MSc thesis in Mechanical Engineering

Predicting Propeller Behaviour Using Machine Learning

Lourens Zuiker
4387015

TU Delft

Boskalis

Additonal info

Predicting Propeller Behaviour Using Machine Learning

Lourens Zuiker
4387015

December 2023

A thesis submitted to the Delft University of Technology in partial fulfillment of the requirements for the degree of Master of Science in Mechanical Engineering

Lourens Zuiker
4387015: *Predicting Propeller Behaviour Using Machine Learning* (2023)

The work in this thesis was carried out on behalf of:

Cognitive Robotics
Delft University of Technology

Royal Boskalis

Supervisors: Dr. Jens Kober
Jop Paauw
Willem de Graaf

Abstract

This thesis focuses on developing a model that effectively captures and generalizes the four quadrant behaviour of propellers, which is crucial for understanding and optimizing propulsion systems in marine vessels. Accurate prediction of four quadrant behaviour offers significant benefits to the industry, including reducing fuel consumption, mapping hull growth accurately, and identifying efficiency losses due to propeller cavitation. Unlike existing approaches that mainly concentrate on modelling the physical interactions between the water and the propeller blade, this thesis investigates different machine learning methods and finds that the ensemble method yields the best model for predicting four-quadrant behaviour. The method is tested on different the different data sets available. The data set whose propeller type is the same as the propellers used in the fleet is used to generate predictions for propellers currently in use in the fleet.

Contents

List of Figures

List of Tables

Acronyms

1 Introduction

1.1 Motivation

The performance of propellers plays a crucial role in the efficient operation of marine vessels. Knowing the formulas that describe the performance of the propeller is very valuable. The behaviour of a propeller in all operating conditions is described in the four-quadrant behaviour. The four-quadrant behaviour describes the connection between the engine's torque and the resulting thrust generated by the propeller is a critical aspect of understanding and optimizing propulsion systems. Tests to measure this behaviour are expensive to carry out and no data is available on the propellers currently in use by Boskalis. Therefore in this thesis, the aim is to provide a machine learning model that effectively captures and generalizes the four-quadrant behaviour of propellers based on the propeller characteristics.

The ability to accurately predict the behaviour of propellers in all four quadrants can aid the industry in several ways. Such knowledge can lead to various improvements, including the reduction of fuel consumption, more precise mapping of hull growth, and the identification of efficiency losses due to propeller cavitation. By addressing these challenges, a contribution can be made to the enhancement of vessel performance, operational cost reduction, and environmental sustainability.

The current state of the art for mapping four-quadrant behaviour is mostly focused on creating models that calculate the physical processes of water interacting with the propeller blade for small autonomous submarines. An addition can be made to the field by providing a new data-driven approach to modelling propeller behaviour that generalizes between different propeller types by employing machine learning techniques.

1.2 Objectives

The overall goal of this thesis can be divided into the following sub-objectives.

- Provide feedback on propeller theory to understand the physical meaning of the tested propellers, the propeller characteristics used for training and the values predicted by the model.
- Choosing an appropriate algorithm to predict the four-quadrant behaviour
- Train and validate the chosen algorithm to create an optimal model for predicting four-quadrant behaviour

1.3 Outline

The thesis is outlined in the following chapters: Chapter 2 will provide the necessary background on propeller theory to understand the different types of propellers used in this research. Furthermore, the physical meaning of the propeller characteristics used for training and the value the model predicts are explained. Lastly, state-of-the-art models predicting four-quadrant behaviour are discussed. Chapter 3 provides an overview of the different tests conducted that make up the data available for training. The problem characteristics that arise from the available data and the type of problem are summarized and listed. These characteristics aid as requirements in the choice of algorithm. Chapter 4 analyzes different algorithms to determine what algorithms meet the requirements set in the previous chapter. An ultimate selection is made for a single algorithm to continue this research. Chapter 5 will offer an overview of the applied methodology, while Chapter 6 discusses the results from the performed tests. A conclusion to the work will be provided in Chapter 7. Lastly, the discussion and recommendation of the work will be provided in Chapter 8.

2 Propeller Theory

This chapter will provide the necessary background on propeller theory to understand the context in which the work of this thesis is carried out. Firstly, Section 2.1 discusses the two different propeller types investigated in this research and how they differ. Furthermore, it explains the physical meaning of the propeller characteristics that are used to train the model. Secondly, in Section 2.2 the physical meaning of the four quadrants is discussed. Lastly, in Section 2.3 the state of the art of four-quadrant analysis is discussed. By comprehensively addressing the propeller characteristics, types, and the four-quadrant analysis, this chapter provides the necessary groundwork for the subsequent chapters.

2.1 Propellers

The goal of a propeller is to translate power generated by the engine into forward thrust. Figure 2.1 shows an overview of a propeller with the corresponding names of all parts. Over the years many different versions of propellers have been developed for propelling ships, however, in this research, only the fixed pitch and controllable pitch propellers will be considered [44, 52].

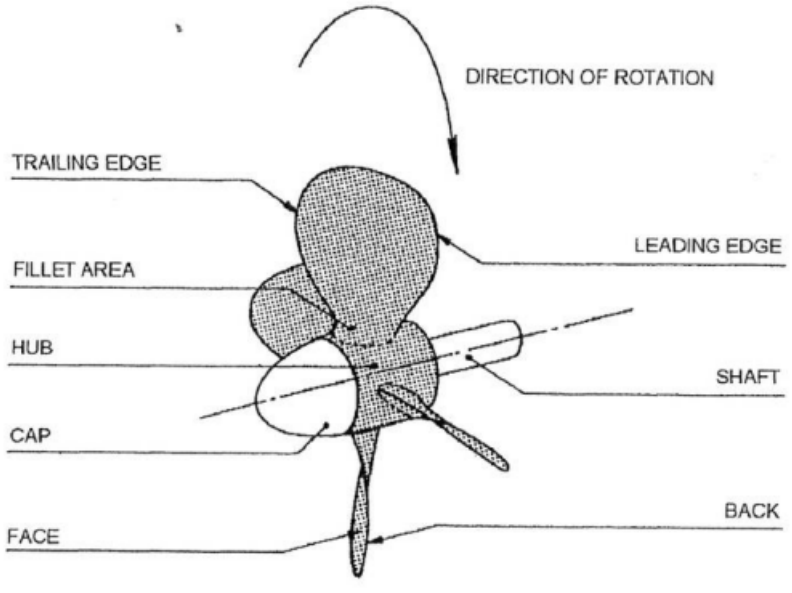


Figure 2.1: Overview of a propeller [34]

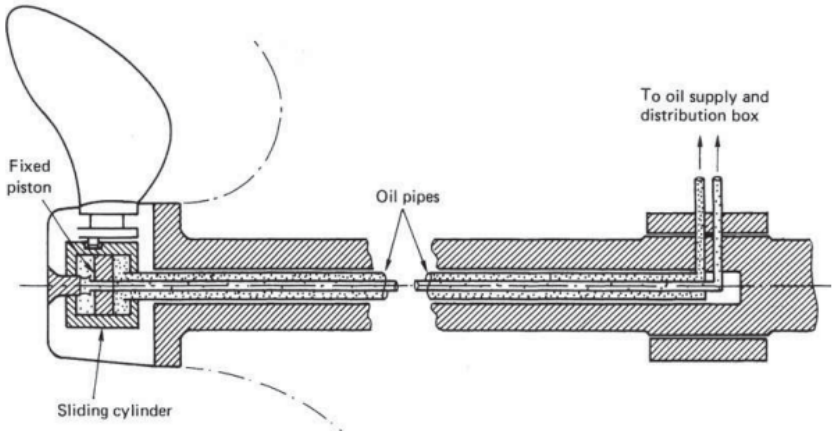


Figure 2.2: The inside mechanics of a CPP [13].

2.1.1 Fixed pitch and controllable pitch propellers

The propeller in Figure 2.1 is an example of a Fixed Pitch Propeller (FPP) and its blades are rigidly attached to the hub. Ships using such a system control their sailing speed by adapting the rotation speed of the engine. This causes the ship to have a need for a special gearbox or an engine capable of rotating two ways for fast stopping or reversing [13, 88]. Fixed pitch propellers have been the standard for centuries but since the 1960's an alternative approach to building propellers was developed, the Controllable Pitch Propeller (CPP) [88]. CPP have the ability to rotate their propeller blades thus changing the pitch angle of the propeller. This allows the engine to operate at a single rotation speed through a large range of sailing speeds. These propellers have an internal hydraulic oil system that is used to rotate the blades. Figure 2.2 shows a possible setup of such a system. CPP are more complex and more expensive. However, it enables very low ship speeds and smooth reversal of the ship [11, 13, 88].

Currently, Boskalis uses both types of propellers on their vessels and therefore this research will focus on both FPP and CPP since a model for each of these propeller types can be useful for the company.

2.1.2 Propeller characteristics

There are several different key characteristics that define different aspects of propeller behaviour when in use. These characteristics will be used as input to the developed model to predict the behaviour of a propeller. The physical meaning and effects of these characteristics are discussed below. The four relevant characteristics for this research are:

The diameter is defined by the diameter of the circle that is traced by the tips of the propeller blades. The generated torque Q and the resultant thrust force T are related through the diameter D and the law of momentum where $Q = T * D$. The depth at which the propeller has to be submerged to obtain the best results, named the draught, is directly related to diameter. The minimum draught depth is equal to the diameter [23].

Propeller Characteristics	
Diameter	D
Pitch diameter ratio	P/D
Number of blades	Z
Expanded Area Ratio (EAR)	EAR

Table 2.1: Propeller characteristics

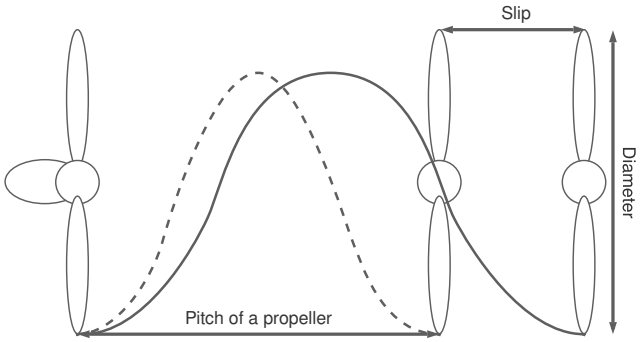


Figure 2.3: Pitch ratio overview [4].

The pitch ratio or pitch diameter ratio is given by the ratio between the pitch and the diameter of the propeller. The pitch is the distance traveled forward during one revolution of the propeller if there is no slip corresponding to the theoretical movement speed V given by the pitch p times the revolution speed n as $V = p \cdot n$. However, since the propeller is moving through a liquid the water accelerates behind the propeller decreasing the apparent forward speed. This effect is called slip and the real traveled distance during one revolution is the geometric pitch of a propeller [23]. Figure 2.3 gives an overview of the physical meaning of these concepts.

One of the determining factors in the choice of number of blades denoted with Z is the need to avoid harmful resonant frequencies of the ship structure and the machinery [15]. Increasing the number of blades helps decrease cavitation at the blade surface. However, it negatively affects the efficiency and optimum propeller diameter. When designing a Controllable Pitch Propeller the number of blades is limited to ensure the blades have enough space to turn their pitch over their full turning range without interfering with one another [14, 15].

The expanded blade area ratio consists of the ratio between two areas determined by the shape of the propeller and its blades. The disc area A_0 is a circle with a diameter equal to the propeller diameter D as given by $A_0 = \pi D^2/4$. The expanded area is the area that is the sum of the area of each propeller blade if they were to be flattened in a single plane. The ratio between the expanded area and the disc area gives the expanded blade area ratio. In general, the expanded blade area is larger than required for operation to decrease cavitation and as a built-in buffer for decrease in expanded area due to decrease of expanded area overtime through material erosion. For Controllable Pitch Propellers the maximum expanded area is

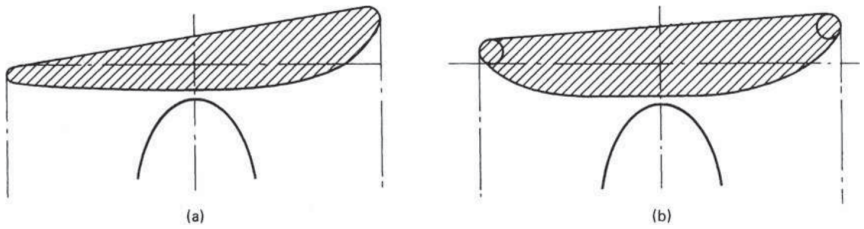
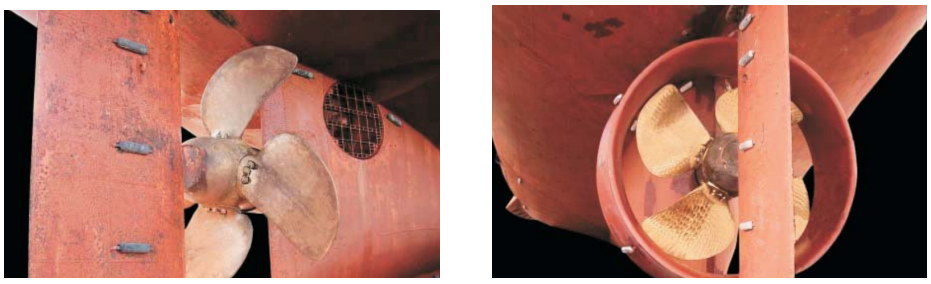


Figure 2.4: Cross section of ducts of type (a) 19a and (b) 37 [13]



(a) (b)

Figure 2.5: Two images side-by-side showing a propeller retrofitted with a duct. (a) Original propeller. (b) Previous propeller fitted with a duct [89].

limited to ensure the blades are able to make full revolution just as for the number of blades [13, 15, 36].

2.1.3 Ducted propellers

The propeller, either a fixed pitch or a controllable pitch propeller, may be placed in a duct. A duct is a ring surrounding the propeller with a cross-section that has a wing-like profile [88]. Figure 2.5a and 2.5b show a propeller that has been retrofitted with a duct by Wärtsilä as a cost-saving measure [89]. Protection of the propeller is a significant benefit of a duct, but more importantly, it improves the characteristics of the propeller such as the ability to generate more thrust at lower speeds [9, 13, 39]. The additional friction between the flow and the duct, however, causes a slightly lower overall efficiency compared to an open propeller [88].

Non Dimensional Coefficients	
The advance ratio	J
The thrust coefficient	K_T
The torque coefficient	K_Q
The nozzle thrust coefficient	K_{TN}

Table 2.2: Non-dimensional coefficients used to describe propeller behaviour

Due to the increased thrust in low-speed operation, the usage of ducted propellers is the current industry standard for dredging ships as dredging operations are executed at low speeds and require sufficient thrust to be generated. The data used in this research for CPP is obtained from a ducted propeller.

2.2 Propeller Analysis

In the previous section information about the different types of propellers and some relevant propeller characteristics were explained. In this section, the analysis of propeller behaviour over different operating conditions will be explored. The behaviour is captured in different non-dimensional coefficients. These coefficients are the quantities we want to predict based on the propeller characteristics with our machine learning method.

To be able to describe the behaviour of propellers it is important to first define the forces and flow speeds clearly. Figure 2.6 gives an overview of all the relevant forces on and flow speeds around a propeller during operation. The resultant of the advance speed V_A and the circumferential speed resulting from the rotation of the propeller n_p and the radius of the propeller r by $2\pi \cdot r \cdot n_p$ is the relative water velocity V_R. This resultant velocity meets the blade at a certain angle which is the angle of attack α. The angle between the resultant water velocity and the circumferential velocity is the hydrodynamic pitch angle β. The sum of both the angle of attack and the hydrodynamic pitch angle is the pitch angle of the blade θ. The generated thrust T and torque Q are also shown and are both parallel to the advance speed and the circumferential speed respectively. The resultant of both these forces is shown in the resultant R. The component of R that is perpendicular to the blade surface is called the lift force and is denoted by L. The component of R parallel to the blade surface is the drag force and is denoted by D [15, 34, 88].

2.2.1 Non Dimensional Coefficients

Much analysis of propellers is done using several non-dimensional coefficients that are obtained using the forces and flow speeds defined in Figure 2.6. Using these coefficients listed in Table 2.2, the behaviour of the propeller can be visualized and analysed [13, 15, 88]:

The advance ratio is defined by Eq. (2.1) where the advance velocity is made non-dimensional by dividing it with the circumferential speed.

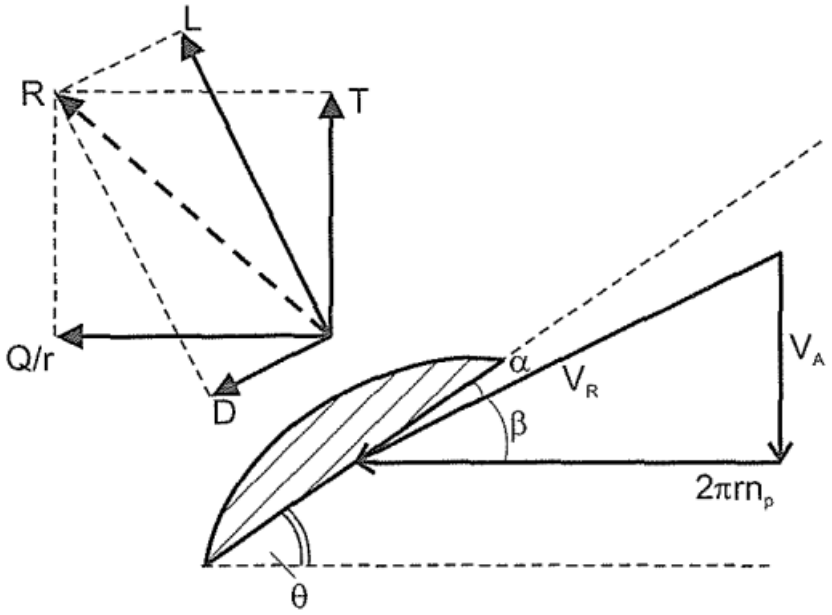


Figure 2.6: Cross section of a propeller blade and forces [88].

$$J = \frac{V_A}{n_p 2r} \tag{2.1}$$

Here V_A is the advance speed of the vessel in $[m/s]$, D is the propeller diameter in $[m]$ and n_p is the rotation speed of the propeller in $[rps]$. The thrust, nozzle thrust and torque can also be expressed in non-dimensional coefficients using the diameter, the rotation speed and the water density ρ using Eq. (2.2), (2.3) and (2.4) [15, 88].

$$K_T = \frac{T}{\rho \cdot n_p^2 \cdot D^4} \tag{2.2}$$

$$K_{TN} = \frac{T_N}{\rho \cdot n_p^2 \cdot D^4} \tag{2.3}$$

$$K_Q = \frac{Q}{\rho \cdot n_p^2 \cdot D^5} \tag{2.4}$$

The open water propeller efficiency can be defined using both the thrust and torque and these non-dimensional coefficients as described in Eq. (2.5) [13, 88].

$$\eta_o = \frac{1}{2\pi} \frac{T \cdot v_A}{Q \cdot n_p} = \frac{1}{2\pi} \frac{K_T \cdot J}{K_Q} \tag{2.5}$$

Figure 2.7 shows an open water propeller diagram from the Wageningen B-series. In this figure the thrust coefficient K_T, the torque coefficient K_Q and the open water efficiency η_o are shown as functions of the advance ratio J. Note that for scale effects the value of the

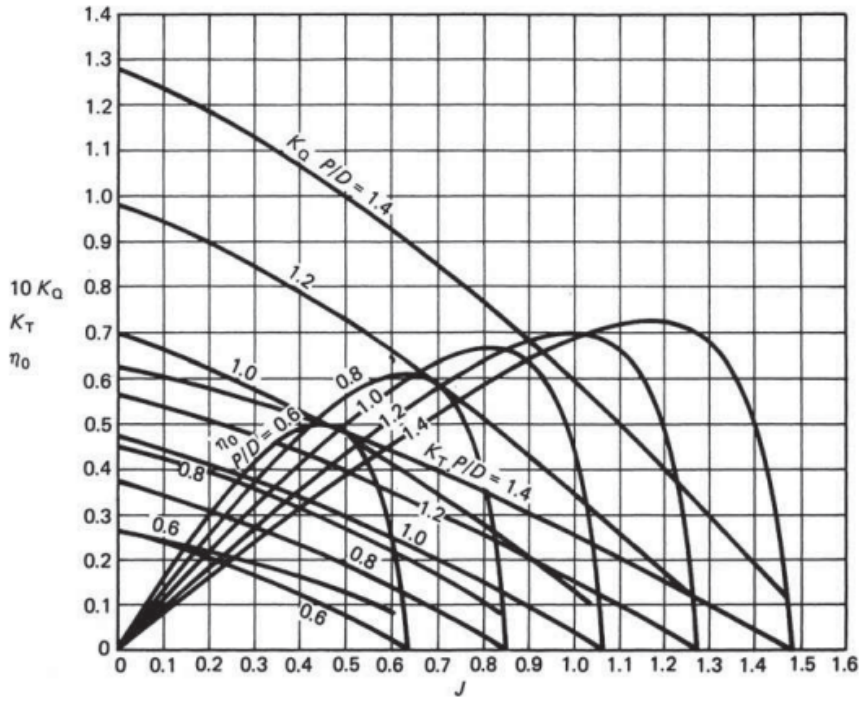


Figure 2.7: Open water efficiency for a range of pitch ratios [88].

torque coefficient is multiplied by 10. There are several lines for all different P/D values that exist in the Wageningen B-series [13, 88].

It can be noted that η_o goes to zero if either K_T or J goes to zero when examining Figure 2.7 and Eq. (2.5). It can also be noted that the maximum value for both K_T and K_Q are obtained when J reaches zero. This operating point is referred to as the *Bollard Condition* and is an important design condition for both tugs and dredging ships. It is defined by the maximum pull a ship can exert on a mooring bollard on a dock when attached by a rope or cable. This shows that the range of operating conditions that these coefficients are able to represent is limited [15, 88].

2.2.2 Four-quadrant diagrams

The open water diagram from Figure 2.7 is only able to capture propeller behaviour in forward motion. However, it is relevant to obtain information about the behaviour in conditions such as running astern or dynamic behaviour such as stopping. These operating conditions are captured by the four quadrants as shown in Figure 2.8 [15, 88].

The relation between the hydrodynamic pitch angle, the advance speed and the forward speed generated by the propeller is shown in Figure 2.9. A full revolution of this angle

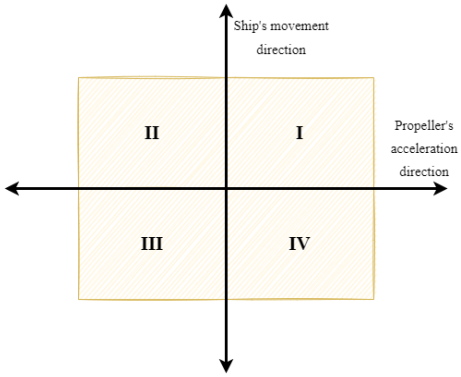


Figure 2.8: The four quadrants of operation [82]

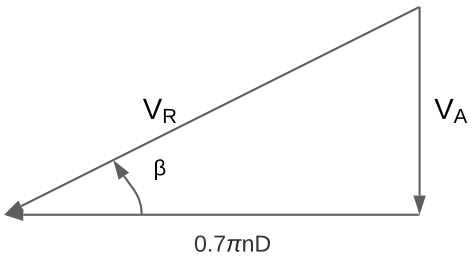


Figure 2.9: The relative speed V_R, hydrodynamic pitch angle β and the advance speed V_A [88].

takes the propeller through all four quadrants of operation. Equation (2.6) describes the mathematical expression of this relation and Table 2.3 shows what ranges of β correspond to what quadrant.

$$\beta = \arctan\left(\frac{V_A}{0.7 \cdot \pi \cdot n_p \cdot D}\right) \tag{2.6}$$

$$C_T = \frac{8K_T}{\pi(J^2 + (0.7 \cdot \pi)^2)} \tag{2.7}$$

$$C_Q = \frac{8K_Q}{\pi(J^2 + (0.7 \cdot \pi)^2)} \tag{2.8}$$

$$C_{TN} = \frac{8K_{TN}}{\pi(J^2 + (0.7 \cdot \pi)^2)} \tag{2.9}$$

Relying solely on the coefficients presented in Section 2.2.1 one is unable to capture the propeller behaviour in all these quadrants. Therefore the MARIN institute has developed two new thrust and torque coefficients C_T and C_Q as defined by Equations (2.7) and (2.8) respectively and a nozzle coefficient C_{T_n} described by Eq. (2.9). By plotting these two coefficients against the hydrodynamic pitch angle β a continuous behaviour can be captured over all four quadrants as shown in Figure 2.10 [13, 15].

Quadrant	Description of quadrant	Hydrodynamic pitch angle range
I	Ahead	$0° - 90°$
II	Crashback	$90° - 180°$
III	Backing	$180° - 270°$
VI	Crashahead	$270° - 360°$

Table 2.3: Ranges of β for each quadrant.

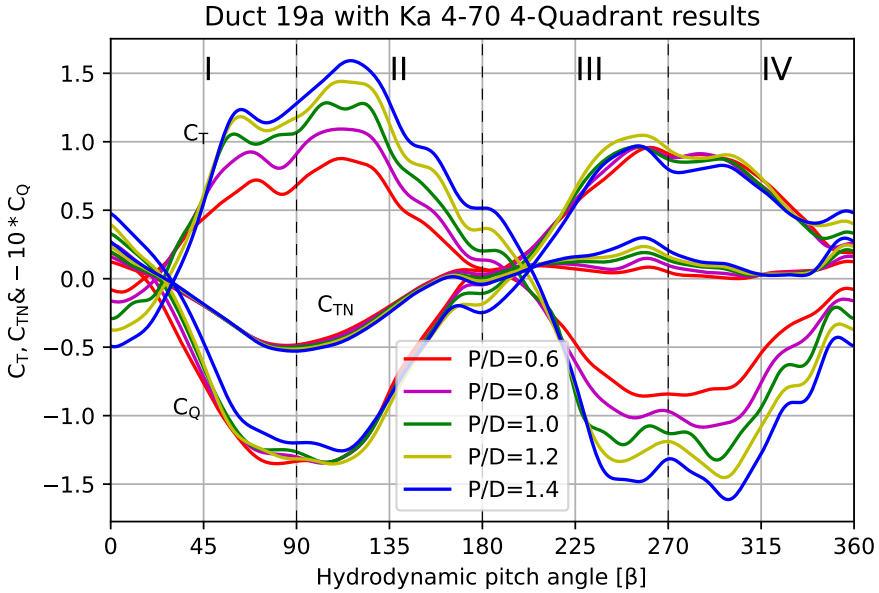


Figure 2.10: Wageningen Ka4-70's four-quadrant behaviour plotted with data from [65].

2.3 four-quadrant analysis

To have accurate data on the propeller in use is a very valuable asset for the design and operation of vessels and can be used for many different optimization efforts as mentioned in Section 1. Several approaches have been tried to obtain a model that can predict the four-quadrant behaviour of C_T, C_{TN} and C_Q.

The four-quadrant behaviour of the Wageningen B-series propeller has extensively been measured over different conditions by van Lammeren et al. in 1969 [46]. They tested for the influence of Pitch ratio P/D, expanded blade area ratio A_E/A_0 and number of blades Z. They represented the obtained data as a 20th-order Fourier series and this data set has served as the starting point for much research that has followed on the topic of predicting the four-quadrant behaviour of propellers.

In the past years several attempts to create models that can predict the four-quadrant behaviour of propellers have been tried for different applications. Different order levels of Fourier functions have been tried to approach the plot found by Oosterveld [77, 83]. They found that the lower order functions are not able to capture the behaviour accurately and the higher order functions (tested from 2nd to 9th order) suffered from overfitting. Ultimately

it was found that 3rd or 4th order gave the best results. The research by Tran et al. [83] focused their model on application for a Autonomous Underwater Vehicle (AUV).

Another approach is to model the forces exerted on the water by the propeller and try to find the generated lift and drag both shown in Figure 2.6 through lift theory [7, 12]. These models often consisted of different models combined that model the engine and the flow individually. These models were found to not be able to create accurate results over the full range of the four quadrants and require knowledge about the surroundings such as the flow speed of water passing through the propeller. For both these investigations the model was compared to data obtained from a test with an AUV propeller.

In his research, Pivano modelled the behaviour through a state space model consisting of two nonlinear dynamic equations [58, 59]. Their research focused on the application of AUVs and used the same approach for the lift and drag relations as [7] and [12] but uses these differently. They used the model created in [33] as the baseline for their research and found that the final version of their model was able to approach the behaviour in all four quadrants although it oversimplified the behaviour. The final model did not depend on information about the surroundings like [7] and [12].

Lastly, Roddy et al. created their model using Feed Forward Neural Networks (FFNN) [65, 66, 67]. They trained and tested their models both on the data from Oosterveld for the Wageningen B-series screws and on data from AUV propellers. Using a shallow network consisting of 2 fully connected layers they trained a model for each quadrant individually and smoothed the results in the border regions by matching the slopes of the functions. The solution they found had little error with both data sets. At the edges of the data set an increase in uncertainty was noted.

The current state of the art sees different approaches to the problem of creating a model that captures the Four-quadrant propeller behaviour. The current models based on modelling the physical processes require more information about the environment [7, 12]. Other approaches did not require this extra information but simplified the outcome [58, 59]. The approaches modelling the physical process and producing the approximation as a Fourier function either simplified the solution [77] or did not cover the entire 360 range of the Four-quadrants [83].

Although some research has already been done using machine learning these models stem from 2006 to 2008 [65, 67, 66] and with the developments the computer sciences have made since then it is to be expected that better results can be obtained by relying on the advancements made in the field. More advanced techniques utilizing data have entered the scene since then and this research tries to investigate all the possibilities. All research thus far has only applied neural networks, therefore other machine learning methods will also be investigated.

In conclusion this chapter in Section 2.1 has provided context on the physical aspects of the problem that shape it. The four relevant propeller characteristics and their physical meaning were explained in detail. The properties and differences of fixed pitch and controllable pitch propellers were discussed providing some background on the propellers used for training the model. Additionally this chapter in Section 2.2 the meaning of the four quadrants is

discussed as well as the non-dimensional coefficients used to describe the four-quadrant behaviour. The information presented in this chapter will serve as a basis for the development of models and methodologies in the following chapters of the thesis.

3 Problem Description

Having gained an understanding of the background on both propellers and the four-quadrant behaviour, this chapter sets out what data is available in this research and how it is obtained. For each propeller type the different tests conducted and values measured during these tests are listed along with possible data augmentations. Furthermore, the relevant problem characteristics that follow from the available data and the desired output are listed. These characteristics will be used in Chapter 4 to help narrow down the selection of appropriate algorithms to use during this research.

3.1 Data

The data describing the Four-Quadrant behaviour is obtained from a 2006 study by Roddy et al. [65]. In their work they mapped the Four-Quadrant behaviour of the Wageningen Ka-series, a ducted CPP and the Wageningen B-series propeller, an FPP, for different conditions. The curves are described as a 20^{th} and a 30^{th} order Fourier series respectively. The coefficients of this Fourier series were obtained from the test conducted by Oosterveld et al. at the Netherlands Ship Model Basin [46]. The curves resulting from these various tests were evaluated from 0° to 360° at 1 degree intervals of the hydrodynamic pitch angle β.

3.1.1 Wageningen Ka-Series

For the test on the Wageningen Ka-series tests on two ducts were conducted measuring the behaviour of the propeller for a range of pitch ratios. The propeller thrust, nozzle thrust, and torque coefficient were measured for each condition. The two duct types tested on the 19*a* and 37 types which are for forward motion improvement and bidirectional improvement respectively [13]. For both propellers different pitch ratios were tested ranging from $P/D = 0.6$ to $P/D = 1.4$ with 0.2 increments. The results of these tests can be seen in Figures B.4 and B.5.

3.1.2 Wageningen B-Series

Three different tests were conducted using the Wageningen B-series each time varying a different propeller characteristic mentioned in Section 2.1.2 while measuring the thrust and torque coefficients C_T and C_Q. Firstly a range of different pitch ratios was tested ranging from $P/D = 0.5$ to $P/D = 1.4$ keeping the number of blades and the expanded area ratio at the same value. The results of this test are shown in Figure B.1. Secondly, the expanded area ratio was varied ranging from 0.40 to 1.00 keeping the number of blades and the pitch ratio at the same value. The results of this test are shown in Figure B.2. Lastly, the number

Measured conditions		
Test	Propeller type	Pitch ratio
Pitch ratio test	B4-70	P/D=0.5
	B4-70	P/D=0.6
	B4-70	P/D=0.8
	B4-70	P/D=1.0
	B4-70	P/D=1.2
	B4-70	P/D=1.4
Expanded area ratio test	B4-100	P/D=1.0
	B4-85	P/D=1.0
	B4-70	P/D=1.0
	B4-55	P/D=1.0
	B4-40	P/D=1.0
Number of blades test	B3-55	P/D=1.0
	B4-40	P/D=1.0
	B5-75	P/D=1.0
	B6-80	P/D=1.0
	B7-85	P/D=1.0

Table 3.1: Tested conditions of the B-series data.

of blades was varied from 3 to 7 blades keeping the pitch ratio at the same value and the expanded area ratio at roughly the same value. The results of this test are shown in Figure B.3. Table 3.1 gives an overview of the propellers and conditions used for each test.

3.1.3 Data representation

For both propeller series a Fourier series describes the measured curve for each training condition. The curve consists of the evaluation of multiple coefficients using Eq. (3.1). The coefficients used in this research are all listed in Appendix B. For each test, these curves are evaluated at each β value creating a data set with 360 data points per test, which means that the data set is relatively small.

$$f(\beta) = \frac{1}{2}A_0 + \sum_{k=1}^{i}[A_k \cos(k2\pi\beta) + B_k \sin(k2\pi\beta)] \tag{3.1}$$

Data augmentation

To help to model capture the behaviour better and to prevent overfitting data augmentation can be applied [21, 38]. Data augmentation has been proven to improve prediction quality on regression problems [32]. Data augmentation is a series of strategies for enlarging and enhancing the size of training data [92]. Because the data is cyclical in nature, sinusoid terms, as a function of β, were added to the training data at different frequencies [65]. Table 3.3 gives an overview of all the augmentation options generated for this research. The choice was made to vary the period of the sinusoids of $\sin(\omega\beta)$ and $\cos(\omega\beta)$ from $\omega = \frac{1}{10}$ to $\omega = 10$. Varying the amplitude of the augmentations would not be effective since the

Original training data
beta
PD
EAR
Z

Table 3.2: Original parameters available for training the model

Available augmentation values			
$\cos(0.1\beta)$	$\cos(0.25\beta)$	$\cos(0.5\beta)$	$\cos(\beta)$
$\cos(2\beta)$	$\cos(3\beta)$	$\cos(5\beta)$	$\cos(10\beta)$
$\sin(0.1\beta)$	$\sin(0.25\beta)$	$\sin(0.5\beta)$	$\sin(\beta)$
$\sin(2\beta)$	$\sin(3\beta)$	$\sin(5\beta)$	$\sin(10\beta)$

Table 3.3: Overview of all augmentations possible to add to the training data.

data will be normalized before training. The normalization of the data and the choice of augmentation will be further discussed in Chapter 5.

3.2 Problem characteristics

Knowing the available input and the desired output the translation between the two can be investigated. With the desired output being a continuous value this is a regression problem [94]. It is assumed that the propeller can be represented as a function dependent on the propeller characteristics listed in Table 2.1 and the hydrodynamic pitch angle β. Equation (3.2) represents the desired input and output of the model.

$$[C_T, C_{TN}, C_Q] = f(\beta, P/D, Z, EAR) \tag{3.2}$$

The order in which different data points are presented to the function in Eq. 3.2 does not influence the output value. Therefore the data is non-sequential in behaviour. The aim is to create a function able to predict the behaviour of a propeller with a certain set of values for P/D, Z and EAR for the full range of β from 0° to 360°. Therefore during training the test set should consist of the data from a single propeller and the train set should consist of all other propellers. One of the challenges in the data is that even when applying augmentation the model only has the value of the propeller characteristic to distinguish between the different data sets since β and its augmentations are equal in all data sets.

In summary, the problem is a regression problem. The model will aim to learn the behaviour of the function relating the propeller characteristics to the non-dimensional coefficients as mentioned in Eq. 3.2. With this information Chapter 4 will review several algorithms to see how well-suited they are to solve the problem described in this chapter.

4 Algorithm investigation

With the necessary background information provided and the problem demarcated the choice for an algorithm can be made. As stated in Section 2.3 this research will focus on using a data-driven model that can accurately predict the four-quadrant behaviour of a propeller given the propeller characteristics listen in Table 2.1. In this section, different algorithms will be reviewed as possible solutions to the posed problem. In Sections 4.1 and 4.2 different algorithms will be reviewed to see if they are appropriate for this research. The methods of measuring and validating the performance of the algorithm are discussed in Sections 4.3 and 4.4.

4.1 Machine Learning

Machine learning is a branch of artificial intelligence that systematically applies algorithms to synthesize the underlying relationships among data and information [5]. In the last decades with the increase of available data and computational power of computers, the application of machine learning has increased in fields beyond computer science. Machine learning techniques are applied in a variety of disciplines and fields for different purposes [1, 29, 48]. A select subset of present solutions is suitable for the issue presented in this research.

The solution has to use a regression approach and output a set of numerical values and therefore has to utilize a supervised approach [18]. The different machine learning algorithms that will be reviewed in this Section are Support Vector Regression, Decision Tree Regression, k-Nearest Neighbour Regression, Gaussian Process Regression, and the Ensemble Method.

4.1.1 Support Vector Regression

Support Vector Regression (SVR) is based on the Support Vector Machine algorithm but used for regression tasks. The core idea behind SVR is to find a hyperplane that best fits the data while minimizing the error between the predicted and actual values. This hyperplane is surrounded by a margin, and the data points that lie on the margin boundaries are called support vectors. SVR aims to minimize the error while ensuring that a certain margin of tolerance is maintained [5, 70, 78].

Typically SVR uses a loss function called the epsilon-insensitive loss function given by Eq. (4.1). This function allows for a certain degree of error between the predicted value and the actual target controlled by ϵ. The objective of SVR is to minimize this loss while finding the optimal hyperplane [5, 17, 91].

$$minimize : \frac{1}{2}||w||^2 + C \cdot \Sigma(\max(0, |y_i - f(x_i)| - \epsilon) \tag{4.1}$$

Here:

- w represents the weights of the hyperplane.
- C is a regularization parameter that controls the trade-off between maximizing the margin and minimizing the error.
- y_i and x_i are the actual target and input features for the i-th data point.
- $f(x_i)$ is the predicted value for the i-th data point.
- ϵ represents the margin of tolerance for errors.

The optimization problem aims to find w and $f(x_i)$ that minimize the loss function while satisfying the constraints.

The SVR method has different strengths and weaknesses [5].

Strenghts:

- SVR is effective for modeling both linear and non-linear relationships in data, thanks to the use of kernel functions.
- It is robust against outliers because the loss function allows for a certain degree of error.
- SVR provides control over the margin of tolerance ϵ and regularization C, allowing fine-tuning of model complexity.
- It works well in high-dimensional spaces, making it suitable for various applications, including financial forecasting and bioinformatics.

Weaknesses:

- SVR can be computationally expensive, especially when dealing with large datasets.
- Selecting an appropriate kernel function and tuning hyperparameters can be challenging.
- Interpretability can be an issue with SVR, as understanding the significance of the hyperplane's coefficients may not be straightforward.
- SVR may not perform well if the choice of kernel and hyperparameters is suboptimal, leading to overfitting or underfitting.

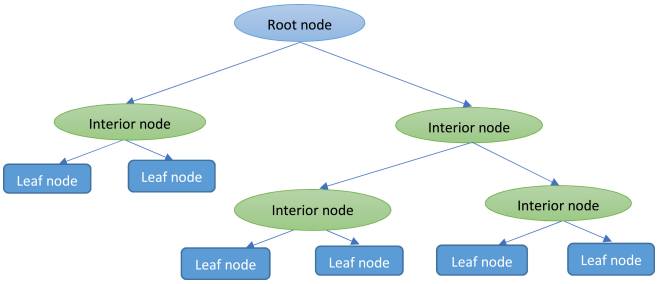


Figure 4.1: Overview of a regression tree [8].

4.1.2 Decision Tree Regression

Decision Tree Regression is a supervised machine learning algorithm used for regression tasks. It's a part of the broader family of Decision Trees used for both classification and regression. The fundamental idea behind Decision Tree Regression is to recursively partition the data into subsets based on the values of input features in a way that minimizes the variance of the target variable within each subset. This process creates a tree-like structure where each internal node represents a feature, each branch represents a decision based on that feature, and each leaf node contains a predicted value [37, 90].

The decision-making process in Decision Tree Regression involves selecting the feature that provides the best split, typically based on criteria like mean squared error or mean absolute error. The tree is built iteratively by choosing the feature and its threshold that results in the best partition. This continues until a predefined stopping criterion is met, such as a maximum depth of the tree or a minimum number of samples per leaf. Figure 4.1 provides an overview of these elements in a regression tree[63].

At each internal node of the tree, it aims to minimize the variance of the target variable within the subsets created by the node's decision. The variance of the target variable within a subset is calculated using Eq. (4.2) [37].

$$Var(Y) = \frac{\Sigma(y_i - \hat{y})^2}{n} \tag{4.2}$$

Where:

- Y is the target variable.
- y_i is an individual data point's target value.
- $\hat{y}$ is the mean target value of the subset.
- n is the number of data points in the subset.

The algorithm evaluates different features and their thresholds to find the split that minimizes the weighted sum of variances in the resulting subsets. The weighted sum is often calculated using the total variance before the split as a reference. This process is repeated recursively for each subset until the tree is constructed.

The decision tree regression method has different strengths and weaknesses [57].

Strengths:

- Decision Tree Regression is simple to understand and interpret, making it suitable for exploratory data analysis.
- It can handle both numerical and categorical data.
- Decision Trees can capture non-linear relationships in the data.
- They are robust to outliers, as the splitting process is based on the relative ordering of values.
- Decision Trees can be visualized, providing insights into the decision-making process.

Weaknesses:

- Decision Trees can easily overfit the training data if the tree is too deep or not pruned properly. Overfitting leads to poor generalization to new, unseen data.
- They can be unstable, meaning small changes in the training data can result in significantly different trees.
- Decision Trees tend to create biased trees if some classes dominate. Balancing the dataset or using ensemble methods like Random Forests can mitigate this issue.
- They may not perform well on tasks where the relationships between features and the target variable are complex and require high-level abstractions.
- In summary, Decision Tree Regression is a versatile algorithm for regression tasks, offering simplicity and interpretability. However, it requires careful tuning to prevent overfitting and may not perform optimally on complex problems without further techniques like pruning or ensemble methods.

4.1.3 k-Nearest Neighbour Regression

k-Nearest Neighbor (k-NN) Regression is a supervised machine learning algorithm used for regression tasks. Unlike traditional regression algorithms that learn a functional relationship between features and the target variable, k-NN Regression makes predictions based on the similarity between the input data point and its neighboring data points in the training data set [26, 55].

Given a new data point for which you want to make a prediction, k-NN identifies the k-nearest data points from the training data set. "Nearest" is typically defined using a distance metric, such as Euclidean distance or Manhattan distance. Once the k-nearest neighbors are identified, k-NN Regression calculates the predicted value for the new data point as the average or weighted average of the target values of these k-nearest neighbors. The choice of the value of k the number of neighbors to consider is a hyperparameter that affects the model's performance. A smaller k value makes the model sensitive to noise in the data, while a larger k value provides a smoother but potentially biased prediction [42, 51].

To make a prediction for a new data point, the algorithm calculates the predicted target value as the average of the target values of its k-nearest neighbors using Eq. (4.3) [41].

$$\hat{y} = \frac{1}{k}\Sigma(y_i) \tag{4.3}$$

Where:

- $\hat{y}$ is the predicted target value.
- k is the number of nearest neighbors.
- y_i represents the target values of the k-nearest neighbors.
- In some cases, you might use weighted averaging, where the neighbors' contributions to the prediction are weighted based on their distance from the new data point.

The k-NN regression method has different strengths and weaknesses.

Strengths:

- k-NN Regression is simple to understand and implement, making it a good choice for quick prototyping and baseline models.
- It is non-parametric, meaning it doesn't assume a specific functional form for the data, making it flexible for different types of relationships.
- k-NN can capture complex, non-linear patterns in the data.
- It's effective when the underlying data distribution is not well-known or changes over time.

Weaknesses:

- Computationally expensive, especially with large data sets, as it requires calculating distances to all data points in the training set.
- Sensitive to the choice of the distance metric and the value of k, which may require hyperparameter tuning.
- Prone to the "curse of dimensionality," where the algorithm's performance degrades as the number of features increases.
- Not suitable for data sets with imbalanced class distributions or when some features are more relevant than others, as all neighbors are treated equally.

4.1.4 Gaussian Process Regression

Gaussian Process Regression (GPR) is a Bayesian non-parametric machine learning technique used for regression tasks. Instead of modeling data with a fixed parametric function, GPR models the relationship between input features and the target variable as a probability distribution over functions. It assumes that any finite number of points from the function follows a joint Gaussian distribution. In simpler terms, Gaussian processes capture uncertainty about the true underlying function by considering multiple possible functions that could fit the data [22, 72, 86].

Given a set of training data points with associated target values, GPR starts by defining a prior distribution over functions. This prior represents our beliefs about the possible functions that describe the data. As new data points are observed, the GP model updates its beliefs about the underlying function by computing the posterior distribution over functions. This is done using Bayes' theorem, which combines the prior distribution with the likelihood of observing the data. To make predictions for new, unseen data points, the GPR model computes a predictive distribution over possible target values. This predictive distribution

provides not only a point estimate but also a measure of uncertainty in the prediction [24, 62].

The core mathematics of GPR can be broken down into the following steps [22, 86]:

1. *Prior:* The prior distribution over functions in a Gaussian process is typically assumed to be a multivariate Gaussian distribution. This distribution is defined by a mean function and a covariance function or kernel, denoted as $k(x, x')$. The kernel function quantifies the similarity between input points x and x'.

2. *Posterior:* When new data points are observed, the Gaussian process model computes the posterior distribution over functions. This posterior is also a Gaussian distribution whose mean and covariance are updated based on the observed data and the prior. The equations for computing the posterior are derived using Bayes' theorem.

3. *Predictive Distribution:* To make predictions for new data points, the Gaussian process model computes the predictive distribution, which is another Gaussian distribution with a mean and variance. The mean of the predictive distribution provides the point estimate, and the variance measures the uncertainty associated with the prediction.

The choice of kernel function is crucial in GP Regression, as it determines the shape and characteristics of the functions that the GP can model.

Strengths:

- Flexibility: GPR can model complex, non-linear relationships in the data and capture uncertainty in predictions.

- Provides uncertainty estimates: Gaussian processes do not only provide point predictions but also quantify the uncertainty associated with those predictions, making them valuable in decision-making under uncertainty.

- Small data efficiency: GPs perform well with small to moderately-sized data sets.

- Non-parametric: GPs do not make strong assumptions about the functional form of the underlying process, making them versatile.

Weaknesses:

- Scalability: GPs can be computationally expensive for large datasets due to the need to invert large covariance matrices.

- Choice of kernel: The effectiveness of GP Regression depends on selecting an appropriate kernel function, which may require domain expertise.

- Interpretability: GPs are often considered as "black-box" models, making it challenging to interpret the learned relationships.

In summary, Gaussian Process Regression is a flexible and powerful regression method that models the relationship between inputs and outputs as distributions over functions. It excels when dealing with small to moderate-sized datasets and provides valuable uncertainty estimates in predictions. However, it can be computationally expensive and may require careful selection of kernel functions.

4.1.5 Ensemble Method Regression

Ensemble method regression in machine learning is a technique that leverages the power of multiple regression models to make more accurate predictions. Instead of relying on a single model, ensemble methods combine the predictions of several base models to produce a stronger and more robust final prediction [64, 84].

The key idea behind ensemble methods is that different models may capture different aspects of the data and have varying strengths and weaknesses. By aggregating their predictions, the ensemble can often achieve better overall performance. There are several popular ensemble methods, including Bagging, Random Forests, Boosting, and Stacking, each with its unique approach to combining predictions explained further below [54]:

- Bagging: Bagging builds multiple base models by training them on different subsets of the training data, which are randomly sampled with replacement. The predictions from these base models are then averaged for regression or voted on for classification to make the final prediction. This technique reduces variance and can help mitigate overfitting.
- Random Forests: Random Forests is an extension of bagging that focuses on decision trees. It builds multiple decision trees, each trained on a bootstrapped sample of the data and considering only a random subset of features at each split. The final prediction is made by averaging the predictions of individual trees. This method is highly robust and can handle high-dimensional data.
- Boosting: Boosting is an iterative technique that aims to correct the errors of previous base models. It assigns weights to data points and focuses on those that were previously misclassified. Boosting sequentially trains base models, giving more weight to the misclassified samples. The final prediction is an aggregation of these weighted base models. Popular boosting algorithms include AdaBoost and Gradient Boosting.
- Stacking: Stacking combines predictions from multiple base models by training a meta-model on top of them. The base models' predictions serve as input features for the meta-model. Stacking allows the ensemble to capture higher-level patterns in the data and often leads to improved performance.

Due to the smaller data set available in this research only the stacking method will be used.

Strengths:

- Improved Predictive Performance: Ensemble methods often outperform individual models, as they can capture different patterns in the data and reduce overfitting.
- Robustness: Ensembles are less sensitive to noise and outliers in the data due to their combination of multiple models.
- Versatility: Ensemble methods can be applied to various regression algorithms, making them versatile for different problem domains.
- Generalization: They tend to generalize well and can adapt to complex relationships in the data.

Weaknesses:

- Complexity: Ensembles can be computationally expensive and complex to train, especially when using a large number of base models.
- Interpretability: The final ensemble model can be challenging to interpret because it combines the outputs of multiple models.
- Overfitting: While ensembles reduce overfitting in many cases, they can still overfit if not properly tuned or if too many base models are used.
- Parameter Tuning: Properly configuring ensemble parameters, such as the number of base models and their hyperparameters, can be time-consuming.

4.2 Deep learning

Deep learning is a sub-field of machine learning. Deep learning is an advanced machine learning approach that is used to make computers able to automatically extract, analyze, and understand useful information from the raw data [18]. The results obtained from deep learning are much improved than conventional machine learning approaches in other disciplines [2, 73, 74]. Deep neural network uses non-linear models of multiple hidden layer architecture that make the system learn about the complex relationship between input and output. The advantage of deep learning over machine learning is that it does not require manually extracted or handcrafted features as in machine learning.

One disadvantage of deep learning algorithms is that they require relatively more data to converge to a solution in comparison to machine learning models. Another possible disadvantage of using deep neural networks is that the reason why the model converges to a certain output can not be understood, this is called the black box effect [35]. The complexity of Deep Learning models hinders the straightforward understanding of the rationale behind their decisions [56]. There are three major reasons why understanding the inner workings of a model is desired [6]. Firstly, the ability to improve the model and solve any problems that arise. When trained models often need troubleshooting or adjusting to obtain the best results and this can be done effectively when the inner workings are comprehensible. Secondly, being able to interpret the hidden layers of a model can give insight into the discovered patterns. Lastly, the results of the model need to be trustworthy. For example, a medical professional might not trust an outcome due to a fear of artifacts or biases in the data [53]. It has been proven that medical models have had biases in the data [16] and the models can be misled by adversarial examples [27].

Deep learning has already been proven to be successful at this task by Roddy et al. [65], therefore this research will also investigate the possibilities of using deep learning applications to find a solution. Firstly, the basic elements of neural networks will be explained. Secondly, different architectures of neural networks and their idiosyncrasies will be discussed.

It is theorized that with sufficient data and a capable neural networ,k any continuous function can be approximated using neural networks in the universal approximation theorem [20]. All neural networks consist of a combination of connected elements called nodes. A node receives several input values, processes these, and produces a single output value [87]. One of the most common ones is the adaptive linear combiner that receives a set of n input values $\mathbf{X} = [x_0, x_1, \ldots, x_n]$ and multiplies each value of $\mathbf{X}$ with their corresponding weight $\mathbf{W} = [w_0, w_1, \ldots, w_n]^T$. The sum of each input value multiplied by their weight is processed

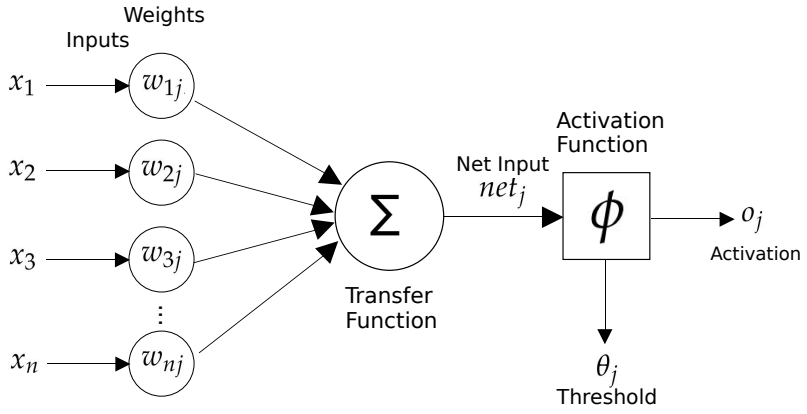


Figure 4.2: Overview of a perceptron [19].

by the transfer or preactivation function and sent to the activation function that uses some non-linear function to output a value Y as shown in Figure 4.2 [43, 68]. One layer of a neural network consists of one or more of these nodes.

A neural network consists of an input layer that reads the data, one or more hidden layers, and an output layer that produces the prediction of the model. For a network to be able to create non-linear distinctions between the input samples a hidden layer is required [43]. The number of hidden layers can vary from a single hidden layer to more than a hundred layers [79]. There is a link between the size and amount of layers and the size of the data set, where more data means a higher upper limit of sufficient size of the network [81]. The lower layers close to the data input learn simple features, while higher layers learn more complex features derived from lower layer features [93]. Therefore it is to be expected that with the small data set available the size of the network will also remain small. The network architecture prescribes both the size of the network and the way the different nodes are connected with each other. Each different network architecture has its own pros and cons and in this section, certain network architectures will be discussed with their applicability to the problem described in Section 3.2.

4.2.1 Feedforward neural networks

In a Feedforward Neural Network (FNN) all information moves from input to output creating a network that effectively has no memory [71]. The number of layers and the number of perceptrons per layer determine the network architecture. Figure 4.3 shows an example of the setup of a feedforward neural network where the arrows represent the flow of information.

The effectiveness of FNNs on the problem set out in Section 3.2 has already been proven by Roddy et al [65, 66, 67]. However, more research on supervised learning solutions using sequential data has been done since. In the description of physical processes FNNs have been used to find the governing equations of physical processes [10, 30, 61] proving that

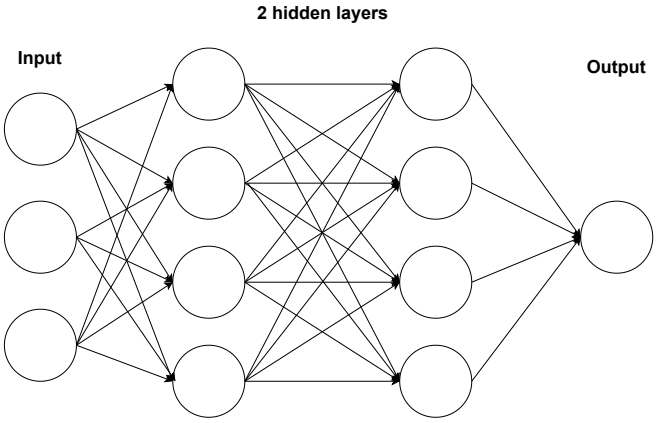


Figure 4.3: Example of a feedforward neural network [49].

FNNs have the ability to capture such continuous physical behaviour similar to the problem in this research. FNNs have been used on data generated by a Fourier transform similar to the input data for this research by Gashler and Ashmore [28] with success.

4.2.2 Recurrent neural networks

A Recurrent Neural Network (RNN) differs from feedforward networks because not all information flows from input to output but some information is fed back into the network. RNNs process one element of an input sequence at a time with a maintaining state vector in their hidden units which contains information on all the past values of elements of that sequence [18]. Figure 4.4 shows an example of the structure of a recurrent neural network, it can be seen that information is handed from the second hidden layer to the first hidden layer with a time delay. This feedback gives the network a memory that enables the network to recognize sequences. However, this method is not an appropriate fit for non-sequential data and therefore in this research only FNNs will be used.

4.3 Losses and metrics

When training an algorithm there are different ways the quality of the prediction can be evaluated. During training the quality of the prediction is measured using a loss function that the model aims to minimize. The ultimate prediction the model generates is evaluated using the chosen metric or metrics.

Losses and metrics are conceptually not the same thing: losses are used by Gradient Descent to train a model, so they must be differentiable and their gradients should not be 0 everywhere. Plus, it is okay if they are not easily interpretable by humans. In contrast, metrics are used to evaluate a model, they must be more easily explicable, and they can be non-differentiable or have 0 gradients everywhere [31].

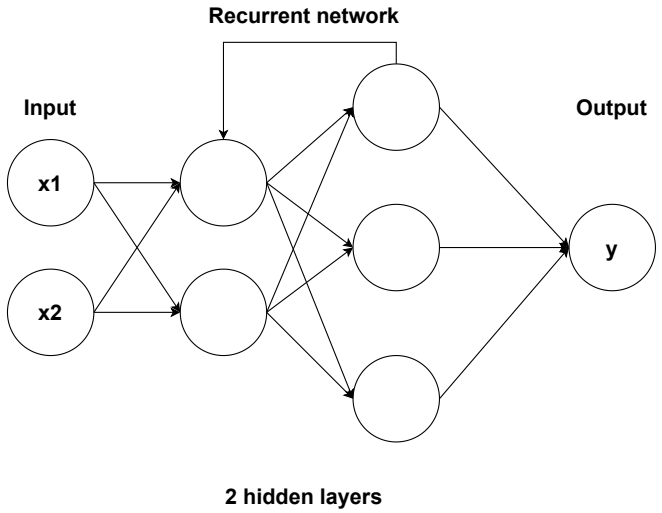


Figure 4.4: Example of a recursive neural network [50]. The connection between the second and first hidden layer has a time delay.

There are several standard loss functions and metrics that are pre-build into TensorFlow and Keras, the machine learning code libraries used in this research.

4.4 Validation

In machine learning, model validation is referred to as the process where a trained model is evaluated with a testing data set [3]. Model validation is carried out after model training. As illustrated in Figure 4.5, together with model training, model validation aims to find an optimal model with the best performance [85]. Note, that the learned model as well as the accuracy of the estimated generalization error depends on the particularly chosen partition of the original sample into training and validation sets and especially on the size of these sets. For example, the more instances we leave for validation, the fewer samples remain for training and hence the model becomes less accurate. Consequently, a bias is introduced to the estimated model error. On the contrary, using fewer instances for validating the model will increase the variance of the estimated model error [47].

Another important method of validating a model's predictions is the use of external validation mentioned as model testing using an independent data set in Figure 4.5. By introducing the model to a new data set issues related to the data quality of the original data set such as a bias can be discovered and external validation is considered to be the gold standard for true estimates of performance and generalizability of prediction models [60, 25].

In this research, the possibility of testing the predictions through external validation exist by using one of the test described in Section 3.1 for training and validating their predictions on other tests. 4 different data sets are available namely the data set with varying pitch ratio, varying number of blades and varying EAR values the B-series propellers and

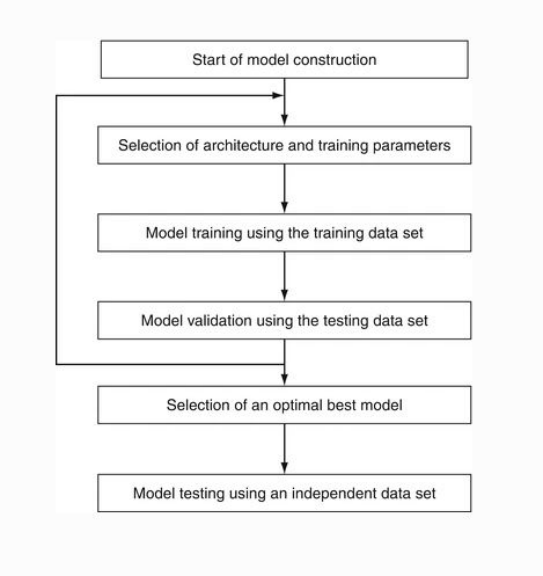


Figure 4.5: Overview of data handling process when training a model [85]

the data set with varying pitch ratio values for the Ka-series propellers. An optimal combination of training and external validation data sets will be chosen to ensure the quality and generalizability of the predictions made by the model developed in this thesis.

Conclusion

Different machine learning methods have been shown and will be tested on the data. For the neural network only a FNN will be investigated since the RNN is unsuitable for the data available. To evaluate the prediction quality of the model, a suitable metric will be chosen. Additionally, external validation will be applied to assess the model's generalizability. This involves combining data from different tests to create training and validation sets to ensure the model's effectiveness in predicting beyond the training data.

5 Methodology

The previous chapters have discussed the background of the problem, the characteristics of the problem, and the different methods chosen to use in making a prediction. This chapter will provide an overview of the methodology used in implementing the different methods for the problem in this thesis. The way the data is pre-processed before training is discussed in Section 5.1. Section 5.2 will discuss the way the different machine learning methods are implemented. Section 5.3 describes how the FNN is built. Lastly, this chapter will provide an overview of all hyper-parameters and the ranges for the hyperparameter tuning. This will be the input for the hyperparameter tuning. The results from this search will be discussed in Chapter 6.

5.1 Data preprocessing

This Section discusses the steps taken in preprocessing the data for training.

Data Augmentation

As shown in Table 3.3 different variables were added to the data set to give the model more information to utilize during learning. Different combinations of augmentation were implemented in preliminary testing and this showed that adding all the listed augmentation options created a more robust network. An overview of the differences between the different augmentation variations is provided in Appendix A.

Normalization

The first step in the process is the normalization of the data. Data normalization is an essential pre-processing step that involves the transformation of features in a common range so that greater numeric feature values cannot dominate the smaller numeric feature values. This practice minimizes a bias towards features with higher numerical values[76]. The data is normalized using the formula from Equation (5.1). For feature x the mean and variance are calculated. Every data point in x noted as x_i is normalized to x_i^{norm} using the formula below.

$$x_i^{norm} = \frac{x_i - mean}{\sqrt{variance}} \tag{5.1}$$

5.2 Machine learning methods

This Section discusses all the machine learning methods used and how they are implemented. The *scikit-learn* module is used for all the models discussed in this Section. Due to the relatively low number of hyperparameters in comparison to neural networks, the methods in this section have been tuned by hand. The chosen hyperparameters are listed for each method.

5.2.1 Support Vector Regression

In Section 4.1.1 the way SVR works was explained. In this research, the *SVR library* was used. The hyperparameters listed below were investigated. The values for C and ϵ will be investigated in the hyperparameter tuning. The only kernel producing valid results is the rbf kernel. This kernel is used for every model using SVR.

- C
- ϵ
- kernel: rbf

5.2.2 Decision Tree Regression

In Section 4.1.2 the way decision tree regression works was explained. In this research the *decision tree regression library* was used. The hyperparameters listed below were investigated. Due to the poor performance of this method in the initial investigation, this method is left out of the hyperparameter tuning search. The results of this method are explained in Appendix A.3.

- criterion
- min samples leaf
- min samples split

5.2.3 k-Nearest Neighbor Regression

In Section 4.1.3 the way k-NN work was explained. In this research the *k-NN regression library* was used. The hyperparameter below was investigated in the hyperparameter tuning.

- number of neighbors

5.2.4 Gaussian Process Regression

In Section 4.1.4 the way GPR work was explained. In this research, the scikit-learn Gaussian process regression library was used. The hyperparameter below was investigated in preliminary testing. The best working kernel was the Matern kernel.

- kernel

5.2.5 Ensemble Method Regression

In Section 4.1.5 the way ensemble methods are implemented in this research was explained. The *Stacking regressor* library was used to build the ensemble. The methods listed below were used to build the ensemble. The hyperparameters predetermined and investigated for each individual method are explained above.

- Support vector regression
- k Nearest neighbor regression
- Gaussian process regression

5.3 Neural Network

This section will discuss how the neural network is built and why certain choices in network architecture have been made.

Activation

The activation function creates a non-linear relation between the weights inputted into a node and the output [43]. Different activation functions have different characteristics. The Rectified Linear Unit (ReLU) is a popular activation function widely used on different types of problems. In contrast, an activation function such as sigmoid activation is better suited for classification problems [75]. This difference is demonstrated in the two figures below. Figure 5.1 shows a heatmap of the weights in every layer for both ReLU and sigmoid activation. Figure 5.1a shows that a selection of nodes light up more than other and the model has learned what node influence the relation between input and output data. In comparison Figure 5.1b shows the heatmap of the weights with sigmoid activation on its hidden layers instead of ReLU. As the heatmap shows rather than learning what nodes need highlighting the majority of nodes have high weight values. Additionally, the prediction shown in Figure 5.2 shows that the model has failed to learn the connection between the input and the output data. For these reasons, all neural networks explored during hyperparameter tuning will use ReLU activation in their hidden layers. The *Keras activation library* was used to implement the activation function in this research.

Optimizer

In this research, the optimizer used is the Adam optimizer from the *keras optimizers library*. It was introduced in 2014 by Kingma and it aims to combine the advantages of two recently popular optimization methods: the ability of AdaGrad to deal with sparse gradients, and the ability of RMSProp to deal with non-stationary objectives [40]. Further tests have shown the optimizer to be reliable chose for optimizer [69]. Due to these facts and to limit the search space during hyperparameter tuning, Adam will be used as the optimizer for all tests. The learning rate will be a hyperparameter and the value range will be 0.0001-0.01.

Network layout		
Network number	Number of layers	Number of nodes per layer
1	8	32
2	8	64
3	8	128
4	16	256

Table 5.1: Different network layout available for training

Loss function and metric

In Section 4.3 it was mentioned that there are several loss functions and metrics pre-build into Keras in the *loss* and *metrics* libraries. All combinations of loss-functions and metrics were tested in preliminary research. Here it was found that the quality of the prediction was more heavily dependent on the loss function than the metric. Three loss functions provided reliable predictions for every metric namely the Mean Squared Error (MSE), the Huber loss and the LogCosh loss. Ultimately it was chosen to use the MSE for the loss function and the metric for all test in this research. This limits the search space for the hyperparameter tuning. By having the metric and loss function the way the ultimate score of the network is determined is the same as the loss the network aims to minimize during training.

Network architecture

The network is built up of several dense layers consisting of multiple nodes from the *Dense library*. Different variations of the number of layers and the number of nodes per layer have been tested. Preliminary testing turned out that the best working networks are block networks that have a constant number of nodes for each layer except for the input and output layers. The input layer is determined by the choice in data augmentation varying from 4 to 20 nodes. For each of the non-dimensional coefficients one network is trained, thus the output layer always consists of 1 node. Table 5.1 provides an overview of the four networks available to train the model on. During hyperparameter tuning one of these four networks is chosen. Therefore the range of network layout is set to 1-4.

Dropout

One regularization approach is dropout. Each activation in a layer has a binary Bernoulli chance of p to be dropped and set to zero. Each node has a $(1 - p)$ chance of remaining unchanged. Figure 5.3 provides an overview of how a network changes when every layer has dropout with $p = 0.5$. By removing a part of the connections in the network the model is less likely to overfit [45, 80]. The *Drop out library* was used in this research. Based on preliminary testing dropout is applied on every fifth layer of the network and the range for the dropout chance p is set to go from 0 to 0.1. A dropout rate of $p = 0$ is effectively no dropout implemented.

5.4 Training Method

As stated in Section 3.2 the data is sequential and therefore the train test split should not be done randomly. Instead in the input parameters one of the tested conditions is specified to be the test set. In the case of the B-series pitch ratio variation test, 6 different tests were conducted for P/D values $[0.5, 0.6, 0.8, 1.0, 1.2, 1.4]$. If the pitch ratio value $P/D = 1.0$ is selected as the test condition, the model will be trained on the data from all 5 tests where the pitch ratio is in $[0.5, 0.6, 0.8, 1.2, 1.4]$. After training it is tested by making a prediction on the data where $P/D = 1.0$. After preparing the data the model is built as specified in the hyperparameters using the *Leave One Group Out* method.

The overall quality of the predictions of a certain model is measured using cross-validation. As explained in Section 5.4 for the varying pitch ratio data set, the data of 6 different propellers was measured. The choice of which of these 6 propellers is to be used as the test data during training is based on the goal of predicting the behaviour of propellers currently in use by Boskalis. Table 5.2 provides an overview of the value ranges of the propeller characteristics both in the data sets from the research by Oosterveld [46] and the ranges of these propeller characteristics on the propellers installed on the vessels in Boskalis' fleet.

Propeller characteristics			
Variable	Tested range	Range in use	Values used in training
P/D	0.5-1.4	0.825-1.15	[0.8, 1.0, 1.2]
EAR	0.4-1	0.6-0.73	[0.55, 0.7, 0.85]
Z	3-7	4	4

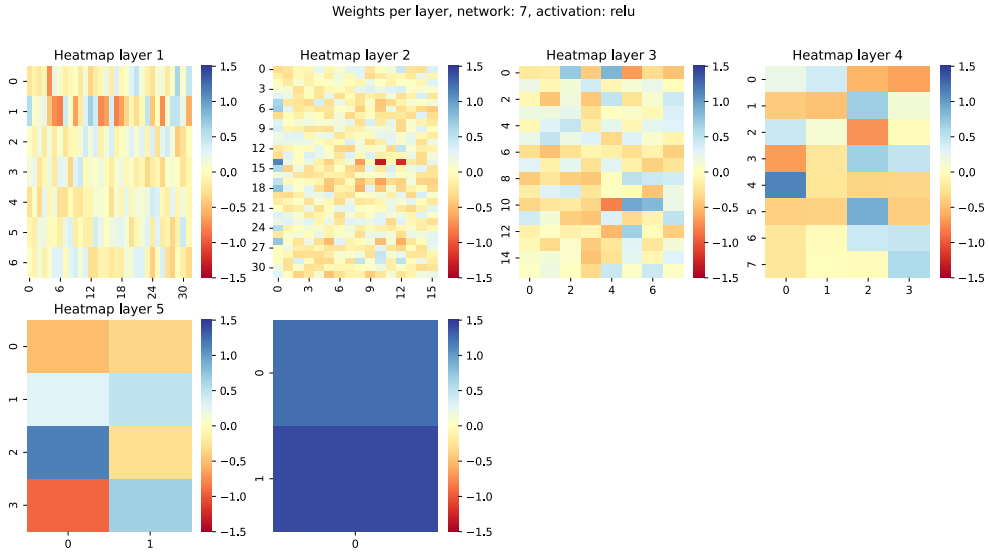
Table 5.2: Ranges of propeller characteristics measured and in use by Boskalis

5.5 Hyperparameter tuning

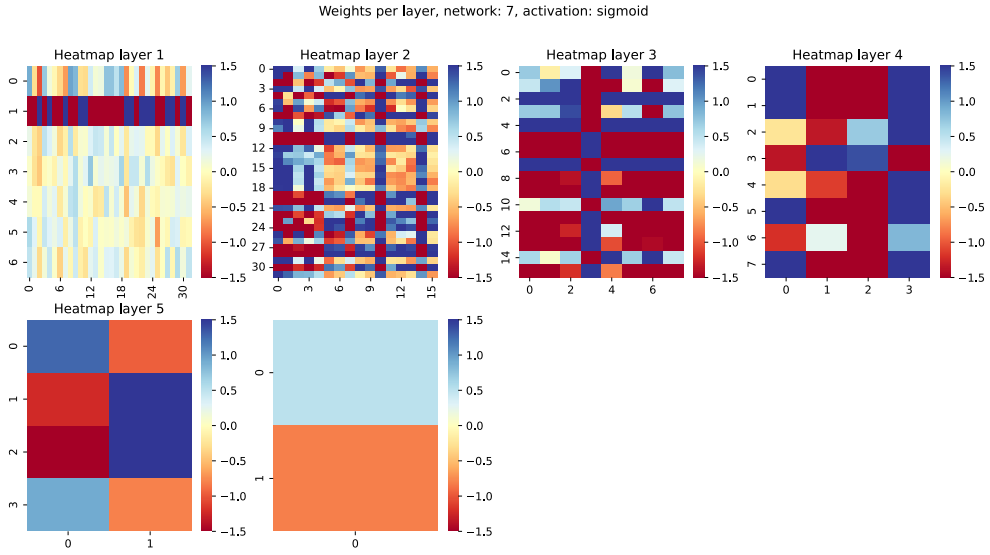
Having demarcated all the methods that will be investigated in this research the process of hyperparameter tuning can be started. In this work, the hyperparameter tuning was carried out using the optuna library. Table 5.3 lists all the hyperparameters, their data types, and the value ranges that will be investigated. The score of one set of hyperparameters is calculated by the sum of the MSE of all tested variations.

Hyperparameters		
Parameter	Data type	Value range
C	float	[0.1 - 1.0]
ϵ	float	[0.01, 0.001, 0.0001]
neighbor number	int	[2-5]
Learning rate	float	[0.0001 - 0.01]
Epochs	int	[50 - 200, step=10]
Validation split	float	[0,05 – 0,2]
Network layout	int	[1 - 4]
Drop out rate	float	[0 – 0.1, step=0,01]
Method	string	[neuralnetwork, SVR, knn, GPR, ensemble]

Table 5.3: Overview hyperparameters



(a)



(b)

Figure 5.1: Heatmap showing the weights per layer using (a) relu activation (b) sigmoid activation

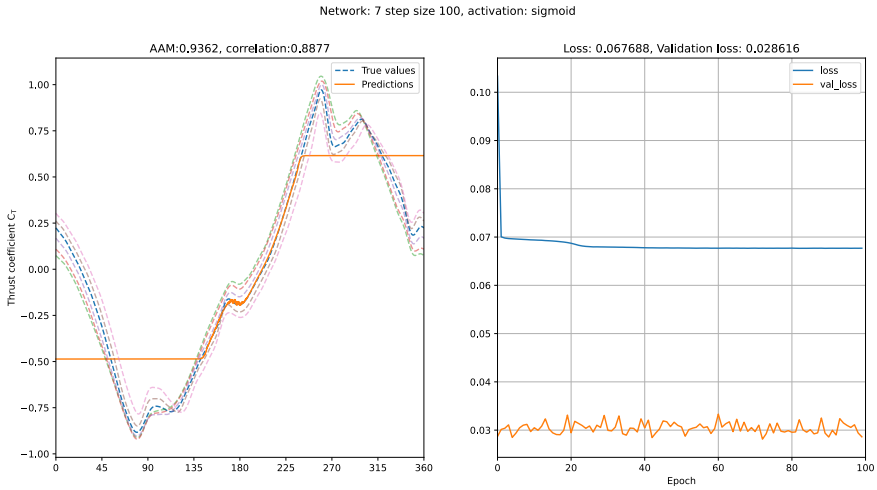


Figure 5.2: Prediction and training loss for a network trained using sigmoid activation

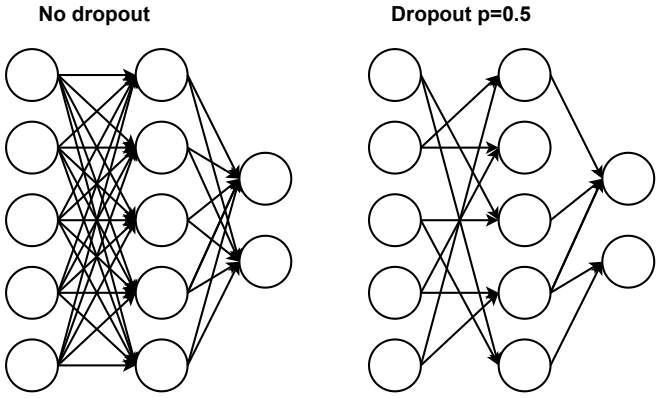


Figure 5.3: Overview of difference between network with and without dropout [45].

6 Results

This section discusses the results obtained from the developed model. Firstly, the tested propellers based on the relevant propeller ranges will be discussed. Secondly, the results from the hyperparameter tuning will be presented. Thirdly, the method obtained from hyperparameter tuning will be applied to all other data sets mentioned in Section 3.1. The results of all different data sets will be presented. Lastly, the performance of the method, resulting from hyperparameter tuning, on propellers from the fleet will be discussed.

Furthermore, these extreme values are not represented in the current propellers used in the fleet. Therefore when applying cross-validation in the training process not every possible measured characteristic is chosen to be used as test data. Rather only the value range representative of the characteristic values used in Boskalis' fleet are used as test sets. Table 5.2's last column shows the measured propeller characteristics that are used during cross-validation to measure the prediction quality of the model. During hyperparameter tuning the sum of the MSE of each model trained on one of these propellers is used to evaluate the score of the method.

6.1 Result hyperparameter tuning

The overall score of a set of hyperparameters during tuning was calculated by taking the sum of the mean squared error of each tested condition as listed in Table 5.2. Table 6.1 shows the outcome of the hyperparameter tuning and figure 6.1a shows the results corresponding to that set of hyperparameters. The ensemble method was found to be the best-performing method. This section will further discuss the performance of this method with these parameters on the data sets of the other propeller characteristics.

The areas around $\beta = 90^\circ$ and $\beta = 270^\circ$ are areas where the behaviour shows the largest change. These areas correspond to transitions from one quadrant to another. For each prediction, the total MSE is calculated as are the MSE scores for these two areas. The β ranges used for calculating these scores are $\beta = [75^\circ : 105^\circ]$ and $\beta = [255^\circ : 285^\circ]$ making each 30° in width and having one of these transition points as their center. An overview of all these scores is provided in Table A.1 for the B-series propellers and Table **??** for the Ka-series propellers.

Hyperparameter values	
Variable	Value
method	Ensemble
C	0.96
epsilon	0.01
neighbor number	3

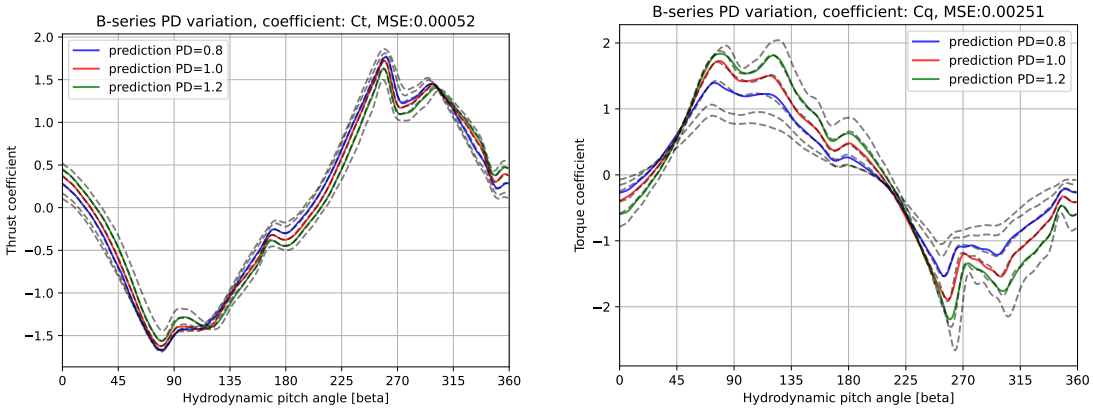
Table 6.1: Final values of hyperparameter tuning.

6.2 B-series

This Section shows a selection of the results of the developed method on data sets that it was not tuned to. The performance of the method on these data sets will indicate how well the method generalizes. For each data set the model was trained using the hyperparameter set tuned to the B-series pitch ratio variation of the thrust coefficient C_T. An overview of all the scores and Figures of the results from predictions on B-series propellers is found in Appendix A.1.1.

Pitch ratio

The performance of the model on both the thrust and torque coefficients, C_T and C_Q respectively, are shown in Figure 6.1. The model was tuned to the performance on the C_T and the model can capture the trend in the data correctly. The score of the model on the C_T coefficient was MSE $= 0.00052$ which is the lowest score obtained. This is to be expected since the model was optimized on this score. The performance of the developed method on the torque coefficient is shown in Figure 6.1b. The behaviour of C_Q has more fluctuations, especially around $\beta = 270°$. The model is capable of following the trend of the data but produces a flattened curve that does not follow the exact fluctuations in this region. This effect is seen when comparing the overall score MSE $= 0.00251$ to the local score around $\beta = 270°$ of $\text{MSE}_{270°} = 0.00757$.



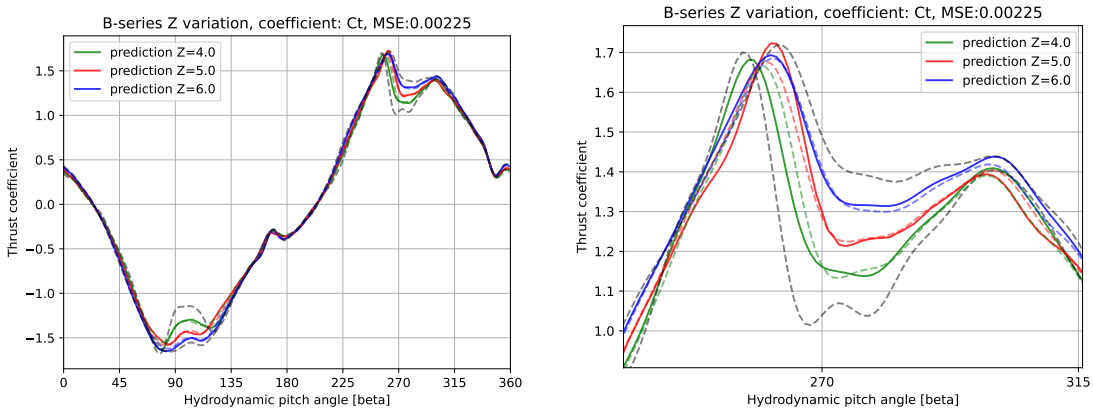
(a) (b)

Figure 6.1: Predictions on B-series PD data for(a) C_T coefficient (b) C_Q coefficient.

Number of blades

In the number of blades variation data, the differences between the behaviour of different propellers are smaller than in the other two data sets. This means that both the thrust and torque coefficient for all measured propellers overlap over the largest part of the four-quadrants as can be noted in Figure 6.2a. Only the areas around $\beta = 90°$ and $\beta = 270°$ show differences between the propellers, which correspond to areas of transition from one quadrant to another. The model can accurately predict the behaviour in the sections with

low differences between propellers and in the valley around $\beta = 90°$. Figure 6.2b shows that there are slight deviations between the predicted value and the true value in the peak around $\beta = 270°$. These trends are reflected in the comparison of the score over the full prediction MSE $= 0.00225$ and the local scores $\text{MSE}_{90°} = 0.00309$ and $\text{MSE}_{270°} = 0.00493$.

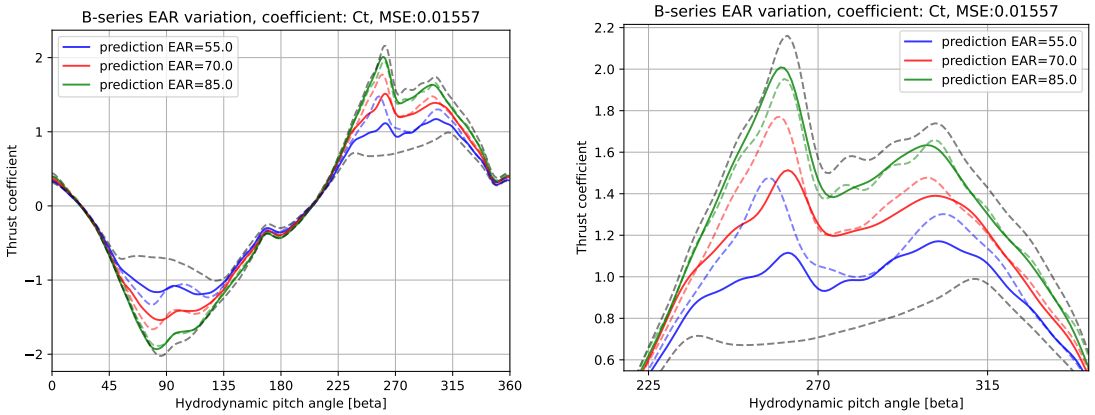


(a) (b)

Figure 6.2: Predictions of C_T for varying number of blades at (a) the full beta range (b) peak at $\beta = 270°$.

Expanded area ratio

When comparing the data set of the EAR variation to the pitch ratio variation data set it can be noted that differences in behaviour between propellers show more variation. While for the pitch ratio variation, the difference between the propellers is near constant over the full 360°, the expanded area ratio shows large variation around $\beta = 90°$ and $\beta = 270°$ and little variation around $\beta = 180°$ and $\beta = 360°$. The performance of the method on the EAR data set is shown in Figure 6.3. In Figure 6.3a the prediction of the whole range is shown. The model can accurately predict the behaviour in the areas with little difference between the propellers. The prediction around $\beta = 90°$ shows deviations between the prediction and the true values. However, the predicted curve is always closest to the true value of that propeller. Figure 6.3b shows the prediction in the peak more clearly. Here the prediction for EAR $= 70.0$ crosses the value for EAR $= 55.0$. This means that the value predicted in this region for a propeller with EAR $= 70$ would correspond to the value of a propeller with EAR $= 55$. The score on the expanded area ratio variation data set for the thrust coefficient for the full prediction is MSE $= 0.01557$ and the local prediction is $\text{MSE}_{270°} = 0.06857$. These scores are two orders of magnitude higher than the score resulting from hyperparameter tuning and one order of magnitude larger than the scores on pitch ratio variation data set for C_Q and number of blades variation data set for both coefficients.



(a) (b)

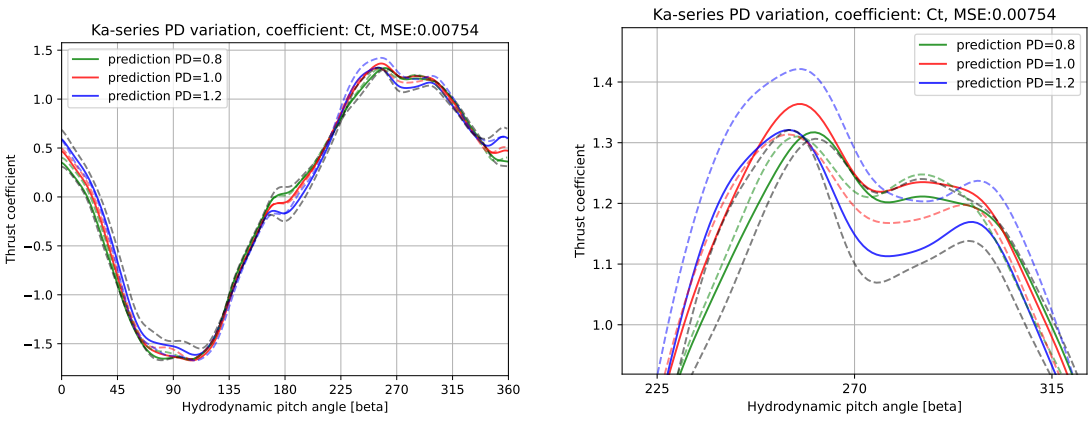
Figure 6.3: Predictions of C_T for varying EAR at (a) the full beta range (b) peak at $\beta = 270°$.

6.3 Ka-series

This Section discusses the results of the model on predictions made on the Ka-series data set. These results are important to Boskalis since all the propellers in use in the fleet are ducted CPPs as are the Ka-series propellers. All the results shown are for the propeller with a 19a duct since this is a similar type of duct to the ducts Boskalis currently uses in their fleet. Results for the propeller with duct type 37 are shown in the appendix A.1.2.

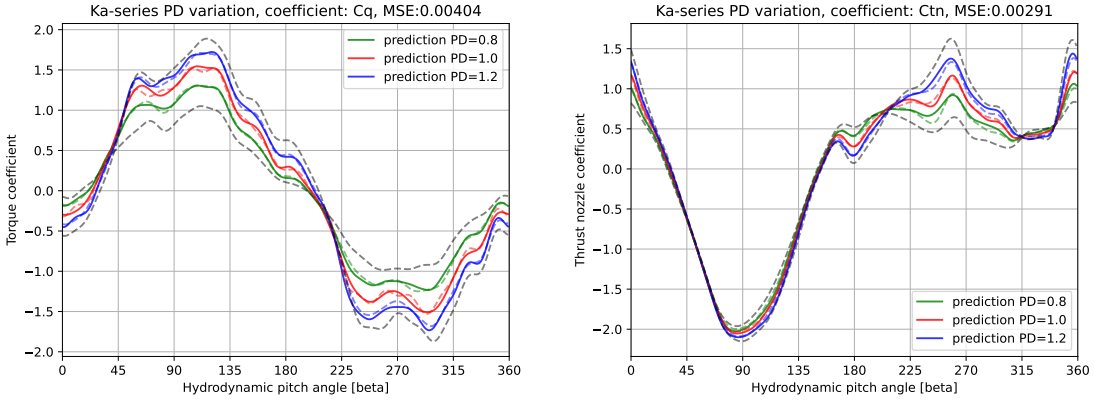
The results of training the model on the Ka-series data are shown in Figures 6.4. Figure 6.4a shows the results over the full range. Since the same propeller characteristic is varied here the overall behaviour of the thrust coefficient is similar to the data set the model was tuned to. The differences between the different propellers are generally equally spaced throughout the four-quadrants. The only exception is that the propeller with pitch ratio $P/D = 1.2$ is the extreme value in the peak around $\beta = 270°$. This can be seen in the prediction quality where the model can predict the behaviour of the propeller correctly except for the peak around $\beta = 270°$. Here all predictions show deviations as can be seen in Figure 6.4b. However, the prediction for $P/D = 1.2$ is unable to capture the behaviour when the model needs to extrapolate rather than interpolate. The overall score of the model on this data set is MSE $= 0.0754$ and the scores for each individual test condition are $\text{MSE}_{P/D=0.8} = 0.00140$, $\text{MSE}_{P/D=1.0} = 0.00130$, and $\text{MSE}_{P/D=1.2} = 0.00483$. This shows that the error is around 4 times higher when the model has to extrapolate.

The results for predictions on the C_Q and C_{TN} are shown in Figure 6.5a and 6.5b. For the torque coefficient, the behaviour the model has to learn is similar to that of the pitch ratio variation of the B-series propeller. Similar to the results in Figure 6.1b the C_Q prediction can capture the general behaviour of the propeller but flattens the prediction where the smaller fluctuations are left out. The overall score of the model on this data set is MSE $= 0.00404$. The performance of the C_{TN} prediction is equally able to capture the general behaviour of the propeller but flattens the behaviour in areas where the changes are more abrupt. The overall score of the model on this data set is MSE $= 0.00291$.



(a) (b)

Figure 6.4: (a) Predictions over the full beta range. (b) Close-up of the predictions around $\beta = 270°$.



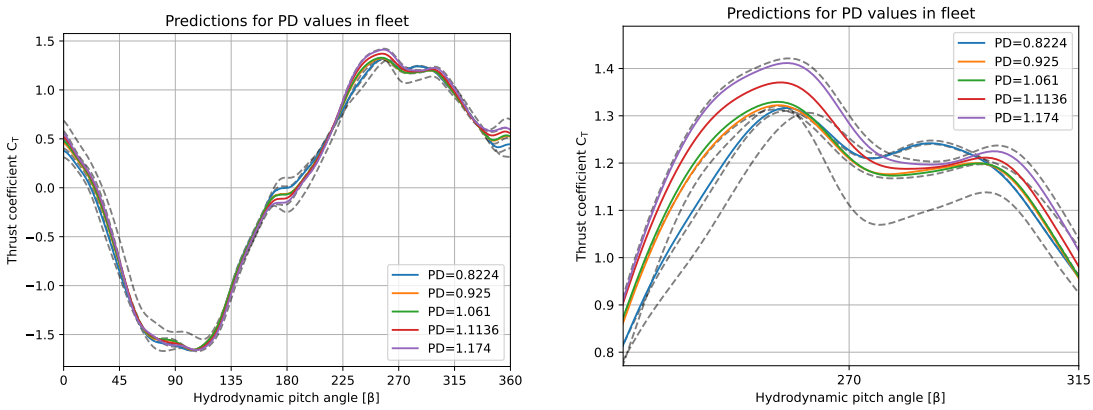
(a) (b)

Figure 6.5: Predictions for Ka-series 19 (a) C_Q coefficient. (b) C_{TN} coefficient.

6.4 Fleet propellers

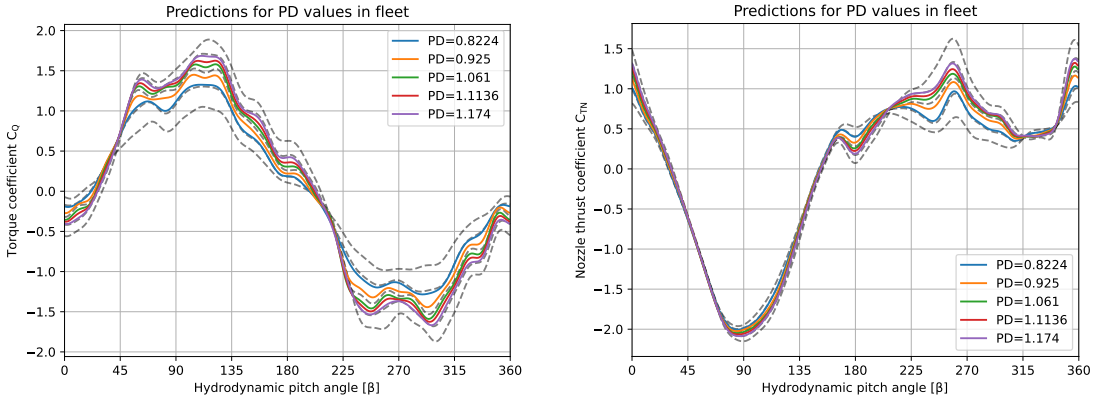
This Section discusses the results obtained from predicting pitch ratio values that are present in the Boskalis' fleet. All predictions are made using the Ka-series propeller with a 19a nozzle due to its similarity to the propeller type used in the fleet. Five different pitch ratios have been predicted by the model in the following range $[0.82 - 1.17]$. All these propellers have EAR values that are near to but differ from the propellers used in the training data having EAR $= 70.0$.

Figure 6.6 shows the predictions for the C_T coefficient. The model was trained on all the avialable data which means that the predictions near to $P/D = 1.2$ are not extrapolation predictions increasing the accuracy. All predicted curves follow the general trend expected of a propeller in that P/D region. The predictions are near the closest P/D value but not on the curve exactly. Figure 6.7 shows the prediction results for the same set of pitch ratios for the C_Q and C_{TN} coefficients. Here the general curve is similar to the expected value and near to the closest P/D value but not on it for both coefficients. The true values of the four-quadrant behaviour of these propellers is not known therefore no MSE score is calculated.



(a) (b)

Figure 6.6: (a) Predictions over the full beta range. (b) Close-up of the predictions around $\beta = 270°$.



(a) (b)

Figure 6.7: Predictions for propellers in fleet (a) C_Q coefficient. (b) C_{TN} coefficient.

6.5 Combined data sets

As mentioned in Section 6.4 the assumption is made that the number of blades is 4 and the EAR $= 70.0$, for the propeller whose four-quadrant behaviour is predicted. There is no information available about the effect of varying EAR values for Ka-series propellers. This information is available for B-series propellers. Therefore a model has been trained on both the varying pitch ratio and varying expanded area ratio test. Predictions made from the model trained on solely the pitch ratio variation data set and the model trained on the combined data set are shown in Figure A.19a. The top part shows the predictions made using both methods for a propeller with the P/D and EAR value of a propeller from Boskalis' fleet. The bottom part of the figure shows the absolute difference between both predictions is largest in the areas around $\beta = 90°$ and $\beta = 270°$. These β values correspond to the transition from the first to the second quadrant and the third to the fourth quadrant. The interaction between the propeller blade and the water is very turbulent during these transitions because the flow of water changes direction. These operating conditions are used less during operations. Therefore, the inaccuracies in these regions do not impact much of nominal operating. Appendix A.4 provides an overview of this comparison for other propellers used by Boskalis whose P/D and EAR values are known. These show that an increase in the difference between the true EAR value and the assumed value of EAR $= 70.0$ leads to an increase in the difference between both predictions.

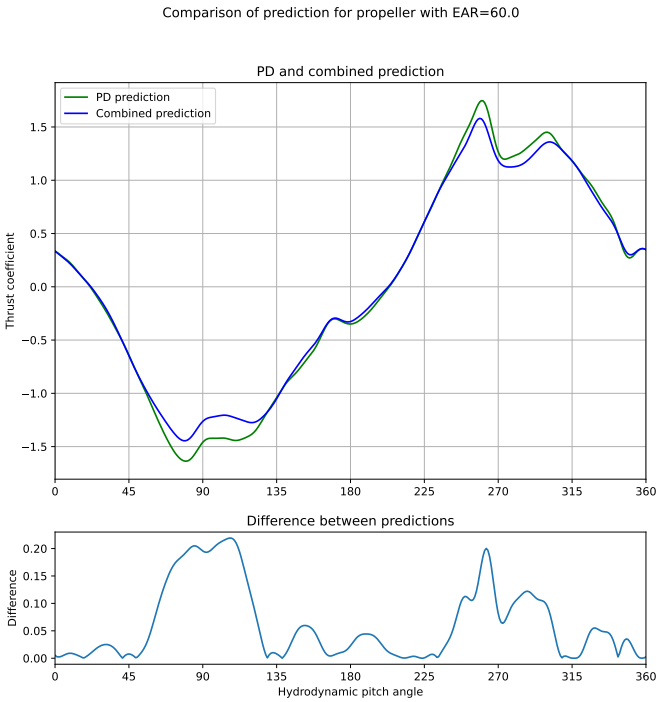


Figure 6.8: Prediction using both P/D data set and combined data set along absolute difference

7 Summary

This Chapter will provide an overview of the conclusion of this thesis work. The overall goal of the research was to develop a model that can create predictions for the four-quadrant behaviour of a propeller based on the propeller characteristics. The conclusion from investigating the state of the art of modelling four-quadrant behaviour is discussed. Next, the data available for training models is outlined. Then the method resulting from hyperparameter tuning is presented. Furthermore, the results obtained from applying this method to all the data sets are shown. Lastly, the performance of the developed method on propellers from Boskalis' fleet is discussed.

State of the art

In Section 2.3 the state of the art of models describing four-quadrant behaviour is discussed. Much work has been done in developing models that simulate the physical processes of water interacting with a propeller. Some promising results were achieved using an FNN to predict propeller behaviour. The choice was made to expand on this work and investigate and compare different machine learning methods on their ability to predict the four-quadrant behaviour of propellers.

Data

Five different data sets were available to train the model which are described in detail in Chapter 3. Three data sets for B-series propellers, one where each of the following three propeller characteristics were varied: pitch ratio, number of blades and expanded area ratio; and two data sets for Ka-series propellers were the pitch ratio is varied for two different duct types. For the B-series propellers, two coefficients were measured during training: the thrust coefficient C_T and the torque coefficient C_Q. For the Ka-series data sets a third coefficient was measured, the thrust nozzle coefficient C_{TN}. For each combination of varied characteristic and coefficient, there is a data set describing a distinct behaviour. For this reason, an individual model needs to be trained on each one of these data sets.

The data set available per model is relatively small with 360 values for β and one distinct value for the propeller characteristic. To provide more information for the model to use in learning the four-quadrant behaviour the data was augmented. Due to the cyclical nature of the behaviour, it was augmented with different terms of $\sin(\beta)$ and $\cos(\beta)$.

Hyperparameter tuning

Different machine learning methods have been investigated for developing a model for this research with hyperparameter tuning and ensemble methods have proven to be the best performing. The hyperparameter tuning was performed on the B-series pitch ratio variation C_T. During hyperparameter tuning, the score of a method was calculated as the sum of the MSE of all tested pitch ratios. Because the goal is to predict propellers used in Boskalis' fleet a relevant range of pitch ratios corresponding to pitch ratios of propellers in use was determined. The pitch ratios in this range were tested with hyperparameter tuning.

B-series results

The method resulting from the hyperparameter tuning was able to produce good predictions on the B-series pitch ratio variation data set for the thrust coefficient C_T. The overall score for this prediction was MSE $= 0.0052$. Since this score was minimized with hyperparameter tuning it is to be expected that the performance of the method will be the best on this data set.

The method was also used to train models for the other data sets and coefficients. The overall score for the torque coefficient C_Q was MSE $= 0.00251$. However, the torque coefficient shows more fluctuations in comparison to the thrust coefficient C_T resulting in a flattening of the true behaviour where the fluctuations are not registered. The largest deviations between the prediction and the true values are around $\beta = 270°$. This is also reflected in the local error score $\text{MSE}_{270°} = 0.00757$.

On the data set where the number of blades Z is varied, the model was able to accurately predict the overall behaviour of the propeller for both coefficients. The four-quadrant behaviour of the measured propellers deviates mostly in the areas around $\beta = 90°$ and $\beta = 90°$. This is is reflected in the scores where the score over full β range is MSE $= 0.00225$ and the local scores $\text{MSE}_{90°} = 0.00309$ and $\text{MSE}_{270°} = 0.00493$.

The prediction results for the B-series EAR variation data set were worse than the other two propeller characteristics with a score of MSE $= 0.01557$. The variation between the different propellers is the greatest for this data set. Over the full range, there are slight deviations between the prediction and the true values. At the peak around $\beta = 270°$ the deviations are so large that the values predicted for one propeller cross the true values of other propellers, which did not happen for the other models. The error for the area around $\beta = 270°$ is $\text{MSE}_{270°} = 0.06857$. This indicates that in this range using the model to predict the propeller behaviour is likely to give less accurate results than using the behaviour of a propeller whose characteristics lie closest to the propeller whose behaviour is desired.

Ka-series results

Results for the Ka-series pitch ratio variation are most interesting for the company because the Ka-series propeller with a type 19a nozzle is most similar to the propellers Boskalis has in use. The results were comparable to the results obtained from the B-series pitch ratio variation data set. The notable difference is that in the peak at $\beta = 270^\circ$ the prediction for $P/D = 1.2$ is further off. This can be explained due to the fact that here the model is aiming to extrapolate what the behaviour should be and this is more difficult than when the model has to interpolate the prediction. This difficulty in extrapolating the prediction is also seen when reviewing the MSE scores of the individual propellers where the scores for each individual test condition are $\text{MSE}_{P/D=0.8} = 0.00140$, $\text{MSE}_{P/D=1.0} = 0.00130$, and $\text{MSE}_{P/D=1.2} = 0.00483$. This means that the test value used during training should be chosen with this in mind.

The results obtained for the torque coefficient are similar to those of the B-series pitch ratio variation data set with $\text{MSE}_B = 0.00251$ and $\text{MSE}_{Ka} = 0.00405$. The overall behaviour is accurately captured by the model however in certain regions with higher fluctuations in the behaviour the model flattens the predictions and some information is lost. The prediction on the thrust nozzle coefficient C_{TN} similarly is able to capture the general behaviour with a score of $\text{MSE} = 0.00291$ and shows deviations in the area with the most rapid changes around $\beta = 270^\circ$.

Fleet propeller results

The model obtained from training on the Ka-series 19a nozzle data set was used to predict the behaviour of propellers with P/D values not present in the data. Therefore the predictions made on propellers present in the fleet can not be compared to a true value, since this is unknown. However, by plotting them on the known values an indication is made of the reliability of these predictions. For each prediction, the prediction lies nearest to the curve corresponding to the nearest pitch ratio. This indicates that no overfitting occurs on the P/D values seen during training and the model scales the prediction in a manner that corresponds with what would be expected.

It should be taken into account that the propellers in the data set available all had an EAR value of 70.0. Data on the influence of varying EAR is available for the B-series propeller. Therefore, to gain some understanding of the size of this effect a model was trained for both the pitch ratio and the expanded area ratio test for the B-series propeller. Predictions made using a model trained on both characteristics and on only the pitch ratio variation were compared to gauge the effect of this assumption. This test shows that the difference between the prediction increases as the difference between the true EAR value and the assumed value grow.

8 Discussion

Based on the literature the choice was made to generate a prediction for the four-quadrant behaviour of propellers using machine learning methods. The best working method proved to be the ensemble method. The developed model was able to provide generally good predictions of the four-quadrant behaviour of the relevant propellers and the results are of sufficient quality for the company.

One of the challenges faced is that there is little information for the model to learn the distinct behaviour. In the original data, only the hydrodynamic pitch angle β and the propeller characteristic were available to learn the behaviour from. Different augmentations based on $\sin\beta$ and $\cos\beta$ were added to the data to enrich the data set. Although this helps the model learn the behaviour better, the only information the model has to distinguish between different propellers in the data set are the varying propeller characteristics which are constant for one propeller. To limit the options only a set amount augmentation options were created. However, since using all the augmentation options gave the best results it would be interesting for future research to expand the augmentation options further.

Another method of expanding the data set that was explored was to increase the number of points at which the Fourier series the data stems from was evaluated. The current data sets are evaluated at 1-degree intervals resulting in 360 data points per measured propeller. The option of evaluating the Fourier series 10 and 100 times per degree has been investigated. However, this did not increase the accuracy of the predictions. This is likely because although more data was created, the amount of information did not increase.

The results indicate that the developed model can provide more information about the four-quadrant behaviour of propellers currently used in Boskalis' fleet. Although there are areas where the model fails to capture the exact value of the four-quadrant behaviour the areas of lower accuracy correspond with points of operation that are rarely used during the operation of a dredging ship. The model is not able to provide an exact prediction of the four-quadrant behaviour. However, the quality of the predictions is sufficient considering the small data set available for training.

There is an overall trend for all the developed models that the prediction quality is lowest around $\beta = 270°$. This area corresponds to the transition from the third quadrant to the fourth quadrant. Consulting Table 2.3 shows us that this corresponds to the ship changing from accelerating backwards while moving backwards to accelerating forwards while moving backwards. This is an area of operation that is rarely used even for dredging operations. Therefore the poor performance in this region will not greatly affect the effectiveness of the model when in use by the company.

A trend could also be noted that the models perform much better when the behaviour they aim to predict lies between characteristic values where the four-quadrant behaviour

is known. This means that with the current data available the range for which accurate predictions can be generated is limited. If Boskalis would purchase new propellers with characteristics outside of the data set it would not be possible to generate a reliable prediction. If the data set were expanded the current method would likely be able to create accurate predictions for a larger range of propeller characteristics.

The generalization is limited by the small data set available. The differences in data between different propellers are only the discreet propeller characteristic values. This means there is little information for the model to learn the differences between the varying propellers. For future work, the information obtained from this research could be combined with live data obtained from the ships. However, the data sets used in this research are obtained from lab experiments where environmental conditions are kept steady. The temperature of the water during dredging operations is likely to influence the interaction between the water and the propeller. Therefore when expanding this research to include live data this effect would need to be accounted for.

A Results

A.1 Results hyper parameter tuning

A.1.1 B-series

Mean Squared error scores						
Coefficient	Thrust coefficient C_T			Torque coefficient C_Q		
propeller characteristic	$\beta = [0:360]$	$\beta = [75:105]$	$\beta = [255:285]$	$\beta = [0:360]$	$\beta = [75:105]$	$\beta = [255:285]$
P/D	0.00052	000069.	0.00078	0.00251	0.00757	0.00257
EAR	0.01557	0.06857	0.01830	0.01648	0.03824	0.03561
Z	0.00225	0.00493	0.00309	0.00351	0.00224	0.00492

Table A.1: Final mean squared error scores B-series data sets for entire range and transition areas

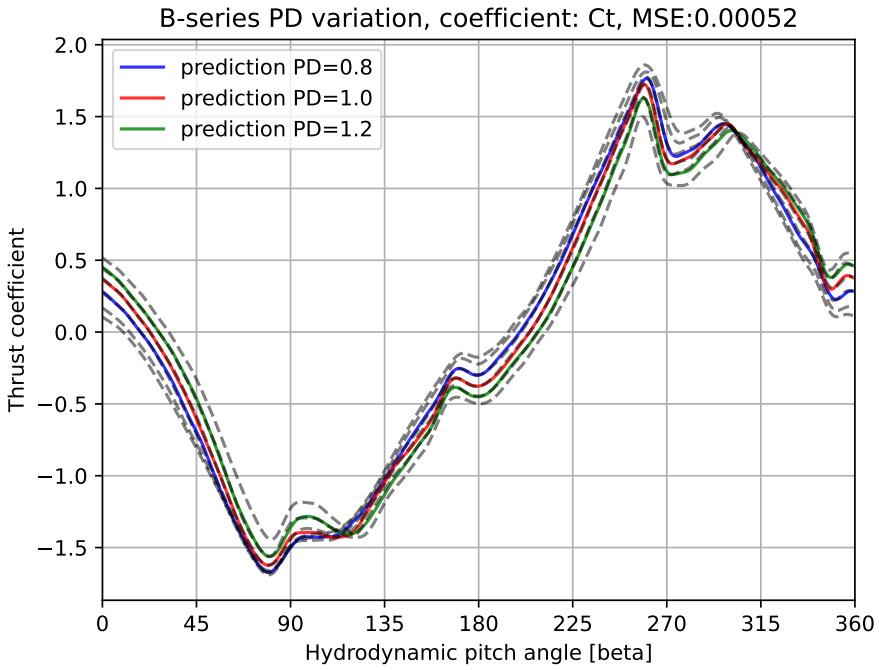


Figure A.1: B-series pitch ratio variation thrust coefficient C_T.

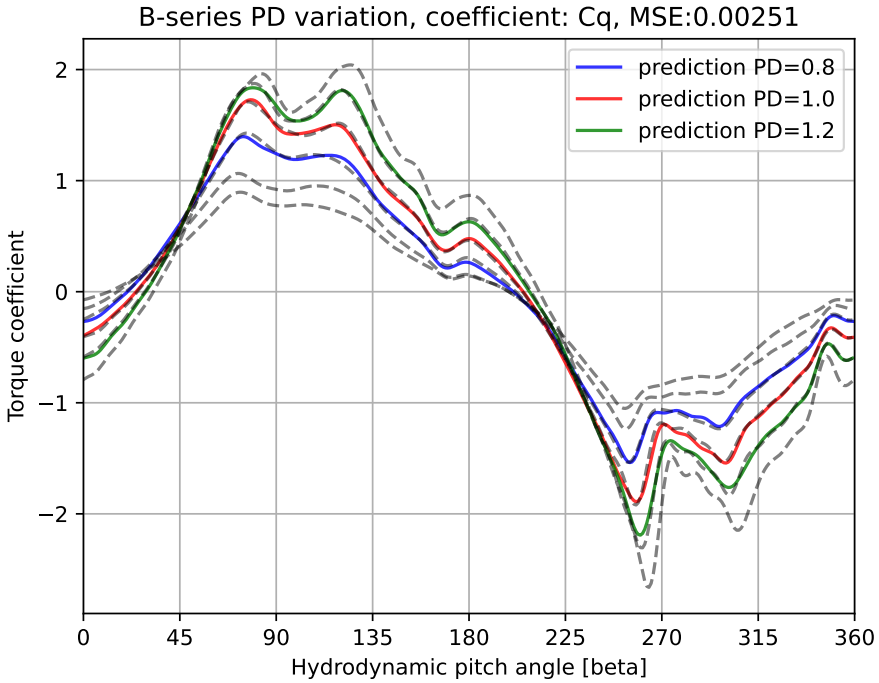


Figure A.2: B-series pitch ratio variation torque coefficient C_Q.

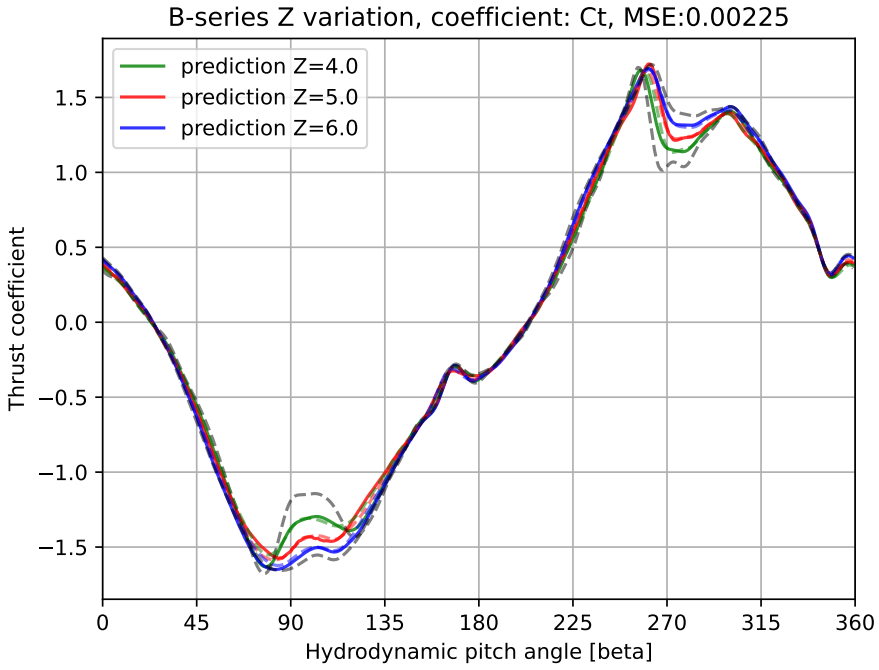


Figure A.3: B-series number of blades variation thrust coefficient C_T.

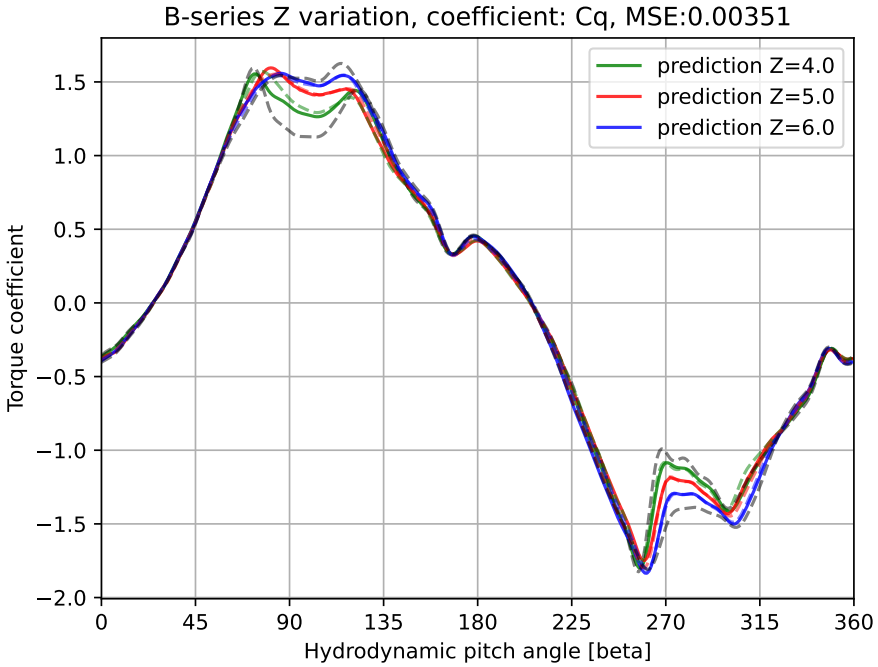


Figure A.4: B-series number of blades variation torque coefficient C_Q.

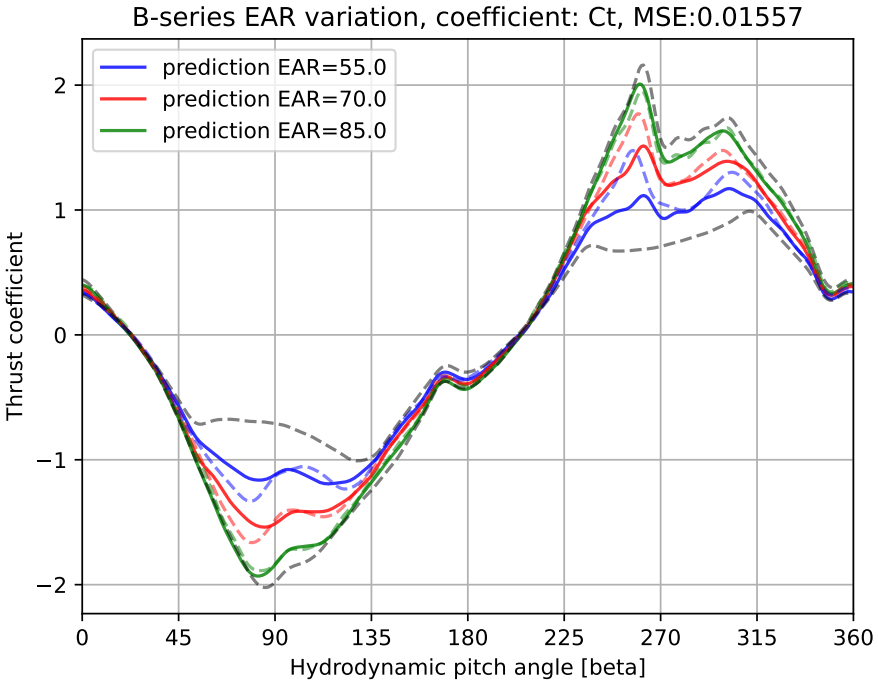


Figure A.5: B-series expanded area ratio variation thrust coefficient C_T.

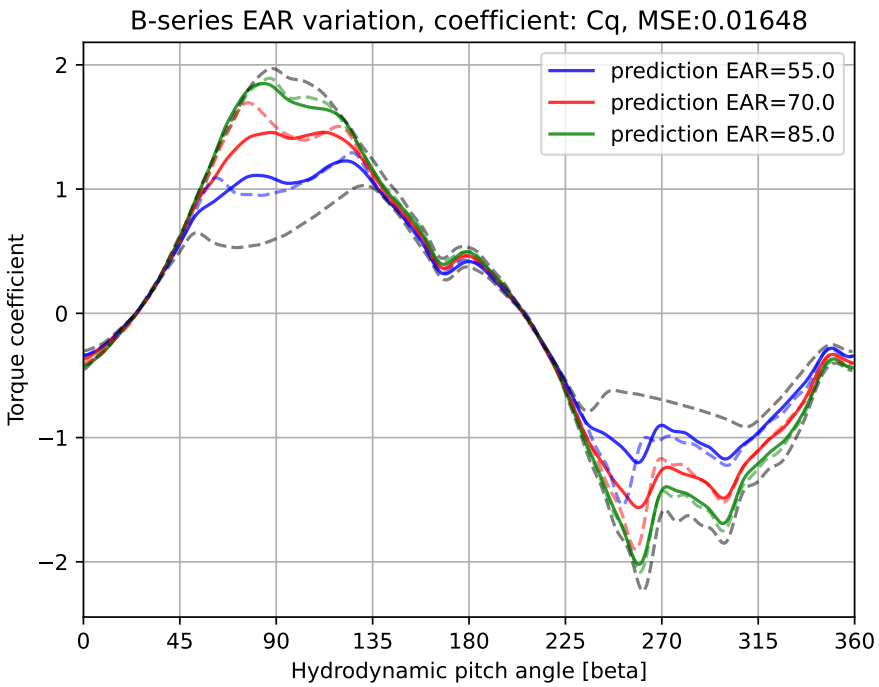


Figure A.6: B-series expanded area ratio variation torque coefficient C_Q.

A.1.2 Ka-series

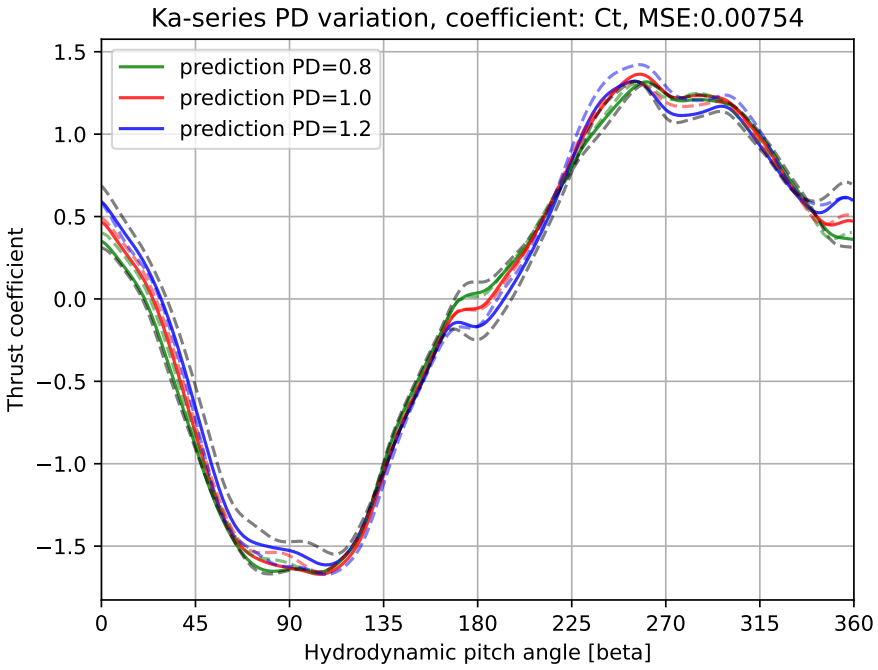


Figure A.7: Ka-series 19a nozzle pitch ratio variation thrust coefficient C_T

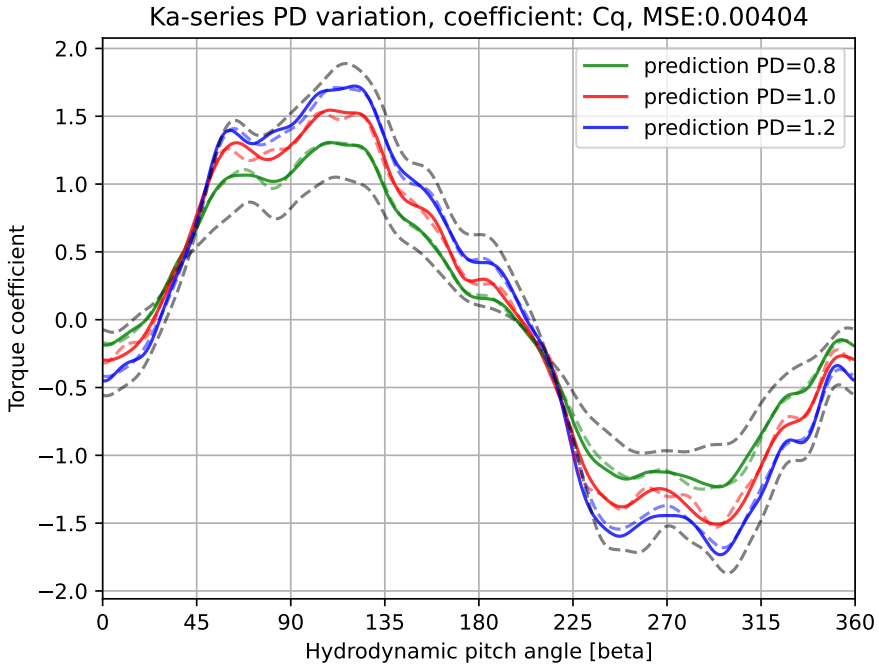


Figure A.8: Ka-series 19a nozzle pitch ratio variation thrust coefficient C_Q

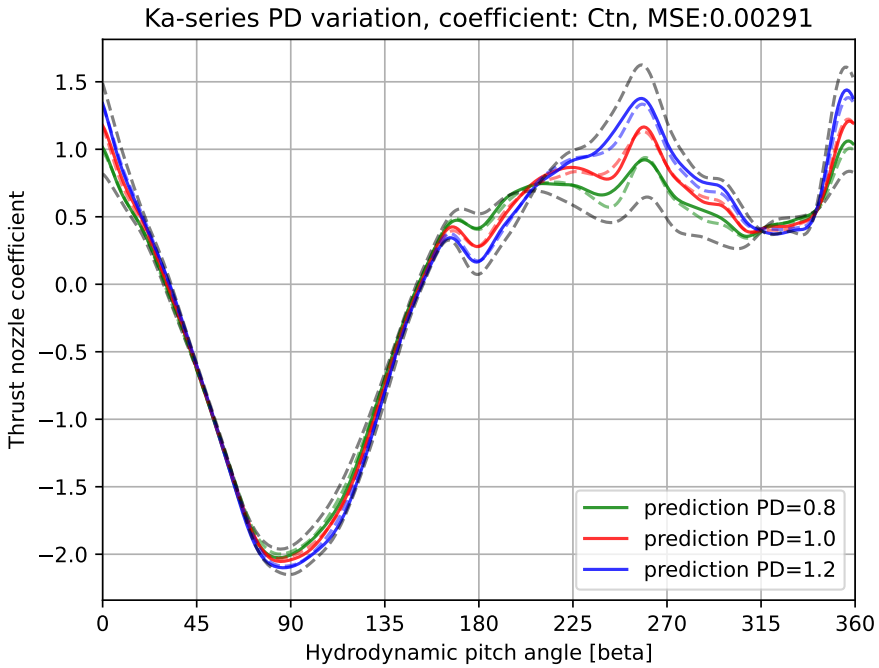


Figure A.9: Ka-series 19a nozzle pitch ratio variation thrust coefficient C_{TN}

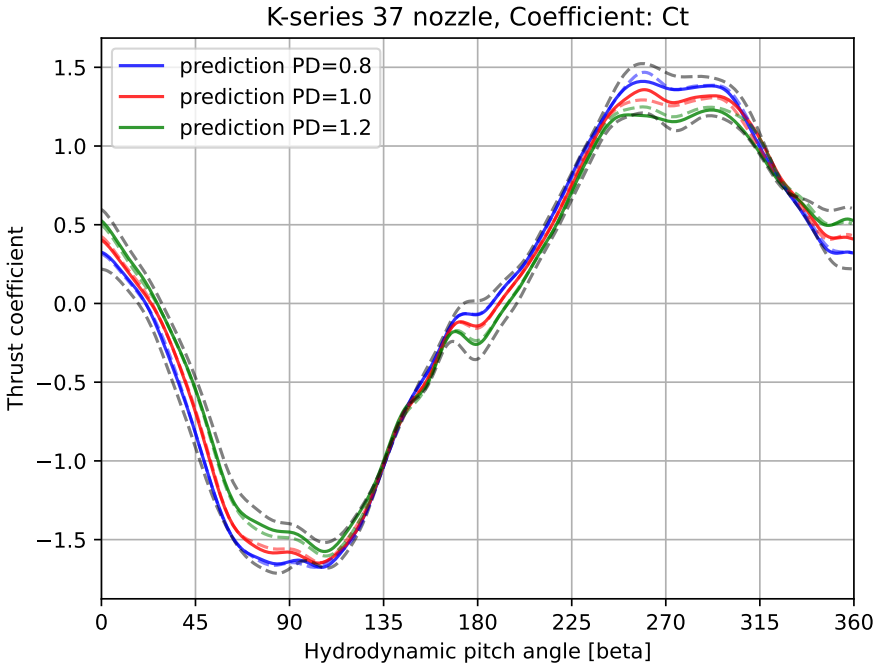


Figure A.10: Ka-series 37 nozzle pitch ratio variation thrust coefficient C_T

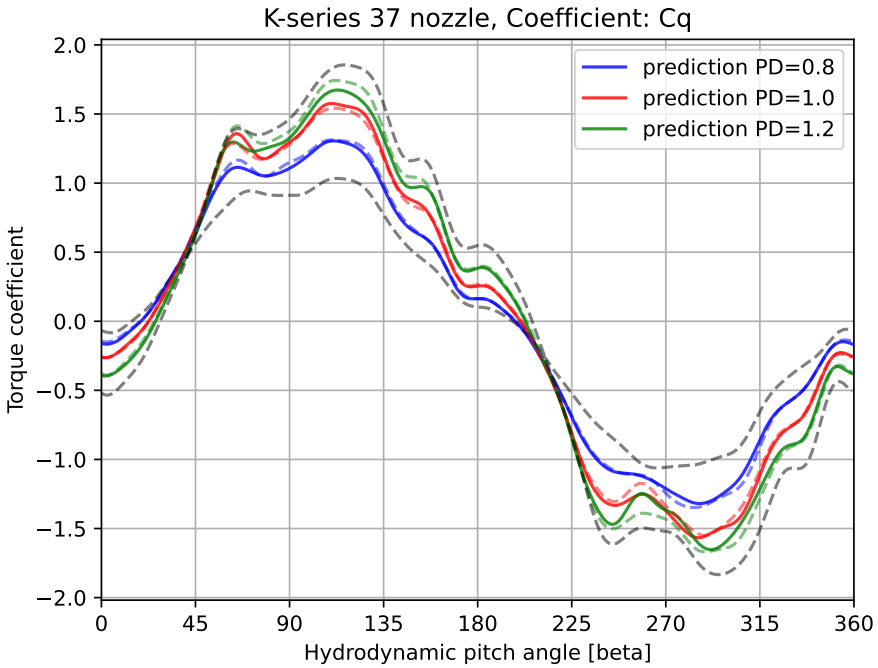


Figure A.11: Ka-series 37 nozzle pitch ratio variation thrust coefficient C_Q

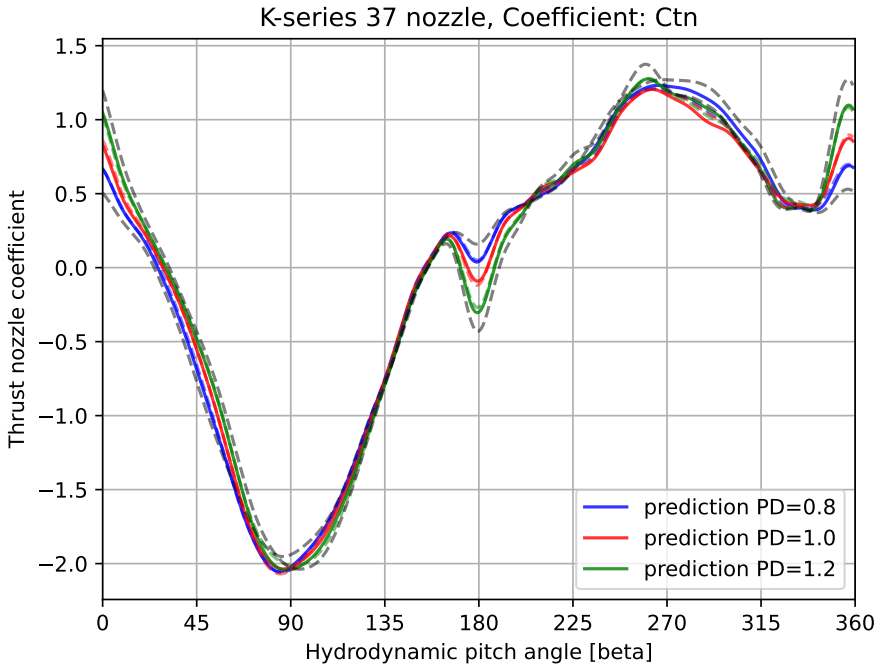


Figure A.12: Ka-series 37 nozzle pitch ratio variation thrust coefficient C_{TN}

A.1.3 Fleet propellers

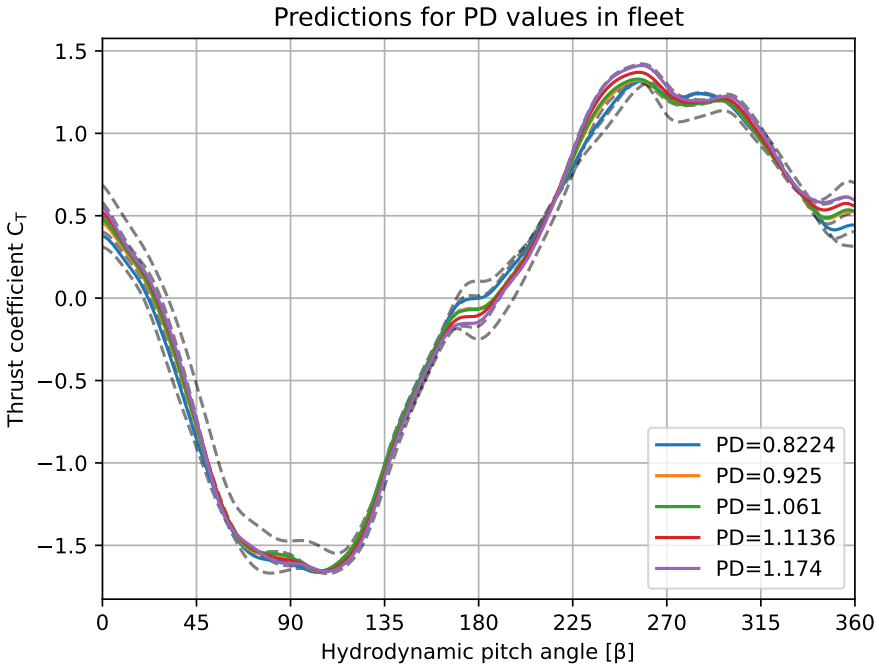


Figure A.13: Predictions of the thrust coefficient of all the propellers in the fleet.

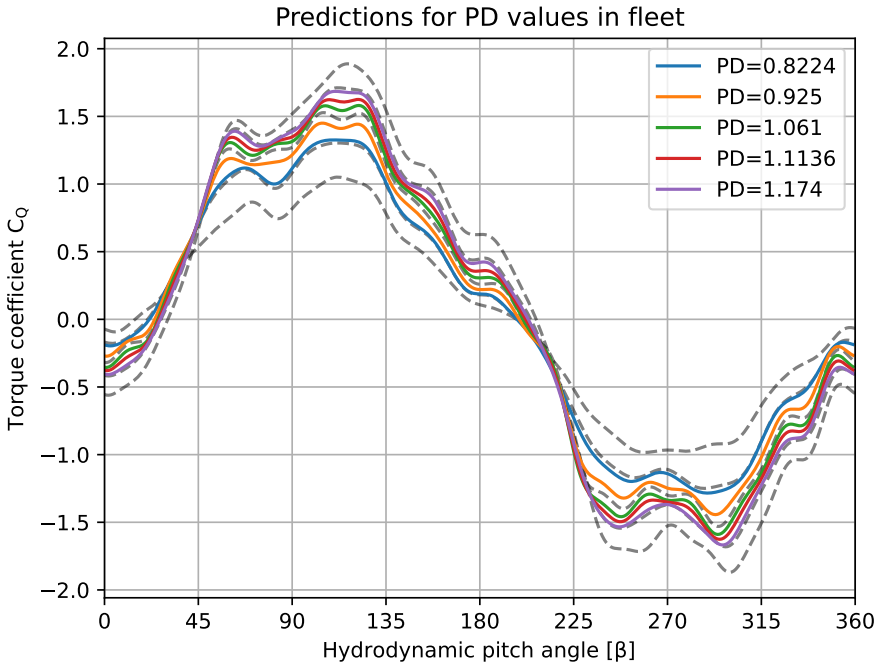


Figure A.14: Predictions of the torque coefficient of all the propellers in the fleet.

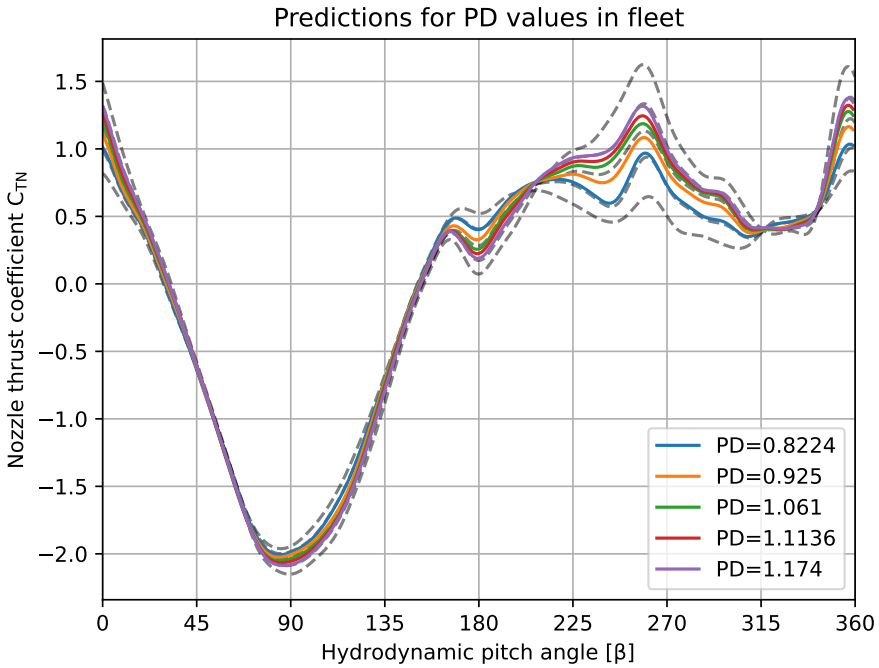
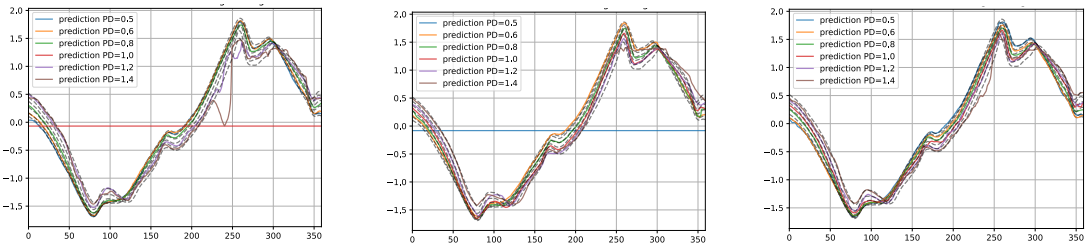


Figure A.15: Predictions of the thrust nozzle coefficient of all the propellers in the fleet.



(a) Augmentation used in Roddy [65]

(b) Augmented with $\sin(\beta)$ and $\cos(\beta)$

(c) All augmentation options from Table 3.3

Figure A.16: Predictions using different combinations of augmentation

A.2 Augmentation variation

For this research the different ways of augmenting the data have been investigated. Due to the cyclical nature of the four quadrant behaviour is augmented with sinusiods $\sin \omega\beta$ and $\cos \omega\beta$. The values of ω implemented are [0.1, 0.25, 0.5, 1, 2, 3, 5, 10]. The amplitude of the sinusoids is not varied since the data is normalized and this would nullify the differences between sinusiods with equal period and different amplitudes. Figure A.19 shows three different augmentation variations side by side. The first tested variation shown in Figure A.16a uses the same augmentation that was applied in the research by Roddy et al. [65]. Here the added values are [$\cos\beta$, $\cos 5\beta$, $\sin\beta$, $\sin 5\beta$, $\sin 10\beta$]. The second tested variation shown in Figure A.16b only adds the sine and cosine of β. Meaning the added values are [$\cos\beta$, $\sin\beta$]. Lastly all the augmentation options listed in Table 3.3 were added to the data sets. Figure A.16c shows the results from this test. It can be noted that prediction set using the full augmentation range provides more reliable results. Therefore all the possible columns listed in Table 3.3 will be added to the training data.

Test value PD	Split values				Predicted PD value
0.5	0.70	0.90	1.10	1.30	0.6
0.6	0.65	0.90	1.10	1.30	0.5
0.8	0.55	0.80	1.10	1.30	1.0
1.0	0.55	0.70	1.00	1.30	0.8
1.2	0.55	0.70	0.90	1.20	1.4
1.4	0.55	0.70	0.90	1.10	1.2

Table A.2: All PD values used for each decision tree for each condition.

A.3 Decision tree regression

The decision tree regression method was unable to effectively capture the behaviour of the test data. It would accurately predict the behaviour of a neighboring plot. This behaviour can be explained when reviewing the P/D values present in the decision tree. For this end the decision tree was exported as a text file and with Regular Expression all the P/D values used in decision making for one test value were collected. Table A.2 provides an overview of all the values used in decision making by the regression tree. It can be noted that the test value not being available during training causes the model to view it as equal to one of the neighboring training values. It can be concluded that the Leave One Group Out method and decision tree regression are an unsuitable match for train/test split and algorithm.

A.4 Combined pitch ratio and expended area ratio data set

One of the inaccuracies inherent to the method of this research is that in each data set only a single propeller characteristic is varied. This means that for all predictions made on the propellers in Boskalis' fleet, the EAR is assumed to be 70.0 while this is not true. There is no data available for the effect of varying EAR values for the Ka-series propellers. However, this is available for the B-series propellers. To investigate the influence of this discrepancy a model has been trained on both the data set varying the pitch ratio and the expanded area ratio. The propeller with $P/D = 1.0$ and $EAR = 70.0$ can be found in both these sets, therefore this propeller was used as the test set during training. By comparing the predictions made using a model trained on solely the pitch ratio variation data set and using the combined data set the effect of this discrepancy can be gauged and visualized. Figure A.19a shows the prediction for the propeller whose EAR values vary the most from 70.0. The difference between both predictions is significant in the regions around $\beta = 90°$ and $\beta = 270°$. These regions correspond to the transitions from the first to the second quadrant and the third to the fourth quadrant. As mentioned in Chapter 6 the region around $\beta = 270°$ is not used much in operations. However, the region around $\beta = 90°$ is used more often. Therefore the effect of this difference in prediction seems significant. This would mean that to improve the prediction quality the effects of the EAR should be further investigated. The graphs of the other propellers and especially Figure A.18b show that with the reduced difference between the actual EAR value and the standard value of EAR = 70.0 the difference is reduced.

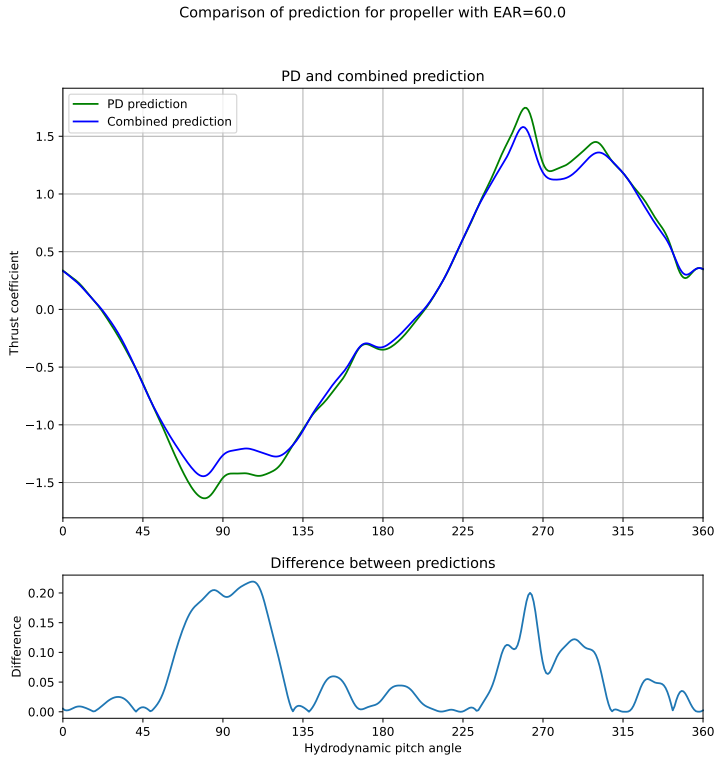
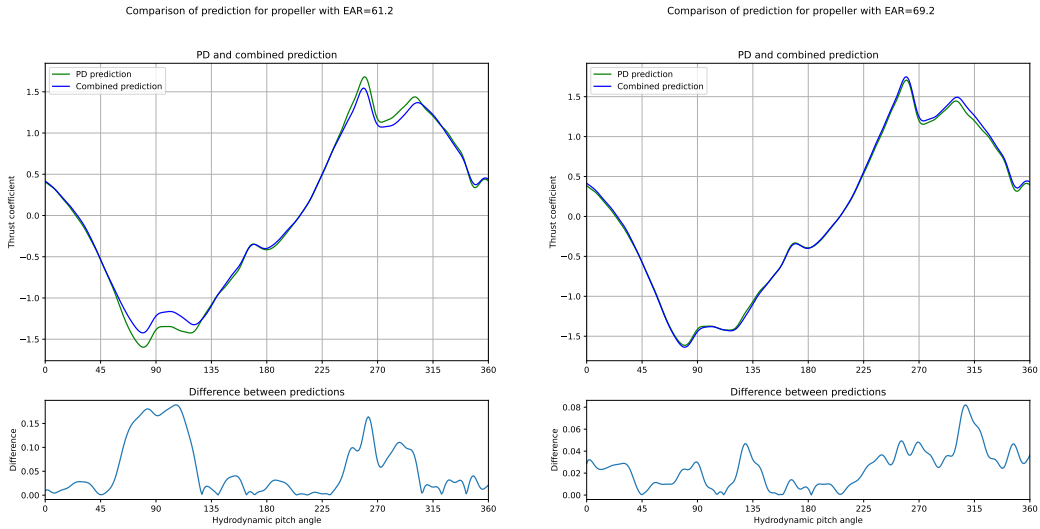
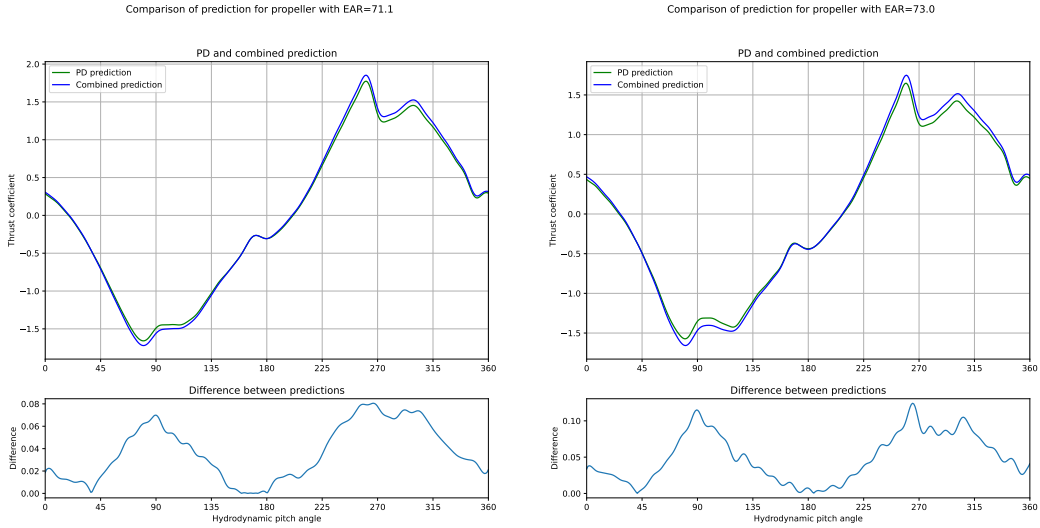


Figure A.17: Prediction using both P/D data set and combined data set along absolute difference



(a) (b)

Figure A.18



(a) (b)

Figure A.19: Predictions using models trained on interpreting P/D and both P/D and EAR for propellers with (a) EAR $= 71.1$ (b) EAR $= 73.0$

B Fourier coefficients

This section contains the coefficients used to generate the different curves describing the four quadrant behaviour obtained from Roddy et al. [65]. Above each section of tables a figure shows the curves resulting from these coefficients.

B.1 Pitch ratio variations

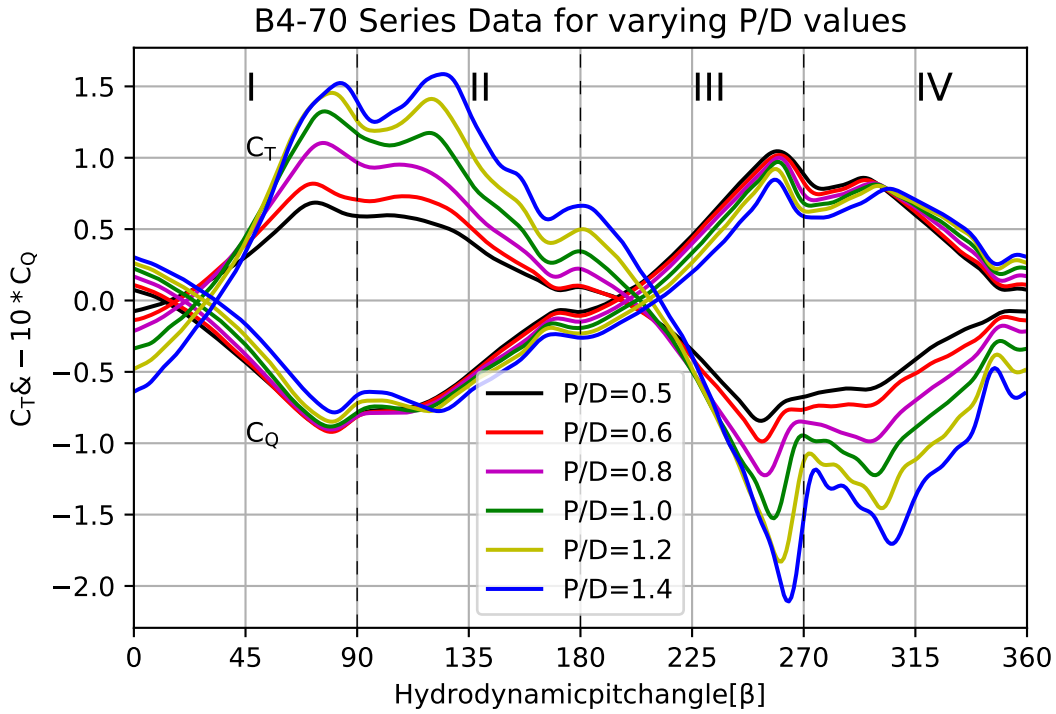


Figure B.1: Four quadrant behaviour for varying P/D values

B4-70 P/D=0.5				
K	C_T A(K)	C_T B(K)	C_Q A(K)	C_Q B(K)
0	5,0001000	0,0000000	-3,2792000	0,0000000
1	5,8760000	-80,2710000	-5,2552000	61,3740000
2	-2,2733000	-1,2337000	2,6413000	-1,1267000
3	2,1575000	11,7620000	-3,4908000	-9,8851000
4	-0,5652700	-1,1263000	-0,1958100	1,1875000
5	-3,9577000	3,9944000	3,1287000	-4,0676000
6	0,4550900	0,8214800	-0,5658100	-0,2152600
7	3,2426000	-0,4088200	-2,0201000	2,0145000
8	-0,4235000	-0,5753900	0,4460600	0,4514600
9	-2,0298000	1,7398000	0,3988000	-1,8924000
10	-0,0023261	0,8361200	0,0355200	-0,2137000
11	1,6667000	-0,0792280	0,3771600	0,2695100
12	0,0962520	-0,5206900	-0,1024100	-0,0090780
13	-0,3764000	0,5858500	-0,8405500	-0,5815200
14	-0,1371000	0,2169700	0,1643200	0,3479300
15	0,6294100	-0,0343270	0,3186400	-0,0974720
16	0,0690470	-0,0931120	-0,0053692	-0,3577100
17	0,1570500	0,3008700	-0,5951100	0,1575600
18	0,1387900	-0,0686190	0,1232500	0,1497200
19	0,2337300	-0,2555500	-0,0228040	-0,2533900
20	0,0537380	-0,0921720	-0,1403700	-0,0371720
21	0,0823360	0,0741180	-0,1102600	0,2051600
22	0,0459030	-0,0387210	0,1523700	-0,0700220
23	0,1006000	-0,2707000	-0,2786300	-0,0550530
24	-0,0105240	-0,1225700	-0,0702560	0,1088300
25	-0,0374950	-0,1190800	0,0255000	0,1739100
26	-0,1201900	0,0347080	0,0526020	-0,0815790
27	0,0209320	-0,0842200	-0,1246600	0,0885180
28	-0,0099185	-0,0621350	0,0629620	0,0587710
29	-0,0563700	-0,0991820	0,0152530	0,0712920
30	-0,1442700	0,0389180	0,0084550	-0,0123610

Table B.1: Fourier coefficients for the B4-70 P/D=0.5 propeller

B4-70 P/D=0.6				
K	C_T A(K)	C_T B(K)	C_Q A(K)	C_Q B(K)
0	3,8804000	0,0000000	-3,0251000	0,0000000
1	8,3653000	-79,3810000	-8,3989000	73,2850000
2	-0,8585900	-1,3907000	0,7169200	-0,4340000
3	2,3607000	11,2510000	-4,0790000	-10,1880000
4	-1,0596000	-0,7885100	-1,1812000	1,8683000
5	-4,1596000	4,2391000	3,7134000	-5,4538000
6	0,4624700	0,9490200	-0,0596700	-0,4419300
7	3,3330000	-0,4307700	-1,6205000	2,4562000
8	-0,3709800	-0,9058100	0,6641900	-0,1915100
9	-1,9768000	1,6585000	0,5094700	-2,8162000
10	-0,1197200	0,8629700	-0,4415300	-0,4921300
11	1,7570000	-0,0790010	0,0147050	0,2699600
12	0,0276790	-0,6021600	0,0804090	-0,0663350
13	-0,4321200	0,8018500	-1,2038000	-0,8003600
14	0,0289390	0,3434400	-0,1036200	0,5180700
15	0,7800900	-0,1632700	0,3769400	-0,2100800
16	-0,0080617	-0,0030060	-0,0401150	-0,7603800
17	0,1942400	0,4405300	-0,9872600	0,2417100
18	0,1205600	0,0549530	0,0821720	0,3479300
19	0,3516800	-0,4002500	-0,0707440	-0,2611500
20	-0,0670990	-0,1760000	-0,1032600	-0,0134310
21	0,1216700	0,1302900	-0,1104600	0,4414700
22	0,1606500	-0,0014661	0,2258300	-0,0735100
23	0,1855500	-0,3040600	-0,3004200	-0,1719400
24	-0,0527390	-0,1716500	-0,2868800	0,1044300
25	-0,0389860	-0,0314440	0,1191100	0,4540800
26	-0,0851510	0,0225160	0,1992700	-0,0890830
27	0,0209610	-0,1070500	-0,1681400	-0,0153840
28	-0,0515490	-0,0956160	0,0230930	0,1710600
29	-0,0830520	-0,0710080	0,2009200	0,0065482
30	-0,0389420	0,0502500	-0,0280480	-0,1121800

Table B.2: Fourier coefficients for the B4-70 P/D=0.6 propeller

B4-70 P/D=0.8				
K	C_T A(K)	C_T B(K)	C_Q A(K)	C_Q B(K)
0	2,8757000	0,0000000	-2,5783000	0,0000000
1	13,0070000	-77,7620000	-15,7270000	93,0800000
2	0,6286900	-1,3032000	-0,6896000	-0,2048200
3	2,8104000	10,5800000	-5,4402000	-13,7860000
4	-1,6832000	-1,0413000	2,4674000	1,5588000
5	-4,4894000	4,5769000	5,0717000	-6,8953000
6	0,3770300	1,4296000	-1,1964000	-0,1977300
7	3,1647000	-0,3961200	-3,1161000	3,1295000
8	-0,1349200	-0,9917000	1,5838000	-0,0113660
9	-1,9905000	2,0759000	1,4925000	-4,0967000
10	0,1757600	0,9015300	-0,9089800	-0,3281000
11	1,9750000	-0,0298770	-1,2135000	0,7953900
12	-0,1106300	-0,5635400	0,5012600	-0,2662400
13	-0,0698280	0,9477300	-1,4143000	-1,4536000
14	0,2566500	0,2368100	-0,1887300	0,6866600
15	0,8068100	-0,5320400	0,1542200	0,4549900
16	-0,1878100	-0,0715370	0,1729400	-0,8479200
17	0,2982100	0,5276800	-1,3330000	-0,1591100
18	0,4864300	0,0174000	-0,2262000	0,5973300
19	0,3389300	-0,6810500	0,1429900	0,2544500
20	-0,2685100	-0,2040000	-0,0453350	-0,2199100
21	0,0960980	0,1408600	-0,4005100	0,4753500
22	0,2145000	-0,0579140	0,2062900	0,1539700
23	0,0664400	-0,4489800	-0,0697050	-0,0736920
24	-0,1321000	-0,1145400	-0,1825400	-0,0226540
25	0,0163780	-0,0120050	-0,0563070	0,4725600
26	0,0123670	-0,1193200	0,2428200	0,1306100
27	-0,1611300	-0,2442700	0,0796060	-0,0613110
28	-0,1053600	0,0498820	-0,0695030	0,0097692
29	-0,0411390	0,0635870	0,0800750	0,1310200
30	-0,0089569	-0,0736110	0,0421560	-0,0190470

Table B.3: Fourier coefficients for the B4-70 P/D=0.8 propeller

B4-70 P/D=1.0				
K	C_T A(K)	C_T B(K)	C_Q A(K)	C_Q B(K)
0	2,66170	0,00000	-2,29450	0,00000
1	17,80200	-74,79700	-26,98900	110,38000
2	1,38280	-1,36320	-1,87580	-0,52852
3	2,78710	10,24600	-6,35520	-16,54400
4	-1,62400	-1,08650	2,16680	1,92900
5	-5,30840	4,73460	7,65300	-8,44800
6	0,19180	1,13550	-0,56516	-0,90714
7	3,83180	-0,93180	-6,26540	3,10420
8	-0,22892	-0,72471	1,92110	0,68648
9	-1,86560	2,37470	3,34560	-4,51960
10	0,34900	0,86208	-1,29560	-1,09280
11	2,23210	0,00702	-3,12940	-0,04497
12	-0,35976	-0,36422	1,24750	0,72404
13	0,13606	0,91758	-1,43510	-1,52840
14	0,47192	0,08938	-0,83201	-0,23983
15	0,87235	-0,59364	0,03487	1,60740
16	-0,35748	-0,07095	0,89981	-0,66184
17	0,35955	0,48825	-1,42470	-1,20840
18	0,52896	-0,09059	-1,10640	0,58384
19	0,21045	-0,81163	0,27359	1,33970
20	-0,27057	-0,05666	0,67590	-0,29095
21	0,10509	0,12011	-0,57328	-0,08961
22	0,27733	-0,13042	-0,22609	0,54988
23	-0,11596	-0,48187	0,70691	0,14626
24	-0,17450	0,04541	-0,03023	-0,48380
25	0,04078	-0,01947	-0,46043	0,38709
26	0,09925	-0,21858	0,25629	0,72114
27	-0,34650	-0,16314	0,76092	-0,13767
28	-0,12357	0,14865	0,00262	-0,45823
29	0,06393	0,10185	-0,19316	0,10520
30	0,06193	-0,16735	0,31487	0,27547

Table B.4: Fourier coefficients for the B4-70 P/D=1.0 propeller

B4-70 P/D=1.2				
K	C_T A(K)	C_T B(K)	C_Q A(K)	C_Q B(K)
0	2,4003000	0,0000000	-3,9384000	0,0000000
1	22,7620000	-71,0420000	-41,1140000	125,7800000
2	1,7143000	-1,2393000	0,5601500	-0,3017200
3	2,1316000	9,2564000	-6,7469000	-19,1930000
4	-1,4846000	-1,3138000	0,7647200	3,0528000
5	-5,9120000	4,9260000	10,2590000	-9,3530000
6	0,0258760	0,7860100	1,6558000	-1,6738000
7	4,2514000	-1,0738000	-9,7787000	2,4956000
8	-0,2876500	-0,3706800	0,3306500	1,6433000
9	-1,8761000	2,8030000	5,0625000	-4,2091000
10	0,5088800	0,4825900	-1,2604000	-2,3535000
11	2,3192000	-0,0532710	-4,5527000	-1,0521000
12	-0,4099900	-0,3979900	0,9581800	1,7638000
13	0,6329500	0,9545600	-0,6588600	-1,4926000
14	0,6453000	0,0006267	-1,3658000	-0,5286100
15	0,5588500	-0,9918800	-1,4995000	2,1916000
16	-0,3096200	0,0319570	1,2968000	0,2960400
17	0,4741300	0,4270300	-0,1265600	-1,4969000
18	0,4005300	-0,3277400	-1,4665000	-0,2346200
19	0,0998380	-1,0834000	-1,1630000	2,1835000
20	-0,1816600	0,0180700	0,6702200	0,7094000
21	0,0408270	0,0384070	0,4820900	-0,4056500
22	0,0382620	-0,3169200	-0,4228200	0,4349300
23	-0,4191100	-0,3939500	0,3920900	1,0100000
24	-0,0832190	0,0662270	0,7166400	-0,1704000
25	0,0703260	-0,0293380	-0,2231200	-0,5032600
26	-0,1010700	-0,2219400	-0,4178800	0,6206400
27	-0,5417800	0,0080383	0,6539600	0,7371100
28	-0,0275070	0,1592600	0,9811000	-0,1759500
29	0,0424680	0,1505600	0,1032400	-0,8032700
30	-0,1378300	-0,0453210	-0,0833620	-0,0360840

Table B.5: Fourier coefficients for the B4-70 P/D=1.2 propeller

B4-70 P/D=1.4				
K	C_T A(K)	C_T B(K)	C_Q A(K)	C_Q B(K)
0	1,8891000	0,0000000	-6,2730000	0,0000000
1	28,0330000	-65,6830000	-59,0670000	137,1800000
2	1,8860000	-0,2306600	3,4811000	-0,0602010
3	0,5997300	7,4446000	-4,3989000	-20,5550000
4	-1,0858000	-1,2398000	-0,4052000	3,3632000
5	-6,4362000	4,7921000	11,8430000	-8,9586000
6	-0,1563100	0,7169500	3,0860000	-2,6638000
7	5,0311000	0,0342760	-13,1370000	-0,9819000
8	-0,0639240	-0,2943400	-1,4057000	2,2511000
9	-2,0315000	2,0101000	5,6834000	-0,8726900
10	0,6086600	0,2709700	0,2612900	-3,1696000
11	1,8917000	0,1648000	-4,5913000	-3,3156000
12	-0,4193500	-0,1798300	-0,5810700	2,1286000
13	0,6718100	1,0989000	-0,1235300	-1,0776000
14	0,6423000	-0,1632600	-1,6486000	-1,3561000
15	0,7271500	-1,3135000	-4,3524000	2,0842000
16	-0,1748300	-0,0030559	0,9560600	2,0193000
17	0,1754800	0,4661900	2,0063000	0,3301100
18	0,3548000	-0,3998800	-0,6776400	-1,0371000
19	0,0797000	-0,7461600	-2,0860000	1,3432000
20	-0,0991090	0,0185520	0,1002100	1,2629000
21	0,2201900	-0,0919380	1,4317000	0,3929200
22	-0,0475300	-0,4169700	-0,4024600	-0,1531600
23	-0,5246100	-0,3053900	-0,3476800	1,2626000
24	0,0534690	0,1421100	1,1234000	1,1060000
25	0,0995100	0,0634860	1,8220000	-0,7667600
26	-0,1619100	-0,2740200	-0,3422400	-0,7106300
27	-0,3416300	0,0092732	-0,6785600	0,0780990
28	0,0252120	0,1106400	0,8999000	0,7047900
29	0,0046286	-0,0177690	0,8462000	-0,6639400
30	-0,2043200	0,0064717	-0,0845670	-0,6184600

Table B.6: Fourier coefficients for the B4-70 P/D=1.4 propeller

B.2 Expanded area ratio variations

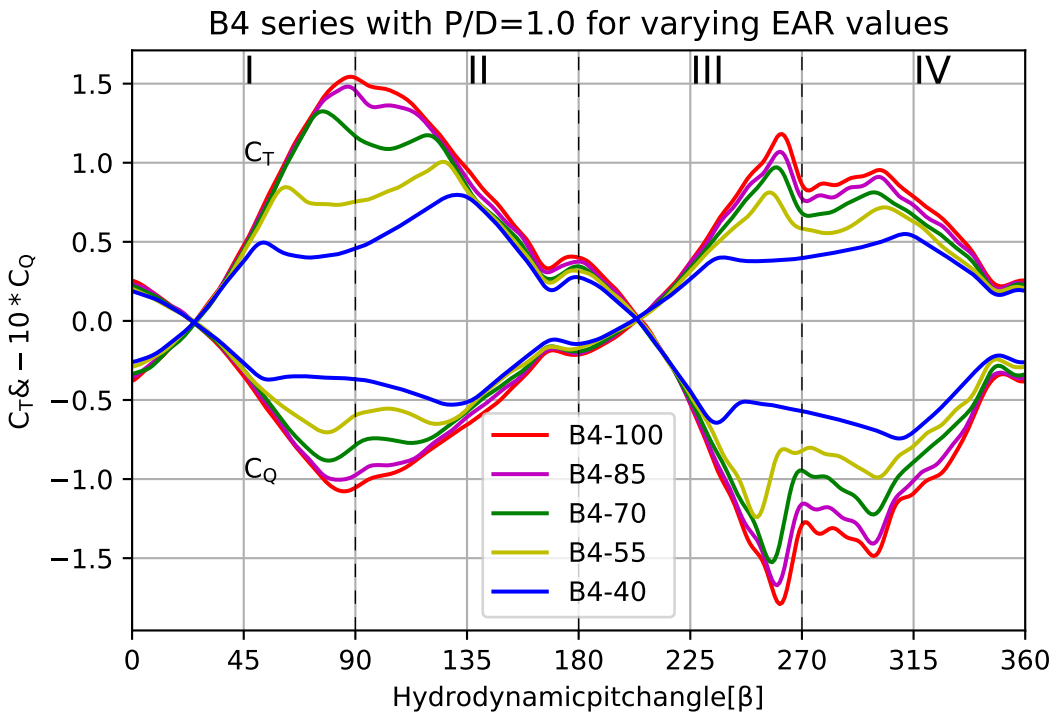


Figure B.2: Four quadrant behaviour for varying EAR values

B4-100 P/D=1.0				
K	C_T A(K)	C_T B(K)	C_Q A(K)	C_Q B(K)
0	3,63130	0,00000	-5,41520	0,00000
1	21,42000	-88,91100	-33,97500	130,80000
2	1,59280	-1,69650	0,53548	0,89413
3	1,84150	14,27300	-3,00960	-23,34100
4	-2,89710	-1,86550	3,91440	4,06150
5	-5,64760	2,68460	6,19930	-3,62920
6	1,30120	2,10620	-1,01950	-3,03960
7	3,20480	1,12800	-3,94800	-0,21287
8	-0,72200	-1,65580	2,21040	1,86640
9	-1,35870	1,66060	1,79210	-3,30240
10	0,30497	1,23110	-0,89254	-2,56670
11	2,51540	0,55586	-4,22710	-0,70419
12	-0,02497	-0,30974	-0,06220	1,22920
13	-0,22675	0,67583	-0,04903	-0,24159
14	0,71969	0,31051	-0,63836	-0,00044
15	1,28060	-0,70452	-1,04590	1,07990
16	-0,63334	-0,24519	1,24210	-0,27960
17	-0,18471	0,59265	-0,34991	-0,93174
18	0,75799	0,29080	-1,37590	-0,05908
19	0,71537	-0,73888	-0,66658	1,55530
20	-0,48378	-0,24769	1,14780	0,29850
21	-0,14651	0,23705	0,45079	-0,60358
22	0,45192	0,07872	-0,84318	-0,43329
23	0,10214	-0,59234	-0,17289	0,75830
24	-0,43181	0,01619	0,38746	-0,19583
25	0,00567	0,32714	-0,36456	-0,27982
26	0,42155	-0,03421	-0,65377	0,57960
27	-0,14898	-0,48710	0,67956	0,70315
28	-0,39834	0,05111	0,62507	-0,39551
29	-0,01311	0,31461	-0,31981	-0,68067
30	0,26745	-0,06266	-0,77599	0,40430

Table B.7: Fourier coefficients for the B4-100 P/D=1.0 propeller

B4-85 P/D=1.0				
K	C_T A(K)	C_T B(K)	C_Q A(K)	C_Q B(K)
0	2,61020	0,00000	-3,77070	0,00000
1	19,78000	-83,38500	-31,22000	123,61000
2	2,54730	-1,38420	-1,19200	0,26523
3	1,93370	13,21400	-3,72570	-21,48000
4	-3,02980	-1,46910	4,27760	3,62570
5	-5,07970	3,38520	6,29120	-5,10650
6	0,96050	1,59360	-1,36150	-2,54140
7	3,20350	-0,02504	-4,63510	1,16680
8	-0,48522	-1,11150	2,04310	1,54370
9	-1,59270	1,98010	2,70370	-3,61520
10	0,06663	0,97636	-1,04690	-1,76560
11	2,28780	0,41025	-4,17300	-0,51881
12	0,00112	-0,27847	0,36137	0,87759
13	0,07396	0,74872	-0,28369	-0,52016
14	0,59500	0,09564	-0,78457	0,17533
15	1,06090	-0,55969	-0,95078	1,12170
16	-0,63953	-0,06817	1,38380	-0,56001
17	0,08212	0,49803	-0,51436	-0,84591
18	0,78204	0,14477	-1,41850	0,30022
19	0,57090	-0,78111	-0,34278	1,56160
20	-0,36914	-0,19997	1,09120	-0,01842
21	-0,03289	0,18140	0,05287	-0,42394
22	0,33895	-0,03431	-0,64971	0,12852
23	-0,11647	-0,55144	0,24277	0,55345
24	-0,25801	0,06329	0,25687	-0,39165
25	0,06205	0,19735	-0,41882	-0,03078
26	0,28105	-0,20256	-0,33611	0,75088
27	-0,33189	-0,37134	0,78826	-0,30283
28	-0,91200	0,16925	0,39635	-0,69050
29	0,06216	0,27936	-0,49340	-0,25673
30	0,15537	-0,15245	-0,26213	0,62653

Table B.8: Fourier coefficients for the B4-85 P/D=1.0 propeller

B4-70 P/D=1.0				
K	C_T A(K)	C_T B(K)	C_Q A(K)	C_Q B(K)
0	2,66170	0,00000	-2,29450	0,00000
1	17,80200	-74,79700	-26,98900	110,38000
2	1,38280	-1,36320	-1,87580	-0,52852
3	2,78710	10,24600	-6,35520	-16,54400
4	-1,62400	-1,08650	2,16680	1,92900
5	-5,30840	4,73460	7,65300	-8,44800
6	0,19180	1,13550	-0,56516	-0,90714
7	3,83180	-0,93180	-6,26540	3,10420
8	-0,22892	-0,72471	1,92110	0,68648
9	-1,86560	2,37470	3,34560	-4,51960
10	0,34900	0,86208	-1,29560	-1,09280
11	2,23210	0,00702	-3,12940	-0,04497
12	-0,35976	-0,36422	1,24750	0,72404
13	0,13606	0,91758	-1,43510	-1,52840
14	0,47192	0,08938	-0,83201	-0,23983
15	0,87235	-0,59364	0,03487	1,60740
16	-0,35748	-0,07095	0,89981	-0,66184
17	0,35955	0,48825	-1,42470	-1,20840
18	0,52896	-0,09059	-1,10640	0,58384
19	0,21045	-0,81163	0,27359	1,33970
20	-0,27057	-0,05666	0,67590	-0,29095
21	0,10509	0,12011	-0,57328	-0,08961
22	0,27733	-0,13042	-0,22609	0,54988
23	-0,11596	-0,48187	0,70691	0,14626
24	-0,17450	0,04541	-0,03023	-0,48380
25	0,04078	-0,01947	-0,46043	0,38709
26	0,09925	-0,21858	0,25629	0,72114
27	-0,34650	-0,16314	0,76092	-0,13767
28	-0,12357	0,14865	0,00262	-0,45823
29	0,06393	0,10185	-0,19316	0,10520
30	0,06193	-0,16735	0,31487	0,27547

Table B.9: Fourier coefficients for the B4-70 P/D=1.0 propeller

B4-55 P/D=1.0				
K	C_T A(K)	C_T B(K)	C_Q A(K)	C_Q B(K)
0	3,40640	0,00000	-4,41640	0,00000
1	16,14500	-63,41100	-24,43400	88,77100
2	0,17506	-1,13890	3,34920	-2,68910
3	2,92210	5,20750	-4,97930	-5,51560
4	-0,56875	-0,82996	-1,36050	3,09920
5	-5,29070	6,42560	4,67430	-12,71400
6	0,15641	0,92855	0,13791	-1,51580
7	3,92110	-0,59179	-1,94810	2,82940
8	-0,50689	-0,37997	2,68480	1,15950
9	-1,44370	0,18518	-1,03960	-2,86520
10	0,55005	0,38178	-2,25050	-0,46642
11	1,19860	-0,06606	0,29114	-1,16780
12	-0,23543	-0,02740	1,13700	-0,91904
13	0,65241	1,00850	-2,51870	0,08695
14	0,28688	-0,13712	0,23806	1,46420
15	0,17300	-0,69713	-0,00422	0,12780
16	-0,21127	0,27074	-0,67432	-1,33320
17	0,67966	0,43689	-0,76630	0,39620
18	0,26391	-0,32628	0,60254	0,75362
19	-0,10945	-0,54754	-0,19228	-0,19235
20	0,02224	0,14242	-0,63017	0,39452
21	0,33699	-0,14012	-0,03974	1,06240
22	-0,13877	-0,23628	0,66197	-0,65197
23	-0,31140	-0,11184	-0,19484	-0,18034
24	0,10828	0,17998	-0,29096	0,61558
25	0,11705	-0,18777	0,69327	0,44397
26	-0,04857	-0,10904	0,01599	-0,27923
27	-0,19210	0,01304	-0,34259	0,09716
28	0,13741	-0,03663	0,36560	0,19218
29	-0,06013	-0,09930	0,50379	-0,41757
30	-0,13257	-0,01575	-0,43807	-0,30054

Table B.10: Fourier coefficients for the B4-55 P/D=1.0 propeller

B4-40 P/D=1.0				
K	C_T A(K)	C_T B(K)	C_Q A(K)	C_Q B(K)
0	2,66140	0,00000	-3,01640	0,00000
1	15,21000	-45,20800	-22,60900	62,79800
2	0,58496	-0,25662	2,80270	-3,55700
3	0,12400	-2,19580	-0,40627	4,32250
4	0,58317	-0,68275	-1,33710	0,74270
5	-1,44970	6,80570	0,00809	-11,47400
6	-0,23586	-0,01359	-0,30625	0,56226
7	0,14112	1,59490	0,68395	-3,26680
8	-0,25652	0,23590	0,08267	-0,03408
9	1,16170	-0,44863	-1,56810	1,22800
10	-0,06902	-0,20488	0,58928	0,31488
11	0,48012	0,72939	-1,39630	-1,06510
12	0,12225	0,11758	-0,06316	0,32814
13	0,01886	0,63909	-0,09963	-0,91986
14	0,04171	0,09434	0,06140	-0,18077
15	0,53829	-0,25580	-0,74530	0,69849
16	0,06739	-0,20823	0,44171	0,08737
17	0,48559	-0,12937	-0,80060	0,06215
18	0,04687	-0,13876	0,08242	0,14682
19	0,03822	-0,06713	0,07970	-0,22208
20	-0,02923	0,02867	-0,02159	-0,27584
21	0,06061	-0,18953	0,01608	0,36942
22	0,00847	-0,07495	0,14863	-0,04301
23	0,09415	-0,14680	-0,21349	0,32712
24	0,02939	-0,10319	-0,06912	0,07399
25	-0,05804	-0,08236	0,09611	0,09050
26	-0,06630	0,00097	-0,10560	-0,18117
27	-0,08928	-0,10354	0,10245	0,13436
28	-0,05915	0,04291	0,05715	-0,07722
29	-0,03228	-0,09839	0,05009	0,10368
30	0,04207	-0,03159	-0,08406	0,09843

Table B.11: Fourier coefficients for the B4-40 P/D=1.0 propeller

B.3 Number of blade variations

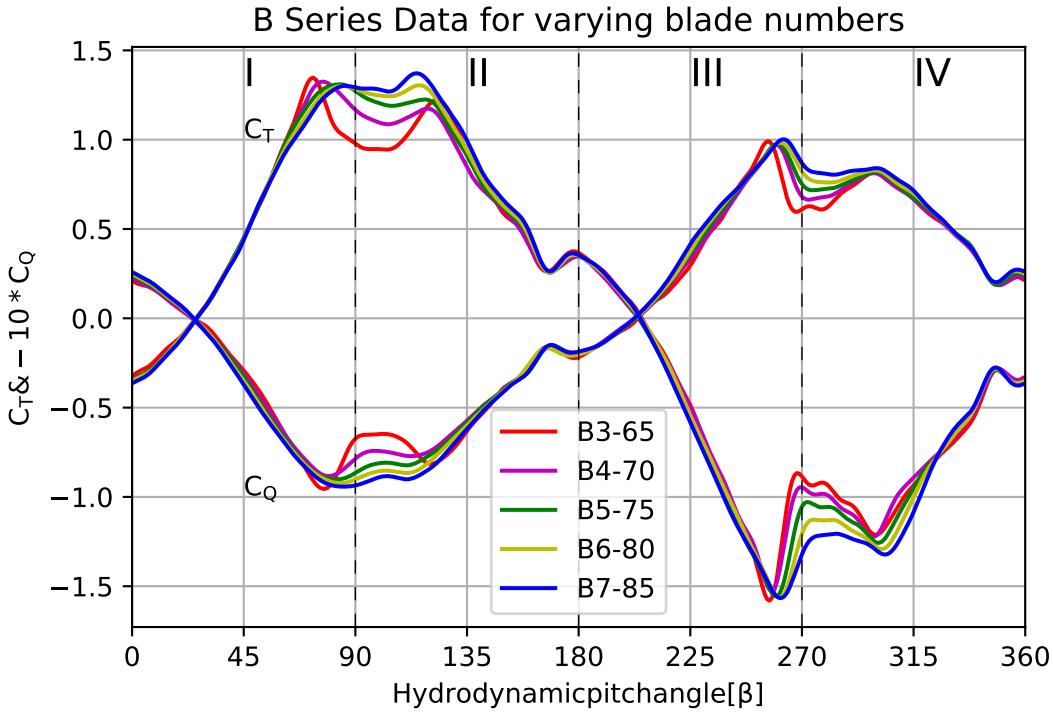


Figure B.3: Four quadrant behaviour for varying Z values

B3-65 P/D=1.0				
K	C_T A(K)	C_T B(K)	C_Q A(K)	C_Q B(K)
0	1,649800	0,00000	-3,360500	0,00000
1	18,754000	-73,236000	-27,953000	107,910000
2	1,466600	-1,080000	0,787960	-0,379110
3	2,784700	8,421700	-6,923900	-12,147000
4	-1,345000	-0,669130	0,586250	2,808900
5	-7,335500	7,418300	10,047000	-12,642000
6	0,164490	-0,578370	1,048400	0,140850
7	6,198200	-2,883700	-8,192900	5,141500
8	-1,322100	0,604580	1,809600	0,441440
9	-2,354600	3,194200	3,179600	-4,831600
10	0,731900	-0,057098	-1,391100	-1,535000
11	1,691000	-0,695680	-1,394100	0,834120
12	-0,363030	0,199270	1,340300	1,203200
13	1,312600	1,554800	-3,191000	-2,248200
14	0,440060	-0,103760	-1,188200	-0,514640
15	-0,139530	-1,430000	0,939990	2,129400
16	-0,180830	0,500440	1,174500	-0,641380
17	1,189900	0,532620	-2,299600	-0,614010
18	-0,505670	-0,441170	-1,252800	0,740750
19	-0,448304	-0,867310	0,641760	0,834040
20	0,096625	0,258830	0,792000	-0,673090
21	0,601400	-0,288610	-0,689390	0,953430
22	-0,140150	-0,523020	-0,336590	0,814200
23	-0,666170	-0,150840	0,924140	-0,041290
24	0,290430	0,304600	-0,217770	-0,936330
25	0,273270	-0,413410	-0,239640	0,672420
26	-0,332170	-0,447250	0,365770	1,190500
27	-0,535510	0,293320	0,602580	-0,412230
28	0,204560	0,093120	-0,379330	-0,681350
29	0,059079	-0,297830	0,008430	0,352050
30	-0,422720	0,034593	0,739710	0,208470

Table B.12: Fourier coefficients for the B3-65 P/D=1.0 propeller

B4-70 P/D=1.0				
K	C_T A(K)	C_T B(K)	C_Q A(K)	C_Q B(K)
0	2,66170	0,00000	-2,29450	0,00000
1	17,80200	-74,79700	-26,98900	110,38000
2	1,38280	-1,36320	-1,87580	-0,52852
3	2,78710	10,24600	-6,35520	-16,54400
4	-1,62400	-1,08650	2,16680	1,92900
5	-5,30840	4,73460	7,65300	-8,44800
6	0,19180	1,13550	-0,56516	-0,90714
7	3,83180	-0,93180	-6,26540	3,10420
8	-0,22892	-0,72471	1,92110	0,68648
9	-1,86560	2,37470	3,34560	-4,51960
10	0,34900	0,86208	-1,29560	-1,09280
11	2,23210	0,00702	-3,12940	-0,04497
12	-0,35976	-0,36422	1,24750	0,72404
13	0,13606	0,91758	-1,43510	-1,52840
14	0,47192	0,08938	-0,83201	-0,23983
15	0,87235	-0,59364	0,03487	1,60740
16	-0,35748	-0,07095	0,89981	-0,66184
17	0,35955	0,48825	-1,42470	-1,20840
18	0,52896	-0,09059	-1,10640	0,58384
19	0,21045	-0,81163	0,27359	1,33970
20	-0,27057	-0,05666	0,67590	-0,29095
21	0,10509	0,12011	-0,57328	-0,08961
22	0,27733	-0,13042	-0,22609	0,54988
23	-0,11596	-0,48187	0,70691	0,14626
24	-0,17450	0,04541	-0,03023	-0,48380
25	0,04078	-0,01947	-0,46043	0,38709
26	0,09925	-0,21858	0,25629	0,72114
27	-0,34650	-0,16314	0,76092	-0,13767
28	-0,12357	0,14865	0,00262	-0,45823
29	0,06393	0,10185	-0,19316	0,10520
30	0,06193	-0,16735	0,31487	0,27547

Table B.13: Fourier coefficients for the B4-70 P/D=1.0 propeller

B5-75 P/D=1.0				
K	C_T A(K)	C_T B(K)	C_Q A(K)	C_Q B(K)
0	2,378900	0,000000	-3,623700	0,000000
1	18,340000	-77,904000	-27,937000	115,960000
2	2,048800	-1,076000	-0,662990	-2,324200
3	3,044100	11,434000	-5,446700	-18,178000
4	-1,890200	-1,341200	2,931300	3,130600
5	-4,579700	4,476900	6,629600	-8,557800
6	0,073021	1,334300	-1,166100	-1,979600
7	3,067500	-0,257930	-6,168200	1,615400
8	0,030931	-0,794160	0,815640	1,976600
9	-1,479200	1,693100	3,102100	-2,549200
10	0,329530	0,952930	-0,958640	-1,411900
11	2,137500	0,488150	-2,822200	-0,919780
12	-0,175340	-0,434600	1,040400	1,026800
13	0,213290	0,679180	-1,078800	-1,127900
14	0,565490	0,053468	-1,115400	-0,131530
15	0,990860	-0,486320	-1,072600	1,265100
16	-0,393620	-0,146060	1,275400	0,055412
17	0,453020	0,090688	-0,426670	-0,684960
18	0,538740	0,070976	-0,709610	-0,232910
19	0,400510	-0,759910	-0,546910	1,504800
20	-0,374300	-0,156240	0,557810	-0,001358
21	0,039350	-0,262950	-0,068255	-0,068908
22	0,130760	-0,029191	-0,454420	0,312620
23	-0,127470	-0,555840	0,385970	0,671130
24	-0,171550	0,070082	0,436410	-0,253230
25	-0,156310	-0,056891	-0,106430	0,043119
26	0,088063	-0,091075	-0,207860	0,205040
27	-0,331430	-0,204190	0,548630	0,382390
28	-0,067347	0,128150	0,340890	-0,323640
29	-0,126430	0,067695	0,035566	-0,328540
30	0,078595	-0,054265	-0,085156	0,254000

Table B.14: Fourier coefficients for the B5-75 P/D=1.0 propeller

B6-80 P/D=1.0				
K	C_T A(K)	C_T B(K)	C_Q A(K)	C_Q B(K)
0	2,388500	0,00000	-3,955400	0,00000
1	18,569000	-80,260000	-29,939000	118,210000
2	2,349000	-0,891440	0,373490	-1,834200
3	2,693100	12,310000	-2,491200	-19,364000
4	-1,583600	-1,640200	1,460400	3,320600
5	-3,703100	4,551100	5,556500	-7,931200
6	-0,172710	1,357000	-0,361610	-2,797500
7	2,812600	0,239260	-5,925300	0,697090
8	0,267600	-0,707180	-0,282010	2,228200
9	-0,873740	1,086300	2,938900	-1,500500
10	-0,166430	0,736460	-0,136550	-1,388400
11	2,316500	0,544700	-2,940500	-1,715000
12	-0,108790	-0,392240	0,845050	0,535820
13	0,105130	0,176340	-0,714100	-0,986080
14	0,471520	0,336590	-0,862310	-0,274180
15	0,934000	-0,350400	-1,561600	1,005200
16	-0,193680	-0,109750	0,856050	0,648880
17	0,246760	-0,134520	-0,315550	-0,208590
18	0,458640	0,115360	-0,410010	-0,670370
19	0,237100	-0,651610	-0,561120	1,231000
20	-0,146600	-0,300740	0,286850	0,185600
21	-0,173950	-0,141600	0,177830	0,056991
22	0,195940	-0,084092	-0,241370	0,126990
23	-0,143690	-0,353060	0,173390	0,738940
24	-0,079146	-0,062500	0,345470	-0,050377
25	-0,165770	-0,016184	0,191010	-0,035304
26	-0,038620	-0,121700	-0,303160	0,021677
27	-0,206060	-0,083304	0,303030	0,275160
28	-0,205220	-0,000911	0,413740	-0,065413
29	-0,132600	0,127610	0,167290	-0,241370
30	0,005434	0,027347	-0,016320	0,046530

Table B.15: Fourier coefficients for the B6-80 P/D=1.0 propeller

B7-85 P/D=1.0				
K	C_T A(K)	C_T B(K)	C_Q A(K)	C_Q B(K)
0	2,438900	0,000000	-5,706400	0,000000
1	18,694000	-82,814000	-31,175000	121,670000
2	2,674400	-0,542620	1,972300	-3,501500
3	2,227500	12,694000	-1,187000	-19,599000
4	-1,190300	-1,373500	1,397100	3,364400
5	-2,984500	4,698500	4,453500	-8,697800
6	-0,277760	1,171800	-0,450630	-2,936100
7	2,304500	0,816970	-5,724800	-0,207540
8	0,883000	-0,513930	-1,303900	2,072800
9	-1,155100	0,722950	2,251800	-0,659470
10	0,015601	0,554510	0,015788	-0,349370
11	1,995300	0,830040	-2,207400	-1,668400
12	-0,051119	-0,516020	0,796380	0,057940
13	0,245840	0,415960	-0,549730	-0,822270
14	0,462080	0,198810	-0,603040	-0,407480
15	0,991170	-0,136580	-1,947600	0,534640
16	-0,175610	-0,428320	0,669140	0,751140
17	0,315670	-0,174650	-0,617790	0,139500
18	0,293110	0,175190	-0,317800	-0,704200
19	0,436650	-0,604530	-0,403650	1,098700
20	-0,249890	-0,245130	0,050233	0,307400
21	-0,014826	-0,230310	0,043532	0,158740
22	0,125220	0,098137	0,012015	0,103320
23	-0,099886	-0,399830	0,205200	0,744560
24	-0,096050	-0,113090	0,091405	-0,128850
25	-0,196690	-0,083862	0,295370	0,072704
26	0,070053	0,028767	-0,241910	-0,030231
27	-0,257640	-0,282350	0,259440	0,187410
28	-0,193630	0,010305	0,162450	-0,082735
29	-0,184120	0,153270	0,416080	-0,090507
30	0,111470	-0,008408	0,092657	0,013142

Table B.16: Fourier coefficients for the B7-85 P/D=1.0 propeller

B.4 Ka-series Duct 37

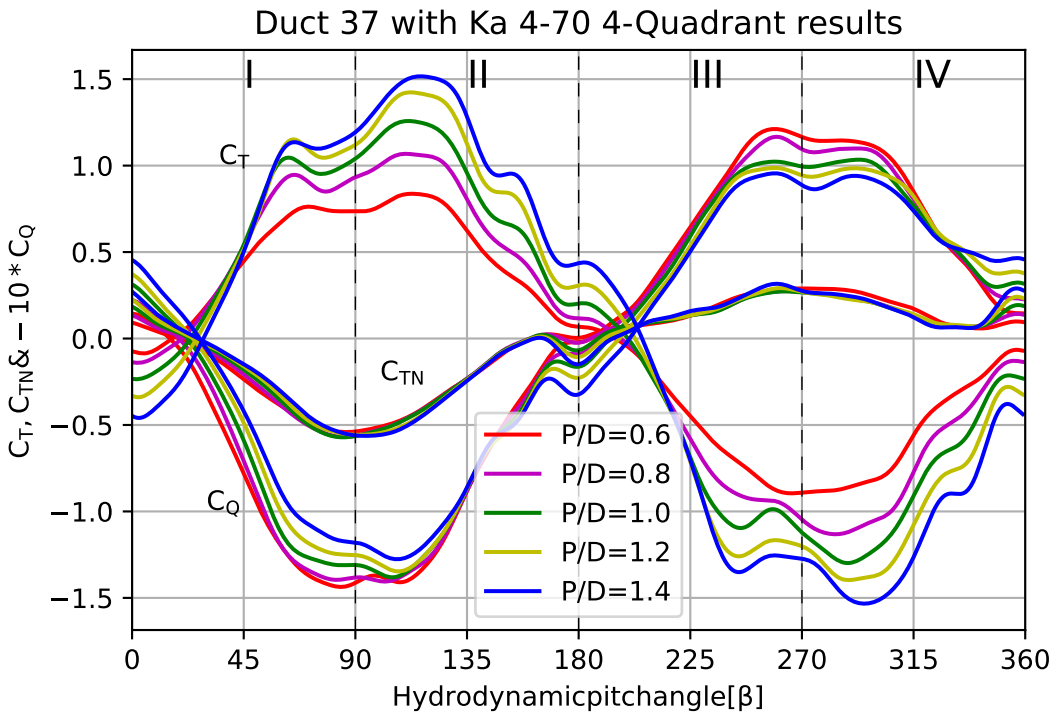


Figure B.4: Four quadrant behaviour for varying P/D values

Ka4-70 P/D=0.6 duct 37						
K	C_T A(K)	C_T B(K)	C_Q A(K)	C_Q B(K)	C_{Tn} A(K)	C_{Tn} B(K)
0	-0,07852200	0,00000000	-0,07585400	0,00000000	0,01488400	0,00000000
1	0,09196200	-1,22410000	0,00911520	-0,34397000	0,10044000	-0,79096000
2	0,09673300	-0,01080500	0,08531600	-0,00608630	-0,02518200	0,01220600
3	-0,00146570	0,16207000	0,00472030	0,08250600	-0,01091800	0,09071800
4	0,01081000	0,00106420	0,00318380	0,00518160	0,02750200	0,00726690
5	-0,02070800	0,07864800	0,00454640	0,00992820	-0,00260720	0,05765300
6	-0,00803160	0,01409800	0,00318280	0,00059292	-0,01140900	0,00011032
7	0,01105200	-0,01132900	0,00954810	-0,00191480	0,00093808	-0,00993880
8	0,00210700	-0,00525960	-0,00194320	-0,00176860	0,00827830	-0,00228920
9	-0,01646600	0,01181500	0,00586070	-0,00078198	-0,01775600	0,00537960
10	0,00085238	-0,00237710	-0,00150470	-0,00304450	-0,00425980	0,00368760
11	0,00393840	0,00641130	0,00380030	-0,00277830	-0,00466640	0,00966030
12	-0,00329050	0,00500270	-0,00063250	0,00017514	0,00102780	0,00041719
13	0,00256720	0,00604670	0,00101020	-0,00060004	0,00206670	0,00729000
14	0,00247700	-0,00282420	-0,00028923	-0,00032771	0,00185010	0,00096970
15	0,00622080	0,00204890	0,00265880	0,00016418	0,00261120	0,00087227
16	0,00034143	0,00031069	-0,00011302	-0,00143230	-0,00325050	0,00021002
17	0,00197800	0,00063925	0,00199660	-0,00114560	-0,00077389	0,00288320
18	0,00060762	-0,00220820	-0,00056644	-0,00074530	0,00132200	0,00070445
19	0,00344880	0,00304210	0,00095964	-0,00083559	0,00168560	0,00395470
20	-0,00171660	-0,00052892	-0,00058527	-0,00005418	-0,00048127	0,00177910

Table B.17: Fourier coefficients for the Ka4-70 P/D=0.6 propeller with duct 37

Ka4-70 P/D=0.8 duct 37						
K	C_T A(K)	C_T B(K)	C_Q A(K)	C_Q B(K)	C_{Tn} A(K)	C_{Tn} B(K)
0	-0,0811690	0,0000000	-0,0851040	0,0000000	0,0200890	0,0000000
1	0,1284900	-1,1842000	0,0151220	-0,3323700	0,1663600	-0,9921900
2	0,1133100	0,0005834	0,1032500	0,0024803	-0,0183880	0,0128920
3	0,0151310	0,1644100	0,0156490	0,0897610	-0,0190510	0,0952720
4	0,0041567	0,0062103	0,0007568	0,0045883	0,0168080	0,0160450
5	-0,0162200	0,0765060	0,0092408	0,0026548	0,0052434	0,0943540
6	-0,0113050	0,0096359	0,0000000	-0,0042819	-0,0110190	-0,0036169
7	0,0093452	-0,0180360	0,0180410	-0,0024385	-0,0269420	-0,0225390
8	0,0017779	-0,0121460	-0,0014737	-0,0045051	0,0094780	-0,0000305
9	-0,0062214	0,0092879	0,0048237	-0,0015233	-0,0019689	0,0021436
10	0,0046290	0,0040488	-0,0009934	-0,0015358	0,0016447	0,0037463
11	0,0069293	0,0073891	0,0048625	-0,0032041	-0,0147660	0,0213980
12	-0,0020445	-0,0040761	-0,0011657	0,0000173	-0,0022670	0,0019168
13	0,0059366	0,0059307	0,0043615	-0,0008087	0,0093023	0,0076358
14	-0,0004406	-0,0022663	-0,0003800	-0,0021661	0,0041823	-0,0033249
15	0,0072301	-0,0015148	0,0032422	-0,0002179	0,0077675	0,0024934
16	-0,0005320	-0,0001326	-0,0003716	-0,0007264	-0,0013283	0,0003921
17	0,0026809	0,0040086	0,0022704	-0,0013466	-0,0009203	-0,0008367
18	-0,0008558	-0,0011431	-0,0011253	-0,0002376	0,0002595	-0,0031653
19	0,0032728	-0,0025883	0,0019759	-0,0010735	0,0023462	0,0041032
20	-0,0011347	-0,0000933	-0,0009698	-0,0000057	0,0006982	0,0020375

Table B.18: Fourier coefficients for the Ka4-70 P/D=0.8 propeller with duct 37

Ka4-70 P/D=1.0 duct 37						
K	C_T A(K)	C_T B(K)	C_Q A(K)	C_Q B(K)	C_{Tn} A(K)	C_{Tn} B(K)
0	-0,07868100	0,00000000	-0,08043200	0,00000000	0,03076700	0,00000000
1	0,17005000	-1,11520000	0,02990400	-0,32774000	0,24472000	-1,13150000
2	0,12604000	0,02037100	0,10546000	0,00819520	-0,01131600	0,03371200
3	0,02444400	0,15275000	0,02127700	0,09307300	-0,00816580	0,09034300
4	-0,00699870	0,00288810	-0,00148180	0,00382890	0,00162080	0,01211300
5	-0,00529980	0,07829900	0,01566700	0,00021950	0,00866320	0,12138000
6	-0,00775000	0,00188650	0,00175920	-0,00630450	-0,00349360	-0,00543220
7	0,00670880	-0,02466500	0,02567100	-0,00231250	-0,04607500	-0,03571200
8	0,00788180	-0,00829560	-0,00090447	-0,00362900	0,00666510	0,00638620
9	0,00830580	0,01508500	0,00882040	-0,00090853	0,00615100	0,01402100
10	0,00208330	0,00158790	-0,00180080	-0,00189970	0,00519470	0,00280950
11	0,00722620	0,00951290	0,00816430	-0,00404800	-0,01370200	0,03282800
12	-0,00058329	-0,00732490	-0,00097356	-0,00034197	0,00012766	-0,00191950
13	0,00884670	0,00349310	0,00690390	-0,00135220	0,00616800	0,00888170
14	-0,00295590	-0,00635700	-0,00056244	-0,00223220	0,00127130	-0,00713090
15	0,01153000	-0,00224740	0,00454750	-0,00148080	0,01457000	0,00309770
16	-0,00083057	0,00250690	-0,00038196	-0,00114810	0,00443600	0,00064517
17	0,00313390	-0,00129900	0,00311600	-0,00286780	0,00181400	-0,00228760
18	-0,00132900	-0,00041905	-0,00118780	-0,00025410	-0,00431270	-0,00009875
19	0,00306660	-0,00282880	0,00191580	-0,00256310	0,00387940	0,00107180
20	-0,00137490	-0,00045929	-0,00131290	0,00002203	0,00014728	-0,00232830

Table B.19: Fourier coefficients for the Ka4-70 P/D=1.0 propeller with duct 37

Ka4-70 P/D=1.2 duct 37						
K	C_T A(K)	C_T B(K)	C_Q A(K)	C_Q B(K)	C_{Tn} A(K)	C_{Tn} B(K)
0	-0,0602560	0,0000000	-0,0723100	0,0000000	0,0443510	0,0000000
1	0,2236000	-1,0687000	0,0472200	-0,3289900	0,3423000	-1,2562000
2	0,1235300	0,0296430	0,1045500	0,0179830	-0,0180870	0,0552980
3	0,0240860	0,1427500	0,0239120	0,0930610	0,0035568	0,0928370
4	-0,0145180	-0,0140160	-0,0032961	-0,0063572	-0,0063786	-0,0049373
5	-0,0062461	0,0734130	0,0248630	0,0007443	-0,0135130	0,1412900
6	-0,0043441	0,0007095	0,0001770	-0,0030252	0,0009457	-0,0031662
7	0,0177260	-0,0267250	0,0309330	-0,0026122	-0,0357930	-0,0451590
8	0,0108200	-0,0103090	0,0015955	-0,0064419	0,0102950	0,0056348
9	0,0089902	0,0153990	0,0124060	0,0003370	0,0022438	0,0280530
10	-0,0024474	-0,0072466	-0,0013918	-0,0029805	-0,0026275	0,0023267
11	0,0051620	0,0093292	0,0102160	-0,0050133	-0,0174270	0,0383090
12	-0,0048962	-0,0039241	0,0000162	-0,0028858	-0,0021233	-0,0069986
13	0,0080184	0,0055616	0,0072150	-0,0028057	0,0098031	0,0142680
14	-0,0025019	-0,0003151	-0,0003925	-0,0011737	0,0038115	-0,0007750
15	0,0149830	-0,0017566	0,0048746	-0,0034043	0,0226080	0,0032200
16	-0,0002822	0,0018409	-0,0021174	-0,0020546	-0,0003723	0,0069956
17	0,0023533	-0,0029180	0,0031669	-0,0031834	0,0001435	-0,0002118
18	-0,0006446	0,0006327	-0,0013671	-0,0011222	0,0016375	-0,0008496
19	0,0012248	-0,0031003	0,0016721	-0,0026716	0,0051490	-0,0014700
20	-0,0017391	-0,0003183	-0,0019020	-0,0004963	0,0026719	-0,0027758

Table B.20: Fourier coefficients for the Ka4-70 P/D=1.2 propeller with duct 37

Ka4-70 P/D=1.4 duct 37						
K	C_T A(K)	C_T B(K)	C_Q A(K)	C_Q B(K)	C_{Tn} A(K)	C_{Tn} B(K)
0	-0,04743700	0,00000000	-0,06389300	0,00000000	0,06403300	0,00000000
1	0,26393000	-1,00040000	0,06026000	-0,33060000	0,45620000	-1,33830000
2	0,11478000	0,04614500	0,10016000	0,02888000	-0,02674700	0,05707500
3	0,04730900	0,14074000	0,03336900	0,09903600	0,01615200	0,08905100
4	-0,01106100	-0,02194000	-0,00177850	-0,01007500	-0,01584600	-0,00977240
5	0,01130800	0,06729400	0,03860400	0,00071186	-0,01933600	0,15575000
6	-0,00083647	-0,00449870	0,00437130	-0,00485840	0,00550300	0,00255990
7	0,02493300	-0,02551800	0,03503500	-0,00186710	-0,02867000	-0,04257200
8	0,00195520	-0,01251800	0,00050841	-0,01009300	0,00028700	0,00296380
9	0,00605310	0,01415100	0,01276400	0,00002444	-0,00496520	0,02485000
10	-0,00287480	-0,00125880	-0,00024911	-0,00227080	0,00239130	0,00306070
11	0,00641180	0,00556180	0,01332000	-0,00773470	-0,02097600	0,04183700
12	-0,00481640	-0,00532890	-0,00185390	-0,00408020	0,00144160	-0,00321160
13	0,00642670	0,00440790	0,00770570	-0,00205230	0,00399530	0,01784800
14	-0,00403580	0,00048467	0,00022148	-0,00210400	-0,00136050	0,00301140
15	0,01605100	-0,00189050	0,00510240	-0,00470280	0,02843700	0,00759770
16	-0,00328160	0,00299650	-0,00266060	-0,00192650	0,00193370	0,00270870
17	0,00302500	-0,00377610	0,00245550	-0,00396150	0,00754720	-0,00132550
18	-0,00135670	0,00327630	-0,00263630	-0,00060252	0,00290070	0,00030258
19	0,00317940	-0,00441090	0,00108830	-0,00276630	0,00323690	-0,00373680
20	0,00044946	0,00104590	-0,00289780	0,00010214	0,00322470	-0,00237940

Table B.21: Fourier coefficients for the Ka4-70 P/D=1.4 propeller with duct 37

B.5 Ka-series Duct 19a

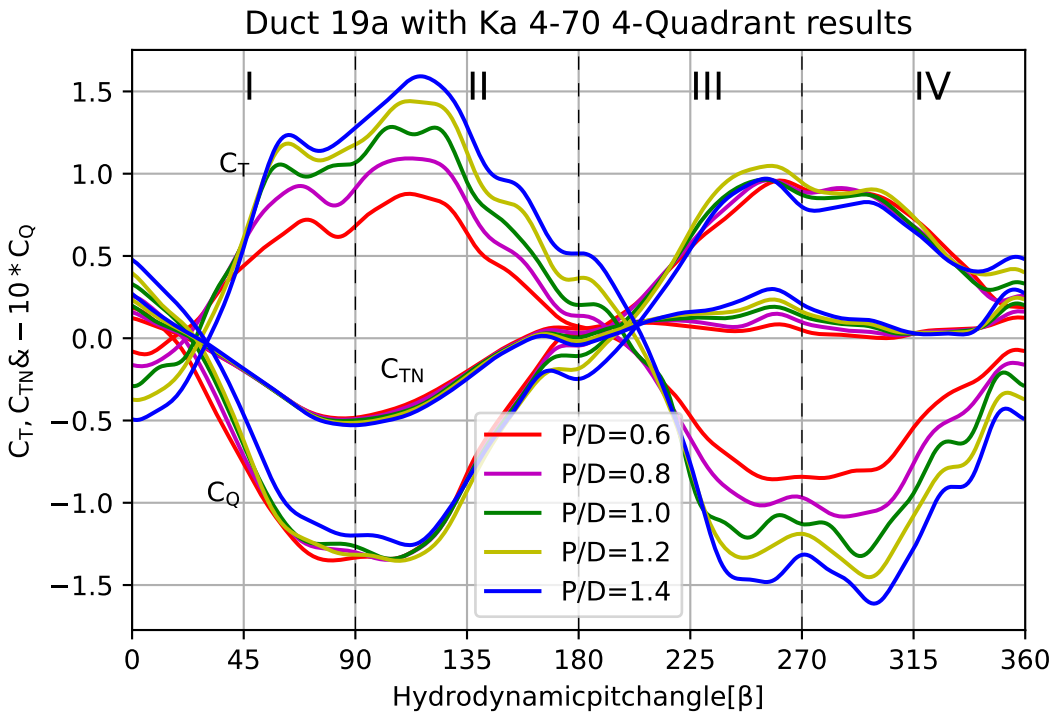


Figure B.5: Four quadrant behaviour for varying P/D values

Ka4-70 P/D=0.6 duct 19a						
K	C_T A(K)	C_T B(K)	C_Q A(K)	C_Q B(K)	C_{Tn} A(K)	C_{Tn} B(K)
0	-0,1482500	0,0000000	-0,1427600	0,0000000	0,0170840	0,0000000
1	0,0846970	-1,0838000	-0,0055945	-0,2187500	0,1055000	-0,7807000
2	0,1670000	-0,0180230	0,1551900	0,0101140	-0,0273800	0,0381340
3	0,0009661	0,1182500	0,0159150	0,0471200	-0,0118270	0,0742920
4	0,0147540	-0,0070713	0,0066633	-0,0005891	0,0286710	-0,0135680
5	-0,0118060	0,0628940	0,0008934	-0,0012958	0,0042504	0,0665950
6	-0,0148880	0,0115190	-0,0038876	-0,0020824	-0,0078835	0,0103300
7	0,0073311	0,0017070	0,0109760	0,0052475	-0,0070981	-0,0178850
8	0,0075022	0,0022990	0,0031959	-0,0015428	0,0076691	-0,0036187
9	-0,0151280	0,0134580	0,0014201	0,0023580	-0,0125060	0,0100150
10	0,0033002	0,0005481	0,0013507	0,0002349	-0,0070343	0,0055926
11	0,0031416	0,0042076	0,0050526	-0,0026868	-0,0102540	0,0069688
12	-0,0021144	-0,0057232	-0,0009086	-0,0045635	0,0025186	-0,0047676
13	0,0029438	0,0074689	0,0000428	-0,0000446	0,0096613	0,0088889
14	0,0003386	-0,0000848	0,0004208	-0,0000186	0,0014934	0,0049081
15	0,0041236	-0,0013374	0,0020269	-0,0008055	-0,0028323	-0,0000582
16	0,0016259	-0,0009193	-0,0007975	-0,0010170	-0,0030360	0,0052044
17	0,0012759	0,0027412	0,0009745	-0,0000467	0,0020889	0,0012522
18	0,0020647	-0,0010198	0,0004890	-0,0001709	0,0031929	0,0033190
19	0,0034157	0,0019845	0,0008435	-0,0006067	-0,0009164	0,0052446
20	-0,0005870	-0,0013980	-0,0003930	-0,0003632	-0,0023922	-0,0020591

Table B.22: Fourier coefficients for the Ka4-70 P/D=0.6 propeller with duct 19a

Ka4-70 P/D=0.8 duct 19a						
K	C_T A(K)	C_T B(K)	C_Q A(K)	C_Q B(K)	C_{Tn} A(K)	C_{Tn} B(K)
0	-0,13080000	0,00000000	-0,12764000	0,00000000	0,01936800	0,00000000
1	0,10985000	-1,07080000	0,00068679	-0,24100000	0,17050000	-0,99912000
2	0,15810000	0,02416300	0,14639000	0,01891900	-0,01190100	0,03192400
3	0,01836700	0,12784000	0,02319500	0,05551300	-0,00256010	0,08138400
4	0,01616800	-0,00140640	0,01029200	-0,00124530	0,01776300	-0,00035096
5	-0,00374020	0,07621300	0,00866510	-0,00147480	0,00820850	0,10631000
6	-0,01173600	0,01325900	-0,00391240	-0,00218990	-0,00343360	0,01511600
7	0,00254830	-0,00423000	0,01598400	0,00566940	-0,02453400	-0,02704500
8	0,00123500	-0,00262460	0,00462950	-0,00499110	-0,00102890	-0,00593890
9	-0,00207720	0,01632800	0,00068371	0,00136310	-0,00749380	0,01108500
10	0,00697490	-0,00033979	0,00167400	0,00128770	0,00154440	0,00837870
11	0,00598380	0,00235060	0,00834950	-0,00351760	-0,01417300	0,01519200
12	-0,00145990	-0,00694970	-0,00077063	-0,00555710	-0,00280340	-0,00515070
13	0,00835330	0,00619250	0,00149360	0,00031438	0,01424600	0,01483600
14	0,00110930	0,00035046	0,00110170	-0,00111790	0,00346630	0,00250410
15	0,00418850	-0,00115710	0,00168040	-0,00245770	0,00207640	-0,00078615
16	-0,00012438	-0,00032566	-0,00103380	-0,00055484	-0,00294240	-0,00245260
17	0,00380340	0,00063420	0,00224090	0,00010968	0,00301490	-0,00032187
18	0,00090073	-0,00227490	0,00110000	-0,00073945	0,00277140	-0,00048551
19	0,00311470	-0,00036805	0,00048406	-0,00154000	0,00018423	0,00354550
20	-0,00010633	-0,00123500	-0,00033008	0,00022408	-0,00008163	-0,00034936

Table B.23: Fourier coefficients for the Ka4-70 P/D=0.8 propeller with duct 19a

Ka4-70 P/D=1.0 duct 19a						
K	C_T A(K)	C_T B(K)	C_Q A(K)	C_Q B(K)	C_{Tn} A(K)	C_{Tn} B(K)
0	-0,1098500	0,0000000	-0,1125700	0,0000000	0,0315890	0,0000000
1	0,1406400	-1,0583000	0,0093340	-0,2626500	0,2440600	-1,1717000
2	0,1578500	0,0472840	0,1378800	0,0275870	-0,0073880	0,0511550
3	0,0455440	0,1312600	0,0332230	0,0652620	0,0282600	0,0890690
4	0,0051639	-0,0077539	0,0126720	-0,0040234	-0,0055959	-0,0065670
5	-0,0025560	0,0935070	0,0142500	0,0010255	0,0002656	0,1420400
6	-0,0060502	0,0092520	-0,0003041	-0,0032045	0,0113680	0,0077052
7	0,0067368	-0,0148280	0,0198880	-0,0021752	-0,0474010	-0,0360910
8	0,0068571	-0,0096554	0,0048334	-0,0059535	-0,0065686	0,0042036
9	0,0047245	0,0096216	0,0028427	0,0009066	-0,0074990	0,0021139
10	0,0023591	-0,0007545	0,0032326	-0,0010222	0,0012873	0,0130950
11	0,0087912	0,0024453	0,0097693	-0,0048133	0,0046502	0,0309610
12	0,0011968	-0,0087981	-0,0002838	-0,0056355	-0,0046676	-0,0099459
13	0,0083808	0,0018184	0,0029395	-0,0018248	0,0033438	0,0179210
14	-0,0008210	-0,0020077	0,0005318	-0,0020263	0,0022046	-0,0081917
15	0,0027371	-0,0033070	0,0016229	-0,0030382	0,0070034	-0,0007843
16	-0,0002612	-0,0007920	-0,0002727	-0,0011128	0,0391470	0,0072661
17	0,0019133	-0,0003631	0,0020276	-0,0015327	0,0073719	-0,0047316
18	0,0003229	-0,0019377	0,0003548	-0,0012433	-0,0009408	-0,0025731
19	0,0015223	-0,0012135	0,0003908	-0,0020069	0,0060560	0,0011136
20	-0,0010151	-0,0003168	-0,0009251	-0,0004884	-0,0004239	-0,0015470

Table B.24: Fourier coefficients for the Ka4-70 P/D=1.0 propeller with duct 19a

Ka4-70 P/D=1.2 duct 19a						
K	C_T A(K)	C_T B(K)	C_Q A(K)	C_Q B(K)	C_{Tn} A(K)	C_{Tn} B(K)
0	-0,0908880	0,0000000	-0,1016600	0,0000000	0,0438000	0,0000000
1	0,1795900	-1,1026000	0,0185930	-0,2776900	0,3529900	-1,2949000
2	0,1495600	0,0614590	0,1340800	0,0354590	-0,0109170	0,0590300
3	0,0656750	0,1371500	0,0437670	0,0723170	0,0470620	0,0935400
4	0,0052107	-0,0172800	0,0136040	-0,0083408	-0,0107790	-0,0061148
5	-0,0068232	0,0965790	0,0186580	0,0044854	-0,0101930	0,1612100
6	-0,0062896	0,0058809	0,0026598	-0,0037642	-0,0008882	0,0146240
7	0,0181780	-0,0225870	0,0249070	0,0007573	-0,0378930	-0,0535490
8	0,0060694	-0,0148190	0,0047924	-0,0088802	-0,0070346	-0,0031589
9	0,0061942	-0,0103980	0,0036556	0,0004054	-0,0080130	0,0143820
10	0,0026482	-0,0029324	0,0039850	-0,0012811	0,0072622	0,0099836
11	0,0121370	0,0040913	0,0106430	-0,0055230	-0,0054390	0,0387810
12	-0,0035705	-0,0044436	0,0002550	-0,0063566	-0,0020068	-0,0046749
13	0,0032985	-0,0012190	0,0029347	-0,0025338	0,0039281	0,0149440
14	-0,0008865	-0,0022551	0,0003660	-0,0020504	-0,0006526	-0,0063253
15	0,0069807	-0,0032272	0,0013115	-0,0038485	0,0154140	0,0022275
16	-0,0001756	0,0017533	-0,0013511	-0,0006391	0,0030356	0,0071826
17	0,0021643	0,0014875	0,0017101	-0,0010819	0,0059073	0,0010229
18	0,0003536	0,0000454	0,0003377	-0,0009632	0,0041433	-0,0059201
19	0,0025772	-0,0008870	-0,0003968	-0,0020969	0,0046102	-0,0014814
20	-0,0018279	-0,0009461	-0,0011814	-0,0001930	-0,0005742	-0,0043092

Table B.25: Fourier coefficients for the Ka4-70 P/D=1.2 propeller with duct 19a

Ka4-70 P/D=1.4 duct 19a						
K	C_T A(K)	C_T B(K)	C_Q A(K)	C_Q B(K)	C_{Tn} A(K)	C_{Tn} B(K)
0	-0,073487	0,000000	-0,086955	0,000000	0,073202	0,000000
1	0,228610	-0,981010	0,030046	-0,297990	0,473010	-1,406200
2	0,148530	0,071510	0,126510	0,043403	-0,033300	0,071683
3	0,075328	0,142170	0,055034	0,083309	0,062786	0,114490
4	0,003408	-0,022675	0,019376	-0,014571	-0,019511	-0,013400
5	-0,001164	0,091082	0,022082	0,004340	-0,027569	0,175470
6	0,000186	-0,004028	0,007628	-0,003926	-0,003830	0,025715
7	0,026970	-0,022759	0,031821	-0,002350	-0,023310	-0,054967
8	0,002062	-0,016727	0,005184	-0,013633	-0,008453	-0,012576
9	0,007867	0,008697	0,003890	-0,001400	-0,004896	0,013084
10	0,004691	-0,004752	0,004930	-0,002821	0,004854	0,010733
11	0,014771	0,002283	0,010731	-0,007736	-0,007195	0,044142
12	-0,007506	-0,004938	0,001139	-0,006867	-0,005319	-0,001095
13	0,001498	-0,002592	0,003138	-0,004239	0,001328	0,012209
14	0,002406	-0,002514	-0,000826	-0,003325	0,006870	-0,001407
15	0,005565	-0,003366	-0,000018	-0,004550	0,018071	0,001784
16	-0,003818	0,002815	-0,003623	-0,001228	-0,001573	0,003795
17	0,002670	-0,000222	-0,000224	-0,001576	0,011527	0,004997
18	0,001575	-0,000537	-0,000584	-0,000001	0,010168	-0,004240
19	0,000245	-0,003519	-0,001281	-0,001779	0,000815	-0,007730
20	-0,000042	-0,000428	-0,001987	0,000496	0,001405	-0,003449

Table B.26: Fourier coefficients for the Ka4-70 P/D=1.4 propeller with duct 19a

Bibliography

[1] H. Afzal. "Prediction of Movies popularity Using Machine Learning Techniques". In: *International Journal of Computer Science and Network Security*, 16 (Aug. 2016), pp. 127–131.

[2] S. Ajala et al. "Comparing machine learning and deep learning regression frameworks for accurate prediction of dielectrophoretic force". In: *Scientific Reports* 12 (July 2022). DOI: 10.1038/s41598-022-16114-5. URL: https://doi.org/10.1038/s41598-022-16114-5.

[3] E. Alpaydin. *MACHINE LEARNING*. Vol. 1. MIT Press Essential Knowledge series, Oct. 2016.

[4] I. D. of Aviation. *pitch ratio*. 2022. URL: https://encyclopedia2.thefreedictionary.com/pitch+ratio.

[5] M. Awad and R. Khanna. "Support Vector Regression". In: *Efficient Learning Machines: Theories, Concepts, and Applications for Engineers and System Designers*. Berkeley, CA: Apress, 2015, pp. 67–80. ISBN: 978-1-4302-5990-9. DOI: 10.1007/978-1-4302-5990-9_4. URL: https://doi.org/10.1007/978-1-4302-5990-9_4.

[6] C. B. Azodi, J. Tang, and S.-H. Shiu. "Opening the Black Box: Interpretable Machine Learning for Geneticists". In: *Trends in Genetics* 36.6 (2020), pp. 442–455. ISSN: 0168-9525. DOI: https://doi.org/10.1016/j.tig.2020.03.005. URL: https://www.sciencedirect.com/science/article/pii/S016895252030069X.

[7] R. Bachmayer, L. Whitcomb, and M. Grosenbaugh. "An accurate four-quadrant nonlinear dynamical model for marine thrusters: theory and experimental validation". In: *IEEE Journal of Oceanic Engineering* 25.1 (2000), pp. 146–159. DOI: 10.1109/48.820747.

[8] A.-L. Balogun and A. Tella. *Modelling and investigating the impacts of climatic variables on ozone concentration in Malaysia using correlation analysis with random forest, decision tree regression, linear regression, and support vector regression*. 2022. DOI: https://doi.org/10.1016/j.chemosphere.2022.134250. URL: https://www.sciencedirect.com/science/article/pii/S0045653522007433.

[9] A. Bhattacharyya, V. Krasilnikov, and S. Steen. "Scale effects on open water characteristics of a controllable pitch propeller working within different duct designs". In: *Ocean Engineering* 112 (2016), pp. 226–242. ISSN: 0029-8018. DOI: https://doi.org/10.1016/j.oceaneng.2015.12.024. URL: https://www.sciencedirect.com/science/article/pii/S0029801815006782.

[10] A. Bihlo and R. O. Popovych. "Physics-informed neural networks for the shallow-water equations on the sphere". In: *Journal of Computational Physics* 456 (2022), p. 111024. ISSN: 0021-9991. DOI: https://doi.org/10.1016/j.jcp.2022.111024. URL: https://www.sciencedirect.com/science/article/pii/S0021999122000869.

[11] L. Birk. *Fundamentals of Ship Hydrodynamics*. Apr. 2019. ISBN: 9781118855485. DOI: 10.1002/9781119191575.

[12] J. P. Brown. *Four quadrant dynamic model of the AUV II thruster*. Sept. 1993. URL: http://hdl.handle.net/10945/27061.

[13] J. Carlton. *Marine Propellers and Propulsion*. Ed. by J. Carlton. 2nd. Butterworth-Heinemann, 2007. ISBN: 9780081003664.

[14] J. Carlton. *Chapter 2 - Propulsion Systems*. Ed. by J. Carlton. Oxford, 2012. DOI: https://doi.org/10.1016/B978-0-08-097123-0.00002-2. URL: https://www.sciencedirect.com/science/article/pii/B9780080971230000022.

[15] J. Carlton. *Chapter 22 - Propeller Design*. Ed. by J. Carlton. Fourth Edition. Butterworth-Heinemann, 2019, pp. 469–497. ISBN: 978-0-08-100366-4. DOI: https://doi.org/10.1016/B978-0-08-100366-4.00022-5. URL: https://www.sciencedirect.com/science/article/pii/B9780081003664000225.

[16] R. Caruana et al. "Intelligible Models for HealthCare: Predicting Pneumonia Risk and Hospital 30-Day Readmission". In: *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. KDD '15. Association for Computing Machinery, 2015, 1721–1730. ISBN: 9781450336642. DOI: 10.1145/2783258.2788613. URL: https://doi.org/10.1145/2783258.2788613.

[17] C.-C. Chang and C.-J. Lin. "Training v-Support Vector Regression: Theory and Algorithms". In: *Neural Computation* 14.8 (Aug. 2002), pp. 1959–1977. ISSN: 0899-7667. DOI: 10.1162/089976602760128081. eprint: https://direct.mit.edu/neco/article-pdf/14/8/1959/815457/089976602760128081.pdf. URL: https://doi.org/10.1162/089976602760128081.

[18] N. Chauhan and K. Singh. *A Review on Conventional Machine Learning vs Deep Learning*. Sept. 2018. DOI: 10.1109/GUCON.2018.8675097.

[19] W. Commons. *Rosenblatt's Perceptron*. CC-BY-SA 3.0. 2012. URL: https://nl.wikipedia.org/wiki/Perceptron#/media/Bestand:Rosenblattperceptron.png.

[20] G Cybenko. *Approximation by superposition of sigmoidal function*. Dec. 1989. DOI: https://doi.org/10.1007/BF02551274.

[21] S. Demir et al. "Data augmentation for time series regression: Applying transformations, autoencoders and adversarial networks to electricity price forecasting". In: *Applied Energy* 304 (2021), p. 117695. ISSN: 0306-2619. DOI: https://doi.org/10.1016/j.apenergy.2021.117695. URL: https://www.sciencedirect.com/science/article/pii/S0306261921010527.

[22] V. L. Deringer et al. "Gaussian Process Regression for Materials and Molecules". In: *Chemical Reviews* 121.16 (2021), pp. 10073–10141. DOI: 10.1021/acs.chemrev.1c00022.

[23] M. Diesel. *Basic Principles of Ship Propulsion*. 2011. URL: https://www.man-es.com/docs/default-source/marine/tools/basic-principles-of-ship-propulsion_web_links.pdf?sfvrsn=12d1b862_10.

[24] D. Duvenaud. *Automatic model construction with Gaussian processes*. Nov. 2014.

[25] I. El Naqa et al. "Machine learning and modeling: Data, validation, communication challenges". In: *Medical Physics* 45.10 (2018), e834–e840. DOI: https://doi.org/10.1002/mp.12811. URL: https://aapm.onlinelibrary.wiley.com/doi/abs/10.1002/mp.12811.

[26] F. Farahnakian et al. *Energy Aware Consolidation Algorithm Based on K-Nearest Neighbor Regression for Cloud Data Centers*. 2013. DOI: 10.1109/UCC.2013.51.

[27] S. G. Finlayson et al. *Adversarial attacks on medical machine learning*. 2019. DOI: https://doi.org/10.1126/science.aaw4399.

[28] M. S. Gashler and S. C. Ashmore. "Modeling time series data with deep Fourier neural networks". In: *Neurocomputing* 188 (2016). Advanced Intelligent Computing Methodologies and Applications, pp. 3–11. ISSN: 0925-2312. DOI: https://doi.org/10.1016/j.neucom.2015.01.108. URL: https://www.sciencedirect.com/science/article/pii/S0925231215017774.

[29] L. Gultchin. "Just for Laughs: Utilizing Machine Learning to Rate and Generate Humorous Analogies". In: (July 2017).

[30] E. Haghighat et al. "A physics-informed deep learning framework for inversion and surrogate modeling in solid mechanics". In: *Computer Methods in Applied Mechanics and Engineering* 379 (2021), p. 113741. ISSN: 0045-7825. DOI: https://doi.org/10.1016/j.cma.2021.113741. URL: https://www.sciencedirect.com/science/article/pii/S0045782521000773.

[31] *Hand-On Machine Learning with Scikit-learn, Keras, and TensorFlow, 2nd Edition*. O'Reilly Media, Inc, 2019. ISBN: 9781492032649.

[32] R. Hao et al. "Data augmentation based estimation for the censored quantile regression neural network model". In: *Expert Systems with Applications* 214 (2023), p. 119097. ISSN: 0957-4174. DOI: https://doi.org/10.1016/j.eswa.2022.119097. URL: https://www.sciencedirect.com/science/article/pii/S0957417422021157.

[33] A. J. Healey et al. "Toward an improved understanding of thruster dynamics for underwater vehicles". In: *IEEE Journal of Oceanic Engineering* 20.4 (1995), pp. 354–361. DOI: 10.1109/48.468242.

[34] J Holtrop. "The design of propellers". In: *Delft University of Technology* (June 2000). URL: http://resolver.tudelft.nl/uuid:929d90ba-391c-43ff-8548-9dcef1b88901.

[35] J. Hussain. *Deep Learning Black Box Problem*. 2019.

[36] ITTC. *ITTC-Recommended Procedures and Guidelines Model Manufacture, Propeller Models Terminology and Nomenclature for Propeller Geometry COMMENTS OF THE PROPULSION COMMITTEE OF 22 nd ITTC*. 2008.

[37] P. Kadavi, C.-W. Lee, and S. Lee. "Landslide-susceptibility mapping in Gangwon-do, South Korea, using logistic regression and decision tree models". In: *Environmental Earth Sciences* 78 (Feb. 2019). DOI: 10.1007/s12665-019-8119-1.

[38] C. Khosla and B. S. Saini. *Enhancing performance of deep learning models with different data augmentation techniques: A survey*. IEEE, 2020.

[39] S. Kim, S. Kinnas, and W. Du. "Panel Method for Ducted Propellers with Sharp Trailing Edge Duct with Fully Aligned Wake on Blade and Duct". In: *Journal of Marine Science and Engineering* 6 (July 2018), p. 89. DOI: 10.3390/jmse6030089.

[40] D. P. Kingma and J. Ba. "Adam: A method for stochastic optimization". In: *arXiv preprint arXiv:1412.6980* (2014).

[41] O. Kramer. *Dimensionality Reduction by Unsupervised K-Nearest Neighbor Regression*. 2011. DOI: 10.1109/ICMLA.2011.55.

[42] O. Kramer. *K-Nearest Neighbors*. Berlin, Heidelberg, 2013. DOI: 10.1007/978-3-642-38652-7_2. URL: https://doi.org/10.1007/978-3-642-38652-7_2.

[43] N. Kriegeskorte and T. Golan. "Neural network models and deep learning". In: *Current Biology* 29.7 (2019), R231–R236. ISSN: 0960-9822. DOI: https://doi.org/10.1016/j.cub.2019.02.034. URL: https://www.sciencedirect.com/science/article/pii/S0960982219302040.

[44] G. Kuik. "The fluid dynamic basis for actuator disc and rotor theories". In: *The Fluid Dynamic Basis for Actuator Disc and Rotor Theories* (Jan. 2018), pp. 1–133. DOI: 10.3233/978-1-61499-866-2-i.

[45] A. Labach, H. Salehinejad, and S. Valaee. *Survey of Dropout Methods for Deep Neural Networks*. 2019. arXiv: 1904.13310 [cs.NE].

[46] W. P. A. van Lammeren, J. D. van Manen, and M. W. C. Oosterveld. *The Wageningen B-Screw Series*. 1969.

[47] S. Lemm et al. "Introduction to machine learning for brain imaging". In: *Neuroimage* 56.2 (2011), pp. 387–399.

[48] C.-J. Lu, T.-S. Lee, and C.-C. Chiu. "Financial time series forecasting using independent component analysis and support vector regression". In: *Decision Support Systems* 47.2 (2009), pp. 115–125. ISSN: 0167-9236. DOI: https://doi.org/10.1016/j.dss.2009.02.001. URL: https://www.sciencedirect.com/science/article/pii/S0167923609000323.

[49] X. Ma et al. "Initial Margin Simulation with Deep Learning". In: *SSRN Electronic Journal* (Jan. 2019). DOI: 10.2139/ssrn.3357626.

[50] C. Maklin. *LSTM Recurrent Neural Netowrk Keras Example*. June 2019. URL: https://towardsdatascience.com/machine-learning-recurrent-neural-networks-and-long-short-term-memory-lstm-python-keras-example-86001ceaaebc.

[51] M. Maltamo and A. Kangas. *Methods based on k-nearest neighbor regression in the prediction of basal area diameter distribution*. 1998. DOI: 10.1139/x98-085.

[52] J. D. van Manen. "The Choice of the Propeller". In: *Marine Technology and SNAME News* 3.02 (Apr. 1966), pp. 158–171. ISSN: 0025-3316. DOI: 10.5957/mt1.1966.3.2.158. eprint: https://onepetro.org/MTSN/article-pdf/3/02/158/2198767/sname-mtsn-1966-3-2-158.pdf. URL: https://doi.org/10.5957/mt1.1966.3.2.158.

[53] T. Miller. "Explanation in artificial intelligence: Insights from the social sciences". In: *Artificial Intelligence* 267 (2019), pp. 1–38. ISSN: 0004-3702. DOI: https://doi.org/10.1016/j.artint.2018.07.007. URL: https://www.sciencedirect.com/science/article/pii/S0004370218305988.

[54] J. Moreira et al. *Ensemble Learning: A Study on Different Variants of the Dynamic Selection Approach*. July 2009. DOI: 10.1007/978-3-642-03070-3_15.

[55] B. Nguyen, C. Morell, and B. De Baets. *Large-scale distance metric learning for k-nearest neighbors regression*. 2016. DOI: https://doi.org/10.1016/j.neucom.2016.07.005. URL: https://www.sciencedirect.com/science/article/pii/S0925231216307354.

[56] C. Panigutti, A. Perotti, and D. Pedreschi. "Doctor XAI: An Ontology-Based Approach to Black-Box Sequential Data Classification Explanations". In: *Proceedings of the 2020 Conference on Fairness, Accountability, and Transparency*. FAT* '20. Association for Computing Machinery, 2020, 629–639. ISBN: 9781450369367. DOI: 10.1145/3351095.3372855. URL: https://doi.org/10.1145/3351095.3372855.

[57] E. Pekel. *Estimation of soil moisture usin gdecision tree regression*. Feb. 2020. DOI: 10.1007/s00704-019-03048-8. URL: https://doi.org/10.1007/s00704-019-03048-8.

[58] L. Pivano, T. I. Fossen, and T. A. Johansen. "NONLINEAR MODEL IDENTIFICATION OF A MARINE PROPELLER OVER FOUR-QUADRANT OPERATIONS". In: Department of Engineering Cybernetics, 2006. DOI: https://doi.org/10.3182/20060329-3-AU-2901.00045.

[59] L. Pivano, A. Johansen, and Øyvind N Smogeli. "A Four-Quadrant Thrust Estimation Scheme for Marine Propellers: Theory and Experiments". In: *IEEE TRANSACTIONS ON CONTROL SYSTEM TECHNOLOGY* 17 (Y 2008), pp. 215–226. DOI: 10.1109/TCST.2008.922602.

[60] N. Polyzotis et al. "Data validation for machine learning". In: *Proceedings of Machine Learning and Systems* 1 (2019), pp. 334–347.

[61] M. Raissi, P. Perdikaris, and G. Karniadakis. "Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations". In: *Journal of Computational Physics* 378 (2019), pp. 686–707. ISSN: 0021-9991. DOI: https://doi.org/10.1016/j.jcp.2018.10.045. URL: https://www.sciencedirect.com/science/article/pii/S0021999118307125.

[62] C. E. Rasmussen and C. K. I. Williams. *Gaussian Processes for Machine Learning*. The MIT Press, Nov. 2005. ISBN: 9780262256834. DOI: 10.7551/mitpress/3206.001.0001. URL: https://doi.org/10.7551/mitpress/3206.001.0001.

[63] S. S. Rathore and S. Kumar. "A Decision Tree Regression Based Approach for the Number of Software Faults Prediction". In: *SIGSOFT Softw. Eng. Notes* 41.1 (2016), 1–6. ISSN: 0163-5948. DOI: 10.1145/2853073.2853083. URL: https://doi.org/10.1145/2853073.2853083.

[64] Y. Ren, P. Suganthan, and N. Srikanth. *Ensemble methods for wind and solar power forecasting—A state-of-the-art review*. 2015. DOI: https://doi.org/10.1016/j.rser.2015.04.081. URL: https://www.sciencedirect.com/science/article/pii/S1364032115003512.

[65] R. F. Roddy, D. E. Hess, and W. Faller. "Neural network predictions of the 4-quadrant Wageningen propeller series". In: (Apr. 2006), p. 78.

[66] R. F. Roddy, D. E. Hess, and W. Faller. "Utilizing Neural Networks To Predict Forces and Moments On a Submarine Propeller". In: *Maneuvering & Control Div* (Jan. 2008). DOI: https://doi.org/10.2514/6.2008-888.

[67] R. F. Roddy, D. E. Hess, and W. E. Faller. "A Tool to Predict the Four-Quadrant Performance of the Wageningen B-Screw Series for Ship Performance Simulations". In: *Ship Technology Research* 54 (3 July 2007), pp. 103–113. ISSN: 0937-7255. DOI: 10.1179/str.2007.54.3.002. URL: http://www.tandfonline.com/doi/full/10.1179/str.2007.54.3.002.

[68] F. Rosenblatt. "Perceptron Simulation Experiments". In: *Proceedings of the IRE* 48.3 (1960), pp. 301–309. DOI: 10.1109/JRPROC.1960.287598.

[69] S. Ruder. *An overview of gradient descent optimization algorithms*. 2017. arXiv: 1609.04747 [cs.LG].

[70] K. S, V. S, and R. R. *A comparative analysis on linear regression and support vector regression*. 2016. DOI: 10.1109/GET.2016.7916627.

[71] I. H. Sarker. *Deep Learning: A Comprehensive Overview on Techniques, Taxonomy, Applications and Research Directions*. Aug. 2021. DOI: 10.1007/s42979-021-00815-1. URL: https://link.springer.com/article/10.1007/s42979-021-00815-1.

[72] E. Schulz, M. Speekenbrink, and A. Krause. *A tutorial on Gaussian process regression: Modelling, exploring, and exploiting functions*. 2018. DOI: https://doi.org/10.1016/j.jmp.2018.03.001. URL: https://www.sciencedirect.com/science/article/pii/S0022249617302158.

[73] J. Sen, S. Mehtab, and A. Dutta. "Stock Price Prediction Using Machine Learning and LSTM-Based Deep Learning Models". In: (Aug. 2021). DOI: 10.36227/techrxiv.15103602.v1. URL: https://www.techrxiv.org/articles/preprint/Stock_Price_Prediction_Using_Machine_Learning_and_LSTM-Based_Deep_Learning_Models/15103602.

[74] S. Setiowati et al. "A review of optimization method in face recognition: Comparison deep learning and non-deep learning methods". In: Oct. 2017, pp. 1–6. DOI: 10.1109/ICITEED.2017.8250484.

[75] S. Sharma, S. Sharma, and A. Athaiya. "ACTIVATION FUNCTIONS IN NEURAL NETWORKS". In: *International Journal of Engineering Applied Sciences and Technology* 04 (May 2020), pp. 310–316. DOI: 10.33564/IJEAST.2020.v04i12.054.

[76] D. Singh and B. Singh. "Investigating the impact of data normalization on classification performance". In: *Applied Soft Computing* 97 (2020), p. 105524. ISSN: 1568-4946. DOI: https://doi.org/10.1016/j.asoc.2019.105524. URL: https://www.sciencedirect.com/science/article/pii/S1568494619302947.

[77] Ø. N. Smogeli. "Control of Marine Propellers: from Normal to Extreme Conditions". In: 2006.

[78] A. J. Smola and B. Schölkopf. *A tutorial on Support Vector Regression*. Aug. 2004. DOI: 10.1023/B:STCO.0000035301.49549.88. URL: https://doi.org/10.1023/B:STCO.0000035301.49549.88.

[79] C. L. Srinidhi, O. Ciga, and A. L. Martel. "Deep neural network models for computational histopathology: A survey". In: *Medical Image Analysis* 67 (2021), p. 101813. ISSN: 1361-8415. DOI: https://doi.org/10.1016/j.media.2020.101813. URL: https://www.sciencedirect.com/science/article/pii/S1361841520301778.

[80] N. Srivastava et al. "Dropout: a simple way to prevent neural networks from overfitting". In: *The journal of machine learning research* 15.1 (2014), pp. 1929–1958.

[81] D. Stathakis. *How many hidden layers and nodes?* Apr. 2009. DOI: https://doi.org/10.1080/01431160802549278.

[82] L. P. Swetha. *Four quadrant operations of DC Motor*. 2023. URL: https://www.electronicshub.org/four-quadrant-operations-of-dc-motor/.

[83] M. Tran et al. "A practical approach to the dynamic modelling of an underwater vehicle propeller in all four quadrants of operation". In: *Proceedings of the Institution of Mechanical Engineers Part M: Journal of Engineering for the Maritime Environment* 233 (1 Feb. 2019), pp. 333–344. ISSN: 20413084. DOI: 10.1177/1475090217744906.

[84] M. I. Uddin et al. *Fake News Detection Using Machine Learning Ensemble Methods*. 2020. DOI: 10.1155/2020/8885861.

[85] H. Wang and H. Zheng. *Model Validation, Machine Learning*. Ed. by W. Dubitzky et al. New York, NY, 2013. DOI: 10.1007/978-1-4419-9863-7_233. URL: https://doi.org/10.1007/978-1-4419-9863-7_233.

[86] J. Wang. *An Intuitive Tutorial to Gaussian Processes Regression*. 2022. arXiv: 2009.10862 [stat.ML].

[87] B. Widrow and M. Lehr. "30 years of adaptive neural networks: perceptron, Madaline, and backpropagation". In: *Proceedings of the IEEE* 78.9 (1990), pp. 1415–1442. DOI: 10.1109/5.58323.

[88] H. K. Woud and D. Stapersma. *Design of propulsion and electric power generation systems*. IMarEST, Institute of Marine Engineering, Science and Technology, 2002, p. 494. ISBN: 1902536479.

[89] Wärtsilä. *Specialist Propulsion Services Service Guide*. 2008. URL: www.wartsila.com.

[90] M. Xu et al. "Decision tree regression for soft classification of remote sensing data". In: *Remote Sensing of Environment* 97.3 (2005), pp. 322–336. ISSN: 0034-4257. DOI: https://doi.org/10.1016/j.rse.2005.05.008. URL: https://www.sciencedirect.com/science/article/pii/S0034425705001604.

[91] F. Zhang and L. J. O'Donnell. *Chapter 7 - Support vector regression*. Ed. by A. Mechelli and S. Vieira. 2020. DOI: https://doi.org/10.1016/B978-0-12-815739-8.00007-9. URL: https://www.sciencedirect.com/science/article/pii/B9780128157398000079.

[92] J. Zhang et al. *3D Reconstruction for Motion Blurred Images Using Deep Learning-Based Intelligent Systems*. 2021. DOI: 10.32604/cmc.2020.014220. URL: http://www.techscience.com/cmc/v66n2/40688.

[93] L. Zhang, S. Wang, and B. Liu. "Deep learning for sentiment analysis: A survey". In: *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery* 8.4 (2018), e1253.

[94] T. Zong, N. Li, and Z. Zhang. "Active Learning Query Strategies for Linear Regression Based on Efficient Global Optimization". In: *Journal of Electrical and Computer Engineering* 2022 (June 2022). DOI: 10.1155/2022/2891463.

myreferences

Colophon

This document was typeset using LaTeX, using the KOMA-Script class `scrbook`. The main font is Palatino.