Prediction of vessel propulsion power from machine learning models based on synchronized AIS-, ship performance measurements and ECMWF weather data

Q Liang1, H A Tvete1 and H W Brinks1
1 Group Technology and Research - Maritime, DNV GL, Veritasveien 1, 1363 Høvik, Norway
Email: Qin.Liang@dnvgl.com

Abstract. In this paper, AIS (Automatic Identification System), ship performance measurement and ECMWF (European Centre for Medium-Range Weather Forecasts) weather data were synchronized as a complete dataset. Detailed processing steps and methods are introduced which can be used as best practice for future related studies. All the data preparation was processed on a spark cluster. The optimization and turning of cluster performance will also be introduced. The synchronized dataset was adapted to train different machine learning models to predict ship propulsion power. The dataset includes 228 container vessels covering a time scale of 50 months. The performance of deep learning models with different architecture was compared and discussed. Compared to previous paper [1] in the same project, this paper is an extended scenario which combines the data adapted in scenario 1 and 2. The analysis of the models’ performance in different scenarios was discussed. More features were included in scenario 3 (this study). Hence, the best performing model from scenario 3 has more complex structure compared to scenario 1 and 2. The overall absolute R² score for test data is slightly lower than scenario 2. However, the performance for individual ships (relative R² score) is much better. This means, models that consider more features (operation, ship characters and environment) are beneficial for individual analysis. For general fleet or wide scope analysis, models that require less data and fewer features are better.

1. Introduction
Shipping is facing the introduction of the global 0.5% sulphur cap this year, one of the important changes in its recent history. Up to 70,000 ships will be affected by this regulation [2]. Shipping is the main logistics channel for the global economy and is responsible for the carriage of around 90% of world trade. With the development of worldwide transaction, the emissions from ships show a significant and growing trend. According to the third greenhouse gas study (GHG) from International Maritime Organization (IMO) [3], shipping emitted 938 Mt CO₂ in 2012 which accounts for around 2.6% of global CO₂ emissions. It is already showing a reduction trend compared to 2007 (1100Mt and 3.5% of global CO₂ emissions) which can be attributed to slow steaming and the increase of vessel size [4]. This means appropriate measures are beneficial for the reduction of GHG emissions. IMO has adopted regulations to address the emission of air pollutants from ships and mandatory energy-
efficiency measures to reduce the emissions of greenhouse gases [5]. As a result of the increased international attention to global warming and air pollution, different stakeholders are beginning to take actions to reduce local and global emissions. E.g., the use of more environmentally friendly fuel, weather optimized routing, power optimization with battery and the possibility of using alternative energy to propel ships. According to [6], the road to greener shipping will see significant uptake of gas-driven propulsion. Most of the fuel will be LNG (liquefied natural gas) and LPG (liquefied petroleum gas), but some will be LBG (liquefied biogas). In order to evaluate the efficiency of these ship emission reduction measures accurately, an emission inventory model is vital.

One step before the calculation of emissions is to calculate the energy consumption or ships’ propulsion power. Various methods have been proposed to estimate ship energy consumption. These methods can be divided into two categories: top-down and bottom up. In [7], the authors presented the definition and an overview of these two methods. The common denominator is that they rely on the relationship between influencing factors and energy consumption. E.g., a detailed ship propulsion model needs to include hull dimension parameters, hydrodynamic performance and different resistance applied to the ships. Many factors can affect the calculation process, e.g. the selection of functions and coefficient. In contrast to these physics-based models, the primary objective of this paper is to evaluate the performance of different machine learning models’ ability to accurately predict ship propulsion power from AIS-, ship performance measurements and weather data.

In the previous paper [1] on this project, the performance of various machine learning models was benchmarked against measurement data and existing physical AIS-based model. In both scenarios in the previous paper [1], generally machine learning models perform better than existing physics-based models. While both scenarios have some limitations. For scenario 1, only 1 container vessel was investigated. Due to the limited access to weather data, only AIS, ship dimension characters and measurement were adapted for scenario 2. For this paper, a comprehensive dataset including AIS, ship dimension characters, weather and measurement was adapted as an additional scenario to explore the performance of machine learning models.

Machine learning is a powerful tool to handle complex problems that using normal methods would be difficult or even impossible. But machine learning should be used properly, it is not a solution to every problem. Data processing and preparation is a crucial step in training a good machine learning model. If data with lots of noise and errors are used for training, the performance of the model is certainly not qualified. In this paper, the data synchronization and processing will be introduced. It includes an illustration of how to combine maritime datasets, best practice for tuning cluster performance and verification of data.

There are many related studies trying to develop data-driven ship performance prediction models. However, most of them adapted a limited amount of data. For example, [8] developed a semi-empirical model on noon reports (manual reports) and sea trial data from 2 oil tankers. [9] analyzed the CO2 emission with AIS data through categorizing of ships. [10] explored the relationship between ship’s engine power, fuel consumption and the vessel speed by using 1 engine test data. [7] developed a Gaussian Process model considering both operation conditions and the impacts of weather but lack of ship dimension information. This paper aims to fill the gaps in these studies by applying machine learning models with data covering all aspects of vessel operation.

2. Methods and data sources

2.1. Methods
Machine learning, especially deep learning, has attracted a lot of attention in recent years. Especially its ability to solve complex problems. Both traditional machine learning and deep learning models have been
adapted in previous study [1], and deep learning models show better prediction results than the traditional machine learning methods. Hence, in this study will focus on exploring deep learning’s ability further.

Artificial Neural Network (ANN), also called deep learning, has been applied widely in recent years, e.g. mathematical prediction, image recognition and speech recognition. There are different types of architecture for ANNs, including Multilayer Perceptron (MLP), Convolutional Neural Network (CNN) and Recurrent Neural Network (RNN). ANN architecture is always improving and global researchers are trying to develop different kinds of ANNs for different applications. The problem being solved in this paper is a regression problem. The model aims to find the relationship between different input features and ship propulsion power. Hence, MLP is capable of solving it properly. The reason why the other complex ANNs were not adapted is already introduced in [1].

2.2. Data sources
In this paper, several main data sources were adapted, including AIS data, IHS Fairplay, ship performance measurement data and ECMWF weather data.

The AIS is a mandatory collision avoidance system required by IMO and the several countries’ Maritime Safety Administrations. For internationally operating ships with a gross tonnage (GT) of 300 or more, passenger ships of all sizes, domestic vessels with a GT of 200 or more traveling in coastal waters, and inland ships with a GT of 100 or more, it is mandatory to equip an AIS system.

IHS Fairplay provides the most thorough details on all sea-going, self-propelled merchant ships of 100 GT and above. For example, ship characteristics, ownership, port, installed engine power, design speed, etc. IHS Fairplay can be combined with AIS data to track live ship positions with unrivalled AIS coverage, analyze the risk profiles of ships’ operation, especially entering or leaving port.

The ship performance measurement data is obtained from a DNV GL in-house database comprised of 4209 ships that have reported their performance data at different granularities, from noon reports to 15-minute measurement intervals.

ECMWF process data including satellite and available observations from non-satellite sources to generate weather forecasts [11]. The forecasts rely on accurate observations of the current weather. From ECMWF data, the sea environment condition can be obtained, e.g. significant wave height, mean wave direction, neutral wind speed, wind direction, etc.

3. Data Processing

3.1. Data processing environment
With the development of new technology, more and more data has become available, and also the computing-power to process it. Different large-scale data processing engines have been under development and improving continuously. Among them the top-two popular engines are Apache Spark and Apache Hadoop. In recent years, Spark has become more popular than Hadoop because of its processing speed. For example, Spark is 100 times faster than Hadoop in solving a logistic regression problem in [12].

The core principle of big data processing is parallel-processing as shown in figure 1. Compared to traditional single node processing which is always limited by the hardware. Parallel-processing has no bottleneck on individual performance. If the computing power or memory was not enough, additional worker nodes will be added automatically by the master node. Both Spark and Hadoop have their advantages and disadvantages. The reason why Spark becomes more popular than Hadoop is because of its in memory processing principle and cloud providers’ flexibility. When spark is processing data, all of the data is loaded
into the cluster’s memory. Thus, it requires much more memory, however, the cloud providers have filled this gap.

![Figure 1. Cluster for parallel computing](image)

There are plenty of computers and associated components ready to be used in a cloud providers’ data center. Different hardware can be ordered and accessed by customers with simple steps. It provides both flexibility and scalability. E.g., a huge cluster including 30 worker nodes, 240 cores, 840 GB memory can be set up in 10 minutes.

All the large-scale data processing in this study is running on Azure Databricks from Microsoft. Databricks is a unified data analytics platform also a company founded by the original creators of Apache Spark. It is available both on Microsoft Azure and Amazon Web Services. Google Cloud has its own similar processing platform called Google Cloud Dataflow.

In this study, a cluster with 2 master nodes and 20 worker nodes was deployed. Master node and worker node will adapt the same type of virtual machine: Standard Ds4 V2 with 8 cores and 28 GB memory. The Databricks Runtime Version is 6.2 with Spark 2.4.4. PySpark is the Python API written in python to support Apache Spark.

3.2. Synchronization of AIS, IHS Fairplay and measurement data

As mentioned in chapter 2.2, there are four data sources that need to be synchronized. The AIS data and IHS Fairplay data are the easiest to synchronize. These two data sources are saved together as Delta Lake tables. For big data storage, normally parquet or orc will be selected as the storage format. Delta Lake is an open source storage layer that brings reliability to data lakes based on parquet format. Delta Lake offers a variety of additional features compared to parquet. E.g., ACID (atomicity, consistency, isolation, durability) transactions, metadata handling, and unifies streaming and batch data processing [13] which normal parquet does not provide.

In DNV GL’s AIS and IHS Fairplay Delta Lake tables, there are a primary key and a foreign key that define the relationship between these two tables. These two tables can be easily connected with a join function based on these two keys.

DNV GL’s performance measurement data is stored on a SQL server. In order to make it accessible with the other data sources, all the available performance measurement data was exported as csv file then loaded into Databricks as Delta Lake table. Because AIS data and measurement data have different sampling times and sampling frequency, it is difficult to synchronize the data without pre-processing. In addition, measurement data is the core dataset in the data processing operation which means no modification will be adapted. Hence, appropriate interpolation was applied to AIS data. Both data sources use the same timestamp format like ‘YYYYMMDD hh:mm:ss’. For normal AIS sampling situations, sample rates can vary from several seconds to several minutes depending on the type of transponder and vessel operation status.
In order to keep all the available records in measurement data, the AIS data has to be expanded to match with AIS data. Typical AIS data for 1 ship and a time scale of 50 months will have around 150,000 records. If this data was interpolated to cover every second for 50 months, there will be 129,600,000 records. In this study 228 container ships were adapted which means around 30 billion records will be generated. This is a huge data set which makes the processing both difficult and time-consuming. On the other hand, when interpolation is introduced, accuracy is also reduced. Several experiments were adapted to check the interpolation accuracy. Based on the analysis of experiment result, the best practice is to interpolate 120 seconds before and after available sample as shown in Figure 2.

After the pre-processing, the AIS data was left joined with the measurement data. The synchronization quality was checked based on the operation speed profile. If the speed profile is not similar or the speed difference between AIS and measurement data was bigger than 2 knots, the measurement was deleted. The synchronized dataset from this step is called the operation dataset in this paper.

3.3. Synchronization of Operation dataset with ECMWF data

ECMWF weather data have different components covering different aspects of weather. In this study, wave and wind information will be adapted. Before synchronization, pre-processing is applied to the operation dataset. The location information in the operation dataset is longitude and latitude which need to be converted to Cartesian coordinate system. After this step, x, y, z location information in Cartesian coordinate system becomes available.

The ECMWF wave and wind data has a measurement point for every hour or three hours. Interpolation was adapted to give full time scale coverage of every hour. The resolution for weather dataset is 0.36° which equals around 40 km. The next step is to create a mesh table, saving the unique location information of wave and wind data. The wave and wind data is saved as a daily parquet file with a file size slightly more than 1 GB. A year of wave and wind data (360 GB) was adapted to generate the mesh table to make sure it covers all the available zones.

All the wave and wind data will be loaded as pyspark dataframe from the daily parquet files with a size of 1800 GB. A special ‘Key’ column was created based on the time and location information. The time information adapted here is hours since 1900. An example is explained in Figure 3.
In order to match the operation dataset with the processed wave and wind data, the same ‘Key’ scheme should also be adapted. The location information in wave and wind data is fixed like the mesh table produced. However, the location information of operation dataset can be anywhere a ship operates. Hence, if wave and wind data need to match with the operation dataset, the operation dataset should have the closest weather information where wave and wind data is available. The mesh table was adapted to add weather location information to the operation dataset. The detailed steps were illustrated in Figure 4.

In addition, the cKDTree function adapted here is based on pandas library and should be applied to pandas dataframe. In order to apply the function with pyspark dataframe, either the pyspark dataframe being converted to pandas dataframe or the cKDTree function being applied as pyspark user defined function like showed in Figure 5.
The first method needs to collect all the data from work nodes to the master node which will overload the master node. The master node was adapted to coordinate workload between worker nodes, not for processing the data. If this method was adapted, the virtual machine size for the master node should be increased. Adapting this method can increase the compatibility between pyspark and pandas, but require more time and resources. In the worst scenario, it might not be possible to collect all the data from work nodes to master nodes. Because the number of worker nodes can be unlimited, but the size of master node is limited. The second method works well when the function is not so complex. If another third-party library was adapted or function with complex calculation, some compatibility problems might arise. The available function packages in spark is expanding continuously. The best practice is attempt to adapt spark original functions and avoid using user defined functions.

After this step, the operation dataset contains closest weather location information and a ‘Key’ can be generated. Until now, both operation dataset and weather data is ready to synchronize. In pyspark there are different join operations as in a normal database, but it also supports other efficient join operation optimized for distributed computing. In this case, weather data has 5.4 million rows per day, in total around 10 billion for 5 years. It is a huge dataset compared to the operation dataset, hence, spark broadcast join was adapted.

Spark broadcast is ideal for joining a large dataframe with a small dataframe, but cannot be used for two large dataframes. Spark will “broadcast” the small dataframe by sending all the data in the small dataframe to all the worker nodes in the cluster like showed in Figure 6. After the small dataframe is broadcasted,
spark can perform the join operation without shuffling the data in the large dataframe. On the other hand, the traditional left join operation could also be implemented but takes more time. Both methods have been tested and got same results in this study.

4. Models performance and result analysis

4.1. Feature selection

For scenario 1 and 2 in the previous paper, limited features were adapted in each scenario due to the limitation of data. After synchronization, a comprehensive dataset with operation, vessel characters and environment becomes available. In total 13 features were adapted in this study as presented in the following Table 1.

Table 1. Features comparison for Scenario 1, 2 and 3

| Scenario 1 | Scenario 2 | Scenario 3 (This study) |
|------------|------------|-------------------------|
| Feature    | Unit       | Feature                 | Unit       | Feature             | Unit       |
| Speed over ground | knots | Speed over ground | knots | Speed over ground | knots       |
| Significant wave height | m | Speed through water | knots | Heading           | degrees   |
| Wave direction to vessel | degrees | Course over ground | degrees |
| Wind speed | m/s | Moulded breadth | m |
| Wind direction to vessel | degrees | Moulded depth | m |
| Length between perpendiculars | m |
| Moulded breadth | m | Deadweight | ton |
| Moulded breadth | m | Significant wave height | m |
| Length between perpendiculars | m | Mean wave period | Hz |
| Wave direction to vessel | degrees |
| Wind speed | m/s |
| Wind direction to vessel | degrees |

The neural network architecture adapted in this study is MLP (Multi-Layer Perceptron). The correlation between the features and propulsion power was calculated in [1]. It is obvious that the direction related features are not closely related to propulsion power. The reason to include these features is the neural network’s ability to extract insights from multi-dimensional or complex relationships.

4.2. Models adapted

As in the previous study, the same number of models with the same architecture were adapted. Compared to the previous 2 scenarios, more features mean more parameters inside the neural network, and longer training times. The present regression problem is not as complex as image recognition, which requires complex layer and deep structure. In general, MLP does not need to be very deep. One to three hidden layers are generally enough to summarize abstract findings from data. The models adapted in this study were shown in Table 2. The model name refers to number of neurons in each layer.
Table 2. Models adapted in the study

| Model  | Model name  | S1 parameters | S2 parameters | S3 Number of parameters (This study) |
|--------|-------------|---------------|---------------|-------------------------------------|
| 1      | M_1000      | 7001          | 12001         | 22001                               |
| 2      | M_100       | 701           | 1201          | 1301                                |
| 3      | M_50        | 351           | 601           | 651                                 |
| 4      | M_30        | 211           | 361           | 391                                 |
| 5      | M_20        | 141           | 241           | 261                                 |
| 6      | M_10        | 71            | 121           | 131                                 |
| 7      | M_5         | 36            | 61            | 66                                  |
| 8      | M_500_100   | 53201         | 55701         | 56201                               |
| 9      | M_100_50    | 5701          | 6201          | 6301                                |
| 10     | M_50_30     | 1861          | 2111          | 2161                                |
| 11     | M_10_5      | 121           | 171           | 181                                 |
| 12     | M_50_20_10  | 1541          | 1791          | 1841                                |
| 13     | M_10_8_4    | 189           | 239           | 249                                 |
| 14     | M_10_8_6_4  | 235           | 285           | 295                                 |

4.3. Model performance and analysis
The prediction of propulsion power is a regression problem. Hence the $R^2$ score, also called coefficient of determination, was selected as the metric to compare the performance of the models. It ranges from 0 to 1, and the higher the $R^2$ score, the better and more accurate prediction was made. If it is negative, it means the prediction does not follow the trend of the data and fits worse than a horizontal line. The detailed function is shown below. $SS_{tot}$ is the total sum of squares. $SS_{rec}$ is the regression sum of squares. $f_i$ is the prediction result from the model.

$$SS_{tot} = \sum_{i}(y_i - \bar{y})^2$$
$$SS_{rec} = \sum_{i}(y_i - f_i)^2$$

$$R^2 = 1 - \frac{SS_{rec}}{SS_{tot}}$$
There are 27 container ships in test data, but they are not equally distributed. Some of the major ships represent more than 50% of the test data. The minor ships represent 1% of the test data. Therefore, the absolute $R^2$ for the entire test data is close to the major ships. As showed in Figure 7, the $R^2$ score of the whole test data (absolute $R^2$) for all the models in scenario 2 is higher than scenario 3. But it does not present the real prediction ability that the model demonstrated for individual vessel. In order to present models’ ability for individual vessel, relative $R^2$ is introduced. The absolute $R^2$ is the $R^2$ for the whole test dataset. The relative $R^2$ is the average of $R^2$ for each test vessel.

![Figure 7. Absolute $R^2$ for ANN models](image)

The number of neurons in the hidden layer should typically be the average of input and output parameters. The model with more neurons than required by the general principle performed well. Hence, the best practice from previous study to choose an ANN model proved correct. Model M50 was selected to benchmark the performance. In addition, model M10_5 was also selected with its performance and number of parameters. Since more features were adapted, especially environmental features, 2 hidden layers can extract deeper relationship than 1 hidden layer.

![Table 3. Absolute and relative $R^2$ for Scenario 2 and 3](image)

| Scenario | Scenario 2 M20 | Scenario 3 M10_5 | Scenario 3 M50 |
|----------|----------------|------------------|----------------|
| Absolute average $R^2$ | 84.6% | 79.2% | 80.7% |
| Relative average $R^2$ | 40.7% | 61.3% | 64.7% |

In Table 3, the average of absolute and relative $R^2$ were compared. Even though the absolute $R^2$ for scenario 2 is higher than scenario 3, the relative $R^2$ for scenario 3 is much better than scenario 2. The $R^2$ for all the test vessels is shown in Figure 8.

![Figure 8. Relative $R^2$ for ANN models](image)
Model S3_M10_5 performed better on 14 of 27 ships than model S2_M20. Model S3_M50 performed better on 19 of 27 ships. Irrespective of the number of ships or the relative average $R^2$, both models in scenario 3 performed better than the model in scenario 2. It means that more features adapted in the training data can help the model learn more complex relationships from it. The added resistance from waves and wind is a major part of resistance affecting the ships’ performance. To calculate the added resistance requires a lot of effort, e.g. adapting empirical functions and statistics. Many factors and index are introduced during this calculation process. It also means that uncertainty and assumptions are introduced. On the contrary, the data-driven methods only rely on the input features and output propulsion power, which avoid bringing in uncontrollable factors. Hence, the data-driven models are more reliable than the physics models.

5. Discussion
The performance of physics-based models and were compared in previous study. In both scenario 1 and 2, the data-driven models performed better. In this paper, an extended study – scenario 3 was applied. The operational data, ship characteristics, performance measurement data and environmental data were synchronized to create a comprehensive dataset.

The comprehensive dataset was used to train ANN models. More features were adapted in scenario 3 which provides not only the ship operation information but also the environment within which the ship operates. Compared to the best performing model (M20 in scenario 2), models with more layers (M10_5) and more neurons in the hidden layer (M50) performed better in scenario 3. Model M10_5 performed better due to the inclusion of environment features, especially features not directly correlated with output, mean that the regression model needs a higher degree of nonlinear regression. Scenario 3 has more features than scenario 2. Model M50 has more neurons in the hidden layer which can summarize and memorize more learnings from the training data.

Because the test data is not evenly distributed, the absolute $R^2$ score for the test data leans towards the major ships. Therefore, both absolute and relative $R^2$ score were calculated and compared. The relative $R^2$ score could represent a model’s ability in handling different ships. The ANN modes in scenario 3 performed much better in relative $R^2$ score than scenario 2. On the other hand, the ANN models in scenario 2 are also good models. First they performed much better than the physics-based model introduced in [1]. Second, they require less data to predict. The weather data is not always available. In addition, even though the weather data is available, it still requires a lot of effort on data synchronization and data quality check. Hence, the ANN models in both scenario 2 and 3 are valuable. When focusing more on the performance prediction of individual vessels and weather is available, ANN models in scenario 3 could be adapted. Models from scenario 2 could be used when less data is available or when trying to predict the performance of a fleet of ships.

The data-driven models provide a short cut for performance prediction when there is a lack of domain knowledge for feature engineering. It also avoids empirical functions and correction factors being introduced which reduce the reliability of the model. The data-driven models trained with different data can be utilized in different scenarios. The best practices for processing and synchronization of the data are introduced in this paper. With the help of cloud technology and the Spark data processing engine, big data processing becomes much easier and applicable. The data-driven model demonstrates its ability to handle complex prediction problem. It is strongly reliant on the data it has been trained on. If the training data is not representative or the data quality is not good, the data-driven models will not necessarily make good predictions.

In this study, container ships were selected as the pilot group. Other ship types will be explored in the continuation of this work. The data-driven models will also be implemented in DNV GL’s production Databricks cluster as a reference to check its ability in a continuous manner. In addition to this, a more
detailed physical model adjusted for added resistance is under development in DNV GL [14]. It will provide further comparison for the performance of data-driven models and physics-based models with the same data.

Acknowledgements
This work was funded by the VERDE (short for VERification for DEcarbonisation) project, which is funded by the Research Council of Norway by grant #282293. Special thanks to Vidyunmala Veldore in DNV GL Digital Solutions for help with initial synchronization work and Håvard Nordtveit Austefjord in DNV GL Maritime Advisory for providing the environment data.

References
[1] Liang Q, Tvete H A and Brinks H W 2019 Prediction of vessel propulsion power using machine learning on AIS data, ship performance measurements and weather data J. Phys. Conf.
[2] Ser. 1357
[3] Chryssakis C, Brinks H, Brunelli A C, Fuglseth T P, Lande M, Laugen L, Longva T, Raeissi B and Tvete H A 2017 Low Carbon Shipping Towards 2050 32
[4] IMO 2014 Greenhouse Gas Emissions
[5] Lindstad H and Eskeland G S 2015 Low carbon maritime transport: How speed, size and slenderness amounts to substantial capital energy substitution Transp. Res. Part D Transp.
[6] Environ. 41 244–256
[7] IMO 2019 Low carbon shipping and air pollution control
[8] DNV GL’s Group Technology & Research 2019 Energy transition Outlook 2019 Waddenl.
[9] Outst. 269–282
[10] Yuan J and Nian V 2018 Ship energy consumption prediction with Gaussian process metamodel Energy Procedia 152 655–660
[11] Lu R, Turan O, Boulougouris E, Banks C and Inccek A 2015 A semi-empirical ship operational performance prediction model for voyage optimization towards energy efficient shipping Ocean Eng. 110 18–28
[12] Rashidia M and Jaswar 2014 Prediction of CO2 emitted by marine transport in Batam-
[13] Singapore channel using AIS J. Teknol. (Sciences Eng. 69 121–126
[14] Kowalak P, Kasyk L and Borkowski T 2011 Assessment of ship’s engine effective power, fuel consumption and emission using the vessel speed J. KONES 18 31–39
[15] ECMWF 2020 ECMWF Official website
[16] Apache 2020 Apache Spark
[17] Lake D 2020 Introduction to Delta Lake
[18] Tvete H A, Guo B, Liang Q and Brinks H 2020 A modelling system for power consumption of marine traffic OMAE2020-18651 (Fort Lauderdale, FL, USA: Conference on Ocean, Offshore and Arctic Engineering) 1–10