CHALMERS
UNIVERSITY OF TECHNOLOGY

Decision Support for Energy Efficiency Operations of Double Ended Ferry

Master's thesis in Naval Architecture and Ocean Engineering

Daniel Vergara

Department of Mechanics and Maritime Sciences

CHALMERS UNIVERSITY OF TECHNOLOGY
Gothenburg, Sweden 2022
www.chalmers.se

Master's thesis 2022

Decision Support for Energy Efficiency Operations of Double Ended Ferry

DANIEL VERGARA

CHALMERS
UNIVERSITY OF TECHNOLOGY

Department of Mechanics and Maritime Sciences
Division of Maritime Technology
Chalmers University of Technology
Gothenburg, Sweden 2022

Decision Support for Energy Efficiency Operations of Double Ended Ferry

Daniel Vergara

Supervisor: Martin Alexandersson, SSPA
Examiner: Wengang Mao, Division of Maritime Technology

Master's Thesis 2022
Department of Mechanics and Maritime Sciences
Division of Maritime Technology
Chalmers University of Technology
SE-412 96 Gothenburg
Telephone +46 31 772 1000

Uraniborg Ship, owned by Rederi AB Ventrafiken

Typeset in LaTeX
Printed by Chalmers Reproservice
Gothenburg, Sweden 2022

Decision Support for Energy Efficiency Operations of Double Ended Ferry

Daniel Alejandro Vergara Escalona
Department of Mechanics and Maritime Sciences
Chalmers University of Technology

Abstract

A year long navigation data was available for the double ended ROPAX ship "Uraniborg" owned by the company Rensiery Ventrafiken AB. Exploratory data analysis shows that allocating most of the engine's load on the stern-most engine has potential for energy savings. Furthermore, a full black box machine learning XG Boost model-based simulator was built in order to forecast the total fuel consumption for a trip, given the meteocean conditions and some engine-related initial conditions (IC), on the assumption that the speed overground and the route remains the same. The simulator acts as a Decision Support System that allows an expert operator to make decisions on how to allocate the Power on the ship for a given trip, as well as to improve the Engine Speed operation of either engine.

Keywords: Decision Support, XGBoost, Machine Learning, Ropax, Power Forecasting

Acknowledgements

I take time now to write about the people who helped me directly or indirectly while working on this research.

First of all I would like to give credit to **Martin Alexandersson**, my project supervisor, I am quite confident that he on his own would have done these calculations with no problem. He did however helped me greatly while learning how to program in Python and how to wrangle data and for that I am truly grateful.

My examiner **Wengang Mao** gave me the chance to work on this project and by his continuous support during the whole research and by all the suggestions he gave me. His support made my code better for sure.

Xiao Lang was a huge help by providing me with the sea current data as well as teaching me how to optimize some of the machine learning models used in this documents.

And **Linus Olsen** CEO of Rederi AB Ventrafiken for trusting me with this data and with carrying out some of the suggestions we made on the operation of the ship.

There are however other people, that, while not being directly involved in the project by any reason still gave me support or encouragement while working on this research.

My cousin **Leonardo Orejarena** happens to be the first mate of a similar ship. He did not hesitate to answer my questions regarding to how the operations were and was extremely supportive about it. He made me understand better what is important from the crew perspective and how decisions are made on board and specially on how did the control of the ship worked.

Finally to **Andrea Savio** with whom I had too many conversations regarding to how the coupling "Propeller-Engine" work and on which variables were important for which. From those discussions I challenged my ways of thinking and my explanations became more clear, specially in what regards to making my explanations simpler and not abusing notation.

Daniel Vergara, Gothenburg, June 2022

Contents

List of Figures **xi**

List of Tables **xiii**

1 Introduction **1**
1.1 Objectives . . . 2
1.2 Limitations . . . 2
1.3 Outline . . . 2

2 Case Study & Theory **3**
2.1 Ship Propulsion Theory . . . 3
2.2 Internal Combustion Engines . . . 5
2.3 Decision Support System . . . 5
2.4 Supervised Machine Learning Models . . . 5
2.4.1 Ordinary Least Squares . . . 6
2.4.2 Support Vector Machines . . . 6
2.4.3 Decision Trees . . . 6
2.4.4 Random Forest . . . 7
2.4.5 XGBoost . . . 7
2.4.6 Artificial Neural Networks . . . 7
2.5 Case Study . . . 9
2.5.1 Ship Specification . . . 9
2.5.2 Engine Specification . . . 10
2.5.3 Route . . . 10

3 Methods **11**
3.1 Output of the work . . . 11
3.2 Software requirements . . . 12
3.3 Machine Learning Pipeline . . . 13
3.3.1 Data Acquisition . . . 13
3.3.2 Data Pre-processing . . . 14
3.3.3 Missing Data . . . 15
3.3.4 Denoising . . . 15
3.3.5 Derived measurements . . . 15
3.3.6 Train Test Split . . . 18
3.4 Model Fitting . . . 19
3.4.1 Machine Learning Pipeline . . . 19

3.4.2 Baseline & Performance . . . 19
3.4.3 Feature Selection . . . 19
3.4.4 Feature Refinement . . . 20
3.4.5 Data Normalization . . . 20
3.4.6 Model Training . . . 21
3.5 Pure Data Analysis . . . 22
3.6 Model Selection . . . 24
3.6.1 Data down-sampling effect . . . 24
3.6.2 Resolution Influence . . . 24
3.6.3 Hyper-parameter tuning . . . 24
3.7 Building the Decision Support . . . 25
3.7.1 Fuel Consumption Predictor . . . 25
3.7.2 Input Validation Block . . . 26
3.7.3 Trip Consumption . . . 28
3.7.4 Additional Blocks . . . 29

4 Performance Modelling Results **31**
4.1 ML - Model Selection . . . 31
4.2 Model Evaluation . . . 31
4.3 Hypter parameter tuning . . . 32
4.4 Model Forecasting . . . 33
4.5 Initial Condition Validation Block . . . 38
4.6 Simulator test . . . 39

5 Decision Making & Trip Simulation Results **41**
5.1 Trip simulations . . . 41

6 Conclusion & Future Work **47**
6.1 Conclusion . . . 47
6.2 Future Work . . . 48

Bibliography **49**

List of Figures

2.1 Example of a multi-layer perceptron (sk-learn) 7
2.2 Ship's General Arrangement. 9
2.3 Ship's Engine Arrangement. 9
2.4 Ship's Engine 10
2.5 Heatmap of one year operation of the ship across the route Ven-Landskrona. 10

3.1 General Structure of the Decision Support System 12
3.2 Example of the SoG profile of a sample trip. 14
3.3 Definition of the acceleration zones. 16
3.4 Engine power curves (Caterpillar, 2009) 17
3.5 Maximum power lookup table (Caterpillar, 2009) 17
3.6 Feature Correlation Matrix 20
3.7 Histogram of the power allocation ratio for each trip over a year. . . . 22
3.8 Total Fuel Consumption for all trips during the year 23
3.9 Total Fuel Consumption Block. 26
3.10 Input Validating Mode 1 - In: RPM_2, PR Out: RPM_1 27
3.11 Input Validating Mode 2 - In: RPM_1, PR Out: RPM_2 27
3.12 Input Validating Mode 3 - In: RPM_1, RPM_2 Out: PR 27
3.13 Integration block 29
3.14 Engine Load Block 29

4.1 Training Comparison between Linear and Black Box Models for Instantaneous Total Fuel Consumption 33
4.2 Training Comparison between Linear and Black Box Models for ME1 Speed (rpm) 34
4.3 Training Comparison between Linear and Black Box Models for ME2 Speed (rpm) 35
4.4 Training Comparison between Linear and Black Box Models for ME1 Power (kW) 36
4.5 Training Comparison between Linear and Black Box Models for ME2 Power (kW) 37
4.6 Initial Conditions input Validation Block. 38
4.7 Output RPM2 from the Validation Block 38
4.8 Initial Conditions for the simulated samples. 39
4.9 Expected total fuel consumption from the simulated operation. 40

5.1 Speed profile of the one sample trip 41
5.2 Initial Conditions for the simulated trip. 42
5.3 Output RPM1 from the validating block 42
5.4 Expected total fuel consumption from the simulated operation. 43
5.5 Speed over ground profile - Second Trip 43
5.6 Initial Conditions for the simulated trip. 44
5.7 Expected total fuel consumption from the simulated operation. 45

List of Tables

2.1 *General Specification Uraniborg* 9

3.1 Required Python libraries and their versions 12
3.2 Available measurements in the raw data-set 13
3.3 Derived Measurements 15
3.4 Control parameter of different Black Box Machine Learning Methods 21
3.5 Default settings XGBoost r^2 coefficient for different down-sampling rates 24
3.6 Normalized Features 25
3.7 Targets 27
3.8 Final Features 28
3.9 Target-Features of the Inverse maps 28

4.1 Linear Models 31
4.2 Black Box Models 31
4.3 Black Box Models 32
4.4 Hyperparameter tuning XGBoost 32
4.5 Integration block result for 1000 samples. 40

5.1 Comparison between the proposed operation and the measurements . 43
5.2 Comparison between the proposed operation and the measurements . 45

1
Introduction

Conventional internal combustion engines are often used as main propulsion engines for most ships. They run on hydrocarbons, which are reservoirs of chemical energy and in their most common and easily available form are derived from Oil and labelled "Fossil Fuels". However, hydrocarbons regardless of their origin are expensive and due to the involved combustion process, contribute to environmental emissions.

The scope of this document is to improve the energy consumption for a case study of one year from different operative measurement records that were provided by Swedish transport company Ventrafiken for the double-ended ROPAX ship Uraniborg. This ship covers the route Ven-Landskrona back and forth nine times a day (18 trips a day total). Improvements in energy consumption for this operation are more about changing the way the ship is operated rather than it is to choose a route for these short trips (c.a. 30 min) or to plan weather routing, as huge changes in weather conditions are not expected within the area during such short trips.

It is of importance to understand what factors can be controlled in ship operation, what can be modified and what is the target of the operation of the ship. From the crew perspective, and the customer, the most important feature is to be able to arrive in time, that is, the Estimated Time of Arrival. Therefore, this research is to find external factors that affect energy expenditure for the case study ship Uraniborg and to use data records to support the ship's operation decision making, while keeping in theory the same quality of service or improving it. For this task a full data preparation, processing and interpretation was executed as well as carrying out data manipulations for extra features not provided in the initial data set. The scope of such manipulations was to find hidden trends within the data that would allow to determine what operational parameters have a higher influence on energy expenditure.

Furthermore, in order to assess the energy savings for future trips and help with the decision making and trip planing, Machine Learning regressions from different models were implemented varying from linear to black-box and deep-learning models, to predict the instantaneous power demand as a function of different operative parameters. The objective of these regressions is to help with Eco-driving, that is, to reduce the energy that is consumed in different trips by making a decision on how to operate based on model predictions. Furthermore, the whole statistical analysis of individual trips may help with post-trip analysis and to assess how good they were compared to similar trips.

1.1 Objectives

Design a Data Driven Decision Support System to test ship's operational strategies that result in energy savings from the provided measurements from the RO-PAX Uraniborg by using Data Analysis.

- Full pre-processing and analysis of the operational data o find trends within the data.
- Feature extraction with the help of physical models and domain knowledge that allows for model fitting.
- Fuel Consumption simulations and tests.

1.2 Limitations

Due to time limitations, validation of the findings on the actual ship fall outside the scope of this analysis. Additional data related to the ship's displacement that would be beneficial for most of the models was also missing and therefore fell outside the scope as well.

1.3 Outline

This document is split into chapters, each of which introduces a different aspect of the analysis carried out during the research. At first the background of the problem with the characteristics of the ship and its main particulars, then the methods that were implemented to analyse the problem and the thought process to replicate the results, followed by the results obtained after the application of the methods, the discussion of the results, the conclusion of the work and finally future aspects to consider in order to improve the research.

2

Case Study & Theory

This chapter introduces the baseline concepts required to understand what the rest of the document is about and aids as a guideline to understand the methodology.

It starts with a brief description of some theoretical aspects of ship's propulsion such as propeller theory and some remarks regarding to internal combustion engines. It then follows up with an overall explanation of different machine learning regression models to show in detail what does each model do. Finally the case study ship is presented including details of its general arrangement and propulsion system and information about the ferry's route.

2.1 Ship Propulsion Theory

In this text, more than to focus in the exact calculations for each of the components of the resistance, thrust and power, a more heuristic approach was considered as it was more important to focus on knowing the different physical quantities that compose those terms rather than to focus on the exact analytical calculations due to the data-driven nature of the machine-learning modelling approach that was carried out for the different targets (Power, Engine's Speed, Ship's speed, etc). Considering these macro dependencies allowed for the selection of the lowest amount of data (features) that were physically related with the phenomena and that were used for the modelling aspect of this work.

For a ship's hull to move through water it has to overcome a resistance force (R_T) that the water applies on it (Birk, 2019). In general this total resistance is furthermore decomposed into two components, a wave making resistance (R_W) and a viscous resistance (R_V). This wave making resistance is a function of Froude's Number (Fr) while the Viscous Resistance (not considering air) is a function of the Reynolds Number (Re) given by:

$$Fr = \frac{U}{\sqrt{(g \cdot L_{WL})}} \tag{2.1}$$

$$Re = \frac{U \cdot L_{OS}}{\nu} \tag{2.2}$$

The macro effect that dominates both non-dimensional numbers is the ship's speed through water. Uraniborg does not measure the speed through water (U) but the

speed over ground (U_g). These quantities however are related by both Wind's and Current's speed. In vector from:

$$\vec{U_g} = \vec{U} + \vec{U_W} + \vec{U_{SC}} \tag{2.3}$$

Another aspect that also influences the resistance is the form factor (k), related with the ship dimensions, in particular the draught. Therefore one can say that:

$$R_T = f\left(U, U_W, U_{SC}, L_{WL}, d, \nu, \rho, ..., \right) \tag{2.4}$$

This resistance force has to be compensated with a "towing force" of equal strength as to achieve a constant speed in steady state conditions. For this "towing force" an "effective power" (P_E) is defined as:

$$P_E = R_T \cdot U \tag{2.5}$$

The "towing force" is a thrust force (T) generated by a propulsion system. Furthermore, to achieve constant speed one must have:

$$T = R_T \tag{2.6}$$

However, the presence of the propulsion system affects the flow around the ship and adds resistance. To compensate for this resistance, extra propulsion power/thrust is required. To account for this effect a thrust deduction factor t is considered. Furthermore, due to changes in the flow field due to the interactions with the propeller, the former moves with an advance velocity (U_a) that is smaller than the ship's speed. To account for this effect, a wake fraction (w) is introduced:

$$t = \frac{T - R_T}{T} \tag{2.7}$$

$$w = \frac{U - U_A}{U} \tag{2.8}$$

Analogous to the effective power, a thrust power can be defined as:

$$P_T = T \cdot U_A \tag{2.9}$$

These quantities allow to introduce the different efficiencies and non-dimensional numbers that allow to evaluate the performance of a propulsion system. In particular one definition:

$$P_D = \omega \cdot Q = \frac{1}{30} \cdot \pi \cdot n \cdot Q \tag{2.10}$$

The delivered power, with n the shaft's speed of the propeller in rpm. Related to the engine operation by the torque Q, which is provided by the engine, and the shaft's speed, which is proportional to the engine's speed by the use of a gear box. Because of these definitions the "Behind efficiency", that is, the efficiency of the propeller behind of the ship, can be defined as:

$$\eta_B = \frac{P_T}{P_D} = \frac{T \cdot V_A}{\frac{1}{30} \cdot \pi \cdot n \cdot Q} \tag{2.11}$$

In summary, one can define all the variables that relate the propeller operation with a function as follows:

$$P_D = f\left(n, V_A, T, Q, ...\right) \tag{2.12}$$

An analysis between which of these quantities were available and which were necessary for modelling of the different sub-systems on this work follows on the "Feature Selection" section.

2.2 Internal Combustion Engines

One important remark regarding to Internal Combustion Engines is that the relationship between the fuel consumption and the power delivered, that in turn, translates into torque (Q_{Eng}) is given by:

$$P_{Eng} = Q_{Eng} \cdot \omega_{Eng} = \eta_{th} \cdot \left(\dot{m_{fuel}} \cdot LHV\right) \tag{2.13}$$

With η_{th} the thermodynamic efficiency of the engine's cycle, $\dot{m_{fuel}}$ the mass flow rate of the fuel, and LHV its lower heating value. However, this equation provides no additional information other than relating the total fuel consumption with the Power Delivered by the means of the torque and the engine's speed.

2.3 Decision Support System

Following the definition from Gottinger and Weimann (1992) a decision support system (DSS for short) is any set of tools that process information and aid with decision making. It takes information from a database and selects what is important and builds a set of rules that would allow a human-expert to make a decision. In this broader sense the use of artificial intelligence (AI) and Machine Learning modelling and simulations would allow for instance to develop predictions from user-input conditions that target a feature that the user wants to be improved.

2.4 Supervised Machine Learning Models

Supervised Learning is a Machine Learning problem where one uses some data (named "training data") and multiple features (x_i) to predict a target variable (y_i). Often the objective is to solve an optimization problem and to minimize an objective function given a set of parameters (θ). It is this objective function that distinguishes different machine learning models. In general, the objective function is defined as:

$$Obj(\theta) = L(\theta) + \Omega(\theta) \tag{2.14}$$

Where L is the training loss function and Ω the regularization therm. The former measures how good model predictions are. The later, how complex the model are. The following Supervised Learning models were programmed and tested as to design the decision support:

2.4.1 Ordinary Least Squares

This is perhaps the simplest machine learning supervised learning model. It fits a linear model with "n" coefficients and aims to minimize as the objective function the sum of squared errors respect to said coefficients, that is, the 2-norm of the distance between the model prediction and the actual measurement. This is given by the following expression:

$$obj(\theta) = min_\theta \left|\left| y(\hat{\theta}, x) - y \right|\right|_2^2 \tag{2.15}$$

$$\theta = \left(\omega_1, \omega_2, ..., \omega_i, ..., \omega_n, \right) \tag{2.16}$$

In this document three different alternatives for ordinary least squares are considered. These are polynomial models determined by different multiplicative combinations of the features, that is, feature combinations given by the following models:

$$y(\hat{\theta}, x) = \omega_0 + \sum_{i=1}^{n} \left(\omega_i \cdot x_i\right) + \sum_{j=1}^{n}\sum_{i=1}^{n} \left(\omega_{ij} \cdot x_i \cdot x_j\right) + \sum_{k=1}^{n}\sum_{j=1}^{n}\sum_{i=1}^{n} \left(\omega_{ijk} \cdot x_i \cdot x_j \cdot x_k\right) \tag{2.17}$$

The model is said to be Linear if $\omega ij = \omega_{ijk} = 0$, Quadratic if $\omega_{ijk} = 0$ and Cubic otherwise. There is no regularization on this formulation. It is robust, however, the higher the order of the model the higher the overfit.

2.4.2 Support Vector Machines

Support vector machines are solutions to a convex optimization problem in higher dimensional spaces. In general they build hyperplanes in high (or even infinite) dimensional spaces and solve for the one with the largest distance to the training set (maximum margin hyperplane). They work on "kernel functions" that map the original space into higher dimensions and that ensures that the dot product between pairs of the new map is defined in terms of the original one. It aims to solve the problem:

$$min_{\omega,b,\zeta} \frac{1}{2}\omega^T\omega + C\sum_{i=1}^{n}\zeta_i \tag{2.18}$$

$$y_i(\omega^T\phi(x_i + b) \geq 1 - \zeta_i \tag{2.19}$$

$$\zeta_i \geq 0, i = 1, ...n \tag{2.20}$$

This approach is computationally demanding and therefore its performance suffers when the data is abundant.

2.4.3 Decision Trees

Decision Trees (DTs) are a supervised learning approach to mimic decision making by humans. They are predictive models that take the observations as branches and yield conclusions about the objective target in the leaves. For the scope of this master thesis a Decision Tree will be used as a synonym of "Regression Tree" as

they are used on this task only for regression problems. The goal of a DT is to assemble a map from $n-$features into at least one target. There are many types of decision trees, however two main approaches were used, although their mathematical formulation will not be discussed due to their complexity, these approaches being "Random Forest" and "Boosted Trees" under the umbrella of "Ensembled Trees":

2.4.4 Random Forest

It is a sub category of decision trees under the umbrella of ensemble trees. They have the characteristic that each branch does not necessarily uses all the information from the input features but a parallel sub-set of them with the purpose of reducing variance in the estimator. The outputs of the estimator are then averaged and a final output is achieved (Breiman, 2001).

2.4.5 XGBoost

Extreme gradient boost is an implementation of gradient boosting trees. Similarly to Random Forest they are also ensemble decision trees. It builds short and simple trees that can predict what the previous ones cannot. These are called "weak learners". On each iteration a new weak learner is created until a stopping criterion is satisfied.

2.4.6 Artificial Neural Networks

Based on the multi-layer perceptron. They are defined as non-linear combinations of $n-$input features into $m-$output targets called input layer and output layer respectively. In between the input and output layer one or more hidden layers are used to produce non-linear combinations between the $(i-1)^{th}$ and the i^{th} layer. These non-linear relations are achieved through "activation functions". If the activation functions are all linear, then the neural network converges to a "fancy" linear model. A representation of a neural network follows:

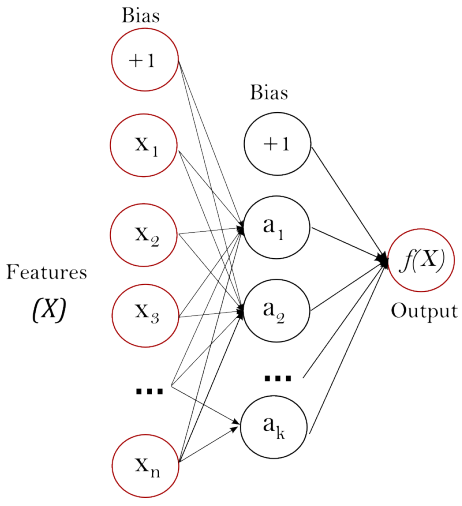


Figure 2.1: Example of a multi-layer perceptron (sk-learn)

The $(n-1)^{th}$ layer then passes through a final activation function which is a map that converges into the dimension of the space of the target objective that processes the final output. The mathematical formulation is more complex than this description but in practice the sk-learn library has a ready-to-use neural network function that can be used for regression problems.

2.5 Case Study

Uraniborg is a double-ended ferry that transits the route Ven-Landskrona 18 times a day. It has symmetric main engines and azimuthal propellers on the fore and aft of the ship that enhances its maneuverability.

2.5.1 Ship Specification

Table 2.1 summarizes the overall specifications of the ship's general arrangement.

Table 2.1: *General Specification Uraniborg*

Gross Tonnage	1143 ton
Lenght Overall	49.95 m
Beam	12.00 m
Depth moulded	3.10 m
Main Engine	2x Caterpillar C32 ACERT 709 kW 1600 rpm each.
Speed	11.50 knots
Propulsion	2x Schottel STP 550. 3+3-bladed twin propeller. Diameter 1.5m.

It is a double ended ferry designed to transit a short route. Once the ship arrives to port the bow becomes the stern and therefore it does not have to rotate.

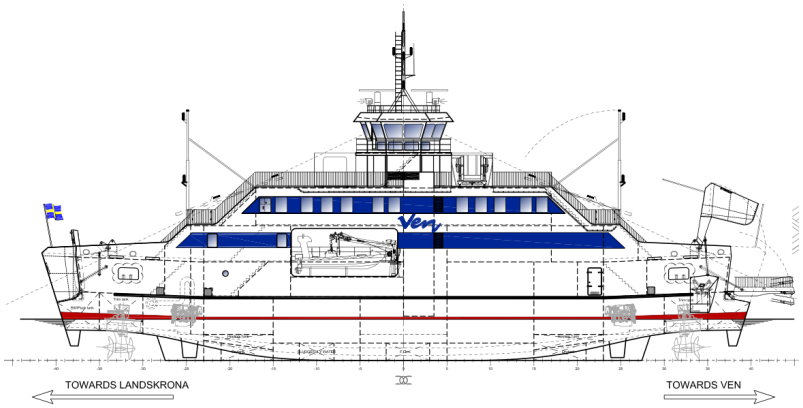


Figure 2.2: Ship's General Arrangement.

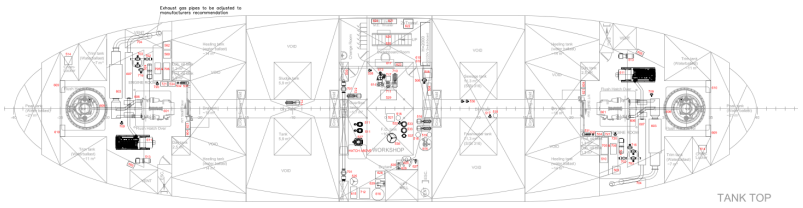


Figure 2.3: Ship's Engine Arrangement.

However, either engine can be used for propulsion as they can be operated independently of each other due to the nature of the double ended ship. Designing a method that allows to find configurations that comprises both performance and energy consumption makes for a good decision support system.

2.5.2 Engine Specification

Figure 2.4: Ship's Engine

Onboard the ship as the propulsion system there are two identical Internal Combustion Engines, each of 746kW. Their specifications follows on figures 3.4 and 3.5. From these engines, measurements regarding to their loading condition and rotation speed were available.

2.5.3 Route

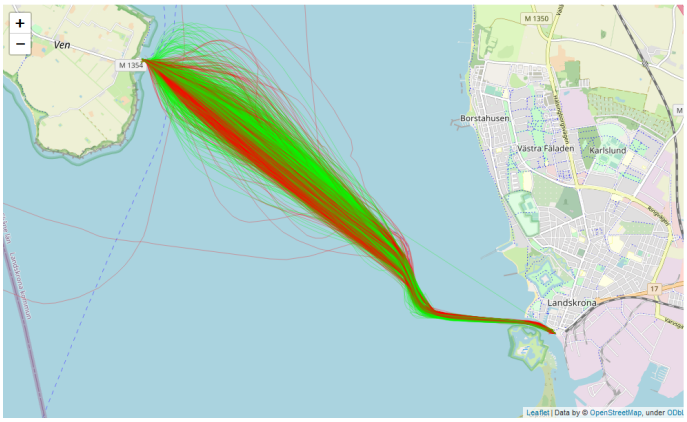


Figure 2.5: Heatmap of one year operation of the ship across the route Ven-Landskrona.

A heatmap of the route between Landskrona and Ven is shown in figure 2.5. The route does not have major changes and the total distance is approximately 4 Nautical Miles between ports. The fact that the route is short enhances the hypothesis that improvements in energy consumption are mostly related to how the engines are operated.

3
Methods

This chapter presents a description of the pipelines used to work the given data for data mining and system modelling in the scope of developing a decision support system that allows to evaluate different operation modes in the ship's engine and hence to make a decision on future trips based on those results.

The data mining part consist on the analysis of the measurements by testing one hypothesis that can be used for trip optimization. The area of competence of this task falls under the category of "Data Analysis" and comprehends different approaches to obtain information from data.

The modelling aspect follows with the implementation and selection of different machine learning regression-models to fit data and the comparison of their performance to choose the most accurate one. This falls under the umbrella of "Machine Learning techniques" as well as "Modelling and Simulation".

The final model was built into a fully functional simulator that acts as the decision vector and that can provide accurate predictions of the trips based on some user defined inputs.

This chapter also includes details regarding to the software and coding required to replicate the work.

3.1 Output of the work

The general objectives of this work were discussed in the introductory chapter. Therefore in here a more in depth explanation of the Decision Support System is presented as well as its design process. The final output of this work is a functional "Total Fuel Consumption" simulator that would allow, from some operational parameters, to evaluate different engine configurations for a given trip while keeping the same Estimated Time of Arrival (ETA), speed over ground and even the same route and meteocean conditions. This permits to evaluate if a given trip could be improved and provides insightful information that would allow to make a future decision on the ship's operation.

The following block diagram shows the inputs and outputs of the simulator as well as the general structure of the decision support system:

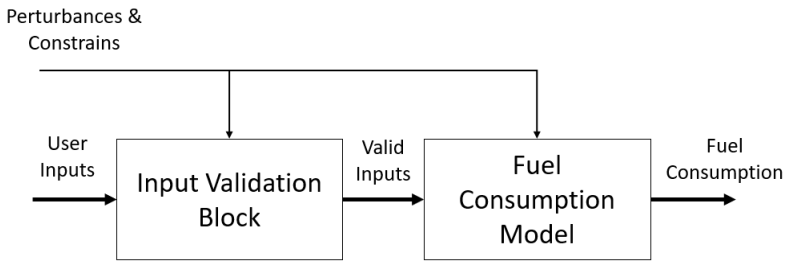


Figure 3.1: General Structure of the Decision Support System

The two main components of the Decision Support System are:

1. Fuel Consumption Predictor
2. Input Validation Block (I.V.B.).

The idea of the following sections is firstly to show steps that were involved in building the different models inside the aforementioned blocks using data analysis and machine learning techniques, as well as introducing an optimization hypothesis to test on sample trips. Then a full explanation of what is inside of the Validation and Predictor blocks is carried out in detail as well as simulations of different scenarios for completeness.

3.2 Software requirements

As mentioned this task was carried out solely in the programming language *Python* version 3.8.8 using Anaconda^{TM} as the distribution and package manager, Jupyter Notebook version 6.4.8. as the working environment. Furthermore, the following libraries are required for repeatability:

Library	Version	Use
Numpy	1.20.1	Algebraical and numerical manipulations.
Pandas	1.4.1	Data wrangling and processing.
Matplotlib	3.3.4	General data visualization.
Seaborn	0.11.2	Data visualization.
Sklearn	0.0	Machine Learning models.
Statsmodels	0.12.2	Statistical analysis.
Joblib		Saving and Loading trained models.

Table 3.1: Required Python libraries and their versions

Each library fulfills a role in this analysis. They are open source and free to use for research and commercial applications. The most important libraries however are Pandas, Numpy and Sklearn as those are the go-to for data processing problems.

3.3 Machine Learning Pipeline

To make models that are useful for the research, a machine learning pipeline was established. The general procedure of a Data Analysis pipeline is as follows (Abebe et al.):

1. Data acquisition
2. Data pre-processing
3. Feature refinement
4. Train and Test Split
5. Feature Engineering & Data Processing
6. Learning Algorithm Selection
7. Hyperparameter Tuning
8. Model Selection & Validation
9. Model Evaluation

3.3.1 Data Acquisition

Measurement	**Units**	**Symbol**
Latitude	deg	Lon
Longitude	deg	Lat
Speed over ground	kts	U_G
Course over ground	deg	Cog
Heading	deg	θ
Depth below keel	m	h
Rate of turn	deg/min	ψ
Fuel Consumption ME$_i$	l/min	FC_{MEi}
Engine speed ME$_i$	rpm	n_{MEi}
Engine load ME$_i$	%	EL_{MEi}
Thruster angle i	deg	Φ_{Thi}
Thruster speed i	%	$\hat{n}_{thi}$
Thruster speed i	rpm	n_{thi}
Trim	deg	ϕ
List	deg	β
Apparent wind speed	m/s	U_{RW}
Apparent wind direction	deg	θ_{RW}

Table 3.2: Available measurements in the raw data-set

The features given correspond to the measurements taken in real time by the company Blueflow onboard the ship Uraniborg. Following ISO 19030-2016 each measurement was recorded with a sampling time (T_s) of at least 15 s.

Proper understanding of the data-set was necessary to reduce redundancy of features and to improve performance of the modelling. Therefore, with the information

provided by the board of Ventrafiken and with the aid of the Main Engine specification, some explanations of the different measurements were found. For example, regarding to "**Engine load ME1/2**" and the general operation of the ship. It was found that "**Engine load ME1/2**" represents the percentage from the "maximum engine load" at a given engine speed.

Additional information regarding to sea currents was imported through the aid of "**Copernicus**". The hypothesis in this case was to hold the sea current constant during each hour of a day. This allows to add extra meteocean information for modelling. Furthermore, physical properties of sea water in the region were assumed to be constant at all times and that had no importance on the modelling.

3.3.2 Data Pre-processing

A first step in the Decision Support System was to assess each individual trip. In this step the raw data was firstly split into individual trips given by the shape of the velocity profile, the closeness to each of the ports and the trip duration. Given the trips schedule it was known that each trip has a duration of approximately 30 minutes in between the geographical coordinates (55.8655º,12.8253º) and (55.9037º,12.7228º) which correspond to Landskrona and Ven respectively. It was therefore simple to determine that for each individual trip the velocity profile was given by a curve:

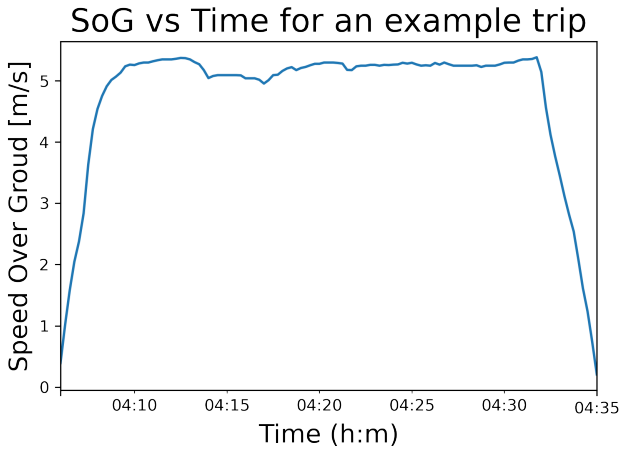


Figure 3.2: Example of the SoG profile of a sample trip.

Based on the shape of the curve, the position, and the history the following filters were applied:

- Trip Start
  - SoG = 0 at time k-1 & SoG > 0 at time k
  - Longitude, Latitude at (12.8253º, 55.8655º) or (12.7228º, 55.9037º).
- Trip Ends
  - SoG > 0 at time k-1 & SoG = 0 at time k
  - Longitude, Latitude at (12.7228º, 55.9037º) or (12.8253º, 55.8655º).
- Minimum transit time 700 seconds.

- Maximum transit time 2400 seconds.

This procedure allowed to isolate the individual trips and to remove every other operation. Furthermore, these splits also allowed to define each trip direction and each trip number.

3.3.3 Missing Data

No measurement is perfect and in particular earlier months data were defective. However, due to the high amount of data missing values ('nan') they were omitted.

3.3.4 Denoising

Downstream the pipeline the data was downsampled even more as to apply a mean value filter. Downsampling achieves to a degree outlier detection and compensates for using outlier data. The dataset however was denoised when it was handled by the company. Engine Operation data lower than the minimum design RPM was filtered out as those were considered outliers.

3.3.5 Derived measurements

Once the dataset was split into individual trips they were also split into three regions based on the governing dynamics as seen from figure 3.3, that is, acceleration, steady state and deceleration. Furthermore new features were computed from the existing ones. These features are:

Table 3.3: Derived Measurements

Measurement	**Unit**	**Symbol**
Acceleration	m/s^2	a
True North Wind	m/s	U_{WN}
True North West	m/s	U_{WW}
Engine Power Demand	kW	P_{MEi}
Engine Power Ratio	(-)	PR
Specific Fuel Consumption	l/kWh	$SFOC$
Trim	m	Tr
Depth Froude Number	(-)	Fn_D

The Acceleration definition follows from figure 3.3:

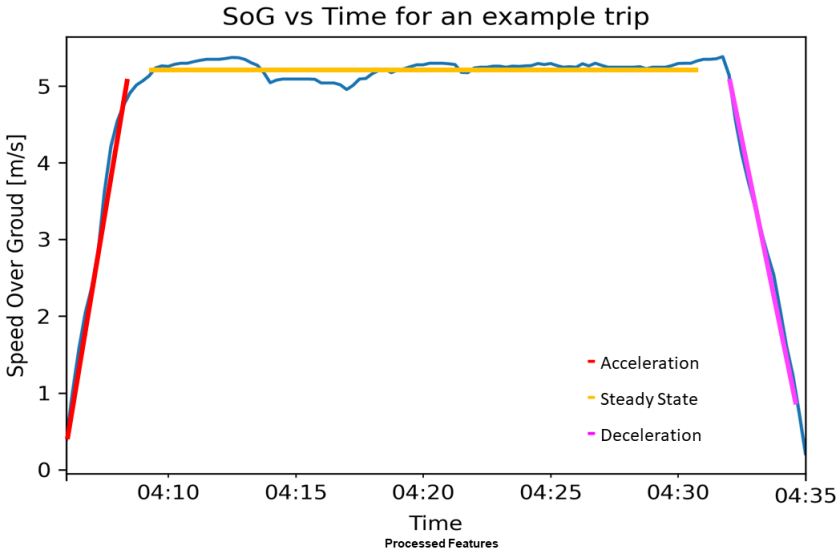


Figure 3.3: Definition of the acceleration zones.

Acceleration, or acceleration in the direction of motion (surge) is computed as the slope of a solution to the least squares line fit and assumed to be a constant value. Surge Acceleration was not directly measured, however it was assumed that it was a constant value until the ship arrives to cruise speed at which acceleration becomes neglect-able and the change in speed is assumed to be zero. For the "acceleration zone" the first 20 measurements were used to solve for the slope. Similarly, for the "deceleration zone" the last 20 measurements were used. Everything in between would then correspond to the Steady State condition.

True North Wind and West are computed following ISO 19030-2016-2, and it was determined by the following equations for apparent wind measurements:
North Wind was defined as:

$$V_{wt} = \sqrt{(V_{wr}^2 + V_g^2 - V_{wr} \cdot V_g \cdot cos(\Phi_{wr}))} \tag{3.1}$$

$$\Phi_{wt} = tan^{-1}\left\{\frac{V_{wr} \cdot sin(\Phi_{wr} - \Phi 0) - V_g \cdot sin(\Phi_0))}{V_{wr} \cdot sin(\Phi_{wr} - \Phi 0) - V_g \cdot cos(\Phi_0)}\right\} \tag{3.2}$$

$$\Phi_{wt} = tan^{-1}\left\{\frac{V_{wr} \cdot sin(\Phi_{wr} - \Phi 0) - V_g \cdot sin(\Phi_0))}{V_{wr} \cdot sin(\Phi_{wr} - \Phi 0) - V_g \cdot cos(\Phi_0)}\right\} + 180 \tag{3.3}$$

Regarding Engine Power Demand and Engine Power ratio, referring to the Ship's Engine Specification and by understanding the measurement under the Label "Engine Load (%)" that represents the percentage of the maximum load the engine is being demanded at current RPM values, the "Engine Power Demand" was computed by interpolation of the Engine Speed and the engine's Speed-Power curve given by:

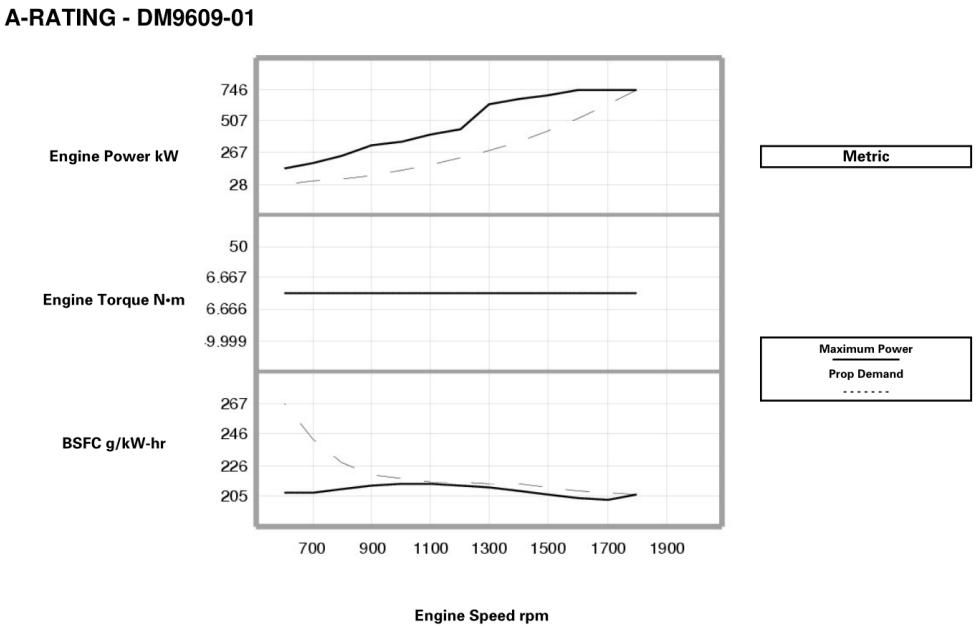


Figure 3.4: Engine power curves (Caterpillar, 2009)

And parameterized by table 3.5:

Maximum Power Data

Engine Speed rpm	Engine Power kW	Engine Torque N·m	BSFC g/kW-hr	Fuel Rate L/hr
1800	746	3958	204.9	182.2
1700	746	4190	202.3	179.9
1600	746	4452	202.7	180.2
1500	707	4501	205.3	173.0
1400	671	4577	208.1	166.4
1300	629	4620	210	157.5
1200	450	3581	211.4	113.4
1100	400	3472	212.5	101.3
1000	355	3390	212.4	89.9
900	315	3342	210.8	79.1
800	245	2924	208.8	61.0
700	190	2592	207.1	46.9
600	145	2308	206.5	35.7

Figure 3.5: Maximum power lookup table (Caterpillar, 2009)

Then the actual power demand is computed as:

$$P_{Eng} = P_{Eng}^{Max}(RPM) \cdot EngineLoad(\%) \tag{3.4}$$

Furthermore, as the Ship is doubled-ended, on each trip it is possible to use either engine for propulsive power. A new operating non-dimensional variable called

"Power Ratio" (PR) was defined. It indicates how much power comes from which engine at every measurement instant and it is given by:

$$PR_{ME1} = \frac{P_{ME1}(RPM_1)}{P_{ME1}(RPM_1) + P_{ME2}(RPM_2)} \tag{3.5}$$

Finally the depth froude number was computed following Laurie et al. (2021) as:

$$Fn = \frac{U_g}{\sqrt{h \cdot g}} \tag{3.6}$$

3.3.6 Train Test Split

Depending on the Machine Learning approach, that is, interpolation or forecasting, different amounts of data were used. Following (Lang et al., 2022), different machine learning training schemes were considered. Furthermore, due to the dynamics of the system and the different influence of sea-currents and wind in either direction, the data was further split into two (2) sets per directions: Westbound and Eastbound trips, and three (3) sets per dynamical region: Acceleration, Steady condition, Deceleration. The first set was straightforward to define from the GPS coordinates of the beginning and the end of the trip, whilst the the dynamical set was defined empirically by evaluation of whole data wrangles of speed from which it was determined that roughly the first five (5) minutes of each trip correspond with the acceleration region, the last five (5) minutes correspond with the deceleration region and everything in between was considered as the steady state region.

For the rest of the document in what regards to prediction models, the data was split using 80% of the processed data as the training set and 20% as the test set. In order to avoid data leaking (Kaufman et al., 2012) the data was split not randomly, but chronologically, this way the model prediction test were not influenced by "future events" ("*it rains on rainy days*").

3.4 Model Fitting

3.4.1 Machine Learning Pipeline

Following (Abebe et. al. 2020), different many machine learning schemes were tested through to find a solution that was as accurate as possible whilst being useful for the ship owners for forecasting (extrapolations) from newer measurements. The final pipeline used in this work is as follows:

1. Down-sampling of the input features to 15s, 1m, 2m or 5m.
2. Correlation analysis of input variables.
3. Random Forest for feature importance.
4. Ordinary Least Squares Models (Linear, and Polynomial)
5. Black Box Models (XGBoost, SVM, Neural Networks)

3.4.2 Baseline & Performance

Establishing a baseline allows for model performance evaluation. In this case, the baseline is the mean value across the whole training/test set. This allows to evaluate the performance of the models by using the coefficient of determination r^2 defined as:

$$r^2 = 1 - \frac{\sum(x - \hat{x})^2}{\sum(x - \bar{x})^2} \tag{3.7}$$

The closer this coefficient is to unity the better the model fits the data. A negative r^2 indicates that the baseline has a better performance than the regression prediction. Improving this coefficient was key for the reliability of the models.

3.4.3 Feature Selection

Feature selection allows to identify how important variables are for a prediction and allow to remove unnecessary measurements. In this way the computational power required to produce the models is reduced whilst improving model performance. With all data already cleansed and refined features were now chosen by using physical models.

Following propulsion theory (Carlton 2007) the following relationships can be stated:

- Thrust is a function of resistance. $T = f(R)$
- Resistance is decomposed in Viscous and Wave Making. Indicatively, it is a function of displacement (Δ), Speed over ground (V_g), Wind Speed (V_w). $R = f(\Delta, V_g, V_w)$
- The power delivered by the propeller is a function of Torque (Q) and rotational speed (n) at a given ship speed and it clearly has to be able to compensate for the resistance. $P_D = f(n, Q)$

The idea with these feature selections was to chose those features that are intrinsically related with the governing physics of the ship. In the available database, for

example, there is no information related to the displacement, therefore it cannot be used as one of the modelling features.

3.4.4 Feature Refinement

All measurements recorded and all derived measurements were not necessary. Following Abebe et al. (2020) a high correlation filter was implemented to reduce measurement pairs highly correlated in the feature set as they may describe the same process. It follows then:

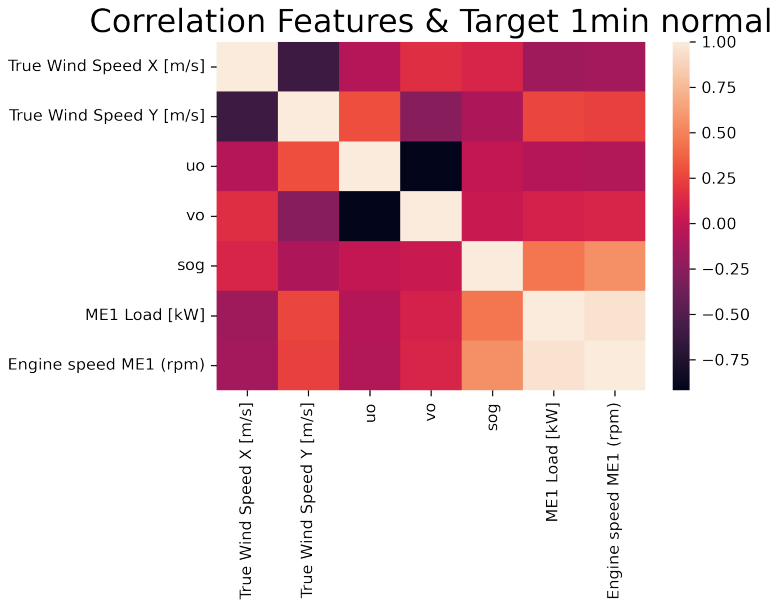


Figure 3.6: Feature Correlation Matrix

High correlation between feature and target was accepted in the case that the final mapping would be useful for the decision support system.

3.4.5 Data Normalization

The features were subjected to Scale Normalization (Min-max Normalization) to redefine their relative relevance before processing. It is defined as:

$$\hat{X}_k = \frac{X_k - min(X_k)}{max(X_k) - min(X_k)} \tag{3.8}$$

Normalization brings the whole map to a non-dimensional 0 to 1 scale. It can be applied regardless of the statistical distribution of the inputs. Normalization however is sensitive to outliers as they will affect the magnitude of the minimum/maximum of the measurement. The result of this data processing approach is a non-dimensional model output that can be re-scaled as:

$$X_k = \hat{X}_k \cdot (max(X_k) - min(X_k)) + min(X_k) \tag{3.9}$$

3.4.6 Model Training

The following machine learning models were trained:

- Random Forest
- Ordinary Least Squares (Linear)
- Ordinary Least Squares (Quadratic)
- Ordinary Least Squares (Cubic)
- Support Vector Machine (Linear Kernel)
- XGBoost
- Artificial Neural Networks (ANN)

As mentioned before the models were trained using 80% of the refined data as the training set whilst 20% was used for testing of the predictions. The models were then "scored" using the r^2 metric and compared to one another for different down-sampling rates. The Ordinary Least Squares models were processed by creating a sklearn pipeline that handled the different polynomial combinations among the input features as well as the training and scoring.

The black box modes required instead more inputs and were tuned as follow:

	Hyper-parameters
Random Forest	Default with 100 estimators
Support Vector Machine	Default with Linear Kernel
XGBoost	Max depth = 100, $\alpha = 0.2$, Max estimators = 100
ANN	Default

Table 3.4: Control parameter of different Black Box Machine Learning Methods

The trained models were then saved using "joblib". That way they did not have to be re-trained every time the code was run. The full code used for the model training can be found in the appendices.

3.5 Pure Data Analysis

An analysis of the ship operation by the means of data analysis for the whole year was carried out.

Engine Power Allocation Hypothesis

Uraniborg being a double-ended ferry, controlled by the fuel throttle in the navigation bridge, can be in theory operated by using either the bow or the stern propeller as the main propulsion unit. As each unit can in theory give the exact same maximum power, it was of interest to check how the engine power was allocated across multiple trips in the same direction considering the trip's fuel consumption and the average power demand during the whole trip. For this the relationship **Fuel Consumption** vs **Engine Power Ratio** was determined computing first a yearly histogram to see how either engine power demand was allocated during the whole year yields:

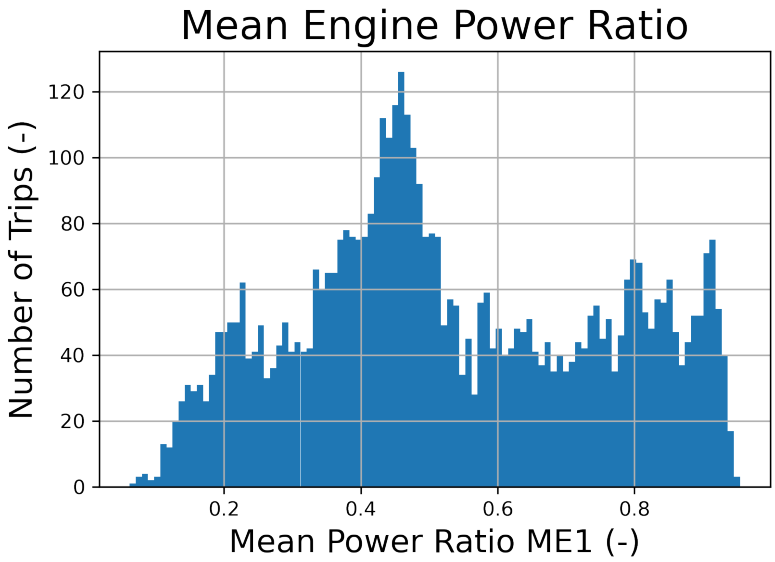
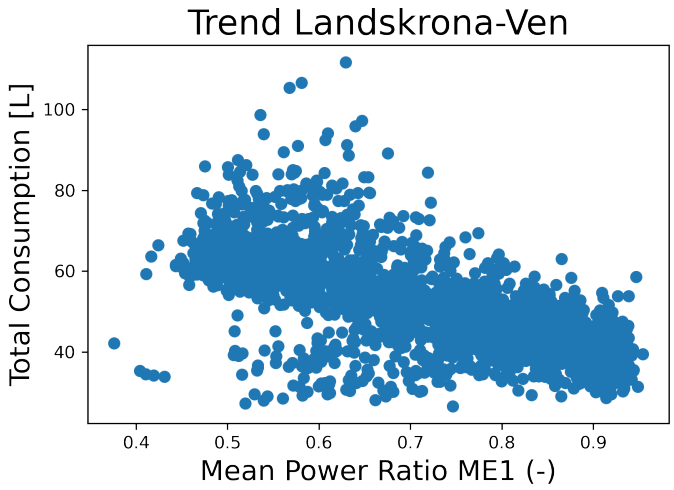


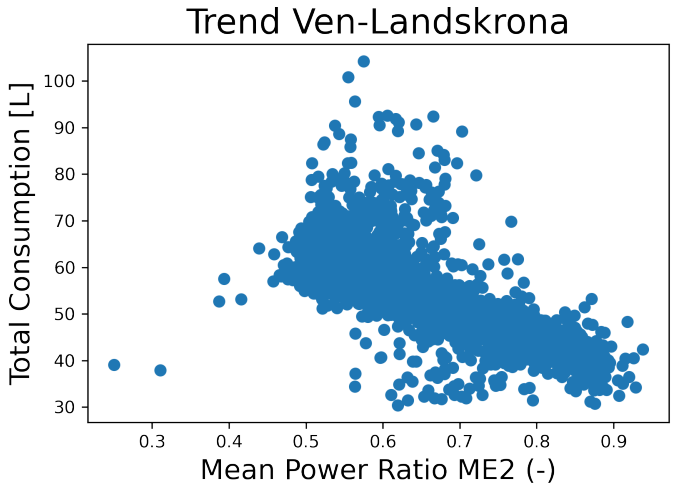
Figure 3.7: Histogram of the power allocation ratio for each trip over a year.

It seems apparent that most trips are located towards the 40 to 60% Mean Power Ratio. It will be shown later that there is a high trend toward using more one engine than the other while travelling in a fixed direction.

By using data mining it was noticed that the choose of the main propulsion unit could have some influence on the total consumption. This resulted in the following two plots. They were created by computing the total fuel consumption of the ship on each individual trip on the same route, that is, Ven-Landskrona/Landskrona-Ven, against the Power Ratio defined by equation 3.5.



(a) Ven - Landskrona



(b) Landskrona - Ven

Figure 3.8: Total Fuel Consumption for all trips during the year

From the previous plots it is apparent that fuel may be saved the most if the stern-wise engine is used as the main propulsive unit. This hypothesis was tested in two simulations and shown in the results. It is also agreed with Ventrafiken that some tests on the actual ship would be carried out to see wether or not this theoretical hypothesis holds true.

3.6 Model Selection

3.6.1 Data down-sampling effect

Having the data sampled every 15s, although beneficial for the machine learning models (as more data is available) is not so great for the computing time. Furthermore, in order to mitigate the influence of possible outliers in the data a "mean value" filter, every $n = 4$, 8 and 20 time steps was applied. This action reduced the number of data by a factor of n and therefore the calculations. However, it had some impact on the resolution of the models.

3.6.2 Resolution Influence

As discussed before, the acceleration and deceleration conditions were assumed to have approximately a 20 time steps (5 minutes) window. Therefore, a down-sample from 15s to 5 minutes would consequently reduce the number of points in those regions to 1 or at most 2 points. Similarly, a 2 minutes down-sampling time would have a negative impact on the resolution. Therefore, a 1 minute down-sampling rate was chosen as the best compromise between resolution and speed. However, calculation of the r^2 coefficients for some models at those sampling rates show:

Prediction	**15s**	**1min**	**2min**	**5min**
TFC	0.970	0.961	0.956	0.857

Table 3.5: Default settings XGBoost r^2 coefficient for different down-sampling rates

Although the use of the lowest (default sampling rate) is the most accurate, one may make an argument to rather use the 1 minute down-sampling rate for the prediction models as it would have to process a quarter of the total data and thus yield faster results without high losses in accuracy. Therefore a 1 minute sampling rate was chosen for the rest of the document.

3.6.3 Hyper-parameter tuning

Hyper-parameters are parameters that configure the settings of a Machine Learning model. They cannot be estimated from the data and must be set a priori before training a model (Yang & Shami, 2020). These parameters can be tuned manually, although that is inefficient. Instead, using a randomized search, the model hyper-parameters were tuned systematically to improve the prediction results. This process was done using sklearn and the function `"RandomizedSearchCV"`. The code used for this purpose can be found in the appendix.

3.7 Building the Decision Support

As stated before the decision support system was built following the block diagram from figure 3.1. Now, after introducing all of the data analysis process and all of the available features a full in depth explanation of what is inside the blocks follows:

3.7.1 Fuel Consumption Predictor

The Fuel Consumption Predictor block is the core of the simulator. It has as inputs the **Engine Speed** for both main engines onboard as well as the **Power Ratio**. Following section 3.4 different machine learning models were trained and fitted using the following normalized features:

Table 3.6: Normalized Features

Target	Unit	Symbol
Wind Speed North	(-)	U_{WN}
Wind Speed West	(-)	U_{WW}
Current Speed North*	(-)	U_{CN}
Current Speed West*	(-)	U_{CW}
Speed Over Ground	(-)	U_G
Engine Speed ME1	(-)	n_{ME1}
Engine Speed ME2	(-)	n_{ME2}
Power Ratio	(-)	PR

These features were selected as they are intrinsically related with the engine operation. The addition of Ocean Currents, that were not available in the original data set, increased the accuracy of the fitness of the prediction (prediction score). The whole feature selection process involved testing different combinations of the input features (refined by considering section 3.4.3). The output of the Machine Learning models is now a map from $f : R^n \rightarrow R$ that produces a prediction of the normalized "Instantaneous Total Fuel Consumption" (TFC) that is later re-scaled again for consistency.

A formal explanation of the simulator follows from the definition:

$$TFC = f_3(X_1, X_2, X_3, ...) \tag{3.10}$$

With X_i each of the control inputs of the system (RPM_2, RPM_1, PR).

The function f_3 represents a non-linear map with 3 control inputs related to the Engine Speed and Power Allocation and 5 exogenous inputs related to external disturbances to the system as well as the constraint speed over ground. This definition is represented as the following block diagram:

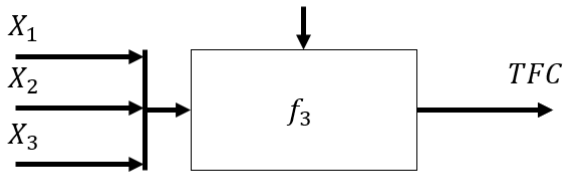


Figure 3.9: Total Fuel Consumption Block.

However, the difficulty of having this block standalone follows from the fact that the three control inputs are not independent of one another as there exists a constrain $G(X_1, X_2, X_3) = 0$ binding them. Such a constraint is given by equation 3.5 and can be written, in general, as:

$$G(X_1, X_2, X3) = X_3 - \frac{f_1(X_1)}{f_1(X_1) + f_2(X_2)} = 0 \tag{3.11}$$

Where each f_i represent the Power Demand to the respective engine and X_i are the Engine Speeds of both ME1 and ME2, and the Power Ratio respectively. Equation 3.11 is the foundation of the validation block(s).

3.7.2 Input Validation Block

As the inputs X_i are constrained by $G = 0$ (eq. 3.11) fixing any two of them also fixes the other. Therefore one shall be careful when selecting the input variables into the system as otherwise one might input contradictory nonsense into the predictor block. To avoid such consistency problems, 3 maps of the form:

$$X_i = g_i(X_j, X_k) \tag{3.12}$$

were computed. These maps are specifically given by:

$$X_1 = g_1(X_2, X_3) = f_1^{-1}\left(\frac{X_3}{1 - X_3} \cdot f_2(X_2)\right) \tag{3.13}$$

$$X_2 = g_2(X_1, X_3) = f_2^{-1}\left(\frac{1 - X_3}{X_3} \cdot f_1(X_1)\right) \tag{3.14}$$

$$X_3 = g_3(X_1, X2) = \frac{f_1(X_1)}{f_1(X_1) + f_2(X_2)} \tag{3.15}$$

Computing the inverse function of a non-linear map is in many cases impossible. Therefore, given the data-driven nature of the simulator, the functions f_i^{-1} where instead fitted by using the same machine learning pipeline from 3.4 and redefined as f_i^*. In this case, four different machine learning models were trained for the following target variables:

Table 3.7: Targets

Target	Unit	Symbol
Engine Speed ME1	(-)	n_{ME1}
Engine Speed ME2	(-)	n_{ME2}
Power Demand ME1	(-)	P_{ME1}
Power Demand ME2	(-)	P_{ME2}

Which correspond to the functions f_i and f_i^* respectively. Furthermore, equations 3.13, 3.14 and 3.15 represent three different modes of validation depending on the initial inputs that the user defines. The following block diagrams show the signal validation process:

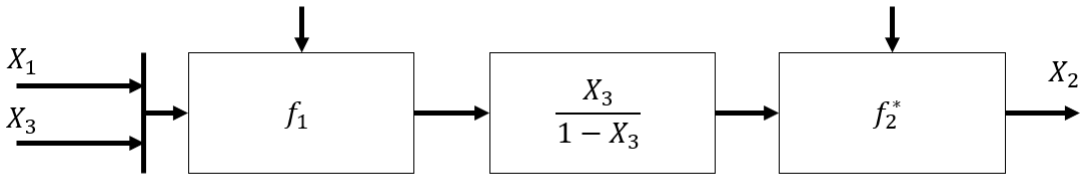


Figure 3.10: Input Validating Mode 1 - In: RPM_2, PR Out: RPM_1

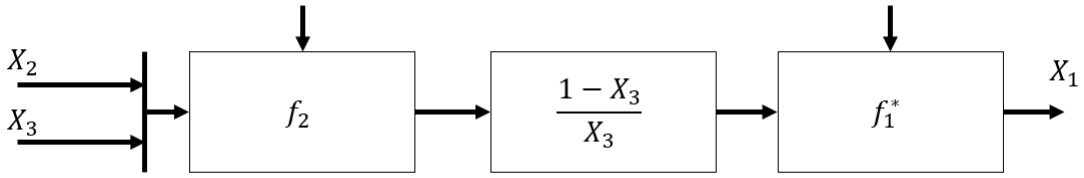


Figure 3.11: Input Validating Mode 2 - In: RPM_1, PR Out: RPM_2

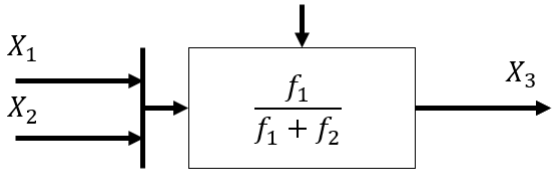


Figure 3.12: Input Validating Mode 3 - In: RPM_1, RPM_2 Out: PR

The features used for the functions f_i and f_i^* were as follow:

Table 3.8: Final Features

Feature	Unit	Symbol
Wind Speed North	(-)	U_{WN}
Wind Speed West	(-)	U_{WW}
Current Speed North	(-)	U_{CN}
Current Speed West	(-)	U_{CW}
Speed Over Ground	(-)	U_G

As well as the following targets/features:

Table 3.9: Target-Features of the Inverse maps

Feature	Unit	Symbol	Note
Engine Speed ME$_i$	(-)	U_{WN}	Target of f_i and a feature of f_i^*
Power Demand ME$_i$	(-)	U_{WW}	Target of f_i^* and a feature of f_i

The advantage of this method is that no inverse function is required and the simulator becomes functional.

3.7.3 Trip Consumption

Adding an integration block after the simulator yields the total consumption in litres for the trip. This integration follows the Forward Rectangular method:

$$X_{TOT} = \int_{t_0}^{t_f} x(t) \cdot dt \approx \sum_{i=0}^{N-1} \Big(x_i \cdot \Delta t \Big) \tag{3.16}$$

With Δt the sampling time and $x(t) = TFC$ the instantaneous total fuel consumption (in l/h). This process can be done before or after scaling, by the following:

$$\begin{aligned} X_{TOT} &= \int_{t_0}^{t_f} x(t) \cdot dt \\ &= \int_{t_0}^{t_f} \Big((\hat{x} - x_{min}) \cdot (x_{max} - x_{min}) + x_{min} \Big) \cdot dt \\ &= (x_{max} - x_{min}) \cdot \int_{t_0}^{t_f} \Big(\hat{x_i} \cdot dt \Big) + x_{min} \cdot (t_f - t_0) \\ &\approx (x_{max} - x_{min}) \cdot \sum_{i=0}^{N-1} \Big(\hat{x_i} \cdot \Delta t \Big) + x_{min} \cdot (t_f - t_0) \end{aligned}$$

The final assembly therefore looks as follows:

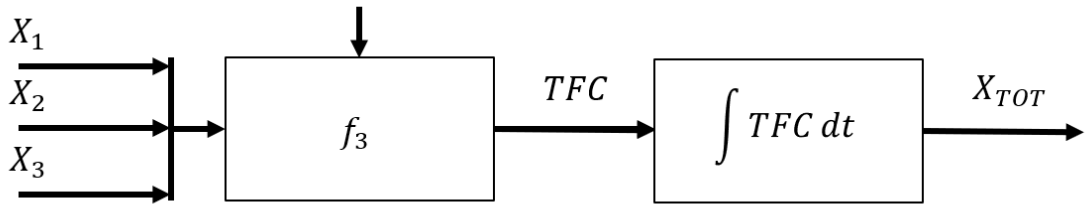


Figure 3.13: Integration block

3.7.4 Additional Blocks

The simulator may be enhanced by the addition of a Engine Load predictor. This predictor follows the definition from equation (3.4) and would be added after the f_i and f_j^* in the initial condition validating block. The general idea looks like this:

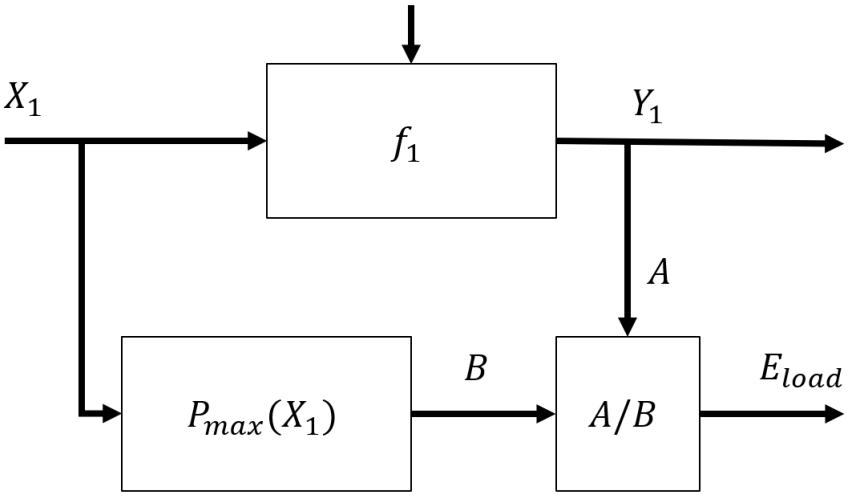


Figure 3.14: Engine Load Block

The P_{max} block is determined by the look-up table from figure 3.5 and defined by equation (3.4).

4
Performance Modelling Results

Previous section showed how the fuel consumption, engine speed and engine load were modelled using different machine learning algorithms. This section evaluate the performance of the different architectures and how do they compare to the actual measurements in the test set.

4.1 ML - Model Selection

The following r^2 scores were computed on the test set for all different ML models for each of the different targets:

Table 4.1: Linear Models

Target/model	OLS-Linear	OLS-Quadratic	OLS-Cubic
TFC	0.937	0.961	0.961
RPM_1	0.925	0.969	0.976
RPM_2	0.934	0.934	0.968
$PowerME_1$	0.920	0.964	0.965
$PowerME_2$	0.928	0.964	0.978

Table 4.2: Black Box Models

Target/model	Random Forest	XGBoost	SVM	ANN
TFC	0.969	0.961	0.940	0.965
RPM_1	0.985	0.987	0.930	0.988
RPM_2	0.991	0.991	0.927	0.988
$PowerME_1$	0.957	0.949	0.921	0.972
$PowerME_2$	0.982	0.978	0.927	0.984

The models with the highest r^2 were selected as the final predictors and then tuned.

4.2 Model Evaluation

The full simulator described by section 3.1 and figure 3.1, is realized as in equations 3.10, 3.11 and 3.12. The inputs are user defined signals for RPM_1, RPM_2 and/or Power Ratio.

4.3 Hypter parameter tuning

Fixing the random-seed for repeatability the following r^2 were determined after this tuning process:

Table 4.3: Black Box Models

Target/model	r^2
TFC	0.966
RPM_1	0.989
RPM_2	0.992
$PowerME_1$	0.950
$PowerME_2$	0.978

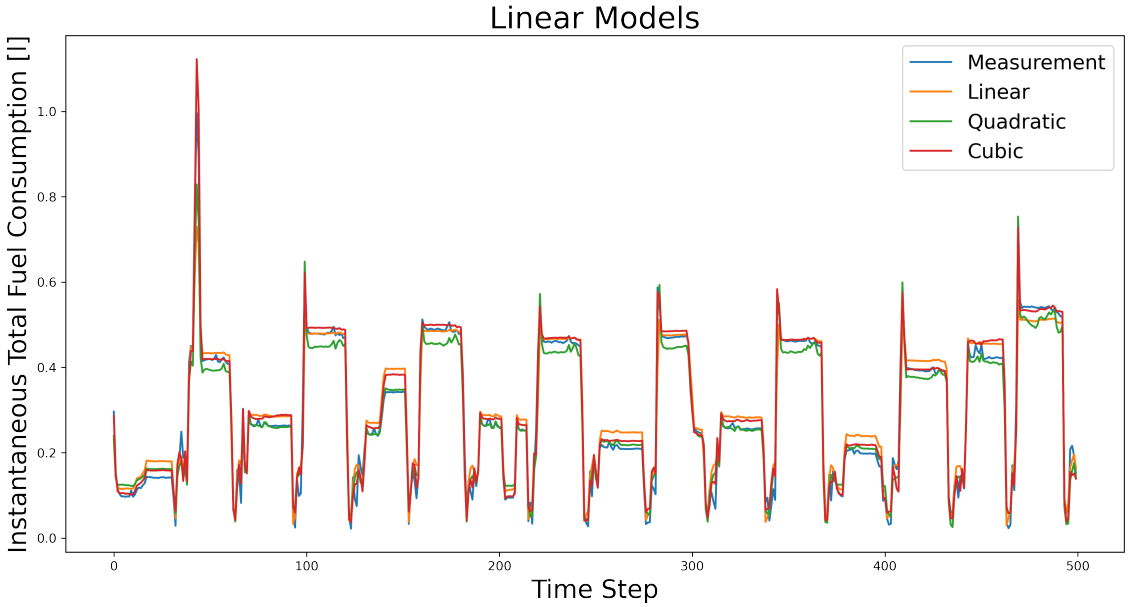
The new hyper parameters focusing on the case of XGBoost follow:

Table 4.4: Hyperparameter tuning XGBoost

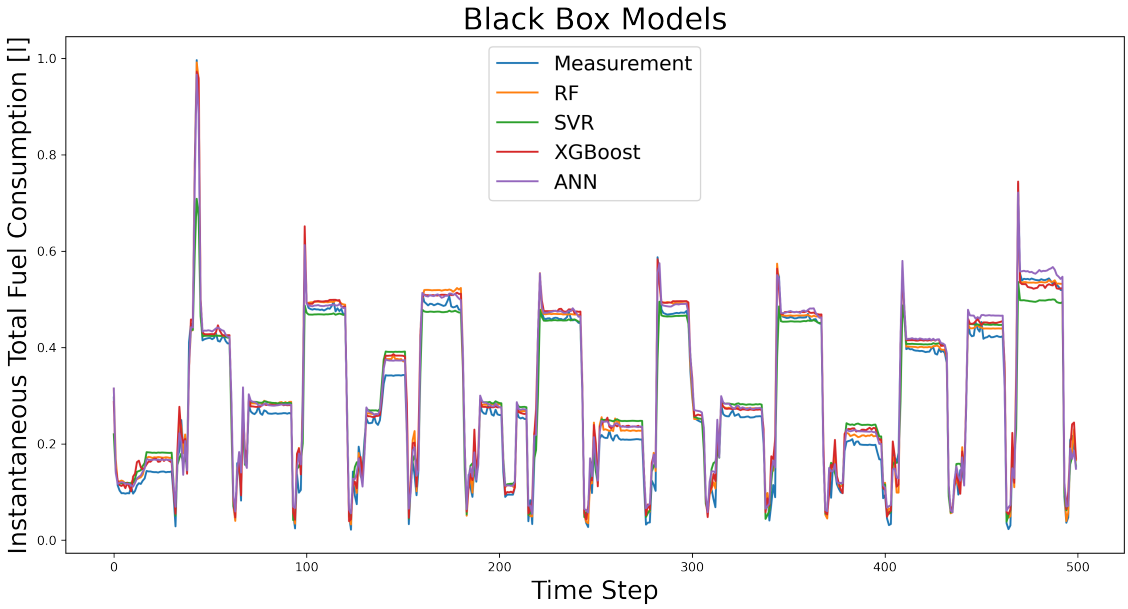
	Hyper-parameters
XGBoost	subsample = 0.7, n_estimators = 50, max_depth = 5, learning_rate = 0.2, colsample_bytree = 0.9, colsample_bylevel = 0.8

4.4 Model Forecasting

For the Total Fuel Consumption forecasting the following plots correspond to 500 time steps (500 minutes after down-sampling). For these conditions it is shown that all the predictive models are able to properly follow the trend described by the measurement in blue.

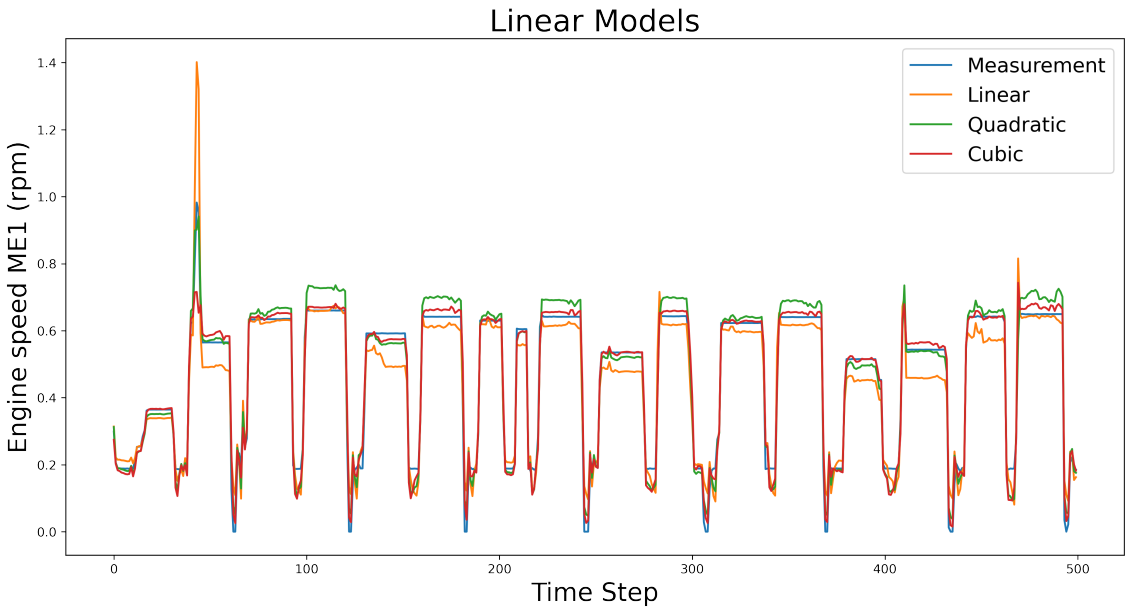


(a) Linear Models

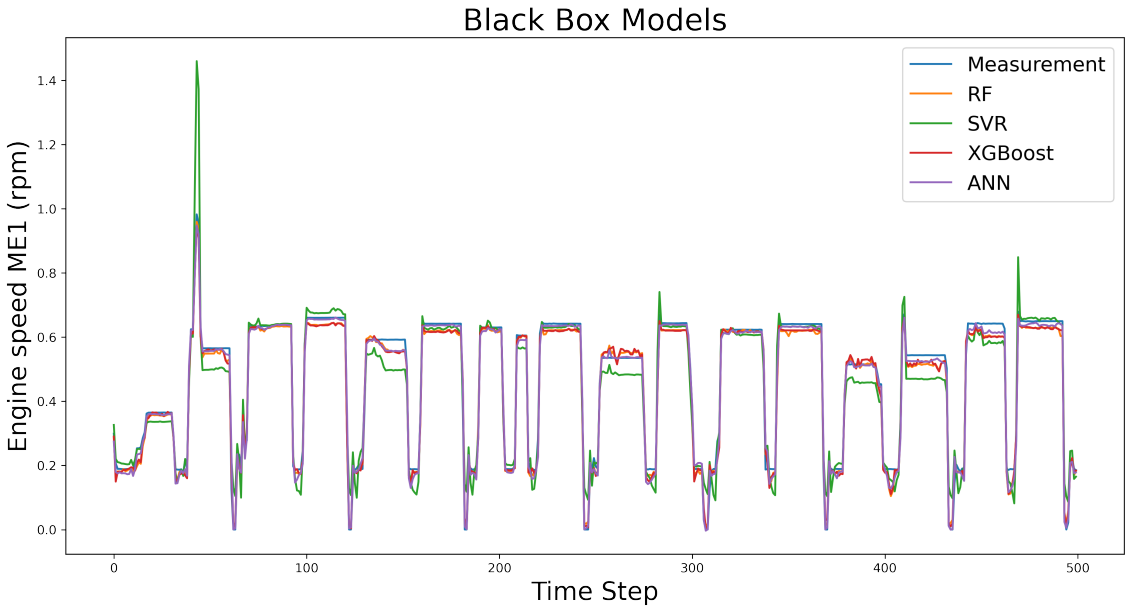


(b) Black Box Models

Figure 4.1: Training Comparison between Linear and Black Box Models for Instantaneous Total Fuel Consumption

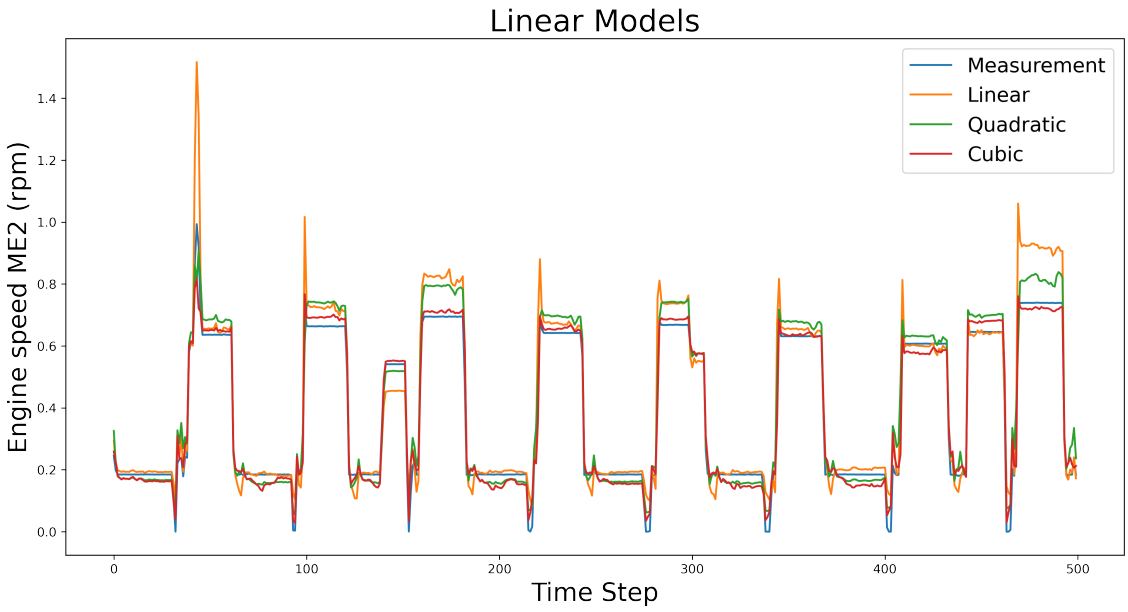


(a) Linear Models

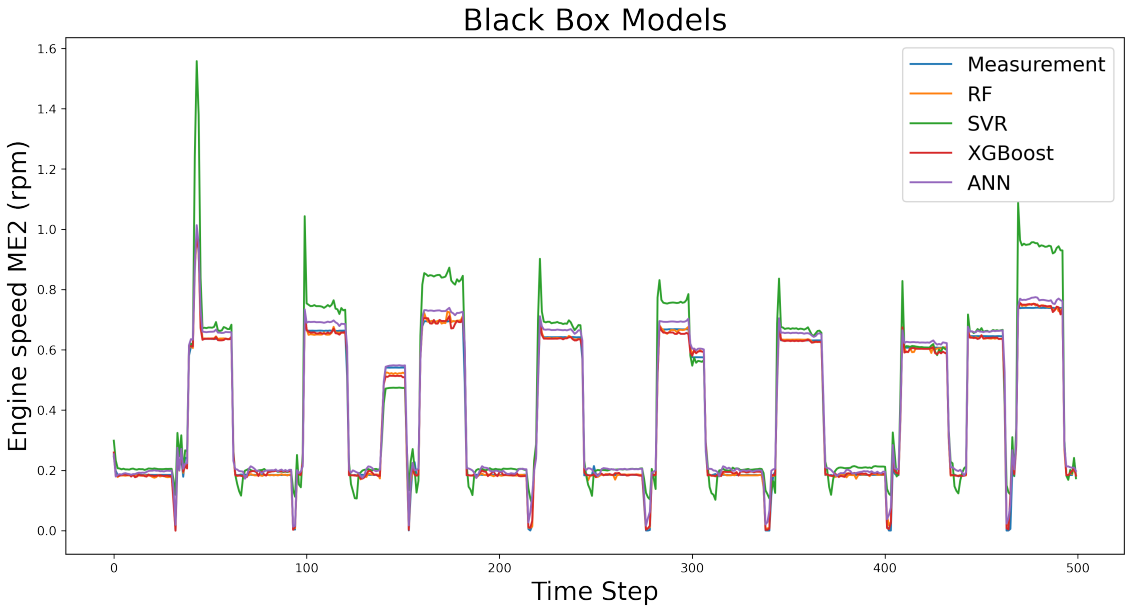


(b) Black Box Models

Figure 4.2: Training Comparison between Linear and Black Box Models for ME1 Speed (rpm)

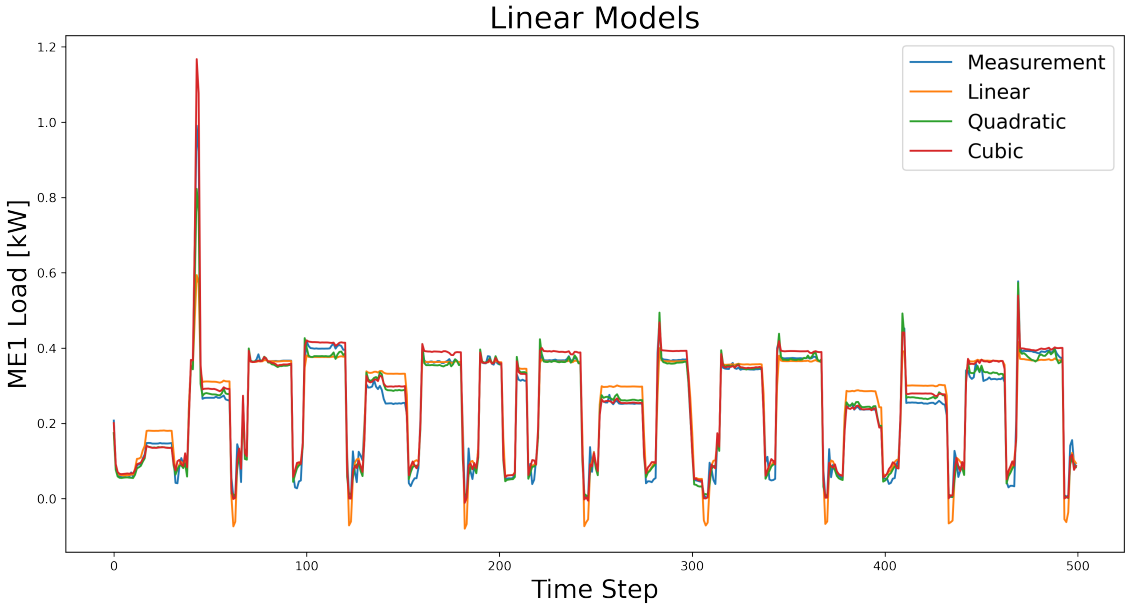


(a) Linear Models

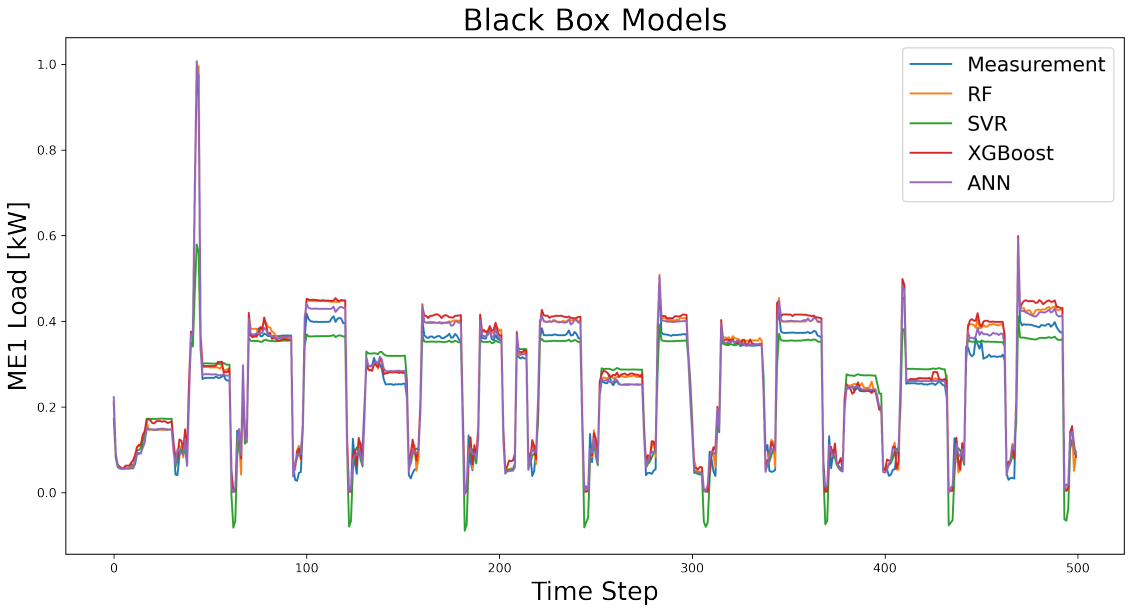


(b) Black Box Models

Figure 4.3: Training Comparison between Linear and Black Box Models for ME2 Speed (rpm)

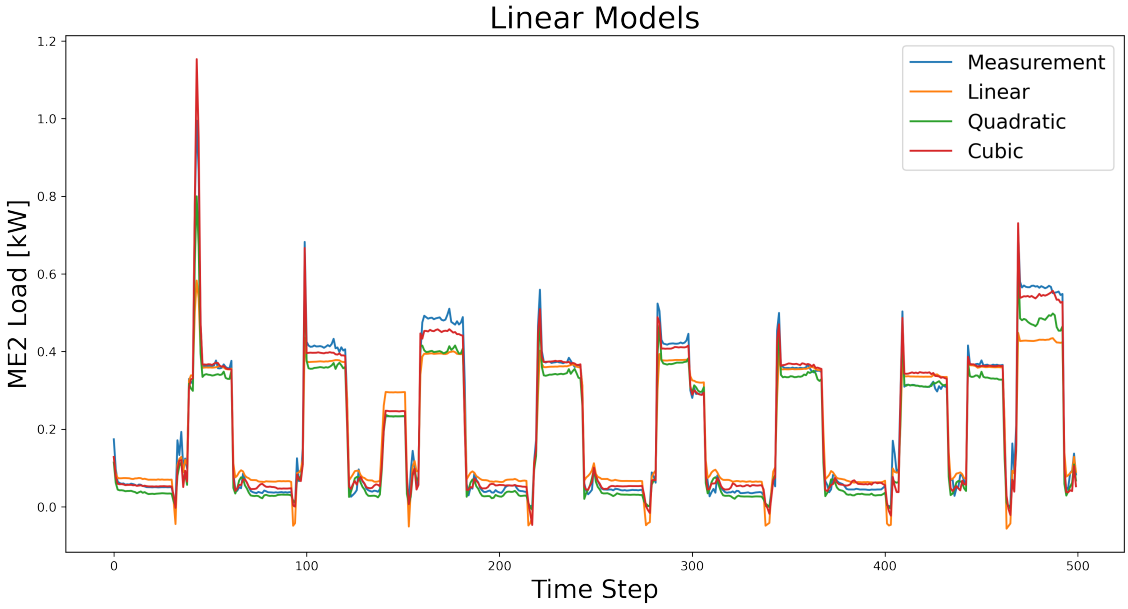


(a) Linear Models

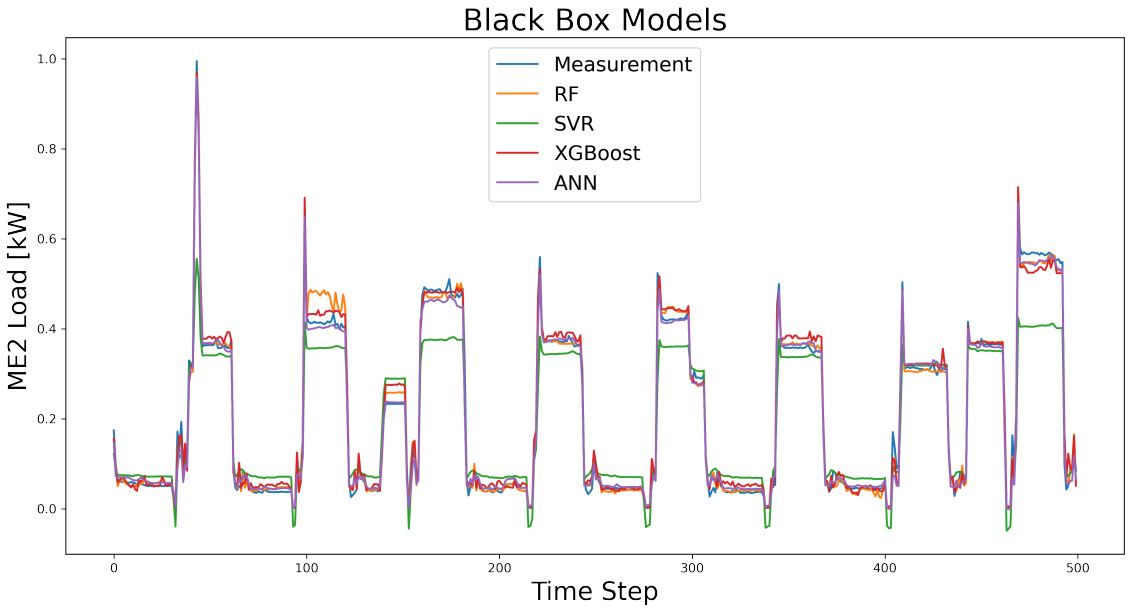


(b) Black Box Models

Figure 4.4: Training Comparison between Linear and Black Box Models for ME1 Power (kW)



(a) Linear Models



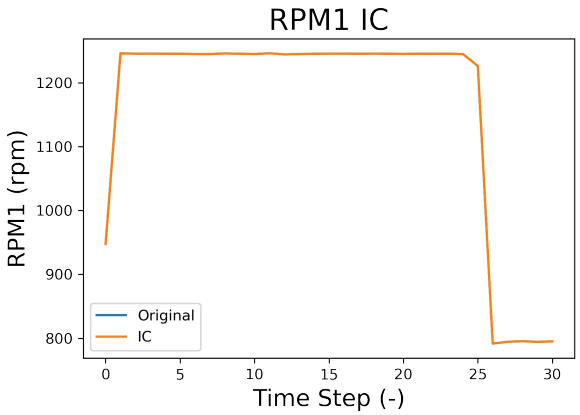
(b) Black Box Models

Figure 4.5: Training Comparison between Linear and Black Box Models for ME2 Power (kW)

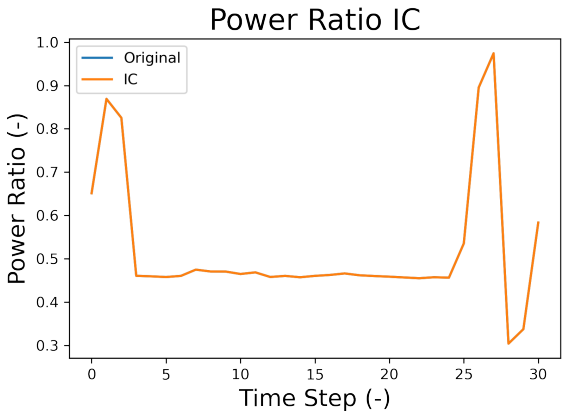
It is apparent from the plots that higher dimensional linear models have better performance alongside Decision Trees as well as Artificial Neural Networks in predicting the different objectives.

4.5 Initial Condition Validation Block

Using as initial conditions (**IC**) PR and RPM_1 the same measurements recorded in a given trip, a third initial condition was computed:



(a) IC RPM1 Validating Block



(b) IC PR Validating Block

Figure 4.6: Initial Conditions input Validation Block.

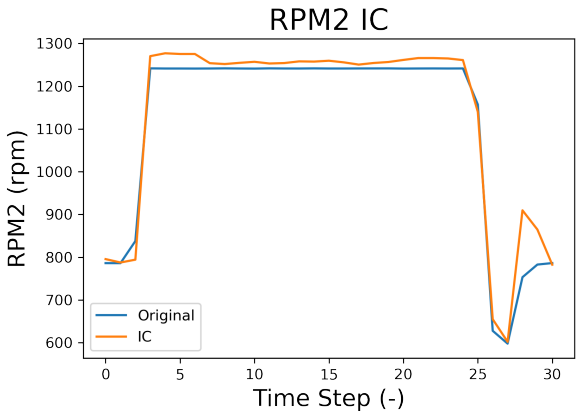
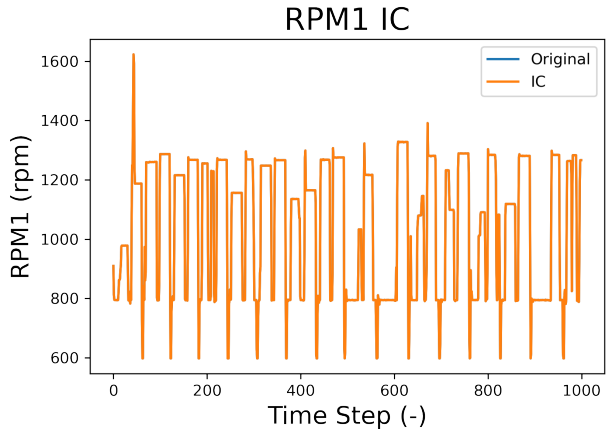


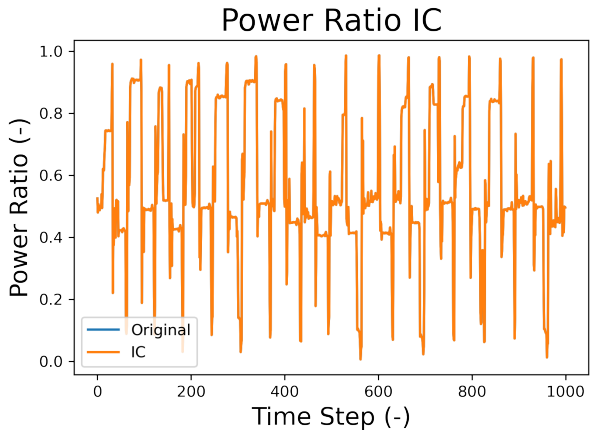
Figure 4.7: Output RPM2 from the Validation Block

4.6 Simulator test

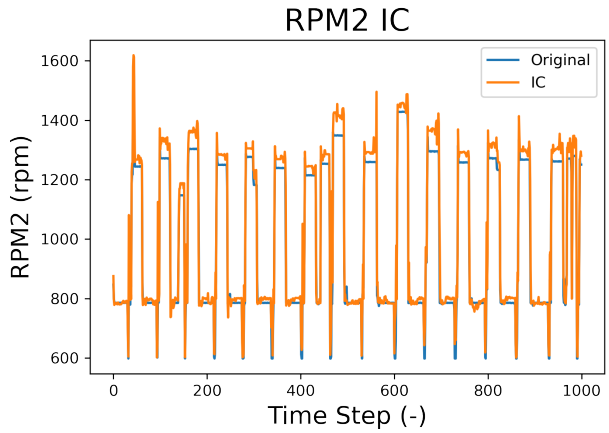
A validation of the whole simulator using 1000 samples from the test set was carried out and the following metrics were computed:



(a) IC RPM1 1000 Samples



(b) IC Power Ratio 1000 Samples



(c) IC Validated RPM2 1000 Samples

Figure 4.8: Initial Conditions for the simulated samples.

The following fuel consumption profile is obtained:

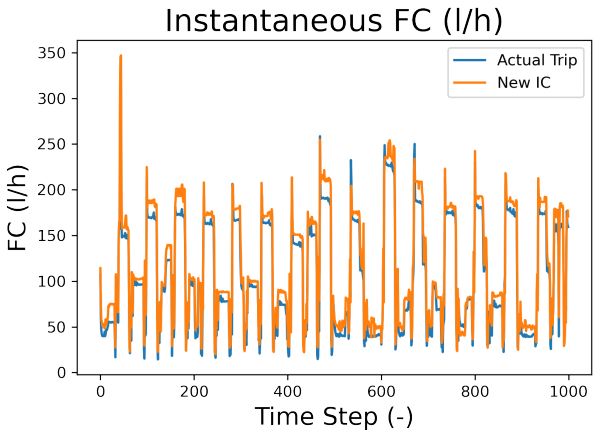


Figure 4.9: Expected total fuel consumption from the simulated operation.

Integration of the total simulated trip as well as the coefficient of determination yields:

Table 4.5: Integration block result for 1000 samples.

	Measurement	**Simulator**
Fuel Consumption [l]	1758.95	1928.57
r^2	-	0.948

5

Decision Making & Trip Simulation Results

The rationale behind the decision making process was explained in chapter 3. Section 3.1 showed the general block diagram of the simulator while section 3.7 showed the inside of each block. In practice the decision support process is as follows:

- Chose two initial conditions from between RPM_1, RPM_2 and PR.
- Input the ICs into the validation block, get a valid third IC.
- The simulator takes all valid ICs and exogenous inputs and predict the total fuel consumption for a trip.

The decision maker is this way able to evaluate the performance of the operation after a given trip and/or prepare the set points for the operation of a future trip. The following sections show tests of the decision support tool applied to two trips.

5.1 Trip simulations

Trip 1
In order to determine the length of a trip its speed profile was plotted. This was a trip in the direction Ven-Landskrona, therefore it was expected that a lower Power Ratio (more power on ME2) would increase the performance.

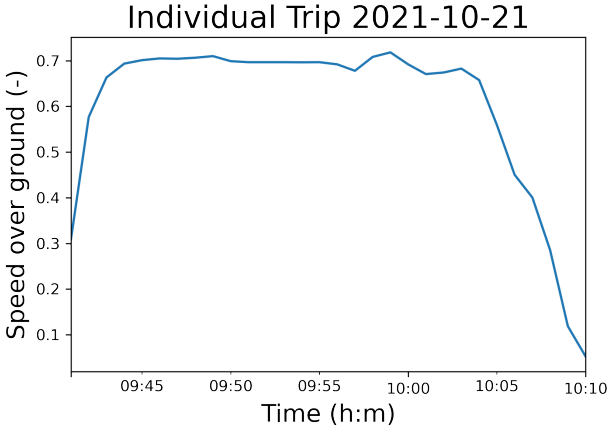
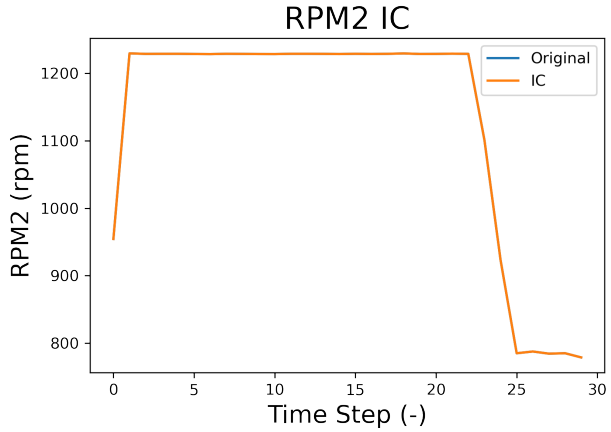
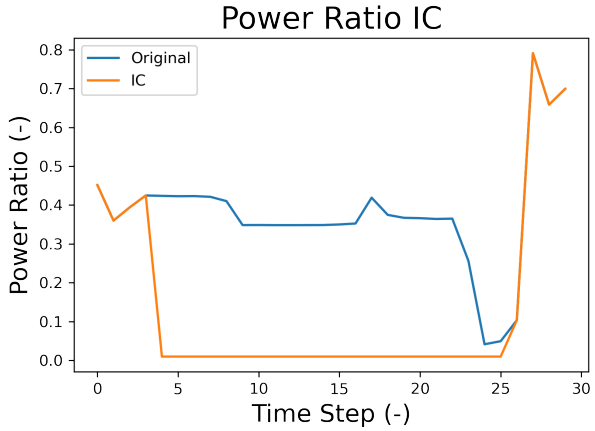


Figure 5.1: Speed profile of the one sample trip

Keeping the same RPM_2 and decreasing, during the voyage, the power ratio to a fixed 1%, then one can input the following ICs into the validating block:



(a) IC RPM2 Trip 1



(b) IC Power Ratio Trip 1

Figure 5.2: Initial Conditions for the simulated trip.

The third initial condition is then given by equation (3.14) and by using the block from figure 3.11:

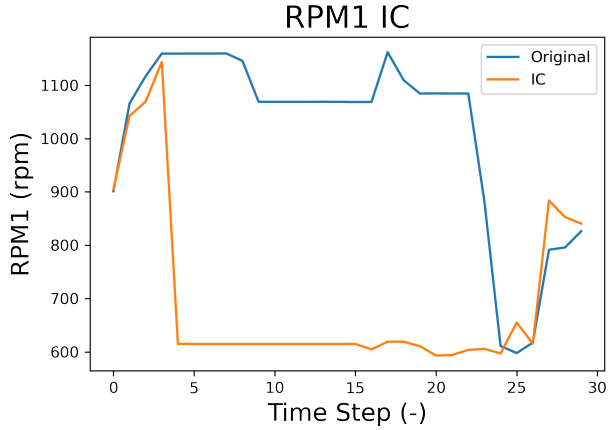


Figure 5.3: Output RPM1 from the validating block

And the following fuel consumption profile is obtained:

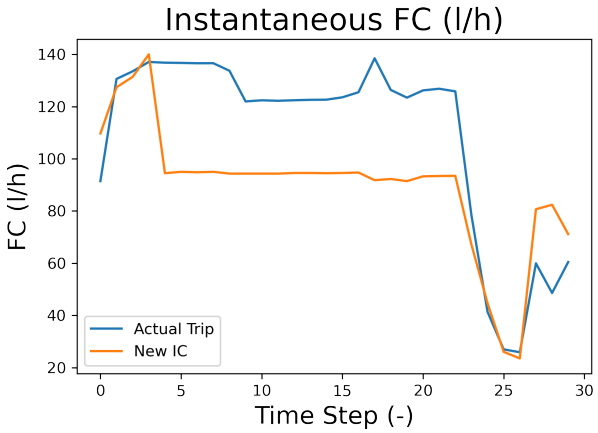


Figure 5.4: Expected total fuel consumption from the simulated operation.

Table 5.1: Comparison between the proposed operation and the measurements

Measured Fuel Consumption (l)	54.41
Proposed Operation (l)	44.81
Fuel Savings (l)	9.60
Fuel Savings (%)	18.67

Trip 2
Similarly a second trip in the direction Landskrona-Ven was analysed, for this operation a high power ratio was expected to have a better performance.

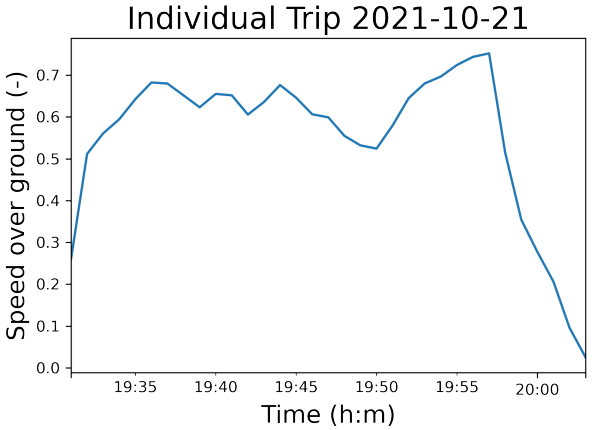
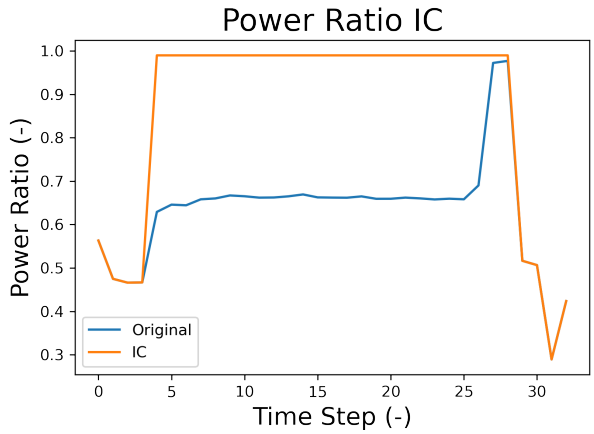
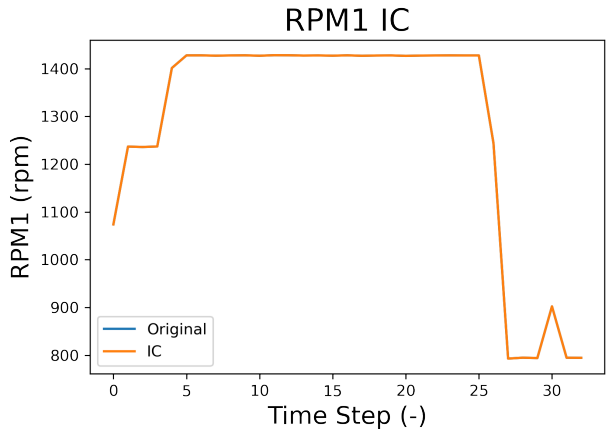


Figure 5.5: Speed over ground profile - Second Trip

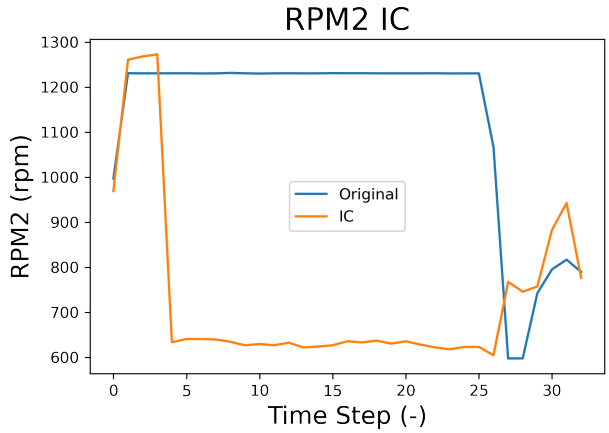
The ICs for this simulation are as follow:



(a) IC Power Ratio Trip 2



(b) IC RPM2 Trip 1



(c) IC Validated RPM2 Trip 2

Figure 5.6: Initial Conditions for the simulated trip.

The following fuel consumption profile is obtained:

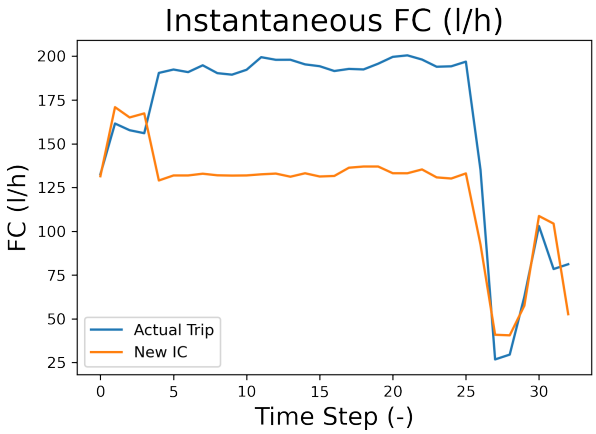


Figure 5.7: Expected total fuel consumption from the simulated operation.

Table 5.2: Comparison between the proposed operation and the measurements

Measured Fuel Consumption (l)	90.09
Proposed Operation (l)	67.53
Fuel Savings (l)	22.56
Fuel Savings (%)	25.04

Showing the possible energy savings from this proposed operation.

6

Conclusion & Future Work

This chapter comprehends the discussion of the different findings with the advantages/disadvantages of each of the approaches and which conclusions can be drawn from those. Furthermore, future work on how to improve the research is also suggested.

6.1 Conclusion

A full data mining of the provided measurements was successfully carried out, determining some trends in fuel consumption related to engine operational parameters.

Validation of the hypothesis regarding to engine power allocation on the ship Uraniborg was evaluated using exploratory data analysis. It was shown that using the stern-most engine for most of the operation would decrease the fuel consumption onboard between 0 to 30 litres.

Using ship-propulsion theory, different features were extracted from within the measurements that allowed to compute the black box modelling of input-output maps of the engine and fuel consumption for the ship.

Different Machine Learning approaches were considered for the forecasting of Power Demand, Engine Speed, and Fuel Consumption. From these models an argument was established to choose XGBoost as the final model for the objectives and then these models were built into a Decision Support System comprised by a simulator as seen in figures 3.1 and 3.13. The different XGBoost models are able to forecast on the Test-set the different objectives with metric $r^2 \approx 0.95$ and are able to compute a possible fuel consumption prediction of the engine for a user defined trip that would allow for Engine Speed optimization and to make a decision on Power Allocation.

The simulator was tested for two sample cases and it was shown that the hypothesis of the power allocation may hold true, that is, that higher power allocation on the stern engine reduces fuel consumption. However, external validation data is expected to be provided by Ventrafiken in order to consider this statement true.

It seems from the many tests in the simulator that the operation of the ship in the direction Landskrona to Ven is already excellent and that most of the fuel savings that the ship can have are from improvements in the operation in the direction Ven

to Landskrona. Furthermore, based on Lang (2020) the models would also benefit from the inclusion of the draught as one of the features, this is in line with section 2.1 in which the ship resistance components have influence from the displacement, which is a function of the draught. So far, no information related to the displacement is used on the models nor is being measured by the company. Furthermore the information of the sea-currents was computed using the database from Copernicus on the same period (january 2021 to january 2022), the models would also benefit from more accurate measurement of the sea-currents with perhaps an on-site floater (a buoy).

This way information of the actual Engine Load, that the captain has information of in the command bridge would be accessible for the decision maker and would perhaps give extra insight about the possible operation of the ship.

6.2 Future Work

As more data is collected by the ship owner a validation of the findings can be performed. Furthermore, it would also be beneficial to collect information regarding to the actual draught of the ship, and the actual sea-currents. The inclusion of these features may then increase the accuracy of the different models.

Based on the power-allocation hypothesis, a proposal was made to Ventrafiken to make some test on the actual ship that would provide some data for further analysis. Once these tests have been carried out more data mining would allow to validate these results, and given that enough data is provided, increase the accuracy of the simulator for extrapolations.

Bibliography

[1] Carlton, J. (2007). Marine propellers and propulsion (2nd ed., pp. 310-316). Oxford: Butterworth-Heinemann.

[2] Abebe, M., Shin, Y., Noh, Y., Lee, S., & Lee, I. (2020). Machine Learning Approaches for Ship Speed Prediction towards Energy Efficient Shipping. Applied Sciences, 10(7), 2325. doi: 10.3390/app10072325

[3] Bassam, A., Phillips, A., Turnock, S., & Wilson, P. (2022). Ship speed prediction based on machine learning for efficient shipping operation. Ocean Engineering, 245, 110449. doi: 10.1016/j.oceaneng.2021.110449

[4] Lu, R., Turan, O., Boulougouris, E., Banks, C., & Incecik, A. (2015). A semi-empirical ship operational performance prediction model for voyage optimization towards energy efficient shipping. Ocean Engineering, 110, 18-28. doi: 10.1016/j.oceaneng.2015.07.042

[5] Mao, W., Rychlik, I., Wallin, J., & Storhaug, G. (2016). Statistical models for the speed prediction of a container ship. Ocean Engineering, 126, 152-162. doi: 10.1016/j.oceaneng.2016.08.033

[6] International Organization for Standardization. (2016). Ships and marine technology —Measurement of changes in hull and propeller performance — Part 2: Default method (ISO/DIS Standard No. ISO 19030-2:2016(E)).

[7] Catepillar (2009), C32 ACERT™ MARINE PROPULSION - 1015 mhp (1000 bhp) 746 bkW.

[8] Pedregosa et al. (2011),Scikit-learn: Machine Learning in Python, JMLR 12, pp. 2825-2830.

[9] Kaufman, S., Rosset, S., Perlich, C., & Stitelman, O. (2012). Leakage in data mining. ACM Transactions On Knowledge Discovery From Data, 6(4), 1-21. https://doi.org/10.1145/2382577.2382579

[10] Lang, X., Wu, D., & Mao, W. (2022). Comparison of supervised machine learning methods to predict ship propulsion power at sea. Ocean Engineering, 245, 110387. https://doi.org/10.1016/j.oceaneng.2021.110387

[11] Yang, L., & Shami, A. (2020). On hyperparameter optimization of machine learning algorithms: Theory and practice. Neurocomputing, 415, 295-316. https://doi.org/10.1016/j.neucom.2020.07.061

[12] Gottinger, H., & Weimann, P. (1992). Intelligent decision support systems. Decision Support Systems, 8(4), 317-332. https://doi.org/10.1016/0167-9236(92)90053-r

[13] Smola, A., & Schölkopf, B. (2004). A tutorial on support vector regression. Statistics And Computing, 14(3), 199-222. https://doi.org/10.1023/b:stco.0000035301.49549.88

[14] Breiman, L. (2001). Machine Learning, 45(1), 5-32. https://doi.org/10.1023/a:1010933404324
[15] 1. Supervised learning. scikit-learn. (2022). Retrieved 26 May 2022, from https://scikit-learn.org/stable/supervised_learning.html#supervised-learning.
[16] 1.1. Linear Models. scikit-learn. (2022). Retrieved 26 May 2022, from https://scikit-learn.org/stable/modules/linear_model.html.
[17] 1.11. Ensemble methods. scikit-learn. (2022). Retrieved 26 May 2022, from https://scikit-learn.org/stable/modules/ensemble.html.
[18] 1.17. Neural network models (supervised). scikit-learn. (2022). Retrieved 26 May 2022, from https://scikit-learn.org/stable/modules/neural_networks_supervised.html.
[19] 1.4. Support Vector Machines. scikit-learn. (2022). Retrieved 26 May 2022, from https://scikit-learn.org/stable/modules/svm.html.
[20] Brownlee, J. (2022). Data Leakage in Machine Learning. Machine Learning Mastery. Retrieved 26 May 2022, from https://machinelearningmastery.com/data-leakage-machine-learning/.
[21] 6.3. Preprocessing data. scikit-learn. (2022). Retrieved 26 May 2022, from https://scikit-learn.org/stable/modules/preprocessing.html.
[22] Bassam, A., Phillips, A., Turnock, S., & Wilson, P. (2022). Ship speed prediction based on machine learning for efficient shipping operation. Ocean Engineering, 245, 110449. https://doi.org/10.1016/j.oceaneng.2021.110449
[23] Lu, R., Turan, O., Boulougouris, E., Banks, C., & Incecik, A. (2015). A semi-empirical ship operational performance prediction model for voyage optimization towards energy efficient shipping. Ocean Engineering, 110, 18-28. https://doi.org/10.1016/j.oceaneng.2015.07.042
[24] Birk, L. (2019). Fundamentals of ship hydrodynamics (1st ed.). Wiley.
[25] Mukhiya, S., & Ahmed, U. (2020). Hands-On Exploratory Data Analysis with Python. Packt Publishing.
[26] Kim, Y., Jung, M., & Park, J. (2021). Development of a Fuel Consumption Prediction Model Based on Machine Learning Using Ship In-Service Data. Journal Of Marine Science And Engineering, 9(2), 137. https://doi.org/10.3390/jmse9020137
[27] Yan, R., Wang, S., & Du, Y. (2020). Development of a two-stage ship fuel consumption prediction and reduction model for a dry bulk ship. Transportation Research Part E: Logistics And Transportation Review, 138, 101930. https://doi.org/10.1016/j.tre.2020.101930
[28] Garcia, G., Michau, G., Einstein, H., & Fink, O. (2021). Decision support system for an intelligent operator of utility tunnel boring machines. Automation In Construction, 131, 103880. https://doi.org/10.1016/j.autcon.2021.103880
[29] Banik, S., Sharma, N., Mangla, M., Mohanty, S., & S., S. (2022). LSTM based decision support system for swing trading in stock market. Knowledge-Based Systems, 239, 107994. https://doi.org/10.1016/j.knosys.2021.107994
[30] Gkerekos, C., Lazakis, I., & Theotokatos, G. (2019). Machine learning models for predicting ship main engine Fuel Oil Consumption: A comparative study. Ocean Engineering, 188, 106282. https://doi.org/10.1016/j.oceaneng.2019.106282

[31] Yan, R., Wang, S., & Psaraftis, H. (2021). Data analytics for fuel consumption management in maritime transportation: Status and perspectives. Transportation Research Part E: Logistics And Transportation Review, 155, 102489. https://doi.org/10.1016/j.tre.2021.102489

[32] Berg, T., Selvik, Ø., Steinsvik, K., & Leinebø, D. (2021). Manoeuvring Study – Norwegian Double-Ended Ferry. Transnav, The International Journal On Marine Navigation And Safety Of Sea Transportation, 15(1), 63-69. https://doi.org/10.12716/1001.15.01.05

[33] Laurie, A., Anderlini, E., Dietz, J., & Thomas, G. (2021). Machine learning for shaft power prediction and analysis of fouling related performance deterioration. Ocean Engineering, 234, 108886. doi: 10.1016/j.oceaneng.2021.108886

[34] Waterhouse J. (2016). Fifty plus years of double-ended ferry design. Marine Log. Elliott Bay Design Group.

[35] Corrales, D., Corrales, J., Ledezma, A. (2018). How to Address the Data Quality Issues in Regression Models: A Guided Process for Data Cleaning. Symmetry, 10(4), 99. https://doi.org/10.3390/sym10040099

[36] Farnsworth, A. (2020). Retrieved 20 June 2022, from https://www.wartsila.com/insights/article/long-live-the-internal-combustion-engine

DEPARTMENT MARITIME TECHNOLOGY
CHALMERS UNIVERSITY OF TECHNOLOGY
Gothenburg, Sweden
www.chalmers.se

CHALMERS
UNIVERSITY OF TECHNOLOGY