applied sciences

MDPI

Article

Assessment of Hull and Propeller Degradation Due to Biofouling Using Tree-Based Models

Nikos Themelis [1,*], George Nikolaidis [1,2] and Vasilios Zagkas [2]

[1] School of Naval Architecture and Marine Engineering, National Technical University of Athens, 15780 Athens, Greece; nikolaidis@mail.ntua.gr
[2] SimFWD, 15232 Chalandri, Greece; vzagkas@simfwd.com
* Correspondence: nthemelis@naval.ntua.gr

Abstract: A hull and propeller biofouling assessment framework is presented and demonstrated using a bulk carrier as a case study corresponding to an operational period of two and a half years. The aim is to support the decision-making process for optimizing maintenance related to hull and propeller cleaning actions. For the degradation assessment, an appropriate key performance indicator is defined comparing the expected shaft power required with the measured power under the same operational conditions. The power prediction models are data-driven based on machine learning algorithms. The process includes feature engineering, filtering, and data smoothing, while an evaluation of regression algorithms of the decision tree family is performed. The extra trees algorithm was selected, presenting a mean absolute percentage error of 1.1%. The analysis incorporates two prediction models corresponding to two different approaches. In the first, the model is employed as a reference performance baseline representing the clean vessel. When applied to a dataset reflecting advanced stages of biofouling, an average power increase of 11.3% is predicted. In the second approach, the model entails a temporal feature enabling the examination of scenarios at different points in time. Considering synthetic data corresponding to 300 days since hull cleaning, it was derived that the fouled vessel required an average 20.5% increase in power.

Keywords: ship energy efficiency; hull and propeller biofouling; data analysis; data-driven models; maintenance optimization

Citation: Themelis, N.; Nikolaidis, G.; Zagkas, V. Assessment of Hull and Propeller Degradation Due to Biofouling Using Tree-Based Models. *Appl. Sci.* **2024**, *14*, 9363. https://doi.org/10.3390/app14209363

Academic Editor: Pedro Couto

Received: 17 September 2024
Revised: 4 October 2024
Accepted: 10 October 2024
Published: 14 October 2024

1. Introduction

To mitigate the effects of global climate change, there are increasing pressures on the maritime industry to reduce the environmental footprint of shipping. As early as 2011, the International Maritime Organization (IMO) adopted mandatory measures with the aim of optimizing the energy efficiency of ships. Since then, the IMO has been taking further action. Specifically, the IMO has already set a strategy to reduce greenhouse gases related to ship operations based on short-, medium-, and long-term measures [1], while the most recent were set out in the July 2023 revised strategy [2], which targets a 20% reduction in ship emissions by 2030 and a 70% reduction by 2040 compared to 2008 emission levels with the ultimate goal of near-zero emissions by 2050.

However, beyond the regulatory framework that steers shipping toward more sustainable solutions, a more general interest is observed from the various parties involved (e.g., ship owners, operators, and charterers) to optimize the operational performance of ships, reduce fuel consumption, and, ultimately, reduce emissions. As many studies show (e.g., [3]), the fuel cost constitutes by far the greatest part (up to 50–70%) of ships' operational cost and, therefore, the reduction of fuel consumption per voyage is a priority for all shipping companies. For example, reducing it by just 1% can result in hundreds of thousands of dollars saved per year in the operation of large ships. An overview of the reduction potential achieved by the candidate measures is presented in [4,5].

An operational measure that has been on the spot in the industry is the continuous monitoring of hull and propeller performance since it significantly affects fuel consumption used for propulsion. According to [6], fuel consumption is strongly related to the ship's drag force and, in particular, to the frictional resistance, which is affected by vessel hull biofouling. Biofouling is generated by the buildup of micro- and macro-organisms onto the ship hull. This results in an increase of the surface roughness and subsequently to an increase in the thickness of the boundary layer deteriorating the frictional resistance. In addition, an increase in the boundary layer thickness reduces the velocity of the water reaching the propeller, causing a reduction in the wake fraction. Finally, the development of biofouling on the propeller blades modifies their original profile and roughness, causing increases in the blade resistance and in the propeller torque, thus changing the propeller characteristics in free flow and resulting in an additional efficiency loss. Therefore, the development of biofouling on the hull and propeller has a significant effect on the overall hydrodynamic performance of a ship, with relevant research showing that thin biofilm and algae growth (slime, algae biofilm) cause a reduction in power efficiency of more than 10%, while the growth of calcareous/hard fouling (barnacles) can lead to an 80% increase in power request [7]. According to [8,9], the generation of biofouling on ship hulls is influenced by several factors, such as the operational conditions (speed and time moored or at anchor), the trading routes, and the seawater characteristics, as well as the maintenance actions concerning the anti-fouling coating as well as the vessel hull cleaning frequency.

A standard approach for monitoring the effect of biofouling development on ship performance is achieved by calculating appropriate key performance indicators (KPIs). The KPIs act as a measure of the ship's actual performance using operational data against the ideal reference performance typically obtained from sea trials. This is the standard approach of ISO 19030 ([10]), which presents a default method using a physics-based approach to address the biofouling effect on hull and propeller performance. However, ISO 19030, when compared with more advanced models based on data-driven methods, is less accurate and provides results with higher uncertainty (see for example [11,12]). A more detailed comparison of physical models (PMs) and data-driven models (DDMs) is provided by the comprehensive review of related works in [13], where it is mentioned that DDMs provide more accurate near- and real-time predictions but require computationally expensive training and can occasionally yield physically implausible results due to their statistical nature.

The current study exploits data-driven models based on machine learning algorithms for the development of a hull and propeller biofouling assessment framework, which could support optimization of the decision-making process for maintenance actions. From a practical viewpoint, the usage of DDMs for the assessment of the fouling status can present several challenges. Firstly, the available dataset can contain several years of operation, including maintenance actions, that affect the ship's performance. The methods and the respective models needed for the assessment of the fouling status require the proper selection of datasets to be used for training. This aspect is not significantly highlighted in previous works and, in the current study, we aim to address several issues related to the choice for model development and dataset selection when defining baseline performance or evaluating performance over time. Specifically, we analyze two methods, each offering a unique approach to address the performance assessment problem. Furthermore, the aspects of data preparation and feature selection, based on the available parameters, also require in-depth analysis. Therefore, the objectives of the study are summarized as follows:

- To prepare a dataset by properly identifying the time periods that would entail meaningful information for the fouling problem by employing a suitable KPI.
- To appropriately compile multiple data sources to enhance, complete, and address any missing data.
- To apply suitable data pre-processing techniques to derive a quality dataset that will focus on the physics of the problem.
- To develop ML models aimed at capturing the impact of fouling development over time.

- To identify the key features of the models by using informed statistical dependencies but also domain knowledge.
- To evaluate various regression algorithms from the decision tree family and critically explore the hyperparameter space.
- To use the models for a practical assessment that will assist in the optimization of maintenance.

Therefore, the key contribution of the paper is the development of a methodology for the assessment of the biofouling impact on hull and propeller performance that systematically tackles the previously mentioned issues. Specifically, the paper aims to highlight the need to consider the physics of the problem for the informed development and application of ML models.

The structure of the paper is as follows. In Section 2, a literature review of related works is provided, while Section 3 presents the ship used in the case study as well as the available sources of data acquisition utilized for the dataset compilation. Moreover, a comparison study of key statistics of the basic parameters of the problem is presented to examine the level of agreement and the activities needed for the completion of the database. Finally, the maintenance actions that occurred during the reporting period are defined, while an initial verification of these actions is performed using a KPI to reflect the performance changes at specific times.

Section 4 describes the steps for developing the data-driven models and, specifically, the feature engineering related to the creation of new parameters and the identification of those more relevant to the problem. Then, the data preparation is presented concerning the application of a proper filtering and smoothing method. The central focus of this section is dedicated to showing the development of the model through the application of multiple ML algorithms and the fine-tuning of hyperparameters using quantified metrics. Ultimately, two distinct models are presented. Section 5 utilizes the models for the fouling assessment. Particularly, the KPI to be used is defined, while the methodology aims to calculate the additional daily fuel cost due to biofouling, considering the time component of biofouling growth. Such information is essential for an efficient decision-making process to select the time of maintenance actions. Finally, a thorough discussion of the limitations and advantages of the study is presented, and the key conclusions follow.

2. Literature Review

Focusing on the development of DDMs, several issues have been examined in previous works related to models aiming at the prediction of fuel consumption. The testing of the prediction accuracy of various machine learning and statistical methods using the same dataset is a typical research approach, as in the work of [14], where XGBoost, artificial neural networks (ANN), support vector regression, as well as statistical methods such as linear and polynomial regression and a generalized additive model are examined. The root mean square error and the complexity degree are used to compare the examined methods. Similarly, [15] examined different predictive models, such as multiple linear regression, ridge and LASSO regression, support vector regression, tree-based algorithms, and boosting algorithms, using data from noon reports and engine logbooks. For the validation of the predictive models, the K-fold cross-validation method was used, and a correlation analysis to identify the relationships among the different variables was carried out.

A significant aspect is the selection of the features of the model. In [16], for the selection of the dependent variables to be used in model development, the "domain knowledge" and "statistical method based on LASSO regularization" methods were examined. The evaluation of these methods was based on the calculation of the mean absolute error, considering several combinations of variable feature selections and model training methods. In [17], the impact of different feature sets on the model's predictive accuracy was evaluated. The authors compared the error metrics obtained through cross-validation and testing of models trained with various combinations of features. On the other hand, in [18] a decision tree (DT) was used for the evaluation of feature importance and the selection of the key

influencing factors of ship fuel consumption. The prediction model was developed by incorporating attention mechanisms into bidirectional long short-term memory (Bi-LSTM) networks. Bi-LSTM networks are a type of recurrent neural network (RNN) that can capture long-term dependencies in time series data. They consist of two LSTM layers that process the data in both forward and backward directions, allowing the network to have more context and a better understanding of the data stream. Attention mechanisms enable neural networks to focus on specific parts of the input data when making predictions. In addition, [19] examined several ML algorithms (multiple linear regression, LASSO, extra trees, XGBoost, LightGBT, and CatBoost) combined with a physics-based model (or else a white box model) for the prediction of ship fuel consumption. The latter is used as a new feature compared to the above-mentioned black box models and thus their combination is referred to as a gray box model. It was derived that the ensemble learning models examined, especially the gray box ones, present better prediction performance and generalization capabilities.

In addition, the data-preprocessing phase is also a demanding task. For data cleaning, a suitable statistical procedure involving median filtering is implemented in [17], while in [20], measurements of each parameter that deviate more than three standard deviations from the mean are discarded. Another significant aspect is the generalization capability of DDMs compared with PMs. For example, in [21], an ANN was developed utilizing a dataset of two sea voyages, thus limiting the model applicability to the encountered operational conditions. Moreover, in [12], several ANN models based on different features for the assessment of hull and propeller degradation were developed. For feature selection, a random forest regressor was applied to evaluate the significance of each feature using p-value criterion. Data pre-processing was also carried out, including outlier removal, using a threshold deviation among two correlated parameters, while smoothing of the signals was performed using the simple moving average. A KPI for the power increase using the ANN models was compared against the KPI of ISO 19030, revealing superior prediction accuracy.

On the other hand, the monitoring of fouling status employing DDMs was explored in [22]. Specifically, this work employed ML algorithms to predict propulsion power and quantify the performance degradation due to fouling using high-frequency data from a containership spanning 12 months. The Days Since Cleaning (DSC) feature was introduced to monitor fouling. To determine the best performing model, four ML algorithms were evaluated: K-nearest neighbors, decision tree, random forest, and ANN. The hyperparameters of each model were tuned using grid search and cross validation. The best performance was achieved by random forest, with a mean absolute percentage error (MAPE) of 1.17% on the test set. The gain in prediction accuracy was examined when DSC and significant wave height features were included. In [17], the number of days passed since the last hull maintenance was incorporated as a feature to account for the time-dependent nature of biofouling growth and its effect on vessel performance. Furthermore, [23] examined the performance degradation expressed by speed loss using a two-stage optimization algorithm that combined a genetic algorithm (GA) and long short-term memory (LSTM) network. Specifically, the GA algorithm was used simultaneously for the optimization of the internal parameters of the LSTM models, such as the number of neurons and the batch size, but also for feature selection. The model predicted a significant speed loss of 1.26 kn within 17 months of ship operation. Finally, [24] developed an advanced approach for the selection of features by implementing domain knowledge as well as correlation and recursive feature elimination with cross-validation (5 folds) to evaluate the importance of variables with the target value. The study presented a model for power prediction, which consisted of three sub-models corresponding to propeller rpm, hull fouling, and environmental conditions. Emphasis was given to the fouling sub-model that predicted hull roughness considering idling times and sea temperature. A comparison of this approach with several data-driven models (such as ridge, ANN, and random forest) using the same input variables was also carried out.

3. The Case Study and Preliminary Assessment

3.1. The Examined Ship and Data Acquisition

The examined ship is a Kamsarmax-type bulk carrier, with the main characteristics shown in Table 1. A dataset generated by the on-board high frequency data-acquisition system (DAQ) was used, spanning a period of two and a half years, specifically from February 2021 to July 2023. The sampling period was one minute, so it consists of 1,311,000 synchronized datapoints. From the available parameters, 268 were derived for analysis by excluding parameters used by the DAQ provider for internal purposes.

Table 1. Main characteristics of the examined ship.

Parameter	Value
Length overall	229.00 m
Length between perpendiculars	225.50 m
Breadth, moulded	32.26 m
Depth, moulded	20.05 m
Summer load line draught, moulded	14.45 m
Deadweight at summer load draught	80,996.1 t
Main Engine	HYUNDAI-MAN B&W—6S60ME-C8.5
Maximum continuous rating (MCRME)	9930 kW × 90.4 rpm

When examining the data, it appeared that some parameters were not available throughout the recording period. For instance, draft measurements were valid until November 2021, since after that date the sensors failed and produced faulty values. Available weather provider data started from April 2022. Moreover, the main engine fuel oil supply volumetric flow meter malfunctioned between November 2021 and July 2022, as well as between September 2022 and January 2023, due to mechanical failure.

In addition to the high-frequency data, a noon report dataset was available for the same period, consisting of 930 datapoints (Table 2). Even though it contained fewer parameters, in the current analysis, noon reports were used to fill in missing values from the high-frequency data in cases where similar parameters existed. As an example, draft and weather parameters from noon reports were filled in per minute based on their daily values to have the same time resolution as the high-frequency data. By merging these filled in noon report parameters with the original high-frequency data, a new dataset was created.

Table 2. Basic information about the available datasets.

	High-Frequency Data	Weather Provider Data	Noon Reports
Period	February 2021 to July 2023	April 2022 to July 2023	February 2021 to July 2023
Sampling interval	1 min	1 h	1 day
Number of points	1,311,000	21,850	930

Weather data from a third-party commercial provider had an actual time resolution of one hour. To align these data with the rest of the dataset, the DAQ provider modified the time resolution, converting it to one minute. Available parameters included sea temperature, air pressure and temperature, wind speed and direction, and total current speed and direction.

As mentioned previously, while automated logging is certainly more reliable due to the higher frequency and accuracy of the measurements, noon report information was available throughout the examined period, and it was decided to be considered in the cases where high-frequency recording was missing. Having created a dataset containing ship

operation parameters from both high-frequency recording and noon reports, the first step of the analysis was to compare the available parameters from both of these sources.

Mean draft is calculated as the average of the fore and aft draft measurements. High-frequency draft measurements were valid from February to November 2021. Figure 1 shows the relative frequency distribution of the difference in the mean drafts obtained by the noon reports and the respective mean draft values from the on-board high-frequency sensors during this period. A mean difference of −0.54 m was calculated, which accounts for bias, as well as a standard deviation of 1.41 m, because of the high-frequency recording variance.

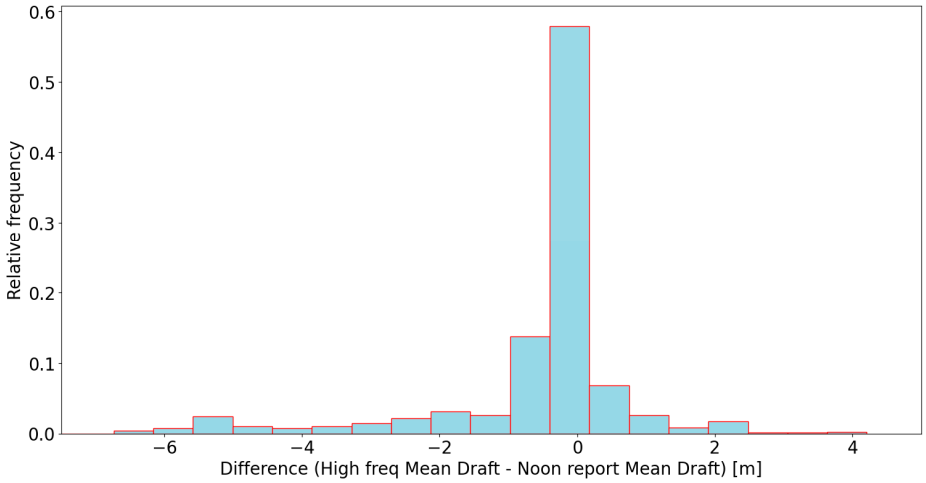


Figure 1. Relative frequency distribution of the difference between the mean draft from high-frequency data and noon reports.

Moreover, data from the weather provider were available from April 2022 to February 2023. Figure 2 shows the relative frequency distribution of the difference of the sea height values from the noon reports and the significant wave height obtained from the weather provider during this period, where both parameters represent the combined effect of swell and wind waves. A mean of −0.02 m and a standard deviation of 0.84 m were calculated. Based on the previous findings, the noon reports draft and sea height parameters were then used instead of their high-frequency counterparts. Another option would be to use high-frequency values where available, and noon reports where high-frequency values are not available, but this would lead to an uneven analysis.

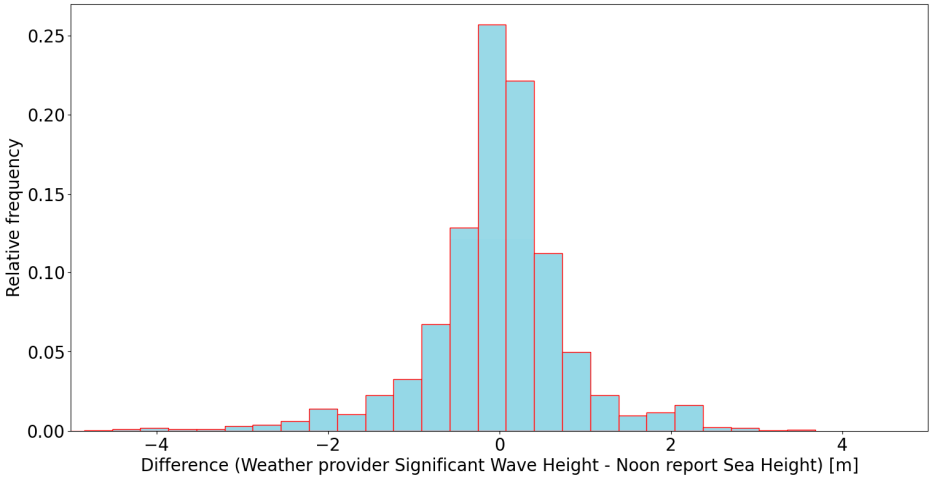


Figure 2. Relative frequency distribution of the difference between the significant wave height from the weather provider and sea height from noon reports.

Another useful tool for exploring the ship's operation is to project its path on the world map by simply superimposing the GPS longitude and latitude signals on the map, as shown in Figure 3. This helps to identify the routes followed by the ship during the two-and-a-half-year recording period, revealing the worldwide ship operation.

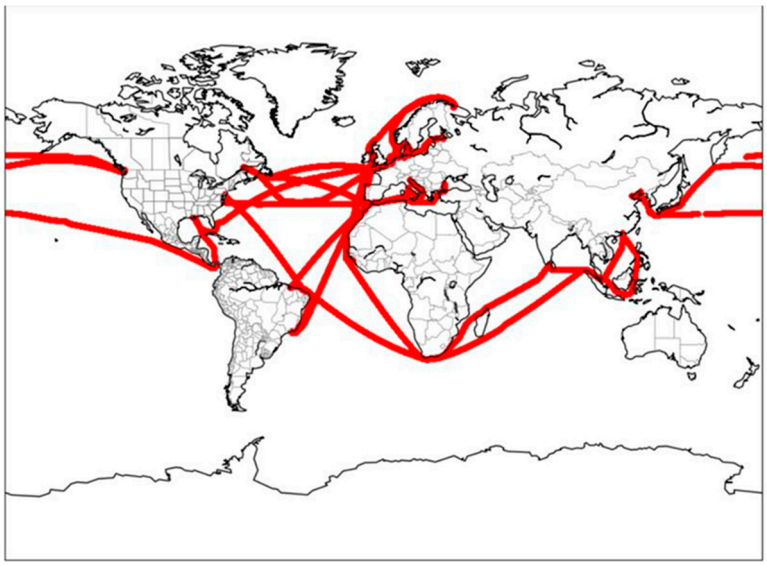


Figure 3. Projection of the ship's path on the world map during the recording period, utilizing high-frequency GPS signals.

3.2. Definition of Maintenance Actions and Respective Datasets

Due to fouling accumulation, vessel hydrodynamic performance may differ drastically at different points in the operation timeline. To assess the changes in vessel performance over time, it is necessary to know the dates of the various maintenance events that took place during the reporting period. In Figure 4, the vessel's operation timeline is shown, containing the date of each important event. The dataset is split, based on maintenance events, into four distinct datasets (A, B, C, D), as described in Table 3. During Dataset A, the propeller was damaged, according to the operator, so the performance was significantly deteriorated and non-representative of the vessel under normal conditions. Dataset B represents a period during which marine growth accumulated. Due to the short duration between cleaning events, Dataset C captures a limited period where fouling build-up was not substantial. Dataset D, spanning over a year, contains the most appropriate data for studying fouling accumulation due to its extended timeframe. Datasets D1 and D2 are subsets of Dataset D: the former spans the first 2.5 months of D, the latter spans the last 3 months. Table 3 also contains the percentage of time during each period the ship remained at port obtained by the noon report data.

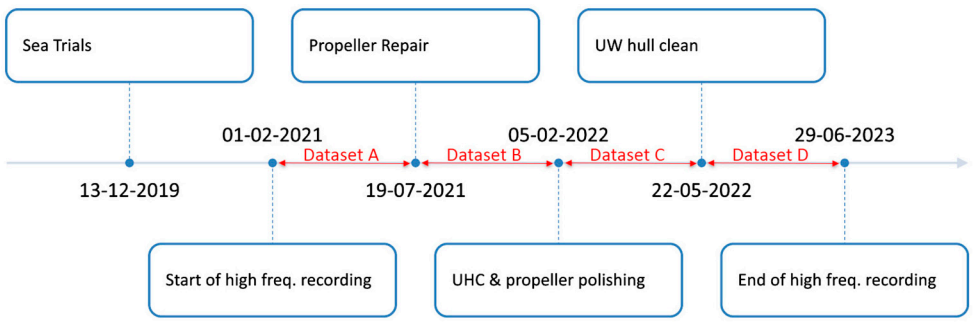


Figure 4. Timeline of the most important events during the vessel's operation.

Table 3. Definition of datasets based on the dates of the main maintenance events. Datasets D1 and D2, are subsets of Dataset D.

Dataset ID	from	to	% of Time in Port
Dataset A	5 February 2021	19 July 2021	50.8
Dataset B	21 July 2021	8 January 2022	54.7
Dataset C	5 February 2022	21 May 2022	42.8
Dataset D	22 May 2022	29 June 2023	37.4
Dataset D1	22 May 2022	8 August 2022	25.6
Dataset D2	23 March 2023	29 June 2023	29.2

As a first approach to assess the impact of hull and propeller fouling on ship performance, a typical KPI was utilized to examine its change over time. The KPI employed in this study was a customary propeller loading coefficient, defined as follows (e.g., [25]):

$$KPI = P/N^3 \quad (1)$$

where P is the ME power in kW and N is the propeller rate of revolution in RPM. This KPI reflects the propeller's loading condition. When the hull and propeller become fouled, the increased resistance leads to a higher thrust requirement, resulting in a rise in the ratio as the power must increase to overcome the additional drag, while the rotational speed may not change significantly to maintain the desired ship speed. Therefore, plotting the KPI over time provides a visual indication of the changes in propeller loading due to the accumulation of fouling on the hull and propeller. In Figure 5, the KPI is plotted throughout the reporting period using separate colors for laden (green) and ballast (blue) conditions, demonstrating that as the draft increases, the KPI increases as well. The maintenance event dates are also shown in the plot, based on which the dataset is split into individual periods. Due to the high variance caused by transients and various loading and weather conditions, it is difficult to directly interpret the variation in the KPI. Therefore, a first order trendline was fitted over the data available in each period in between two consecutive maintenance events. The uptrend observed in each period trendline accounts for the hull and propeller deterioration due to fouling. In addition, the drop in the KPI noticed after each maintenance event represents the improvement in the vessel performance achieved because of the maintenance itself.

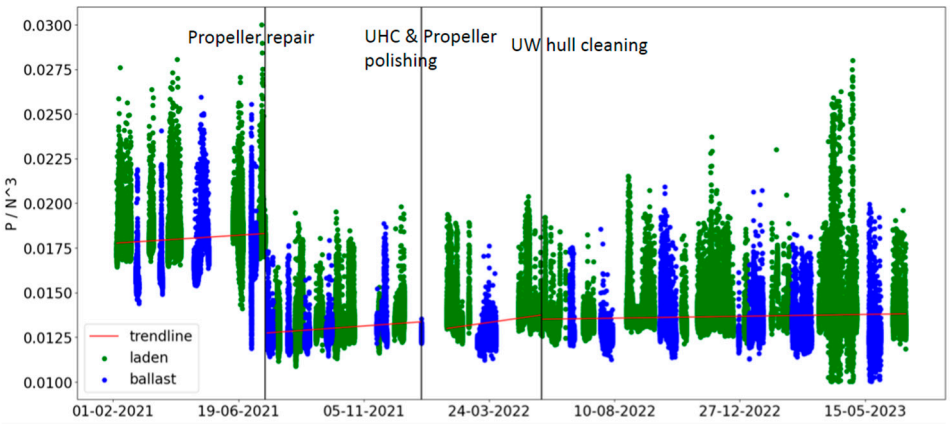


Figure 5. Utilization of the propeller loading coefficient (Equation (1)) to monitor the status of the hull and the propeller performance, as well as to track the main maintenance events.

Despite being intuitive and easy to implement, this approach presents a few shortcomings. First, it fails to answer the crucial question of how much power is saved because of a maintenance event, and second, it does not account for the various factors affecting

ME power consumption. Power normalization methods for environmental factors exist (e.g., [10]), but they serve as a coarse approximation since they rely on assumptions that introduce uncertainty. For a more accurate and holistic approach to the problem, sophisticated data-driven methods ought to be employed. Using a well-tuned, highly accurate black box model, the vessel's behavior in various conditions can be predicted.

4. Development of Data-Driven Models

4.1. Methodology

Data-driven models developed using the available operational data and suitable machine learning (ML) algorithms are henceforth referred to as ML models. Figure 6 illustrates a generic workflow of the ML-based framework for assessing ship biofouling performance. Given the compromised state of the fuel oil consumption (FOC) measurements, shaft horsepower (SHP) was selected as the output variable of the ML model. The input variables that feed the model to predict ME SHP are termed features. Therefore, following data acquisition, the next step was feature engineering, which consists of feature construction and feature selection. The data preparation step addresses smoothing and filtering to derive a dataset free from anomalies that focuses on the desired operational conditions. Subsequently, various ML algorithms were evaluated, and different hyperparameter combinations were explored using state-of-the-art techniques to develop a model with high predictive accuracy. Hull and propeller performance assessment was achieved by employing the ML model to evaluate the expected ship performance.

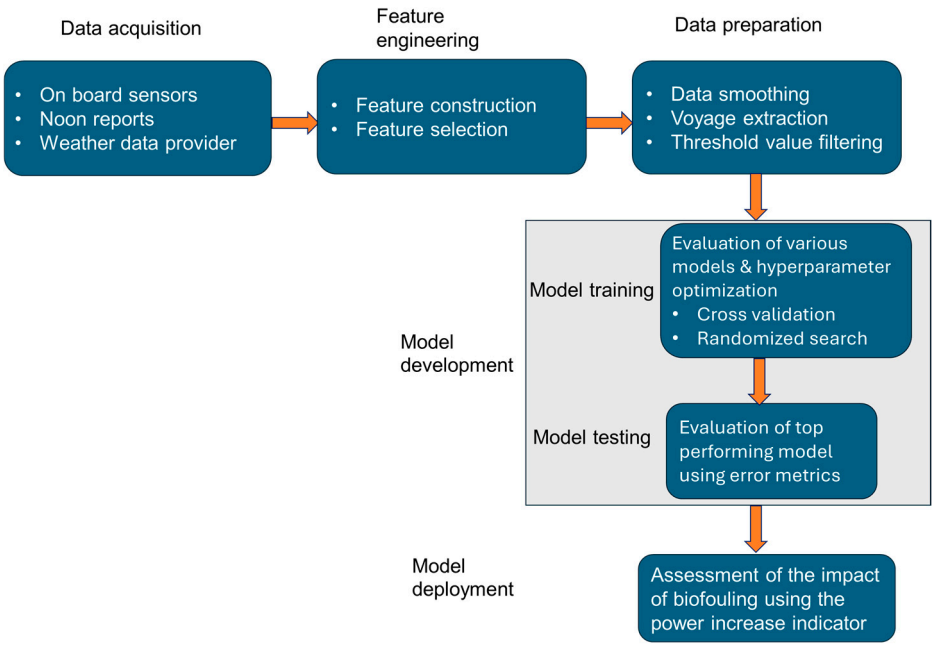


Figure 6. Flow diagram of the calculation framework for the biofouling assessment.

4.2. Feature Engineering

Feature engineering requires domain knowledge and comprises feature construction and feature selection. In feature construction, new variables are built using existing ones for a more accurate physical description of the problem. Specifically, aft and fore draft measurements are combined to calculate the mean draft and trim of the vessel, according to Equations (2) and (3). In addition, the sea current speed is calculated as the difference between speed over ground (SOG) and speed through water (STW), as stated in Equation (4). The relative wind speed and direction measurements are combined to determine the longitudinal and transverse components of the relative wind speed, according to Equations (5) and (6). Similarly, in Equation (7) the relative wave angle is calculated to determine the longitudinal and transverse components of the significant wave height,

according to Equations (8) and (9). To account for the strong time dependence of fouling growth, the Days feature was added, which is equal to the number of days passed since the last hull and propeller maintenance event. For each data point recorded on a given date, the Days feature was calculated according to Equation (10). Thus, for each dataset mentioned in Table 3, the Days feature was created separately, based on the respective starting date.

$$\text{Mean draft} = \frac{\text{Fore Draft} + \text{Aft Draft}}{2} \tag{2}$$

$$\text{Trim} = \text{Fore Draft} - \text{Aft Draft} \tag{3}$$

$$\text{Current} = \text{SOG} - \text{STW} \tag{4}$$

$$\text{Wind x} = \text{Relative Wind Speed} \cdot \cos(\text{Relative Wind Angle}) \tag{5}$$

$$\text{Wind y} = \text{Relative Wind Speed} \cdot \sin(\text{Relative Wind Angle}) \tag{6}$$

$$\text{Relative Wave Angle} = \text{abs}(\text{Vessel Heading} - \text{WaveDirection}) \tag{7}$$

$$\text{Wave x} = \text{Wave Height} \cdot \cos(\text{Relative Wave Angle}) \tag{8}$$

$$\text{Wave y} = \text{Wave Height} \cdot \sin(\text{Relative Wave Angle}) \tag{9}$$

$$\text{Days} = \text{Date} - \text{Date of last maintenance event} \tag{10}$$

In feature selection, the most important input variables are determined with the help of appropriate statistical tools, whereas irrelevant or redundant input variables are eliminated to simplify the model and prevent overfitting. To determine the necessary features, two criteria are used: comprehension of the physical problem and correlation analysis. Based on domain knowledge, among all available variables, a few were shortlisted as feature candidates for SHP prediction. Then, correlation analysis was performed on those candidates using the distance correlation method, with the results presented in Table 4. Distance correlation was presented by [26] and constitutes a measure of statistical dependence between two random variables that captures both linear and nonlinear relationships.

Table 4. Calculation of the distance correlation coefficient between SHP and the candidate features.

Feature	Unit	Distance Correlation Coefficient
RPM	rpm	0.8620
STW	kn	0.4509
Mean Draft	m	0.3014
Trim	m	0.3724
RoT	deg/min	0.0778
Wave	m	0.1434
Wind	m/s	0.1816
Wave x	m	0.2007
Wave y	m	0.1115
Wind x	m/s	0.2252

Table 4. *Cont.*

Feature	Unit	Distance Correlation Coefficient
Wind y	m/s	0.0939
Current	kn	0.1737
Days	days	0.2895

While traditional correlation measures such as Pearson's correlation coefficient are effective for capturing linear relationships, they often fail to detect complex dependencies that may exist in the data. Distance correlation overcomes this limitation by considering the distances between observations rather than assuming a specific functional form. It provides a robust measure of dependence that can reveal intricate relationships that may be missed by traditional methods. The concept of distance correlation is based on the notions of distance covariance and distance variance. The distance covariance quantifies the similarity of paired observations in terms of their distances to other observations, while the distance variance measures the dispersion of the distance between paired observations. By normalizing the distance covariance with the square root of the product of the distance variances, the distance correlation is obtained. The resulting measure ranges between 0 and 1, with 0 indicating independence and 1 indicating perfect dependence. This makes the distance correlation a valuable tool for feature selection, as it captures a wide range of relationships and provides a comprehensive understanding of the data. Specifically, the distance covariance between two random variables X and Y is defined as the square root of their distance variance, as follows:

$$dCov(X,Y) = \sqrt{dVar(X,Y)} \tag{11}$$

The distance variance between X and Y measures the dispersion of the distances between paired observations. It is calculated as follows:

$$dVar(X,Y) = \frac{1}{n^2}\sum_{i=1}^{n}\sum_{j=1}^{n} S(X_i,X_j)S(Y_i,Y_j) \tag{12}$$

where $S(X_i,X_j)$ and $S(Y_i,Y_j)$ represent the Euclidean distances between observations X_i and X_j, and between Y_i and Y_j, respectively, and n represents the sample size. The distance correlation between X and Y is obtained by normalizing the distance covariance $dCov(X,Y)$ with the square root of the product of the distance variances $dVar(X)$ and $dVar(Y)$ of X and Y, respectively, as follows:

$$dCor(X,Y) = \frac{dCov(X,Y)}{\sqrt{dVar(X)dVar(Y)}} \tag{13}$$

The correlation analysis confirmed what was expected: RPM and STW were the candidate features with the highest correlation values with the SHP variable, which highlighted their importance as predictors. However, the RPM variable was not included as a feature because it would limit the applicability of the model as an emulator, even though it would increase its predictive accuracy. It is worth mentioning that the correlation coefficient values of the longitudinal components of the relative wind speed and the significant wave height were higher than the respective values of both the transverse components and the original variables. In addition, it is worth noting that although the correlation coefficient of the rate of turn (RoT) variable had a small value, it was useful to include a variable related to maneuvering. On the other hand, current speed, resulting in the difference between SOG and STW, did not directly impact the hydrodynamics and ship speed-power performance. However, according to [27], the wave–current interaction can affect the wave height, as the wave field significantly changes as it interacts with following and opposing

currents. Therefore, it might be assumed that the current speed can indirectly affect the ship's resistance and propulsion power requirements. On the other hand, the correlation analysis revealed a small but non-negligible relationship between current speed and SHP, with a correlation coefficient similar in magnitude to other environmental factors like wave height and wind speed. This suggests that current, as a proxy for wave-current interaction, may provide useful information to the model in predicting the ship's performance.

Therefore, feature selection was based on a combination of domain knowledge and correlation analysis using the distance correlation method. Eventually, two sets of features were selected, one including the Days feature and the other not including it, as listed in Table 5. The models derived by each feature set will be used in different ways, as will be described next, to monitor the hull and propeller condition.

Table 5. List of the selected features with or without the Days feature.

Feature Set w/o Days	Feature Set w/Days
STW	STW
Mean Draft	Mean Draft
Trim	Trim
RoT	RoT
Wind x	Wind x
Wave x	Wave x
Current	Current
-	Days

4.3. Data Preparation

Before feeding the data to an ML algorithm, suitable preparation is required to ensure their quality. In the present study, data preparation was implemented in three steps:

- Data smoothing
- Voyage extraction
- Threshold value filtering

These processes are described hereafter.

Data smoothing aims to remove signal noise while preserving the main trend. To do this, first, a simple moving average (SMA) filter is applied to each signal (features and output variables). The signal is denoted as X(i), where i represents the data sample at a certain time. The equation for calculating the SMA at time i, denoted as SMA(i), with a window size of k can be expressed as follows:

$$SMA(i) = \frac{X(i) + X(i-1) + X(i-2) + \cdots + X(i-k+1)}{k} \quad (14)$$

Then, each SMA-smoothed signal is decimated by a factor of k, i.e., only one sample out of k samples is retained, where k is equal to the window size used for the moving average. In Figure 7, window sizes of 5, 10, and 15 for two variables are compared to show the effect of k. The selected sizes are commonly used in the literature; for example, [11,12,17,24] chose a 5 min window, while [28,29] chose 10 and 15 min windows, respectively. Clearly, the wider the time window, the more intense the effect of the smoothing. In the current study, we visually compare the effect of these window sizes by inspecting the time series of the original signals and the smoothed signals. Based on our observations, a window size of 5 min effectively reduces the sharpest spikes while maintaining sufficient resolution for accurate model training. Since it is the smallest window of the three, it preserves higher data resolution, providing more data points for the machine learning models.

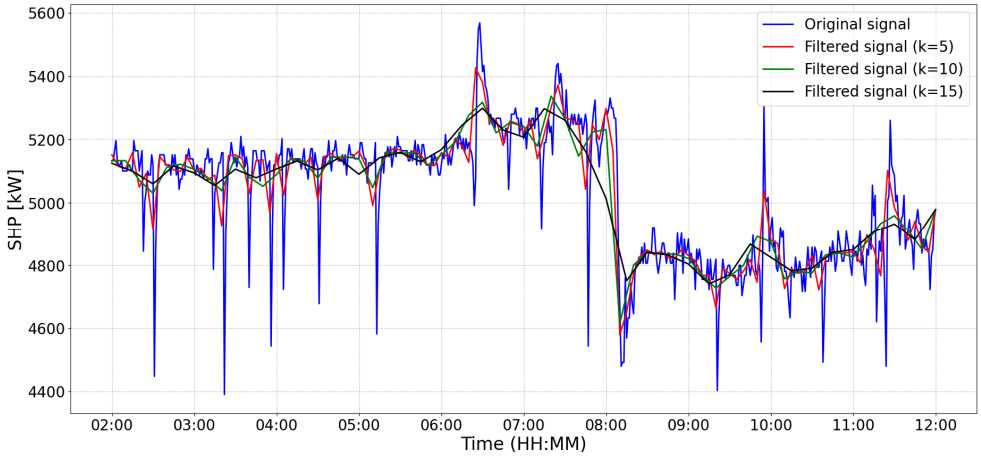


Figure 7. Comparison of SHP signals over a 10 h time interval, smoothed using a simple moving average with different window sizes (k = 5, 10, 15 min).

Voyage extraction: An assessment of the vessel's hydrodynamic performance makes sense only in sailing conditions. Thus, it is important to identify in the data records the time intervals during which the ship was traveling. With voyage extraction, the identification of time intervals in the dataset corresponding to actual ship voyages, therefore excluding port calls, is achieved. Thus, a list of the start and end date/time of voyages during the reporting period is derived from noon reports. Subsequently, a dataset containing only voyages is extracted.

Threshold value filtering involves the rejection of data points based on threshold values. In such a way, only the values inside the operational range of choice are kept, ensuring that the data analysis is focused on specific conditions related to open-sea operation. Therefore, certain criteria are subsequently applied to the dataset, as presented by the lower and upper limits of Table 6. For the definition of these values, the histogram of each variable is examined to identify the tails of the distributions, as these represent unrepresentative conditions occurring over short periods ([30]). Therefore, the chosen thresholds ensure that only data points within the operational range of interest are retained.

Table 6. Threshold values used for filtering the ship's navigation variables.

Variable	Unit	Lower Limit	Upper Limit
SHP	kW	3000	-
STW	kn	6	-
RoT	deg/min	−6	6
Current	kn	−3	2

These data preparation steps caused a progressive reduction of the number of data points with respect to the raw data, as shown in Figure 8, where the number of points after each step is indicated, showing that the greatest reduction occurred during the smoothing process. This reduction was a side effect deemed necessary to eliminate spikes and reduce the noise present in the raw data.

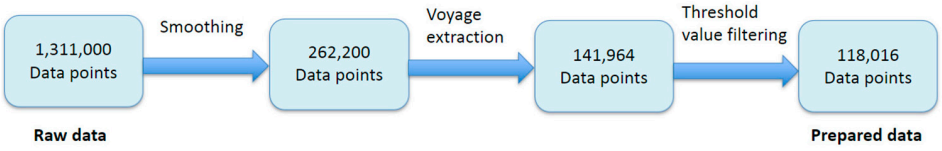


Figure 8. Reduction in the number of datapoints at each stage of data preparation for the entire dataset.

4.4. ML Model Calculations

The main idea for the development of Model-1 was to employ it as a reference performance baseline, representing the vessel in clean condition, to evaluate conditions with fouling, as presented by Datasets B and D2. Model-1, based on the feature set without the Days feature, was trained on data corresponding to a few months after hull cleaning and propeller polishing (Dataset D1) so that it emulated the behavior of the vessel when it is clean. A basic assumption in this approach was the unchanged fouling status of the vessel during this period. Another issue to consider was the relatively short duration of Dataset D1 used for model training. This limited training period may not have captured the operational conditions encountered by the vessel during Datasets B and D2. Figures 9 and 10 present the relative frequency distributions of STW and mean draft, two key operation parameters for the problem studied, while Table 7 shows the basic statistics for STW. The STW values did not differ significantly among the examined datasets, even though Dataset B presented higher STW values. However, the under-representation of Dataset D1 in mean drafts in the range of 8 m and 11 m may have led to possible inaccurate extrapolated predictions for these drafts. Therefore, a second approach was also examined, as discussed next.

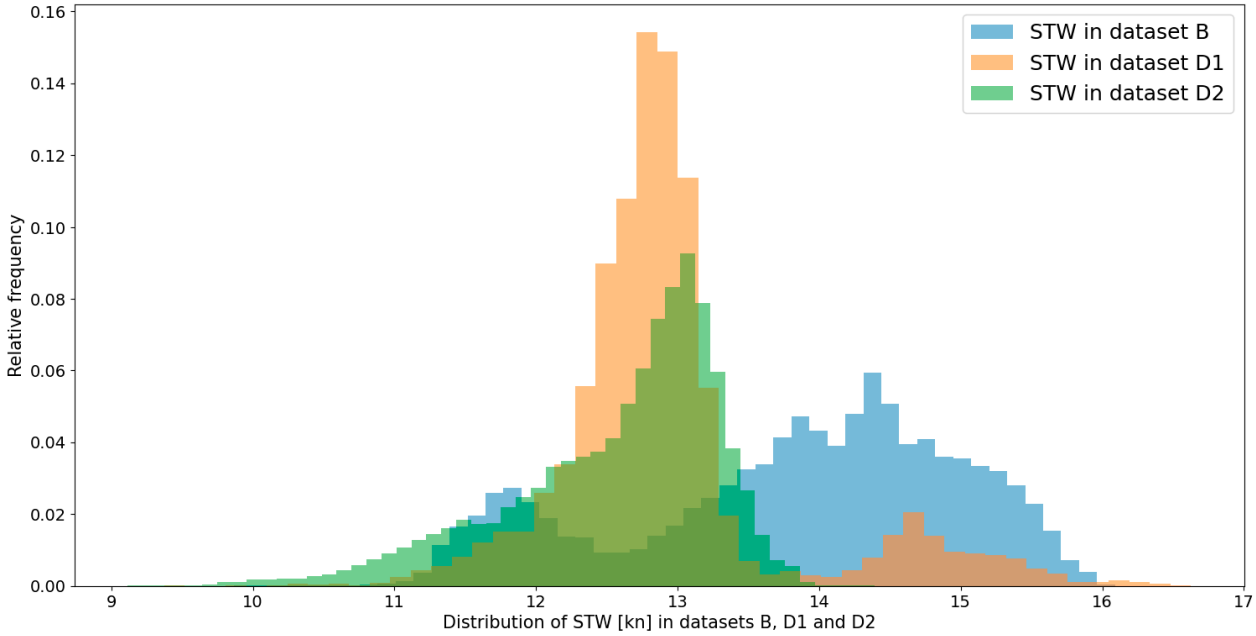


Figure 9. Relative frequency distribution of STW in Datasets B, D1, and D2.

Table 7. Basic statistics of STW in kn per dataset.

	STW Mean (kn)	STW Stand. Deviation (kn)
Dataset B	13.8	1.2
Dataset D1	12.9	0.8
Dataset D2	12.6	0.7

It was also noticed that Dataset C was another candidate dataset for reference performance. However, as shown in Table 3, idle time, referred as time in port, comprised 42.8% of Dataset C, in contrast to 25.6% in the case of Dataset D1. Due to the much greater percentage of idle time in Dataset C, more significant fouling accumulation was expected during Dataset C compared to Dataset D1, as idle time facilitates biofouling growth (see for example [9]). Furthermore, Dataset C contained one less ballast loading case compared

to Dataset D1, as shown in Figure 5. Hence, Dataset C was not considered as a reference performance baseline for the vessel in clean condition.

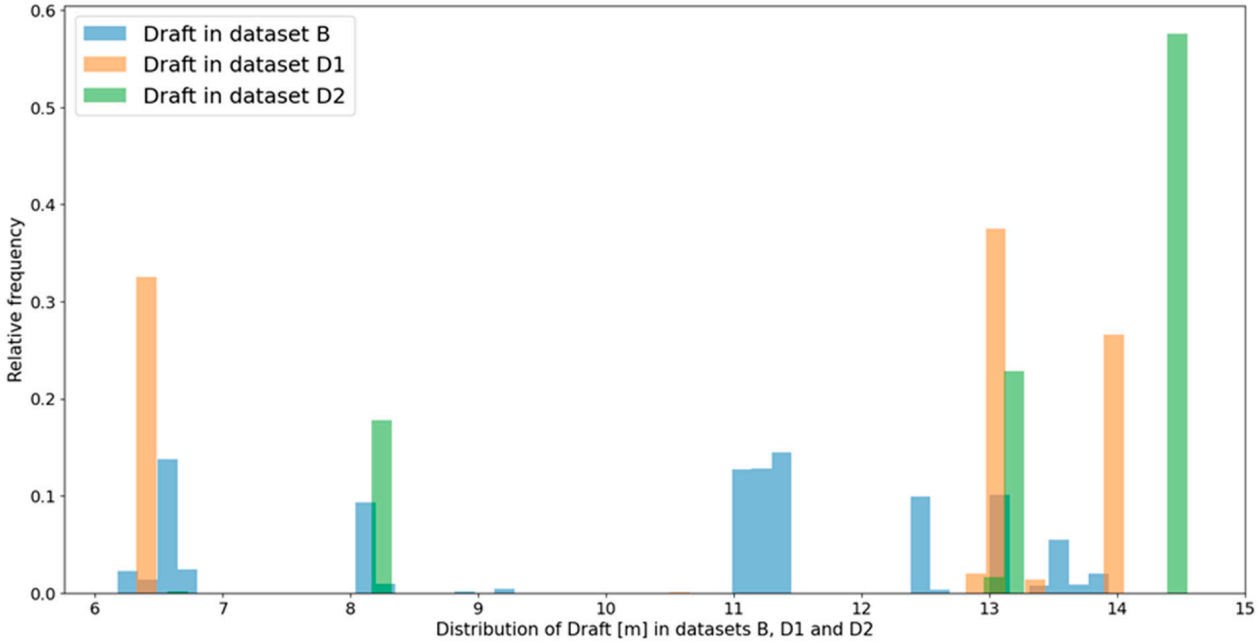


Figure 10. Relative frequency distribution of mean draft in Datasets B, D1, and D2.

The second approach used Model-2, which was employed to monitor the advance of biofouling in the period corresponding to Dataset D spanning over a year after maintenance. The distinctive feature of Model-2 was the Days feature, which enabled the emulation of scenarios referring to different points in time. To achieve this, synthetic datasets were created by assigning specific values to the model features. Details of these models are given in Table 8, while Figure 11 illustrates the two approaches for the assessment of hull and propeller fouling using Model-1 and Model-2, respectively.

Table 8. Overview of the two ML models, their feature sets (see Table 5), and training datasets (see Table 3).

	Model-1	**Model-2**
Feature set	Without Days (7 features)	With Days (8 features)
Training set	Dataset D1 (22 May 2022 to 8 August 2022)	Dataset D (22 May 2022 to 29 June 2023)

The procedure for the development of both models is as follows. First, the dataset is randomly shuffled and split into a training set and a test set using a ratio of 80% to 20%, respectively. Then, various ML algorithms, suitable for regression tasks, are evaluated on the training set. The ML algorithms considered are decision trees ([31]) and ensemble methods based on decision trees, i.e., random forest ([32]), extremely randomized trees (extra trees; [33]), and gradient tree boosting ([34]). This decision was motivated by the superior performance exhibited by tree-based algorithms compared to other algorithms, such as artificial neural networks (ANN), as demonstrated in relevant studies ([22,35]).

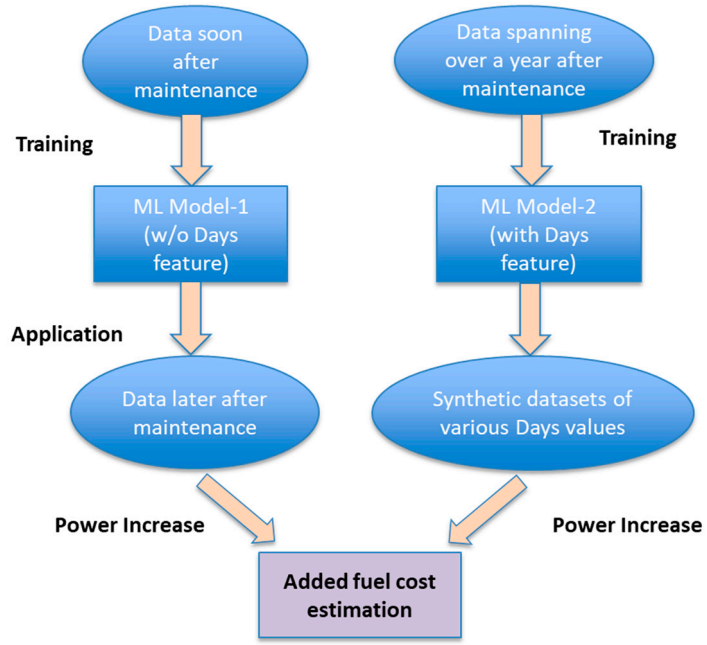


Figure 11. Flowchart illustrating the training and application process of Model-1 and Model-2 on separate datasets, within the context of hull and propeller condition monitoring.

In summary, decision trees can capture complex interactions and patterns, thus they can handle a wide range of regression problems, including those with nonlinear or non-monotonic relationships. Ensemble methods combine multiple decision trees, each one trained on different subsets of the data, providing multiple predictions. The combination of the information coming from these multiple predictions improves the overall accuracy and allows a generalization of the model. Decision trees are prone to overfitting, which occurs when the model becomes too specific to the training data and performs poorly on unseen data, i.e., instances that were not part of the training dataset. Ensemble methods alleviate this issue by reducing overfitting through techniques like random subspace sampling (random forests) or gradient-based optimization (gradient boosting). These methods introduce randomness and regularization, leading to better generalization and improved performance on unseen data.

Moreover, the averaging employed in ensemble methods helps to mitigate the influence of individual noisy predictions and leads to more robust overall predictions. In ensemble methods, multiple models are combined, and their predictions are averaged to produce a final prediction. By doing so, the impact of individual predictions that may contain noise or errors is reduced, resulting in more reliable and stable predictions across the ensemble. More details are presented in Appendix A.

Given that the most suitable algorithms for any given problem cannot be known a priori, a comparative analysis is required. In addition, the performance of each algorithm heavily depends on the configuration of the hyperparameters. To determine a set of well-performing hyperparameters, a search across the hyperparameter space is conducted. An exhaustive exploration of the search space is impossible, so only a few selected combinations are tested, either with grid search or a randomized search. Grid searches evaluate all of the possible combinations of the specified hyperparameter values, whereas randomized searches select a random value for each hyperparameter in each iteration, so that a set number of random combinations are evaluated. For a more extensive exploration of the search space, the randomized search is performed over a wide range of hyperparameter values, coupled with 5-fold cross-validation, to account for the intrinsic randomness in training. Randomized searches allow for broader exploration of the hyperparameter space while maintaining computational efficiency.

Additionally, randomized searches are coupled with 5-fold cross-validation to account for the intrinsic randomness in training and to help mitigate the risk of overfitting. The use of 5-fold cross-validation ensures that the model's performance is evaluated on multiple subsets of the data, providing a more reliable estimate of its generalization capacity.

The hyperparameter optimization was conducted separately for the two models. The best configuration for Model-2 is presented in Table 9, as it plays a more critical role in this study.

Table 9. Overview of the hyperparameter values explored for each ML algorithm using randomized search combined with cross-validation.

	Hyperparameter	Min	Max	Step	Best
Decision Tree	maximum depth of the tree (max depth)	5	30	1	23
	minimum number of samples required to be at a leaf node (min samples leaf)	1	20	1	3
Random Forest	max depth	5	30	1	20
	min samples leaf	1	20	1	4
	number of trees in the forest (n estimators)	20	200	5	110
Extra Trees	max depth	5	30	1	20
	min samples leaf	1	20	1	4
	n estimators	20	200	5	120
Gradient Tree Boosting	max depth	5	30	1	14
	number of boosting stages to perform	20	200	5	135
	learning rate	0.05	1	0.01	0.3

After hyperparameter tuning, a comparative evaluation of the four ML algorithms was performed using 10-fold cross-validation on the test set, which enabled a robust assessment of model performance. The goal was to compare the RMSE values across the 10 folds for each algorithm. The respective mean cross-validated RMSE values are reported in Table 10. To determine whether the observed differences in performance were statistically significant, we applied the Friedman test ([36]), followed by the post-hoc pairwise Wilcoxon signed-rank test ([37]).

Table 10. Mean cross-validated RMSE for each ML algorithm.

ML Algorithm	Model-1 Mean Cross-Validated RMSE [kW]	Model-2 Mean Cross-Validated RMSE [kW]
Decision Tree	184.3	151.6
Random Forest	154.4	134.2
Extra Trees	148.0	116.9
Gradient Tree Boosting	157.4	128.4

The Friedman test indicated a statistically significant difference between the RMSE values of the four algorithms for both models:

- Model 1: $\chi^2 = 26.04,\ p < 0.001$
- Model 2: $\chi^2 = 28.92,\ p < 0.001$

Since the Friedman test showed significant differences, the post-hoc pairwise Wilcoxon signed-rank test was conducted to assess the differences between individual algorithms. In Model-1, pairwise comparisons showed that decision tree was significantly different from random forest, extra trees, and gradient boosting (p = 0.002 for all comparisons). Extra

trees also significantly outperformed random forest (p = 0.004) and gradient tree boosting (p = 0.002), while random forest and gradient tree boosting were marginally different (p = 0.049).

The results suggested that, in both models, the extra trees algorithm significantly outperformed the other methods, while decision tree consistently exhibited worse performance compared to all ensemble methods. The pairwise comparisons also revealed that random forest and gradient boosting performed similarly, with only marginal differences between them in certain cases. The statistical analysis reinforced the conclusion that extra trees was the most suitable algorithm for this regression task, as it consistently showed the lowest RMSE and significantly better performance across the folds compared to the other methods. Hence, the extra trees algorithm was selected.

The model development process was completed by evaluating the selected, top-performing, ML algorithm (extra trees ensemble method) on the test set, with the objective of assessing the model's generalization capacity. The test set provided a reliable final measure of how well the model performed when applied to unseen data, because it had not been involved in either training or validation. Thus, error metrics suitable for regression tasks were used, and the results are reported in Table 11. The model's performance was evaluated using error metrics on the test set, which consisted of data absent from the training and validation process. Let $\hat{y}_i$ be the value predicted by the model for the i-th sample, y_i be the respective actual value, and $\epsilon_i = y_i - \hat{y}_i$ be the corresponding error. Thus, the following error metrics are defined, which are suitable for regression analysis ([17]):

- Coefficient of determination, R^2

$$R^2(y, \hat{y}) = 1 - \sum_{i=1}^{n} (y_i - \hat{y}_i)^2 / \sum_{i=1}^{n} (y_i - \overline{y})^2 \tag{15}$$

where $\overline{y} = \frac{1}{n}\sum_{i=1}^{n} y_i$ and $\sum_{i=1}^{n} (y_i - \hat{y}_i)^2 = \sum_{i=1}^{n} {\epsilon_i}^2$

- Mean absolute error, MAE

$$MAE(y, \hat{y}) = \frac{1}{n}\sum_{i=1}^{n} |y_i - \hat{y}_i| = \frac{1}{n}\sum_{i=1}^{n} |\epsilon_i| \tag{16}$$

- Root mean squared error, RMSE:

$$RMSE(y, \hat{y}) = \sqrt{\frac{1}{n}\sum_{i=1}^{n} (y_i - \hat{y}_i)^2} = \sqrt{\sum_{i=1}^{n} {\epsilon_i}^2} \tag{17}$$

- Mean absolute percentage error, $MAPE$:

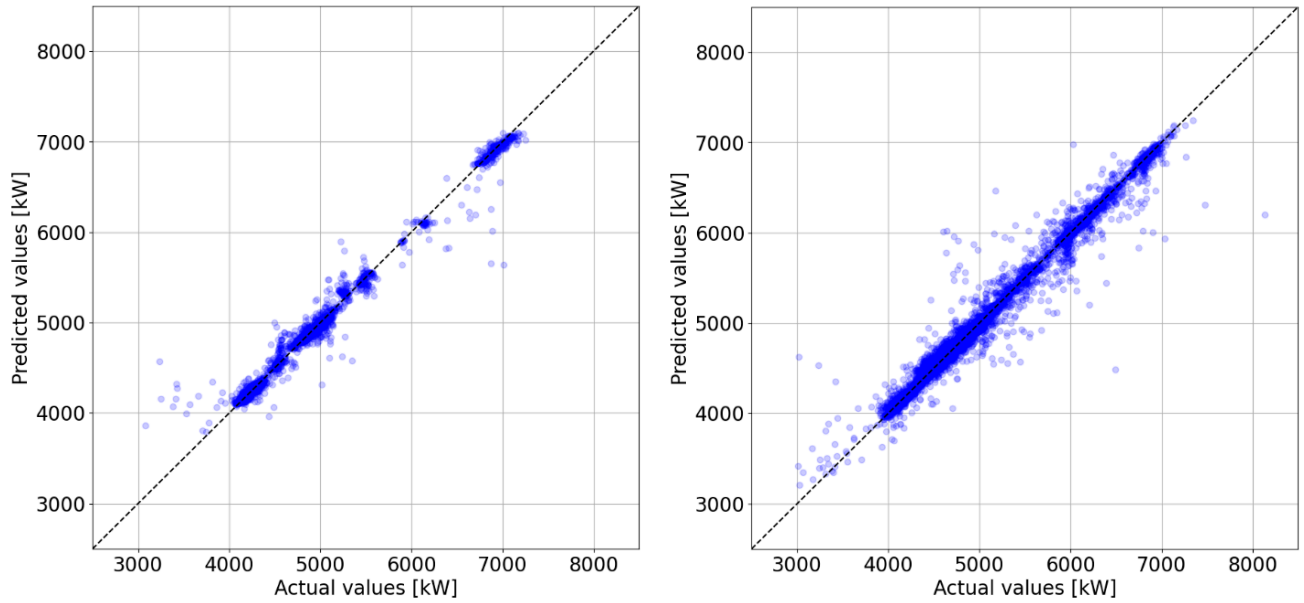
$$MAPE(y, \hat{y}) = \frac{1}{n}\sum_{i=1}^{n} |y_i - \hat{y}_i| / max(\varepsilon, |y_i|) \cdot 100\% \tag{18}$$

where ε is an arbitrarily small positive number to avoid zeroing the denominator.

Moreover, in Figure 12, the SHP values predicted by the model in the test set are plotted against the actual SHP values, and the points are clustered around the 45-degree line. Similarly, in Figure 13, the residual SHP values (actual minus predicted) in the test set are plotted against the predicted SHP values, and the points are clustered around the horizontal line through zero. A comparison of the results of Model-1 and Model-2 in both the test set and cross-validation revealed slightly better performance of the model with the Days feature, which was due to the additional information provided by the temporal feature.

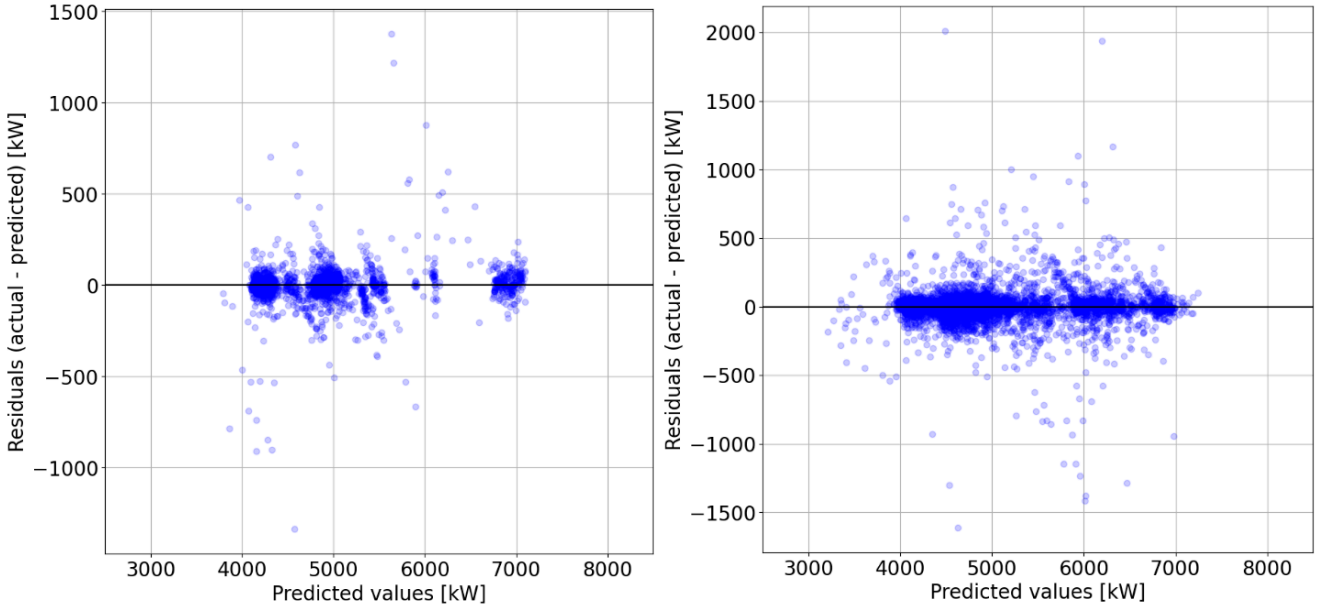
Table 11. Performance evaluation of the extra trees ensemble method for Model-1 and Model-2 on the test set, using a variety of error metrics.

	Model-1	**Model-2**
R^2	0.979	0.981
MAE [kW]	56.6	54.1
MAPE	1.12%	1.11%
RMSE [kW]	115.1	109.9



(**a**) (**b**)

Figure 12. Plot of the predicted SHP values against the actual SHP values of the test set: (**a**) for Model-1, and (**b**) for Model-2.



(**a**) (**b**)

Figure 13. Plot of residual SHP values against the predicted SHP values on the test set: (**a**) for Model-1, and (**b**) for Model-2.

5. Assessment of the Impact of Biofouling and Maintenance Optimization

As mentioned previously, as time elapses, the ship's frictional resistance increases due to the accumulation of biofouling on hull and propeller surfaces. This results in an increase in the thrust that must be delivered by the propulsion system to maintain a given ship speed, which leads to increased shaft power demand. The main assumption behind this reasoning is that, apart from the fouling condition of the vessel, the other conditions affecting the power demand remain the same. The goal of the data-driven modeling approach followed in the present study was to satisfy this assumption. Thus, two methods were proposed to assess the fouling condition using Model-1 and Model-2, respectively. Moreover, the power increase indicator (PI) was used, which is defined as follows:

$$PI = 100\frac{P_f - P_c}{P_c} \tag{19}$$

where P_c is the SHP in clean vessel condition and P_f is the SHP in fouled vessel condition, with both values corresponding to the same operational conditions.

5.1. Results of Model-1

Model-1 acts as a reference performance baseline, representing the clean vessel condition, and it be used to evaluate conditions with fouling. According to Table 3, the two datasets corresponding to advanced biofouling are Datasets B and D2. The difference observed between the actual power and the values predicted by Model-1, when applied to these datasets, serves as an estimate of the hull and propeller deterioration due to biofouling.

Dataset B was about two years away from sea trials and no hull cleaning or propeller polishing had been performed in the meantime. Thus, it represents a period in which severe marine growth has occurred, and, as a result, the hydrodynamic performance has deteriorated. In Figure 14, a scatter plot of the actual power and that predicted by Model-1 is shown, as well as a scatter plot of the power increase indicator.

Dataset D2 was extracted about a year away from the previous hull cleaning event, three months before the propeller was polished. In this period, biofouling was expected to have caused a considerable power penalty, even if not as significant as that in Dataset B. Indeed, when Model-1 was applied to Dataset D2, the actual power values were greater than the predicted values, which translated into a significant power increase, as shown in Figure 15. Moreover, Table 12 presents the mean power increase (PI) for both cases, which was estimated to be 21.7% and 10.8%, respectively. However, the limitations of Model-1 should be noted, as described in the previous section. Specifically, Figure 15b shows a negative power increase from May 19 to June 1, 2023, which was likely due to the model making predictions outside its training data range. Upon further investigation, it appears that during this voyage, the vessel was sailing in ballast condition with a single mean draft value of 8.2 m. However, the training Dataset D1 used to develop Model-1 did not contain any samples with draft values close to 8.2 m, according to Figure 10.

Table 12. Application results of Model-1 to Datasets B and D2, showing the mean values of the actual and predicted SHP, as well as the mean PI value.

Parameter	Value (Dataset B)	Value (Dataset D2)
Mean actual SHP [kW]	6342	5546
Mean predicted SHP [kW]	5209	4984
Difference in means [kW]	1133	562
Mean PI	21.7%	11.3%

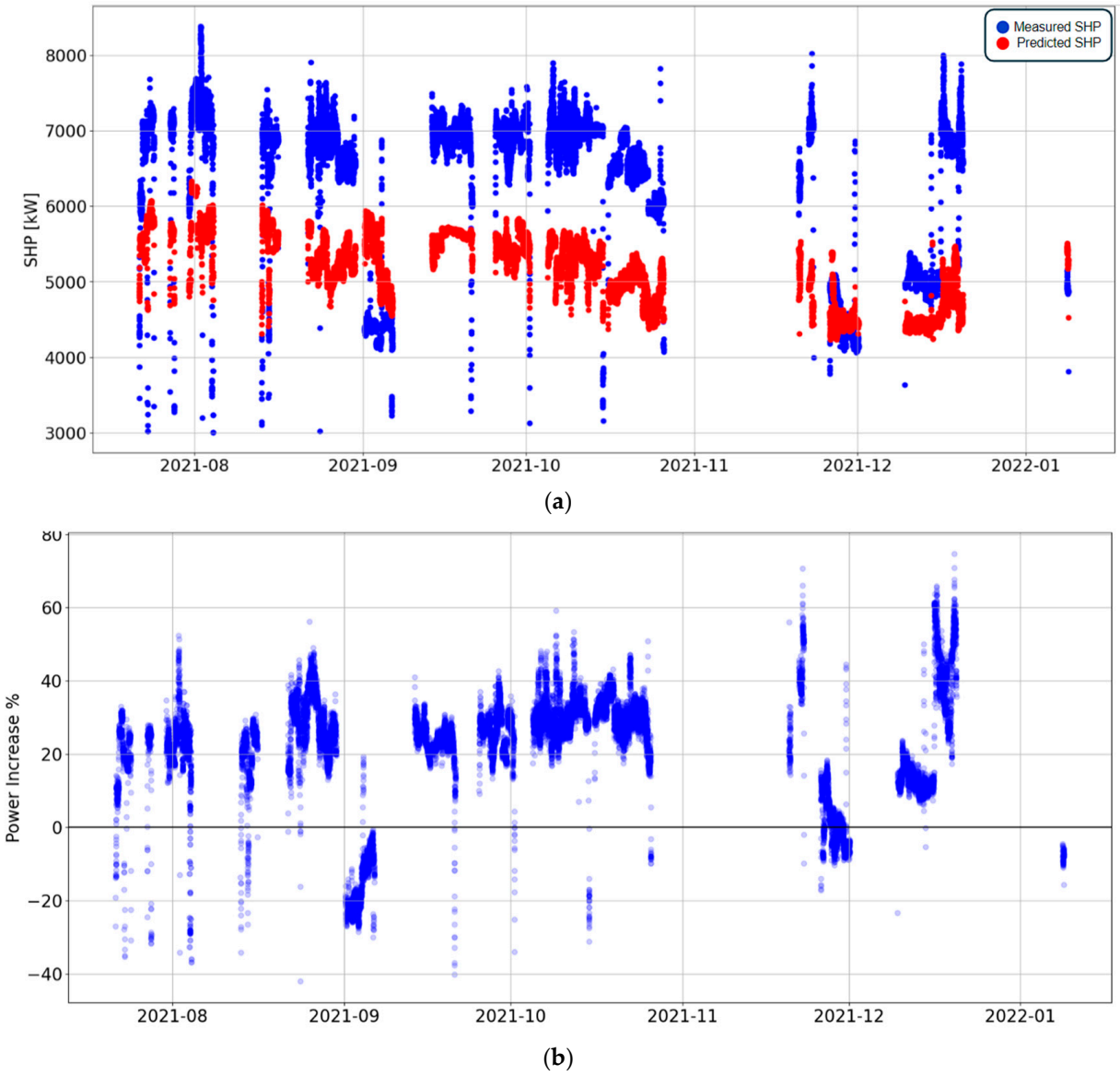


Figure 14. Application of Model-1 to Dataset B. (**a**) Scatter plot of the actual (blue) and predicted (red) SHP over time, and (**b**) scatter plot of PI over time.

5.2. Results of Model-2

In this approach, Model-2 was employed to monitor the advance of biofouling in the period corresponding to Dataset D. To achieve this, synthetic datasets were created by assigning specific values to the model features. Although the goal here was to evaluate the effect of the Days feature, other input variables played a crucial role in the output, namely the STW, the Draft, and the Trim. Thus, two loading conditions, one laden and one ballast, were considered, setting the Draft and Trim values equal to the respective mode values of Dataset D. Then, for each condition, values within the range of 12 to 14 kn were considered for the STW, since most values of Dataset D were in this range (see [30]). Furthermore, a calm sea condition with no current effect was considered, so the wave, wind, and current variables were set equal to zero. Additionally, the RoT variable was assumed to be zero, indicating that the vessel was traveling on a steady course. Lastly, for each loading condition, three values for the Days feature, those most frequently found in Dataset D for each loading condition, were selected over the entire period of Dataset D. The feature values listed above are summarized in Table 13, which describes the synthetic datasets of laden and ballast conditions.

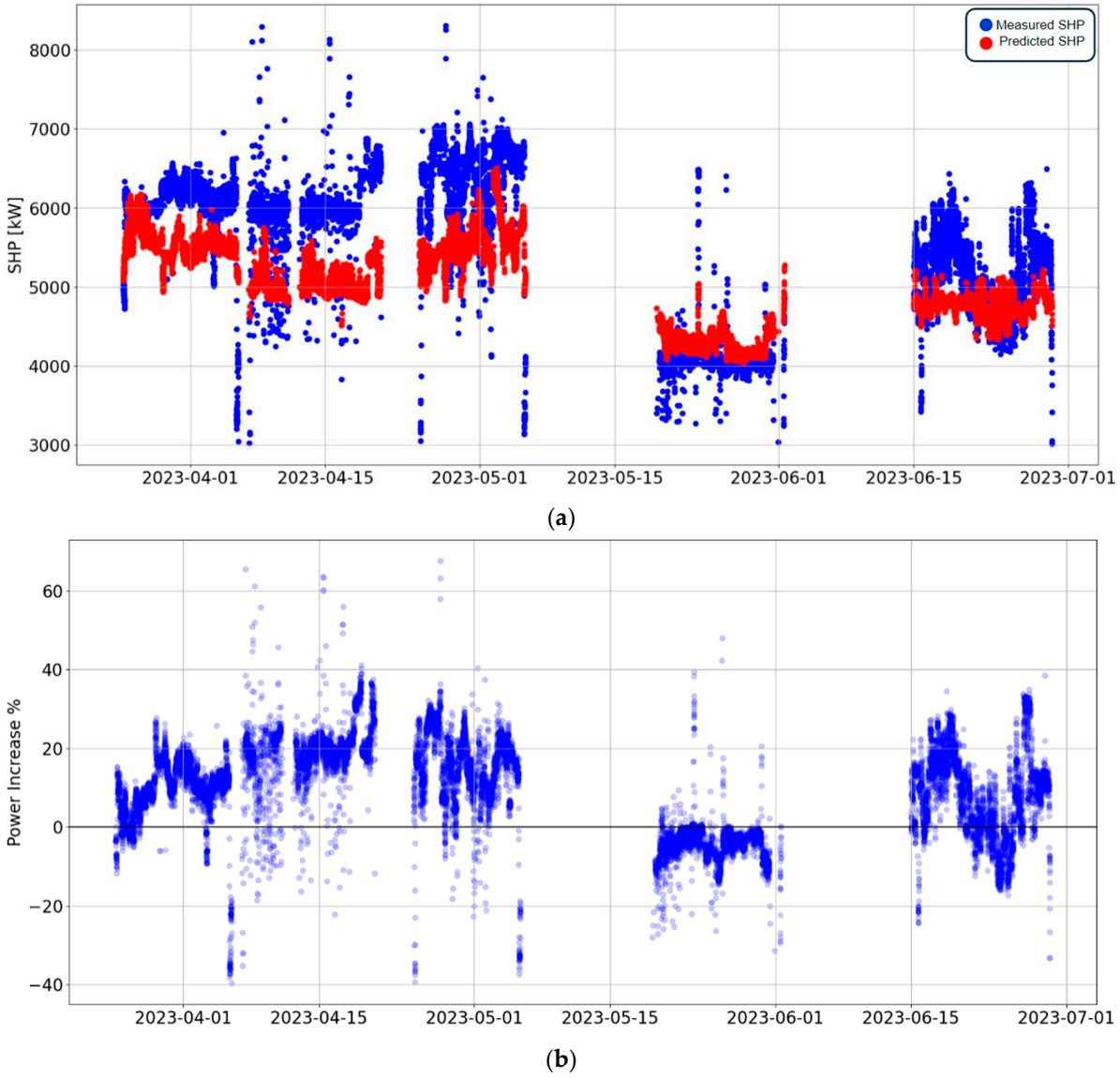
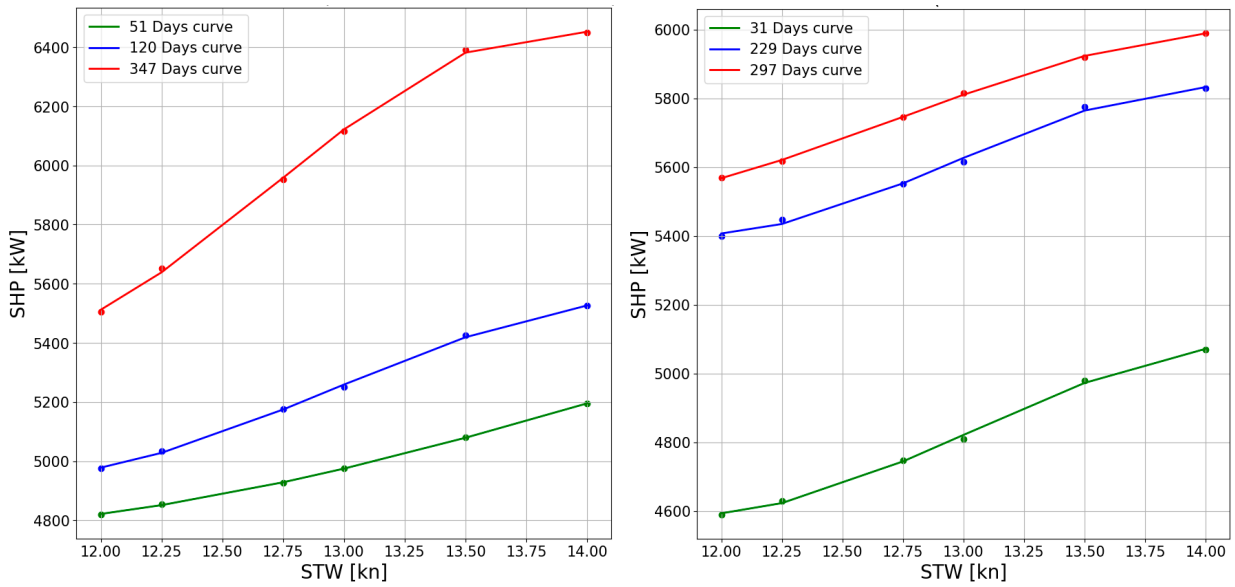


Figure 15. Application of Model-1 to Dataset D2. (**a**) Scatter plot of the actual (blue) and predicted (red) SHP over time, and (**b**) scatter plot of PI over time.

Table 13. Overview of the synthetic datasets designed separately for laden and ballast conditions.

Variable	Laden Condition	Ballast Condition
STW [kn]	12.0, 12.25, 12.75, 13.0, 13.5, 14.0	12.0, 12.25, 12.75, 13.0, 13.5, 14.0
Draft [m]	14.4	6.81
Trim [m]	−0.13	−2.56
RoT [deg/min]	0	0
Wind x [m/s]	0	0
Wave x [m]	0	0
Current [kn]	0	0
Days [days]	51, 120, 354	31, 229, 297

Then, Model-2 was applied to the synthetic datasets, and the predictions were grouped by loading condition and day. In Figure 16, the predicted power is plotted against the STW, and a least squares polynomial fit of 3rd degree is applied to each group of points. This fitting process yielded an estimate of the propeller curves for laden and ballast conditions, respectively. The mean predicted power was calculated in each group of points, as listed in Tables 14 and 15. In Figure 17, for each loading condition, the values of the mean predicted power are plotted against the corresponding values of the Days feature, along with the corresponding best-fit straight lines.



(**a**) (**b**)

Figure 16. Predicted SHP values on the synthetic datasets plotted against STW, with respective least squares polynomial fits of 3rd degree: (**a**) laden condition and (**b**) ballast condition.

Table 14. Mean predicted SHP and corresponding PI, when Model-2 is applied to the synthetic dataset representing the laden condition.

Days	Mean Predicted SHP on Synthetic 'Laden' Dataset [kW]	Mean PI
51	4975	-
120	5231	5.1%
347	6011	20.8%

Table 15. Mean predicted SHP and corresponding PI, when Model-2 is applied to the synthetic dataset representing the ballast condition.

Days	Mean Predicted Power on Synthetic 'Ballast' Dataset [kW]	Mean PI
31	4805	-
229	5603	16.6%
297	5776	20.2%

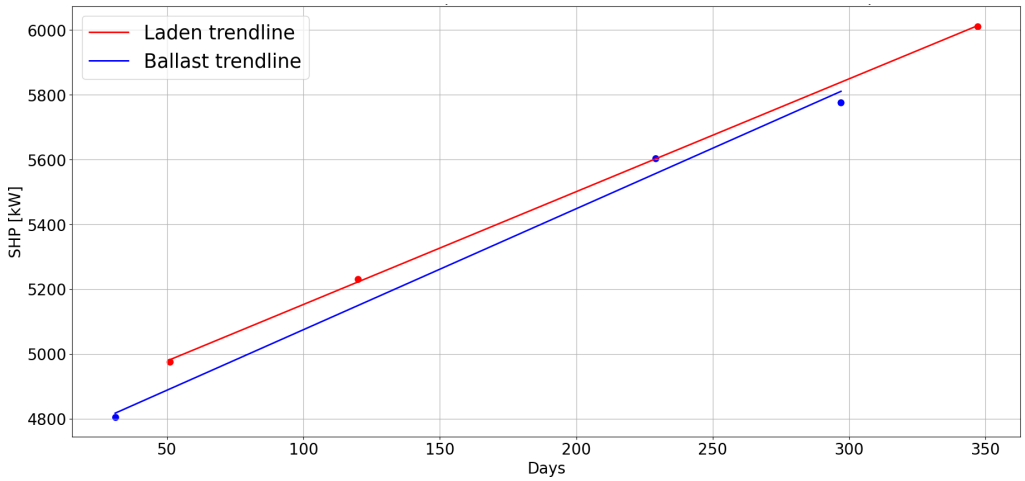


Figure 17. Mean predicted SHP values on the synthetic laden and ballast datasets in terms of days, plotted with respective linear fits.

5.3. Estimation of Added Fuel Cost

For maintenance optimization, the added fuel cost due to biofouling must be evaluated. The methods described above provide an estimate of the power penalty caused by biofouling growth over a certain period after hull cleaning and propeller polishing. To calculate the consequent increase in FOC (dFOC), the approximate values of specific fuel oil consumption (SFOC) at different loads were used based on ME's shop tests, where no degradation of the performance of the main engine was considered. Since the power values were known, the dFOC was calculated as follows:

$$dFOC = P_f \cdot SFOC\left(P_f\right) - P_i \cdot SFOC(P_i) \tag{20}$$

where P_i and P_f are the initial and final SHP values respectively, and $SFOC(P_i)$ and $SFOC\left(P_f\right)$ are the corresponding SFOC values. Based on the operator's data, the average fuel cost per ton during the examined period was 556 \$/ton, so the added fuel cost per day (dCost) was derived from the following equation:

$$dCost = dFOC \cdot FuelCost \tag{21}$$

For instance, by applying this reasoning to the power values predicted with Model-2 for each loading condition and for different ship speed scenarios, the added fuel cost per day due to biofouling that accumulated during the corresponding period was estimated and is reported in Table 16. Hence, the added fuel cost when the vessel operated under the conditions assumed in Model-2 was estimated to be on average equal to \$2228 per day. Considering that hull cleaning and propeller polishing for a Kamsarmax vessel ranges between \$15,000 and \$18,000, where the higher cost corresponds to the laden condition, it is concluded that, on average, in eight days of fouled operation, the accumulated additional fuel cost equaled the maintenance cost. Furthermore, the mean predicted power at each loading condition was linearly extrapolated using the straight-line fit to the points in Tables 14 and 15 to predict the power for an assumed value of the Days feature equal to 400. Thus, based on the behavior of the vessel in the past, on Day 400, an average added fuel cost of \$2719 per day was predicted. This information can be used by the shipping company to decide the right time to perform the next maintenance action, although other factors may influence this decision, such as the ship's operation schedule.

Table 16. Estimated added fuel cost per day for the laden and ballast loading conditions, considering different values of the Days feature.

	From Day	To Day	dCost at STW = 14 kn	Mean dCost at STW = 12–14 kn	Mean dCost at STW = 12–14 kn and Day 400
Laden	51	347	$2791	$2317	$2724
Ballast	31	297	$1807	$2138	$2714

6. Discussion

In this study, the models developed are data-driven and based on machine learning (ML) algorithms. This approach was chosen due to the large volume of available operational data from various sources and the improved accuracy that ML algorithms can provide in analyzing such data. Additionally, with Model-2, a quasi-static approach was followed to capture the long-term impact of biofouling on the ship's performance by incorporating time as an input feature.

Specifically, the two-and-a-half-year period of available operational data with three maintenance events in between provided an opportunity to examine various hull and propeller conditions of varying performance and decide the most appropriate periods of vessel operation to perform data analysis and train and implement ML methods. It is emphasized that knowledge of maintenance events is significant when selecting the time periods to be used for training ML models as it prevents including data that entail physical processes not related to the targeted one. For example, the inclusion of the time period during which the propeller was damaged would disturb the analysis of the effect of fouling development as it would include data points corresponding to different propulsion characteristics.

Moreover, feature engineering and data preparation are important steps in data-driven studies. Domain knowledge and understanding of the physics of the problem is necessary for insightful usage of parameters by combining or creating new ones. Furthermore, extensive filtering was applied to keep only the operational conditions related to open-sea navigation and within a range of key parameters that will be meaningful to the problem data in the ML models. However, considering that this a static approach for threshold definition, methods related to adaptive thresholding could be explored in future work to potentially enhance the robustness of this procedure. Data smoothing was also carried out using a simple moving average to reduce the noise of the signals. The aim was to reduce the sharp fluctuations that do not add significant information to the physical process of the problem. However, at the same time, we need to stay close to the original data to be able to capture environmental loading and other feature variations that can reveal important patterns in the data. The proper time window for smoothing in this analysis was selected based on domain knowledge; however, the addition of a statistically quantified metric to support the justification is planned in a future study. Even though a significant percentage of the initial dataset was removed from the final dataset, this is a necessary step to derive a quality dataset.

Having prepared the dataset, the approach used to evaluate the biofouling condition of the hull and propeller presents a novel methodology that considers the temporal aspect and incorporates separate loading conditions in the generation of synthetic datasets. However, the study has certain limitations. Firstly, due to the unavailability of high-frequency measurements for draft and weather data during specific time periods, the corresponding noon report data had to be used, resulting in lower accuracy and increased uncertainty of the actual values.

While minimal variation in the draft over a day is plausible, the wave conditions can change rapidly due to factors such as shifting wind patterns and swell conditions. Using a single daily wave height value from the noon reports does not fully capture this temporal variability of the wave conditions experienced by the vessel over a day. To quantify the potential impact of this limitation, a comparative analysis was conducted between the noon

report sea height and the high-frequency significant wave height data from the weather provider during their overlapping period. This analysis revealed a mean difference of −0.02 m and a standard deviation of 0.84 m, suggesting that while the noon report data provides a reasonable approximation, there is still a notable degree of variability around the daily average value. To further address the limitations of using noon report wave data, a sensitivity analysis could be conducted to explore the impact on the performance of the ML model. This could involve comparing the results obtained using noon report wave heights against an alternative approach that utilizes high-frequency wave data. However, such data in the current study were not available for the entire set of periods used for the training and application of the ML models. Future research may benefit from access to higher-resolution weather data, which could provide a more accurate representation of the vessel's operating environment. Moreover, following the discussion of current speed as an input feature, the use of SOG instead of STW as an input feature is worth exploring, as SOG would inherently capture the effect of the current. In future research, we will compare model performance with and without the inclusion of the current feature to verify if similar or improved results can be achieved without its direct incorporation.

Another limitation of Model-1 is the fact that the model was trained on data from only a 3-month period; thus, it has not encountered enough operational conditions to fully represent the complete profile of the vessel, especially its loading condition. Therefore, when the model is applied to operating conditions significantly different from what it was trained on, the model can produce inaccurate extrapolated predictions. This is likely the cause of the unexpected decreasing power increase trend shown in Figure 12b. On the other hand, a longer period for this approach would deviate from the assumption of the unchanged fouling status.

Moreover, in the development stage of the ML models, only tree-based algorithms were evaluated, given their superior performance in the literature. The model used in this study is a quasi-static model, not a time series-based model. Therefore, more sophisticated, temporally aware algorithms, like LSTMs, were not explored in this study. The reason for using time (specifically, the Days feature) is that, in the biofouling problem, time has a long-term impact on the ship's performance and, consequently, on the power requirements. Thus, the model can capture the gradual, long-term deterioration of performance due to biofouling accumulation even though the model itself is not a dynamic, time series-based model.

However, it would be beneficial to explore ANNs as well as LSTMs and compare their performance with tree-based methods, as both have demonstrated strong capabilities in similar applications (see for example [11,18,38]). Such an examination will be carried out in a future study. In addition, it could be interesting to explore a feature that directly reflects the number of days the ship is idle in port while at anchor or berthed. Depending on the environmental conditions in the port, biofouling can accumulate significantly during these idle periods. In the current study, the Days feature was used to capture the effects of both sailing and idle time on biofouling accumulation. The data preparation excluded port berthing periods, so the model was trained only on active voyage data. The Days feature indirectly accounts for idle periods, as its value cumulatively incorporates the time spent in port.

7. Conclusions

In conclusion, a hull and propeller condition monitoring tool for a bulk carrier ship using operational data collected over a period of two-and-a-half years is presented. The analysis utilized high-frequency data and noon reports, while a preliminary assessment of the vessel's hydrodynamic conditions was conducted using a suitable KPI, which allowed for the identification of performance degradation due to fouling and the impact of maintenance events.

Furthermore, the study implemented a data-driven approach to predict the SHP in realistic operational conditions. Following feature engineering and data preparation,

various machine learning algorithms from the decision tree family were evaluated, and an extensive exploration of hyperparameters was performed to identify the model with the best generalization capability. As a result, a MAPE of 1.1% was achieved on previously unseen test data. A model trained on data collected shortly after maintenance was applied to data collected at a later stage after maintenance. The PI, defined as the normalized difference between the actual and predicted SHP, was utilized to describe the deterioration of the vessel's condition. When applied to a dataset reflecting advanced stages of biofouling accumulation, the model predicted that if the vessel were in a clean state and operated under identical conditions, there would be an estimated average increase of 11.3% in power requirements. In a separate approach, the Days feature was incorporated to indicate the elapsed time since the previous cleaning event, thus enabling emulation of scenarios that referred to different points in time. A model incorporating this feature was trained using a dataset spanning over a year, and it was then applied to synthetic datasets representing both laden and ballast conditions across a range of vessel speeds, with various values of the Days feature. By comparing the predictions made for different Days feature values, while keeping the other features constant, the PI resulting from biofouling was derived. By comparing the model predictions on synthetic data 300 days apart, while keeping other parameters constant, it is determined that the fouled vessel required an average 20.5% increase in power compared to a clean state.

These data-driven approaches provide an estimation of the PI caused by marine biofouling growth within a specific timeframe. By incorporating SFOC and fuel cost information, the subsequent daily increase in fuel cost could be determined. As a result, it was estimated that over a span of approximately 10 months, biofouling led to an average daily fuel cost increase of $2228. Shipping companies can compare the cumulative additional fuel cost with the cost of maintenance, considering the vessel schedule, to strategically determine the optimal timing to perform vessel cleaning. Hence, by providing valuable insights and practical implications, the findings of this study enable cost-effective decision making within the shipping industry. These insights support the development of more informed and efficient maintenance strategies, ultimately leading to improved operational efficiency.

Author Contributions: Conceptualization, N.T.; Methodology, N.T. and G.N.; Validation, G.N.; Formal analysis, N.T. and G.N.; Investigation, G.N.; Resources, N.T. and V.Z.; Data curation, G.N.; Writing—original draft, N.T. and G.N.; Writing—review & editing, N.T., G.N. and V.Z.; Visualization, G.N.; Supervision, N.T. and V.Z.; Project administration, N.T. and V.Z.; Funding acquisition, N.T. and V.Z. All authors have read and agreed to the published version of the manuscript.

Funding: The present work was supported by the Retrofit55 project (Retrofit solutions to achieve 55% GHG reduction by 2030), which has received funding from the European Union's Horizon Europe Research and Innovation Program under Grant Agreement No. 101096068.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The original contributions presented in the study are included in the article, further inquiries can be directed to the corresponding author. The dataset employed in the study is unavailable, as the authors do not have permission to share it.

Acknowledgments: The authors would especially like to thank Laskaridis Shipping Co. LTD. for the productive conversations and for sharing the operational data of the ship examined as part of the Retrofit55 project.

Conflicts of Interest: Authors George Nikolaidis and Vasilios Zagkas were employed by the company SimFWD. The remaining authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Appendix A. Tree-Based ML Algorithms

Decision trees are powerful and interpretable models used for both classification and regression tasks. In this report, we focus on decision trees for regression, specifically those that employ the CART (classification and regression trees) algorithm ([31]).

A decision tree for regression is a predictive model that uses a tree-like structure to make predictions on continuous numerical target variables. The CART algorithm constructs the decision tree by recursively partitioning the feature space into smaller regions, with the goal of minimizing the sum of squared errors (SSE) within each partition. The CART algorithm starts with the entire dataset and selects a feature v to split the data. It evaluates different split points for the selected feature and chooses the one that results in the lowest SSE. The splitting process divides the dataset into two subsets A_1 and A_2 based on the feature value f_v, creating left and right child nodes. The cost function minimized with the spilt is given by the following:

$$J(v, f_v) = \frac{k_1}{k} SSE_1 + \frac{k_2}{k} SSE_2 \tag{A1}$$

where k, k_1, and k_2 are the numbers of instances in the original dataset and in subsets A_1 and A_2, respectively. Thus, the SSE is given as follows:

$$SSE_j = \sum_{i \in A_j} \left(y_i - \overline{y}_j \right)^2, \text{ for } j = 1, 2 \tag{A2}$$

$$y_j = \frac{1}{k_j} \sum_{i \in A_j} y_i, \text{ for } j = 1, 2 \tag{A3}$$

The splitting process is applied recursively to each child node until a stopping criterion is met. This criterion can be a maximum tree depth, a minimum number of samples per leaf node, or a minimum reduction in SSE. The recursive nature of the algorithm allows complex relationships to be captured between features and the target variable. When the recursion stops, the resulting tree contains terminal nodes called leaf nodes. Each leaf node represents a prediction for the target variable in the corresponding region of the feature space. In the case of regression, the prediction is typically the mean or median value of the target variable within that region. The predictions from the leaf nodes form the final output of the decision tree model.

Typically, the "max_depth" and "min_samples_leaf" hyperparameters are tuned when creating a decision tree model. Specifically, the "max_depth" hyperparameter determines the maximum depth or levels of the decision tree. It controls how many splits or branches the tree can have from the root node to the leaf nodes. A higher value can result in a more complex tree that can potentially overfit the training data, while a lower value can lead to underfitting. On the other hand, the "min_samples_leaf" hyperparameter sets the minimum number of samples required in a leaf node. If the number of samples in a potential leaf node is less than the specified value, further splitting is halted, and the node becomes a leaf. A higher value can prevent the tree from creating very specific rules for a small subset of data, reducing overfitting. These hyperparameters play a crucial role in controlling the complex and generalization ability of a decision tree model for regression tasks. A pseudocode for the Decision Tree regression algorithm (CART) is presented in Algorithm A1.

Random forests for regression combine the strength of decision trees with ensemble techniques ([32]). Instead of relying on a single decision tree, random forests aggregate predictions from multiple trees, providing more robust and accurate regression results. The CART algorithm is employed to construct individual decision trees within the random forest.

Random forests are constructed through the following steps. First, the algorithm begins by bootstrapping, where subsets of the original training data are randomly selected with replacement. These subsets, called bootstrap samples, have the same size as the

original dataset but can contain duplicate instances. Next, for each bootstrap sample, a decision tree is constructed using the CART algorithm. To introduce diversity among the trees in the ensemble, feature randomness is incorporated. At each split during decision tree construction, a random subset of features is considered. Finally, once all of the trees are built, predictions are made by aggregating the individual tree predictions. This is achieved by calculating the mean of the predicted values from each tree. The "n_estimators" hyperparameter refers to the number of individual decision trees that are created and combined in the ensemble.

Extremely randomized trees build upon the idea of random forests by introducing additional randomness during the tree construction process ([33]). This randomness leads to further diversity among the trees in the ensemble, which can enhance their predictive performance. The CART algorithm is utilized to construct individual decision trees within extremely randomized trees.

The construction of extremely randomized trees involves several steps. Firstly, bootstrapping is employed, similar to random forests, to create multiple bootstrap samples from the original training data. Each bootstrap sample is generated by randomly selecting instances with replacement. Secondly, feature randomness is introduced in extremely randomized trees. At each split point during the tree construction process, thresholds for each feature are randomly selected without evaluating the optimal split points. This random selection of thresholds adds more randomness and reduces correlation between the trees in the ensemble. Thirdly, individual decision trees are constructed using the CART algorithm. The feature space is recursively partitioned, and each tree is grown by selecting random thresholds for each feature and choosing the split that minimizes the SSE within each leaf node. Finally, after constructing all of the trees, predictions are made by taking the mean of the predicted values from each tree. The hyperparameter "n_estimators" specifies the number of individual decision trees that are generated and aggregated as an ensemble.

Gradient tree boosting, or gradient boosted regression trees (GBRT), is an ensemble learning method that combines the predictive power of decision trees with the concept of boosting ([34]). It builds an ensemble of weak prediction models, typically decision trees, in a stage-wise manner. Based on the principle of gradient boosting, each subsequent model tries to correct the residuals (errors) of the previous models.

The construction of GBRT entails a series of distinct steps. Firstly, the predictions are initialized by using a selected value, such as the mean of the target variable. This value acts as the starting point for subsequent iterations. An iterative process then begins: the difference between the actual target values and the prediction from the previous iteration is calculated, resulting in residuals that represent the errors to be rectified by the current model. Following that, for each iteration, a new decision tree is built using the CART algorithm. After constructing the tree, the predictions from the tree are multiplied by a learning rate, which governs the contribution of each tree to the final prediction. These predictions are then aggregated to update the overall prediction of the GBRT. This iterative process continues with each iteration aiming to reduce the remaining residuals. Multiple decision trees are added to the ensemble, and their predictions are combined with the previous predictions to progressively refine the overall model.

Algorithm A1. Pseudocode for the Decision Tree regression algorithm (CART)

```
# Function to compute SSE for a given set of target values
function compute_SSE(target_values):
    mean_value = mean(target_values)
    SSE = sum((target_value - mean_value) ^ 2 for target_value in target_values)
    return SSE

# Function to split data into two subsets based on a feature and split point
function split_data(dataset, feature, split_value):
    A1 = subset of dataset where feature <= split_value
    A2 = subset of dataset where feature > split_value
    return A1, A2

# Function to find the best split (feature and split value) for a given node
function find_best_split(dataset, target_values):
    best_feature = None
    best_split_value = None
    min_cost = infinity

    for each feature in dataset:
        for each split_value in unique values of feature:
            # Split the dataset
            A1, A2 = split_data(dataset, feature, split_value)

            # Compute SSE for both subsets A1 and A2
            SSE1 = compute_SSE(target_values corresponding to A1)
            SSE2 = compute_SSE(target_values corresponding to A2)

            # Calculate the weighted cost of this split
            k1 = number of instances in A1
            k2 = number of instances in A2
            k = number of instances in original dataset
            cost = (k1/k) * SSE1 + (k2/k) * SSE2

            # Check if this is the best split so far
            If cost < min_cost:
                min_cost = cost
                best_feature = feature
                best_split_value = split_value

return best_feature, best_split_value

# Recursive function to build the Decision Tree
function build_tree(dataset, target_values, max_depth, min_samples_leaf, current_depth):
    # Check stopping criteria: max depth or min samples per leaf
    if current_depth >= max_depth or len(dataset) <= min_samples_leaf:
        return create_leaf_node(mean(target_values))

    # Find the best split
    best_feature, best_split_value = find_best_split(dataset, target_values)

    # If no valid split, create a leaf node
    if best_feature is None:
        return create_leaf_node(mean(target_values))

    # Split the dataset into two subsets
    A1, A2 = split_data(dataset, best_feature, best_split_value)
    target_A1 = target_values corresponding to A1
    target_A2 = target_values corresponding to A2

    # Recursively build the left and right subtrees
    left_child = build_tree(A1, target_A1, max_depth, min_samples_leaf, current_depth + 1)
    right_child = build_tree(A2, target_A2, max_depth, min_samples_leaf, current_depth + 1)

    # Return a decision node with the best split and its children
    return create_decision_node(best_feature, best_split_value, left_child, right_child)

# Function to create the leaf node (terminal node)
```

Algorithm A1. *Cont.*

```
54.  function create_leaf_node(value):
55.          return {"type": "leaf", "prediction": value}

56.  # Function to create a decision node
57.  function create_decision_node(feature, split_value, left_child, right_child):
58.          return {"type": "decision", "feature": feature, "split_value": split_value, "left": left_child, "right":
     right_child}

59.  # Main function to train the Decision Tree model
60.  function train_decision_tree(dataset, target_values, max_depth, min_samples_leaf):
61.          return build_tree(dataset, target_values, max_depth, min_samples_leaf, current_depth = 0)
```

References

1. IMO. *Initial IMO Strategy on Reduction of GHG Emissions from Ships*; Resolution MEPC.304; International Maritime Organisation: London, UK, 2018.
2. IMO. *IMO Strategy on Reduction of GHG Emissions from Ships*; Resolution MEPC.377(80); International Maritime Organization: London, UK, 2023.
3. Barreiro, J.; Zaragoza, S.; Diaz-Casas, V. Review of ship energy efficiency. *Ocean Eng.* **2022**, *257*, 111594. [CrossRef]
4. Brynolf, S.; Baldi, F.; Johnson, H. Energy efficiency and fuel changes to reduce environmental impacts. In *Shipping and the Environment: Improving Environmental Performance in Marine Transportation*; Andersson, K., Brynolf, S., Lindgren, J.F., Wilewska-Bien, M., Eds.; Springer: Berlin/Heidelberg, Germany, 2016; pp. 295–339. [CrossRef]
5. Bouman, E.A.; Lindstad, E.; Rialland, A.I.; Stramman, A.H. State-of-the-art technologies, measures, and potential for reducing GHG emissions from shipping—A review. *Transp. Res. Part D Transp. Environ.* **2017**, *52*, 408–421. [CrossRef]
6. Yusim, A.K.; Utama, I.K.A.P. An Investigation into The Drag Increase on Roughen Surface due to Marine Fouling Growth. *J. Technol. Sci.* **2017**, *28*, 73–78. [CrossRef]
7. Carhen, A.; Altar, M. Four KPIs for the assessment of biofouling effect on ship performance. *Ocean Eng.* **2020**, *217*, 107971. [CrossRef]
8. Arndt, E.; Robinson, A.; Hester, S.; Woodham, B.; Wilkinson, P.; Gorgula, S.; Brooks, B. *Factors That Influence Vessel Biofouling and Its Prevention and Management*; Final Report for CEBRA Project 190803; Center of Excellence for Biosecurity Risk Analysis: Melbourne, VIC, Australia, 2021.
9. Uzun, D.; Demirel, Y.K.; Coraddu, A.; Turan, O. Time-dependent biofouling growth model for predicting the effects of biofouling on ship resistance and powering. *Ocean Eng.* **2019**, *191*, 106432. [CrossRef]
10. *ISO 19030*; Ships and Marine Technology—Measurement of Changes in Hull and Propeller Performance. International Organization of Standardization: Geneva, Switzerland, 2016.
11. Karagiannidis, P.; Themelis, N. Data-driven modelling of ship propulsion and the effect of data pre-processing on the prediction of ship fuel consumption and speed loss. *Ocean Eng.* **2021**, *222*, 108616. [CrossRef]
12. Spandonidis, C.; Paraskevopoulos, D. Evaluation of a Deep Learning-Based Index for Prognosis of a Vessel's Propeller-Hull Degradation. *Sensors* **2023**, *23*, 8956. [CrossRef]
13. Valchev, I.; Coraddu, A.; Kalikatzarakis, M.; Geertsma, R.; Oneto, L. Numerical methods for monitoring and evaluating the biofouling state and effects on vessels' hull and propeller performance: A review. *Ocean Eng.* **2022**, *251*, 110883. [CrossRef]
14. Lang, X.; Wu, D.; Mao, W. Comparison of supervised machine learning methods to predict ship propulsion power at sea. *Ocean Eng.* **2022**, *245*, 11687. [CrossRef]
15. Uyanık, T.; Karatuğ, Ç.; Arslanoğlu, Y. Machine learning approach to ship fuel consumption: A case of container vessel. *Transp. Res. Part D Transp. Environ.* **2020**, *84*, 102389. [CrossRef]
16. Kim, Y.R.; Jung, M.; Park, J.B. Development of a fuel consumption prediction model based on machine learning using ship in-service data. *J. Mar. Sci. Eng.* **2021**, *9*, 137. [CrossRef]
17. Nikolaidis, G.; Themelis, N. Examining the performance of retrofit measures in real ship operation using data-driven models. *Ship Technol. Res.* **2022**, *69*, 170–180. [CrossRef]
18. Zhang, M.; Tsoulakos, N.; Kujala, P.; Hirdaris, S. A deep learning method for the prediction of ship fuel consumption in real operational conditions. *Eng. Appl. Artif. Intell.* **2024**, *130*, 107425. [CrossRef]
19. Ma, Y.; Zhao, Y.; Yu, J.; Zhou, J.; Kuang, H. An Interpretable Gray Box Model for Ship Fuel Consumption Prediction Based on the SHAP Framework. *J. Mar. Sci. Eng.* **2023**, *11*, 1059. [CrossRef]
20. Gkerekos, C.; Lazakis, I. A novel, data-driven heuristic framework for vessel weather routing. *Ocean Eng.* **2020**, *197*, 106887. [CrossRef]
21. Farag, Y.B.; Ölçer, A.I. The development of a ship performance model in varying operating conditions based on ANN and regression techniques. *Ocean Eng.* **2020**, *198*, 106972. [CrossRef]
22. Laurie, A.; Anderlini, E.; Dietz, J.; Thomas, G. Machine learning for shaft power prediction and analysis of fouling related performance deterioration. *Ocean Eng.* **2021**, *234*, 108886. [CrossRef]

23. Huang, G.; Liu, Y.; Xin, J.; Bao, T. Assessment of Hull and Propeller Performance Degradation Based on TSO-GA-LSTM. *J. Mar. Sci. Eng.* **2024**, *12*, 1263. [CrossRef]
24. Kim, H.S.; Roh, M.I. Interpretable, data-driven models for predicting shaft power, fuel consumption, and speed considering the effects of hull fouling and weather conditions. *Int. J. Nav. Arch. Ocean* **2024**, *16*, 100592. [CrossRef]
25. Logan, K.P. Using a ship's propeller for Hull condition monitoring. *Naval Eng. J.* **2012**, *124*, 71–87.
26. Székely, G.J.; Rizzo, L.M.; Bakirov, N.K. Measuring and testing dependence by correlation of distances. *Ann. Stat.* **2007**, *35*, 2769–2794. [CrossRef]
27. Kumar, A.; Hayatdavoodi, M. On wave–current interaction in deep and finite water depths. *J. Ocean Eng. Mar. Energy* **2023**, *9*, 455–475. [CrossRef]
28. Pedersen, B.P.; Larsen, J. Modeling of ship propulsion performance. In Proceedings of the World Maritime Technology Conference WMTC 2009, Mumbai, India, 21–24 January 2009; The Institute of Marine Engineers: Mumbai, India, 2009.
29. Coraddu, A.; Oneto, L.; Baldi, F.; Cipollini, F.; Atlar, M.; Savio, S. Data-driven ship digital twin for estimating the speed loss caused by the marine fouling. *Ocean Eng.* **2019**, *186*, 106063. [CrossRef]
30. Themelis, N.; Nikolaidis, G.; Zagkas, V.; Tsoulakos, N. Operational data analysis to aid the optimization of Retrofit solutions within the RETROFIT55 framework. In Proceedings of the 17th Annual Meeting of the Marine Technology, Palaio Faliro, Greece, 14–15 November 2023; pp. 59–73.
31. Breiman, L.; Friedman, J.; Stone, C.J.; Olshen, R.A. *Classification and Regression Trees*; Chapman and Hall: Boca Raton, FL, USA, 1984.
32. Breiman, L. Random Forests. *Mach. Learn.* **2001**, *45*, 5–32. [CrossRef]
33. Geurts, P.; Ernst, D.; Wehenkel, L. Extremely randomized trees. *Mach. Learn.* **2006**, *63*, 3–42. [CrossRef]
34. Friedman, J.H. Greedy function approximation: A gradient boosting machine. *Ann. Stat.* **2001**, *29*, 1189–1232. [CrossRef]
35. Gkerekos, C.; Lazakis, I.; Theotokatos, G. Machine learning models for predicting ship main engine Fuel Oil Consumption: A comparative study. *Ocean Eng.* **2019**, *188*, 106282. [CrossRef]
36. Friedman, M. The Use of Ranks to Avoid the Assumption of Normality Implicit in the Analysis of Variance. *J. Am. Stat. Assoc.* **1937**, *32*, 675–701. [CrossRef]
37. Wilcoxon, F. Individual comparisons by ranking methods. *Biometrics* **1945**, *1*, 80–83. [CrossRef]
38. Theodoropoulos, P.; Spandonidis, C.C.; Themelis, N.; Giordamlis, C.; Fassois, S. Evaluation of different deep learning models for the prediction of a ship's propulsion power. *J. Mar. Sci. Eng.* **2021**, *9*, 116. [CrossRef]