-5

Figures 4.10a and 4.10b are the proposed path by the algorithm and the optimal paths regarding the different objectives. Further is Figure 4.10c showing an overview of the three paths beside each other.



(a) Fig. Proposed path - Björkö

(b) Fig. Pareto optimal - Björkö

(c) Fig. Comparison - Björkö

Figure 4.10: Final results - Björkö

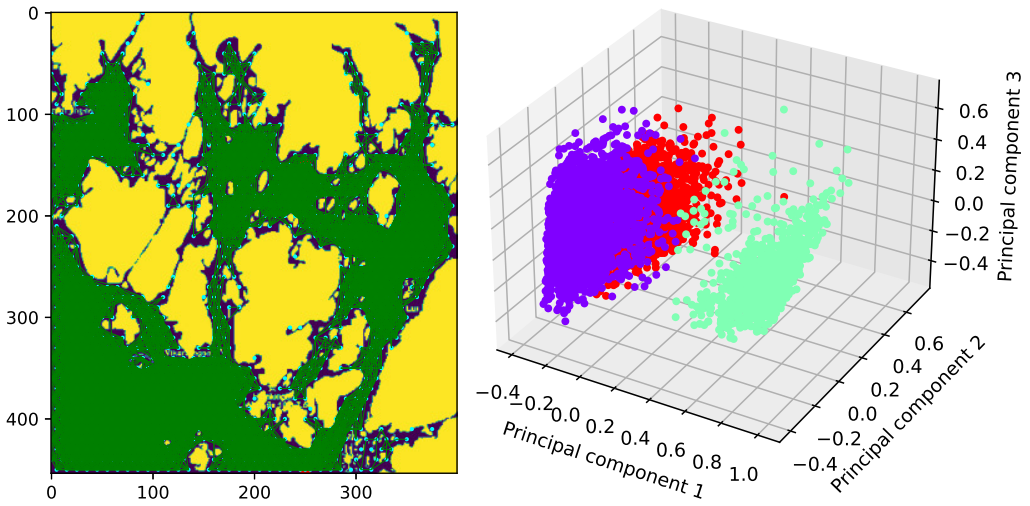
Firstly, Table 4.7 contains the evaluated fitness score of the proposed path and the corresponding scores of the paths drawn by the candidates given the same weather forecast. Secondly, it also presents the resulting fuel consumption when running real-world testing of the paths and their relative consumption.

Table 4.7: Fitness score - Proposed path - Björkö

	Proposed path	**Candidate 1**	**Candidate 2**
Score:	2.50e-5	1.79e-5	1.93-5
Real consumption[Liter]:	19.90	27.20	23.65
Relative consumption:	1.00	1.37	1.19

4.1.3 Complex test environment

The roadmap used by the algorithm is seen in Figure 4.11a and the partition of the initialized paths in Figure 4.11b.

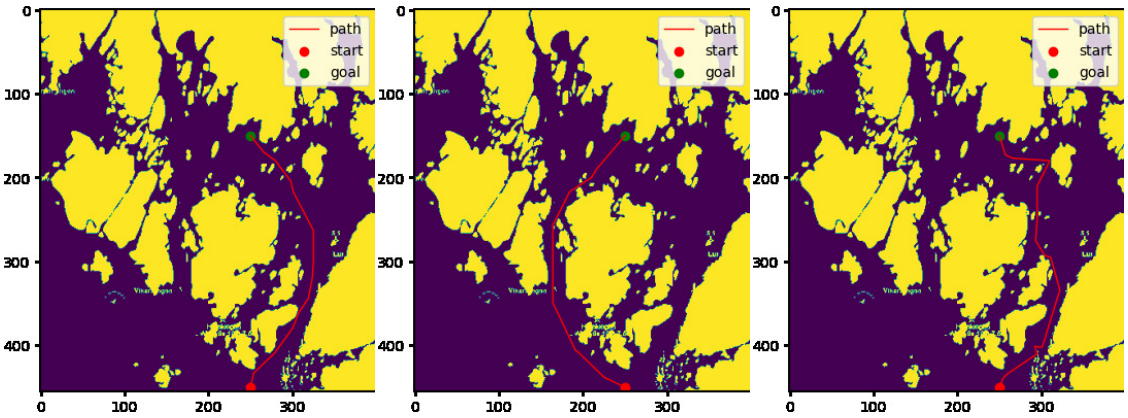


(a) Fig. Roadmap - Complex. The cyan dots correspond to the vertices and the green lines to the edges between the vertices.

(b) Fig. K-means partition - Complex. Every point corresponds to a path, and the color of the point corresponds to the assigned cluster.

Figure 4.11: Initialization of path planning - Complex

Figure 4.12 show the resulting best path of each subpopulation after they each have passed through the GA. Table 4.8 contain the resulting fitness score of each subpopulation. These are the scores used to compare the subpopulation between each other.



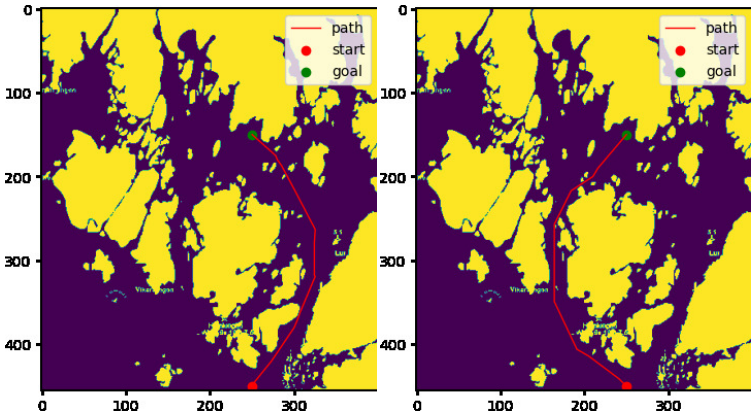
(a) Fig. Subpopulation 1 **(b)** Fig. Subpopulation 2 **(c)** Fig. Subpopulation 3

Figure 4.12: Iteration 1 - Complex

Table 4.8: Fitness score - Iteration 2 - Complex

	Subpopopulation 1	**Subpopopulation 2**	**Subpopopulation 3**
Score:	1.57e-5	1.52e-5	1.37e-5

Figure 4.13 show the resulting best path of each subpopulation after they each have passed through the GA the second time. Table 4.9 contain the resulting fitness score of each subpopulation. These are still used as the basis to compare the subpopulation between each other.



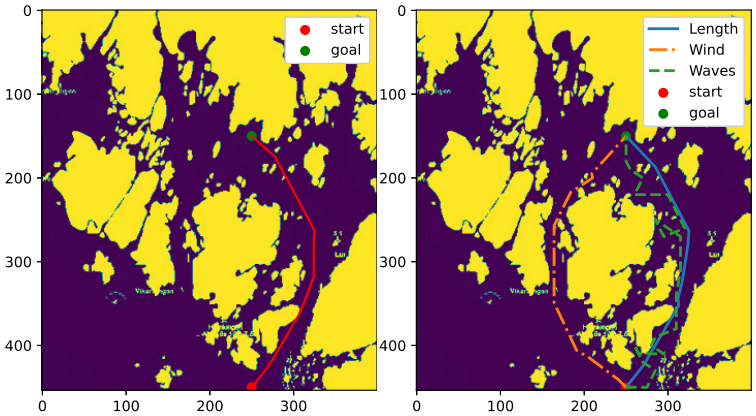
(a) Fig. Subpopulation 1 **(b)** Fig. Subpopulation 2

Figure 4.13: Iteration 2 - Complex

Table 4.9: Fitness score - Iteration 2 - Complex

	Subpopopulation 1	**Subpopopulation 2**
Score:	1.59e-5	1.53e-5

Figures 4.14a and 4.14b are the proposed path by the algorithm and the optimal paths regarding the different objectives. Figures 4.14a and 4.14b are the proposed path by the algorithm and the optimal paths regarding the different objectives. No routes were drawn by the candidates to compare the results.



(a) Fig. Proposed path **(b)** Fig. Pareto optimal - Grötö

Figure 4.14: Final results - Complex

Table 4.10 contains the evaluated fitness score of the proposed path.

Table 4.10: Fitness score - Proposed path - Complex

	Proposed path
Score:	2.94 e-5

4.2 Speed profile

The result consists of models for five unique vessels. The first two vessels show an adequate use of the system. Following this are two models built for the RPM versus fuel rate in order to convey further validity. Lastly, a model consisting of the three dimensions of speed, fuel rate, and interceptor angle is presented.

4.2.1 Vessel 1

The performance indicators and figures of the model are presented with two different distinct factors. This is due, as previously mentioned in Section 3.3.2, that the model is solely built by the usage of data from the starboard engine. In contrast, the sea trials are recorded for both engines collectively. Hence, the purpose of evaluating the model's performance needs to be adjusted with a factor to ensure a fair comparison. The model is compensated with a factor of 2.2 for every value. The second with a factor of two. The cause for these measures and their difference will be considered in the discussion.

The hyperparameters yield by the Bayesian optimization are given in Table 4.11.

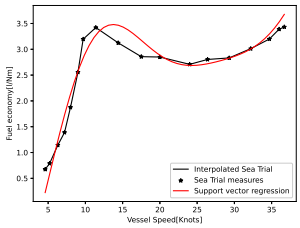
Table 4.11: Selected hyperparameters by Bayesian optimization for vessel 1

Gamma	1
C	8.438088062579128

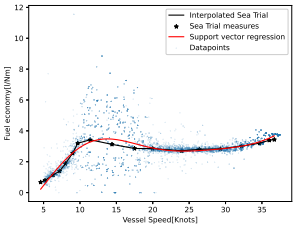
Furthermore, the performance indicators of model with a factor of 2.2 can be found in Table 4.12. In addition to this, can a graph of the SVR model, sea trial and data points be observed in Figure 4.15.

Table 4.12: Performance indicator of the SVR model in comparison with the sea trial for vessel 1 with a factor of 2.2

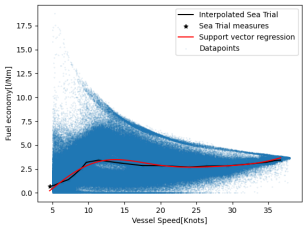
	MAE	**RMSE**	**MAD**
Mean	0.170	0.225	0.170
Standard deviation	0.006	0.011	0.010



(a) SVR model and with the interpolated sea trial.



(b) sea trial, SVR model and the 8000 data points utilized to build the model.



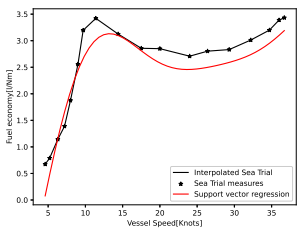
(c) All data points togheter with the sea trial and SVR model.

Figure 4.15: Overview of sea trials, SVR model and data points for vessel 1 with a factor of 2.2

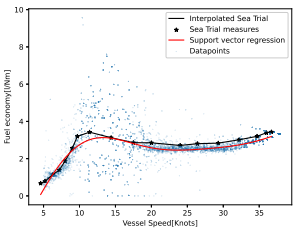
Here, the results are shown from when the model was compensated with a factor of two instead. Table 4.13 provides the performance indicators. Figure 4.16 shows an overview of the visual results obtained.

Table 4.13: Performance indicator of the SVR model in comparison with the sea trial for vessel 1 with a factor of two

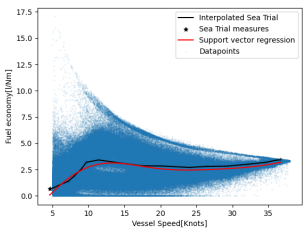
	MAE	**RMSE**	**MAD**
Mean	0.249	0.306	0.263
Standard deviation	0.007	0.008	0.009



(a) SVR model and with the interpolated sea trial



(b) sea trial, SVR model and the 8000 data points utilized to build the model.



(c) All data points togheter with the sea trial and SVR model.

Figure 4.16: Overview of sea trials, SVR model and data points for vessel 1 with a factor of 2

An example for the intended use case are given in Figure 4.17. The speed, optimized by the PSO, was subject to time constrains. In order to arrive on time the vessel need to have a speed of between 22 and 30.5 kn. On this basis, the PSO found the most fuel efficent speed to be 23.766 kn with an estimated fuel consumption of 407.794 Liter.

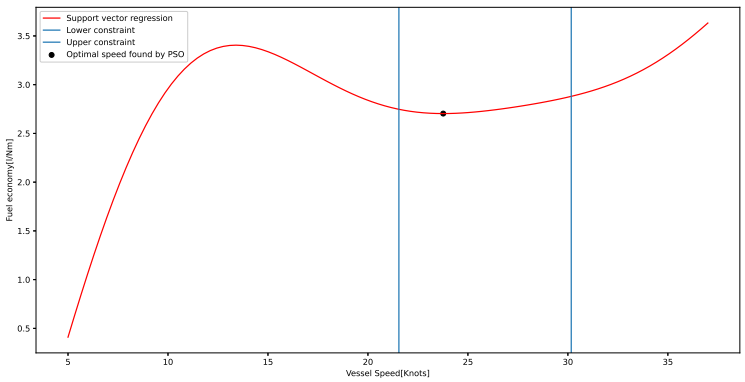


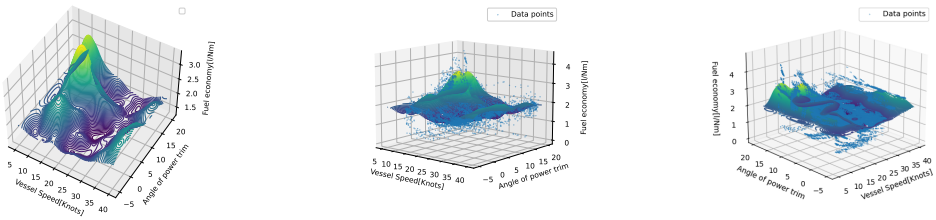
Figure 4.17: The SVR model, subject to time constraints, and the optimal speed for vessel 1 is given as the black dot.

4.2.2 Vessel 2

Vessel 2 was investigated for three dimensions. This was speed, fuel rate and power trim. In difference to the previous result, this model could not be validated. The hyperparameters are presented in Table 4.14 and a overview of the three dimensional model can be viewed in Figure 4.18

Table 4.14: Selected hyperparameters by Bayesian optimization for vessel 2

Gamma	0.1
C	0.1



(a) Fig. The SVR model.

(b) Fig. The SVR model with the data points used to build the model.

(c) Fig. The SVR model with all data points collected.

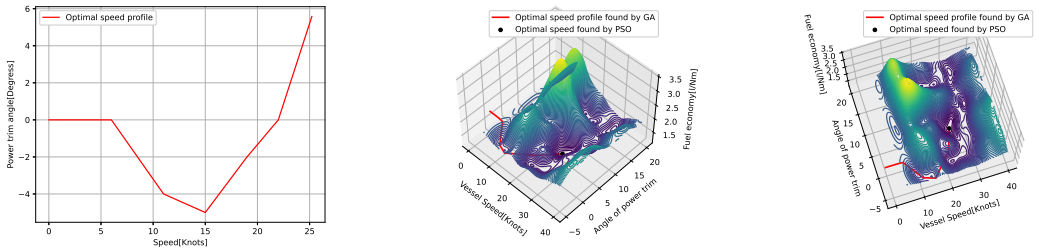
Figure 4.18: Overview of sea trials, SVR model and data points for Sessan Key Lago 30

The system was applied with bounds set between ±6 degrees for the power trim and 15 to 40 kn for the velocity. The optimal speed for these bounds was set to 25.216 kn by the PSO. The nodes given by applying a GA, from the initial position

(0,0) to the optimum, are shown in Table 4.15. A graphical view of the path in two dimensions speed versus power trim angle is given in Figure 4.19.

Table 4.15: Proposed nodes for the MEP by GA

Speed[Kn]	Power trim[Deg]
0	0
2	0
4	0
6	0
11	-4
15	-5
19	-2
22	0
25.216	5.581



(a) Fig. The proposed MEP by the GA.

(b) Fig. The proposed MEP together with the SVR model.

(c) Fig. The proposed MEP together with the SVR model from another view.

Figure 4.19: The SVR model and the proposed MEP for vessel 2.

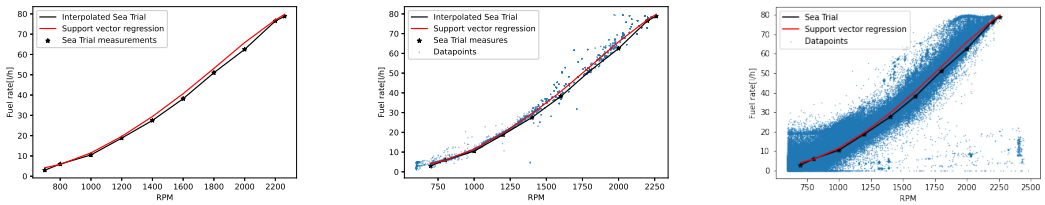
4.2.3 Vessel 3

Next, vessel 3 was tested to build a model for fuel rate versus RPM. As previously mentioned, this was not the intended usage of the system. However, it aims at a way to show the result of building a model for the vessel. The selected hyperparameters are given in Table 4.16.

Table 4.16: Selected hyperparameters by Bayesian optimization for vessel 3

Gamma	1e-6
C	9.244234053405913

Moreover, are the visual result shown in Figure 4.20. All the performance indicator for the model can be viewed in Table 4.17



(a) [SVR model and with the interpolated sea trial.

(b) sea trial, SVR model and the 8000 data points utilized to build the model.

(c) All data points togheter with the sea trial and SVR model.

Figure 4.20: Overview of sea trials, SVR model and data points for vessel 3

Table 4.17: Performance indicator of the SVR model in comparison with the sea trial for vessel 3

	MAE	**RMSE**	**MAD**
Mean	2.616	2.793	2.053
Standard deviation	0.502	0.945	0.498

4.2.4 Vessel 4

The model of vessel 4 was also made for RPM versus fuel rate. In Table 4.18 the optimal hyperparameters from the Bayesian optimization are presented.

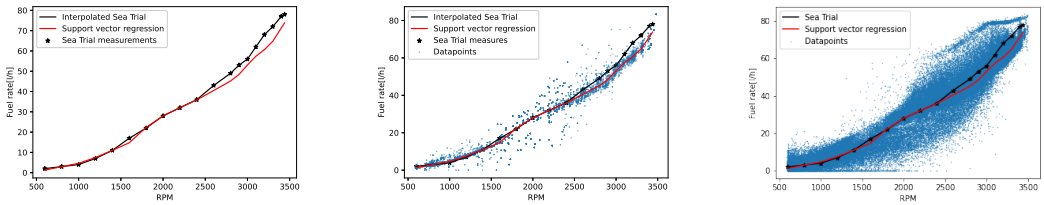
Table 4.18: Selected hyperparameters by Bayesian optimization for vessel 4

Gamma	1e-4
C	2.922508826781333

Moreover, the performance indicator are shown in Table 4.19 and a visual overview of the outcome in Figure 4.21.

Table 4.19: Performance indicator of the SVR model in comparison with the sea trial for vessel 4

	MAE	**RMSE**	**MAD**
Mean	2.871	3.829	2.870
Standard deviation	0.073	0.120	0.121



(a) SVR model and with the interpolated sea trial.

(b) sea trial, SVR model and the 8000 data points utilized to build the model.

(c) All data points together with the sea trial and SVR model.

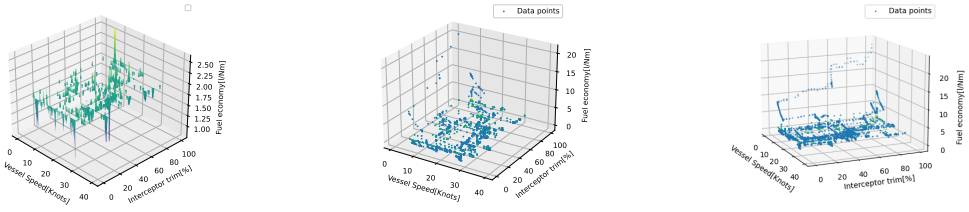
Figure 4.21: Overview of sea trials, SVR model and data points for vessel 4

4.2.5 Vessel 5

The interceptors effect on fuel consumption was investigated for the last vessel. The model's parameters are given in Table 4.20. The resulting model can be seen in Figure 4.22.

Table 4.20: Selected hyperparameters by Bayesian optimization for vessel 5

Gamma	100
C	0.1



(a) Fig. The SVR model.

(b) Fig. The SVR model with the data points used to build the model.

(c) Fig. The SVR model with all data points collected.

Figure 4.22: Overview of sea trials, SVR model and data points for vessel 5

5
Discussion

This chapter analyze and discuss the results produced by this thesis. The discussion deals with the quality of the results and their coupling to real-world scenarios for the path planning and speed profile. Further, will methods applied throughout the thesis be dealt with, especially the up and downsides of these, regarding potential improvements that could have been made. Lastly, there is a discussion regarding the possibility of future work. Areas in which this thesis did not have the resources ot time to investigate. However, they could still be of great interest to explore, given the results of this thesis.

5.1 Path planning

By beginning to study the drawn routes by the two candidates in Figures 4.2a and 4.2b, it is evident that there are differences between the two. In the case of Grötö, the candidates drew similar routes regarding the general area of travel but still not identical. Regarding *Björkö* they opted to take two completely different ways of travel towards the goal position. This outcome is of great interest. One functionality of the planning system is to analyze different ways around an island, like these, and optimize those that seemed most beneficial, given the on-route conditions.

Moreover, the system succeeds in solving the path planning around Grötö. The roadmap in Figure 4.3a shows that a network was able to form that included several ways of travel from start to goal. The k-means partition in Figure 4.3b has successfully divided the initialized population into three separate subpopulations. The partition does not contain any clear hyperplanes that divide the different types of paths. Nevertheless, the general behavior seems to have been identified. Figure 4.4 display the results after one pass through the GA. It shows that the algorithm has successfully found different types of travel between the two points. More interesting is that the paths drawn by the candidates are very much alike those of subpopulations 2 and 3. The different scores for the subpopulations indicate that the algorithm prefers subpopulations 1 and 2. This conclusion will result in the exclusion of subpopulation 3 from the MLSGA.

Following another pass through the GA; see Figure 4.5, the general appearance is not that much different from before. Except for some more minor alterations. At this point, two clear options can be considered. The fitness score of the remaining two populations shows that the algorithm prefers subpopulation 1. This alternative

is of great interest as it differs significantly from the ones drawn by the candidates. But as previously discussed, the algorithm has evaluated paths in the same area as the candidates and taken an active choice to not proceed with them. It is further shown that the path with the least wind resistance and distance seems to belong to subpopulation 1. Given this information, the final proposed path is expected. The optimal path concerning waves appears to be very unrealistic to how one would behave at sea and need further work to give any relevant information.

Moving over to the case of Björko. As previously mentioned, the candidates drew two completely different types of paths for this case; see Figure 4.2b. The roadmap manages to find ways through most feasible passages in the environment just as before. More interestingly, the k-means partition in Figure 4.7b, shows that there is a more clear distinction between different groups of data points. This partition in the data, the initialized paths, indicates that apparent differences have been found. The difference is likely a result of paths following the trend of either candidate 1 or 2, i.e., paths passing north or south of Björkö in the two-dimensional map. The hypothesis is strengthened by analyzing the results in Figure 4.8. In this figure, the different subpopulations have done just so, i.e., followed different trends of the candidates.

The scores in Table 4.5 indicate that the trend of candidate 2 is strongly preferred. Following the second iteration, subpopulation 1 has further optimized its best path while the score of subpopulation 3 has decreased, resulting in subpopulation 1 now having the higher score and being chosen. The phenomenon of a decreased score is not uncommon. This is due to elitism only being applied regarding the score of the best solution and not the population as a whole. The final proposed path can be reassembled as an optimized version of that drawn by candidate 2, and thus have a better resulting fuel consumption.

Continuing and analyzing the additional test environment. The reason for using a different test environment was to evaluate the algorithm's ability to efficiently find paths in a more challenging environment concerning the FC-Space. The new environment was set on the east coast of Norway and seemed fitting due to the combination of a few large islands and several smaller islets. The results of the testing are seen in Section 4.1.3. The results show that the algorithm can find multiple ways of travel even in these more complex environments. However, as a cause of the more complex FC-Space, the proposed paths are seen to be more jagged than previous results in the early phase of the run. The MLSGA therefore needed to use a higher amount of iterations inside of the GA to create more reasonable and smoother paths. Nonetheless, the algorithm still manages to solve the problem and propose a feasible solution that should be the most efficient concerning fuel consumption. The result is expected, given the outcome of the previous test cases.

Lastly, looking back at the first research question stated in the introduction, i.e., "What is the performance of a path planning system that optimizes concerning given objectives, compared to traditional voyage planning?". When following the paths

drawn by the two candidates, the resulting fuel consumption was, on average, 38% and 24% higher than those suggested by the path planning system. The proposed paths have further been able to outscore the candidates in both the case of Grötö and Björkö when comparing their fitness scores. This indicated that the optimization model, even though a simplification, manages to represent the optimization problem. It is reasonable to question the validity of these numbers. Due to the sample size being so small, no statistical significance can be concluded. However, these numbers did greatly exceeds the expectations and are seen as a success.

5.2 Speed profile

The results of the derived model in Figure 4.18 show that there are clear local minimums and maximums for different regions of the model. Hence, the model shows clear differences regarding the configuration of trim and speed and the resulting fuel economy. Moreover, one can, by visual examination of Figure 4.19b conclude that the PSO succeeds in finding the minimum of regions within the bounds. However, if the model manages to resemble the real-world condition is still unsaid. Thus, the model's performance has not been evaluated. This is due to the lack of data that could be utilized for verification, as stated in Section 3.3.2.

Furthermore, other models in the result were possible to validate. One example of this is in Section 4.2.1. The model utilized fuel economy versus speed. Hence, the dimensions correspond to the suggested purpose of the system. In order to build the model, only data from the starboard engine was utilized. Compensation had to be made because the fuel consumption recorded during the sea trial was a total for both engines. This was introduced by adding a factor. This will coincide with a factor of two, if one assumes that the motors are equal. That seems reasonable due to the motors being the same model. However, adding a factor of 2.2 the performance indicators are heavily decreased. The significant increase in performance can be due to the engine's load. This is supported by the fact that the sea trial was fully fueled and had ten passengers. This raises the question of what the load was for the data points. If the mean of the routes are substantially lower, it might explain the reason for the extra 0.2 in the factor.

Moreover, the intended use case of the models, in this thesis, does not depend on predicting the resulting fuel consumption perfectly. Instead, the usefulness still exists as long as the characteristics are replicated. Thus, a factor will not cause harm to the usability of the model. In the end, the system only serves as a way of finding the most energy-efficient configuration.

In the case of building a model for RPM versus fuel rate, the result, seen in Table 4.19, and 4.17, are similar for all test cases. Undoubtedly, the model succeeds in replicating the underlying measures. Nevertheless, the model is still deviating from the measures to various extent. This raises the question of to what extent the model can be allowed to deviate. Currently, no models are used for this pur-

pose. Therefore, it has not been evaluated, and no standards exist. The sea trial measures are the closest, but the intention is only to record the vessel's performance.

A statistical model is, at its core, just a approximation. Thus, in its essence only succeed in modeling the outcome for the variables included in the model. Other variables, uncorrelated with the included variables, can affect the outcome. For the models mentioned above, both the torque and load of the motor will affect the resulting fuel consumption. Since these were not included in the model, it is clear that measurements will have some variance. The sea trial will simultaneously be subjected to at least a particular load by forces acting on the vessel, e.g., the weight and sea conditions. This can be a cause to why the resulting SVR deviates from the sea trial. Besides this, the motor's torque affects the used data, which also causes variance. The torque variable was not available for usage and could hence not be included in the SVR model.

A possible explanation for the deviations from the sea trial may be due to several sources of error. The tester manually writes down the sea trial measurement. Thus, it is possible that a measurement can be wrong due to the human factor. In addition to this is the measurement read from the displays. This can also cause inaccuracy if the displays are not displaying all decimals. Indeed, the sensors will also play a key aspect here. The variance and accuracy of the sensors are fundamental for building a statistical model. If the data recorded varies due to inaccurate or faulty sensors, it will affect the model's performance.

Moreover, the MEP for vessel 2, i.e., the suggested speed and trim path, could not be validated due to time limitations. However, the GAs proposed path, seen in Figure 4.19a, can be concluded to be realistic. Nevertheless, since a stochastic algorithm is used, the algorithm can never guarantee the solution corresponding to the global optimum. Further, the thesis also investigated the dimensions interceptor and speed. Surprisingly, this combination did not yield the expected result. A possible explanation for this might be due to faulty sensors. Another possible reason could be that the interceptor can only optimize the fuel economy during speeds close to the planning threshold.

It is interesting to note that all hyperparameters, except for the interceptor model, are in the range of one to ten for C and $1 * 10^{-4}$ to 1 for gamma. The optimal value of gamma being lower than one is consistent with previous findings [62]. They utilized numerical and statistician considerations to find the optimal gamma and C for predicting energy consumption. The best solutions were gamma=0.02 and C=0.71 with ε=0.05. However, the utilized ε in this project was set to 0.001. This can explain the difference compared to the previous study.

It is not surprising that the models yield low deviation from the sea trial measures and hence succeed to predict the fuel consumption. This finding is consistent with what previous studies, although not the exact same application, have found using black-box modelling [20]. This thesis has demonstrated how the optimization

of engine configuration can be made, especially how to build a model with no more than 8000 data points and find the optimal configuration with respect to constraints.

5.3 Future work

The different scenarios previously tested are of reasonable short voyages. The weather conditions with such a short time horizon can often be assumed accurate. However, if the path planning is to be applied for longer voyages at sea, the conditions could change over time. It would be interesting to enable the possibility of updating the proposed path over time concerning new information regarding the weather forecast or additional stopping points to refuel or similarly. Further research could also be conducted to analyze the performance of other clustering algorithms, such as density-based spatial clustering of applications with noise or spectral clustering, so that the number of clusters does not have to be specified.

Looking at the usage area for the path planning system, it is limited to motor vessel and the usual way of travel for these kinds of vessels. If one instead is traveling by sailboat, the vessel's dynamics and propulsion is vastly different from that of motor vessels. An extension of the system to optimize path planning regarding sailboats should both be applicable and feasible. The same methodology should be suitable, but the objective functions and the corresponding evaluation should be changed.

Furthermore, having access to raw ENC files should enable more precise environmental information. Previous research has shown that such an application is viable[10]. The access to raw ENC files should erase the need for any image processing as there would not be any insecurities about the area of interest. Additionally, it could enable information on potential slow-speed areas or similar real-world circumstances that can be of great interest when planning voyages. Moreover, this should allow following the COLREG to a greater extent.

There also exists a possibility of creating a more advanced model for the forces acting on the vessel. The model used in this thesis was known to be imperfect. However, there are many challenges to creating a good representation of a vessels dynamics. Research shows that even some of the most commonly used mathematical models have their flaws [61]. However, a more reliable result can be expected from a guidance system by making a more in-depth vessel model.

Regarding the speed profile, extensions can be made by investigating several other dimensions. As previously mentioned, the vessel's load will affect the fuel consumption. However, this is currently not included in the model. The reason for this is that the parameter was not measured. In addition to this, if the dimension acceleration were included, which is not the case at present for the models of vessels 2-5, it might improve the performance. The MEP has not been developed to its fullest potential in this thesis. Currently, the GA applied for finding the path is basic, and the algorithm may fail to find the MEP. However, the combination of speed and trim can be formulated as a graph. Hence, applying a graph-searching algorithm, such

as A*, would always yield the MEP. It would also be interesting to extend the MEP system to take acceleration into account. Hence, the optimal power trim, speed, and acceleration would be given either to an autopilot or driver assistant to the vessel user. If the planned path contained information regarding eventual low-speed areas, this could also be employed to develop a more advanced speed profile.

Lastly, further validations are needed to implement the system in real-world conditions. The path planning system has been validated by two persons drawing routes. A more extensive validation could consist of comparing the system to historical routes logged from the AIS of the vessels, with a path planning system using historical weather data. Moreover, the speed profile system has only been validated using three different vessels, where only one did validate the intended use case. Here a more extended validation could be comprised of an adapted model under operation, where sea trial has been collected in near time.

6
Conclusion

This thesis aimed to produce a marine guidance system. The work was carried out by two distinct systems that collaborate. The first gives the optimal configuration of the vessel, given constraints, by the usage of Bayesian optimization and SVR. The optimal speed, given in the configuration, is utilized by the path planning system to find routes optimized against weather conditions. This system functions by the usage of images processing algorithms, a proposed C-Space algorithm, PCA, k-means, and MLSGA. The proposed C-Space algorithm for complex environments is inspired by the PRM algorithm.

The overall results of the path planning, given in Table 4.4 and 4.7, showed that the path planning system was able to generate more fuel efficient routes. However, as the sample size was too small, no statistical significance could be concluded. Nevertheless, the real-life test shows that it can assist drivers at sea in their voyage planning to help reduce fuel consumption. Additionally, it has proven to coupe with different environments with varying complexity.

This study has further identified a potential way for constructing the dynamics of individual vessels. The data-driven approach was compared to previous manually recorded measurements from Volvo Penta. It can be concluded that the application showed promising performance for future implementation. Models optimizing for more than two dimensions could not be validated due to a lack of data and resources. Hence, the potential is still not validated but showed great promise.

Bibliography

[1] I. M. Organization, "Guidelines for voyage planning„" 1999. [Online]. Available: https://wwwcdn.imo.org/localresources/en/KnowledgeCentre/IndexofIMOResolutions/AssemblyDocuments/A.893(21).pdf

[2] Kunskapsfordringar avseende båtpraktik (dag) för fritidsbåt gällande från 2020-05-19. [Online]. Available: https://batlivsutbildning.se/wp-content/uploads/2020/09/Kunskapsfordringar-gallande-Batpraktik-Dag-redigerad-2020-09-14.pdf

[3] M. Hinostroza and C. Guedes Soares, "Global and local path-planning algorithm for marine autonomous surface ships including forecasting information," in *Global and local path-planning algorithm for marine autonomous surface ships including forecasting information*, 11 2020.

[4] P. Chen, Y. Huang, E. Papadimitriou, J. Mou, and P. van Gelder, "Global path planning for autonomous ship: A hybrid approach of fast marching square and velocity obstacles methods," *Ocean Engineering*, vol. 214, p. 107793, 2020. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0029801820307691

[5] R. JAMES, "Application of wave forecasts to marine navigation," 1957. [Online]. Available: https://www.proquest.com/openview/494678c1e524a1b8e12fe9f2cda75fd1/1?pq-origsite=gscholar&cbl=18750&diss=y

[6] K. Karur, N. Sharma, C. Dharmatti, and J. E. Siegel, "A survey of path planning algorithms for mobile robots," *Vehicles*, vol. 3, no. 3, pp. 448–468, 2021. [Online]. Available: https://www.mdpi.com/2624-8921/3/3/27

[7] M. Nazarahari, E. Khanmirza, and S. Doostie, "Multi-objective multi-robot path planning in continuous environment using an enhanced genetic algorithm," *Expert Systems with Applications*, vol. 115, pp. 106–120, 2019. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0957417418305165

[8] P.-W. Tsai, T.-T. Nguyen, and T.-K. Dao, "Robot path planning optimization based on multiobjective grey wolf optimizer," in *Genetic and Evolutionary Computing*, J.-S. Pan, J. C.-W. Lin, C.-H. Wang, and X. H. Jiang, Eds. Cham: Springer International Publishing, 2017, pp. 166–173.

[9] G. Wu, I. Atilla, T. Tahsin, M. Terziev, and L. Wang, "Long-voyage route planning method based on multi-scale visibility graph for autonomous ships," *Ocean Engineering*, vol. 219, p. 108242, 2021. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0029801820311641

[10] S. Reed and V. E. Schmidt, "Providing nautical chart awareness to autonomous surface vessel operations," in *OCEANS 2016 MTS/IEEE Monterey*, 2016, pp. 1–8.

[11] W. Du, Y. Li, G. Zhang, C. Wang, B. Zhu, and J. Qiao, "Ship weather routing optimization based on improved fractional order particle swarm optimization," *Ocean Engineering*, vol. 248, p. 110680, 2022. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0029801822001408

[12] D. Sen and C. P. Padhy, "An approach for development of a ship routing algorithm for application in the north indian ocean region," *Applied Ocean Research*, vol. 50, pp. 173–191, 2015. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0141118715000206

[13] H. Wang, X. Lang, and W. Mao, "Voyage optimization combining genetic algorithm and dynamic programming for fuel/emissions reduction," *Transportation Research Part D: Transport and Environment*, vol. 90, p. 102670, 2021. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1361920920308555

[14] J. Z. T. L. He Yan Kang, Di Zhang, "Ship route planning using historical trajectories derived from ais data," *TransNav*, 2019. [Online]. Available: http://dx.doi.org/10.12716/1001.13.01.06

[15] S. Khan, P. Grudniewski, Y. S. Muhammad, and A. J. Sobey, "The benefits of co-evolutionary genetic algorithms in voyage optimisation," *Ocean Engineering*, vol. 245, p. 110261, 2022. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0029801821015705

[16] P. Lundblad, O. Eurenius, and T. Heldring, "Interactive visualization of weather and ship data," in *2009 13th International Conference Information Visualisation*, 2009, pp. 379–386.

[17] E. Artusi, F. Chaillan, and A. Napoli, "Path planning for a maritime suface ship based on deep reinforcement learning and weather data," in *OCEANS 2021: San Diego – Porto*, 2021, pp. 1–8.

[18] O. B. Öztürk and E. Başar, "Multiple linear regression analysis and artificial neural networks based decision support system for energy efficiency in shipping," *Ocean Engineering*, vol. 243, p. 110209, 2022. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0029801821015249

[19] W. D. Salma Sherbaz, "Ship trim optimization: Assessment of influence of trim on resistance of moeri container ship," *The Scientific World Journal*, 2014. [Online]. Available: https://doi.org/10.1155/2014/603695

[20] A. Coraddu, L. Oneto, F. Baldi, and D. Anguita, "Vessels fuel consumption forecast and trim optimisation: A data analytics perspective," *Ocean Engineering*, vol. 130, pp. 351–370, 2017. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0029801816305571

[21] L. Moreira, R. Vettor, and C. Guedes Soares, "Neural network approach for predicting ship speed and fuel consumption," *Journal of Marine Science and Engineering*, vol. 9, no. 2, 2021. [Online]. Available: https://www.mdpi.com/2077-1312/9/2/119

[22] Y. Du, Q. Meng, S. Wang, and H. Kuang, "Two-phase optimal solutions for ship speed and trim optimization over a voyage using voyage report

data," *Transportation Research Part B: Methodological*, vol. 122, pp. 88–114, 2019. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0191261517305738

[23] C.-H. Wu, J.-M. Ho, and D. Lee, "Travel-time prediction with support vector regression," *IEEE Transactions on Intelligent Transportation Systems*, vol. 5, no. 4, pp. 276–281, 2004.

[24] P.-S. Yu, S.-T. Chen, and I.-F. Chang, "Support vector regression for real-time flood stage forecasting," *Journal of Hydrology*, vol. 328, no. 3, pp. 704–716, 2006, the ICWRER - Symposium in Dresden, Germany. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0022169406000473

[25] A. Gretton, A. Doucet, R. Herbrich, P. Rayner, and B. Scholkopf, "Support vector regression for black-box system identification," in *Proceedings of the 11th IEEE Signal Processing Workshop on Statistical Signal Processing (Cat. No.01TH8563)*, 2001, pp. 341–344.

[26] M. H. D. M. Ribeiro, R. G. da Silva, V. C. Mariani, and L. dos Santos Coelho, "Short-term forecasting covid-19 cumulative confirmed cases: Perspectives for brazil," *Chaos, Solitons Fractals*, vol. 135, p. 109853, 2020. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0960077920302538

[27] L. Yang and A. Shami, "On hyperparameter optimization of machine learning algorithms: Theory and practice," *Neurocomputing*, vol. 415, pp. 295–316, 2020. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0925231220311693

[28] M. S. Alam, N. Sultana, and S. Z. Hossain, "Bayesian optimization algorithm based support vector regression analysis for estimation of shear capacity of frp reinforced concrete members," *Applied Soft Computing*, vol. 105, p. 107281, 2021. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1568494621002040

[29] P. Wadhwani and P. Saha, "Recreational boating market," 2020. [Online]. Available: https://www.gminsights.com/industry-analysis/recreational-boating-market

[30] Recreational craft sector. [Online]. Available: https://ec.europa.eu/growth/sectors/maritime-industries/recreational-craft-sector_en

[31] Boat dynamics - powertrim assistant. [Online]. Available: https://www.volvopenta.com/marine/products/easy-boating-solutions/boat-dynamics/powertrim-assistant/

[32] T. Tveitdal, "Market barriers towards electric boats," 2017. [Online]. Available: https://www.ntnu.edu/documents/139799/1279149990/33+Article+Final_trulstv_fors%C3%B8k_2017-12-07-17-25-18_TPD4505+Truls+Tveitdal.pdf/4523bd20-1024-4179-8dde-878d847a7e29

[33] Stena line challenges the shipping industry – by going electric. [Online]. Available: https://www.stenaline.com/media/stories/stena-line-challenges-the-shipping-industry-by-going-electric/

[34] Z. Liu, Y. Zhang, X. Yu, and C. Yuan, "Unmanned surface vehicles: An overview of developments and challenges," *Annual Reviews in Control*, vol. 41, pp. 71–93, 2016. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1367578816300219

[35] E. R. Dougherty and R. de Alencar Lotufo, *Hands-on Morphological Image Processing*. SPIE Press, 2003.

[36] B. Heinold, "A simple introduction to graph theory," 2018. [Online]. Available: https://www.brianheinold.net/notes/A_Simple_Introduction_to_Graph_Theory_Heinold.pdf

[37] R. Truax, "Elementary graph theory," 2020. [Online]. Available: https://web.stanford.edu/~truax/notes/Graph_Theory.pdf

[38] L. Kavraki, P. Svestka, J.-C. Latombe, and M. Overmars, "Probabilistic roadmaps for path planning in high-dimensional configuration spaces," *IEEE Transactions on Robotics and Automation*, vol. 12, no. 4, pp. 566–580, 1996.

[39] Probabilistic roadmaps (prm). [Online]. Available: https://www.mathworks.com/help/robotics/ug/probabilistic-roadmaps-prm.html

[40] I. T. Jolliffe and J. Cadima, "Principal component analysis: a review and recent developments," *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, vol. 374, no. 2065, p. 20150202, 2016. [Online]. Available: https://royalsocietypublishing.org/doi/abs/10.1098/rsta.2015.0202

[41] D. Kalman, "A singularly valuable decomposition: The svd of a matrix," *The College Mathematics Journal*, vol. 27, no. 1, pp. 2–23, 1996. [Online]. Available: https://doi.org/10.1080/07468342.1996.11973744

[42] A. Likas, N. Vlassis, and J. J. Verbeek, "The global k-means clustering algorithm," *Pattern Recognition*, vol. 36, no. 2, pp. 451–461, 2003, biometrics. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0031320302000602

[43] M. Wahde, *Biologically Inspired Optimization Methods*. WIT press, 2008.

[44] P. A. Grudniewski and A. J. Sobey, "Behaviour of multi-level selection genetic algorithm (mlsga) using different individual-level selection mechanisms," *Swarm and Evolutionary Computation*, vol. 44, pp. 852–862, 2019. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2210650218304826

[45] A. Zingl, "A rasterizing algorithm for drawing curves," 2012. [Online]. Available: http://members.chello.at/easyfilter/Bresenham.pdf

[46] D. Kalyanmoy, *Multi-objective Optimisation Using Evolutionary Algorithms: An Introduction*. Springer London, 2011. [Online]. Available: https://doi.org/10.1007/978-0-85729-652-8_1

[47] T. I. Fossen, *Guidance and control of ocean vehicles*. John Wiley Sons Inc, 1994.

[48] "Mariems learning material - trim optimization and 3.2 hull and propeller condition," 02 2017.

[49] A. F. Molland, S. R. Turnock, and D. A. Hudson, *Ship Resistance and Propulsion: Practical Estimation of Propulsive Power*. Cambridge University Press, 2011.

[50] A. Kukner and A. M. Yasa, "High speed planning hulls resistance prediction methods and comparasion," 10 2011.

[51] H. J. and M. G.G.J., "An approximate power prediction method," 1982. [Online]. Available: https://repository.tudelft.nl/islandora/object/uuid:ee370fed-4b4f-4a70-af77-e14c3e692fd4?collection=research

[52] D. Savitsky, "Hydrodynamic Design of Planing Hulls," *Marine Technology and SNAME News*, vol. 1, no. 04, pp. 71–95, 10 1964. [Online]. Available: https://doi.org/10.5957/mt1.1964.1.4.71

[53] J. Wu, X.-Y. Chen, H. Zhang, L.-D. Xiong, H. Lei, and S.-H. Deng, "Hyperparameter optimization for machine learning models based on bayesian optimizationb," *Journal of Electronic Science and Technology*, vol. 17, no. 1, pp. 26–40, 2019. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1674862X19300047

[54] J. Joyce, *Bayes' Theorem*, E. N. Zalta, Ed. Metaphysics Research Lab, Stanford University, 2021.

[55] E. Brochu, V. M. Cora, and N. de Freitas, "A tutorial on bayesian optimization of expensive cost functions, with application to active user modeling and hierarchical reinforcement learning," *CoRR*, vol. abs/1012.2599, 2010. [Online]. Available: http://arxiv.org/abs/1012.2599

[56] V. N. Vapnik, *The Nature of Statistical Learning Theory.* Springer New York, NY, 1995.

[57] S. B. Smola, A.J., "A tutorial on support vector regression," *Statistics and Computing 14*, p. 199–222, 2014.

[58] A. Mariette and K. Rahul, *Efficient Learning Machines*, 1st ed. Apress Berkeley, CA.

[59] Sjökort, båtsportkort och e-sjökort. [Online]. Available: https://www.sjofartsverket.se/sv/tjanster/sjokortsprodukter/kopa-sjokort2/

[60] När vanliga sjökort inte räcker... [Online]. Available: https://hydrographica.se/

[61] E. LINDBERGH and F. AHLSTRAND, "Methods to predict hull resistance in the process of designing electric boats," 2020. [Online]. Available: http://www.diva-portal.org/smash/get/diva2:1465549/FULLTEXT01.pdf

[62] L. Lu, "Optimal and c for -support vector regression with rbf kernels," 2015. [Online]. Available: https://arxiv.org/abs/1506.03942

Department of Mechanics and Maritime Sciences
CHALMERS UNIVERSITY OF TECHNOLOGY
Gothenburg, Sweden
www.chalmers.se

CHALMERS
UNIVERSITY OF TECHNOLOGY