# Data-driven time series forecasting of offshore wind turbine loads

Muhammad Amri, Hafiz Ghazali Bin; Marramiero, Daniela; Singh, Deepali; Van Wingerden, Jan Willem; Viré, Axelle

**DOI**
10.1088/1742-6596/2767/5/052060

**Publication date**
2024

**Document Version**
Final published version

**Published in**
Journal of Physics: Conference Series

**Citation (APA)**
Muhammad Amri, H. G. B., Marramiero, D., Singh, D., Van Wingerden, J. W., & Viré, A. (2024). Data-driven time series forecasting of offshore wind turbine loads. *Journal of Physics: Conference Series*, *2767*(5), Article 052060. https://doi.org/10.1088/1742-6596/2767/5/052060

# Data-driven time series forecasting of offshore wind turbine loads



Journal of Physics: Conference Series **2767** (2024) 052060 doi:10.1088/1742-6596/2767/5/052060

# Data-driven time series forecasting of offshore wind turbine loads

**Hafiz Ghazali bin Muhammad Amri[1], Daniela Marramiero[2], Deepali Singh[1], Jan-Willem Van Wingerden[3], Axelle Viré[1]**

[1] Wind Energy Section, Faculty of Aerospace Engineering, TU Delft, Kluyverweg 1, 2629 HS Delft, The Netherlands
[2] RWE Offshore Wind, World Trade Center Utrecht, 3521 AZ Utrecht, The Netherlands
[3] Delft Center for Systems and Control,Faculty of Mechanical Engineering, TU Delft, Mekelweg 2, 2628 CD Delft, The Netherlands

E-mail: h.binmuhammadamri@tudelft.nl

**Abstract.** Long Short-Term Memory Recurrent Neural Networks (LSTM) are used to build surrogate models to forecast time-series blade loads for both fixed and floating offshore wind turbines. In this paper, we train surrogate models on datasets generated with OpenFAST on the IEA-15MW-RWT under a range of metocean conditions. The aim of the surrogate models is to generate load forecasts inexpensively and accurately such that they can be used in a model predictive controller. Two cases are investigated with different model inputs: one with only measurements available to typical PI controllers and another one with additional wave elevation and deflection measurements (alongside the endogenous variable). The model performances are evaluated and compared. It was found that for the fixed turbine, the models predicted all three blade loads to a high degree of accuracy. The floating turbine surrogate models performed relatively worse, but edgewise and pitching moments are still reasonably accurate. The surrogate model forecasts the flapwise moment to a satisfactory accuracy only in 58% out of 400 test cases. The addition of wave elevation and blade deflection features did not significantly improve the prediction performance of the surrogate, demonstrating that just the information used by current PI controllers may be sufficient for forecasting blade loads.

## 1. Introduction

Reducing the levelised cost of electricity of offshore wind farms is important to further incentivize their installation and help us fulfill our renewable energy targets. For this purpose, it is important that the power output and operational life of offshore wind turbines is improved, which can be achieved through the use of data-driven model predictive control (MPC) over proportional integral controllers (PI). This can be a significant advantage especially for floating offshore wind turbines, where platform motion can cause unsteady aerodynamics, resulting in losses in energy production and overall turbine lifetime [1, 2].

MPC involves the use of a fast model predicting the future behaviour of a system. For wind turbines, the model used is typically a linearised representation of a physics-based model about an operating point [3, 4]. However, when the dominant dynamics is highly nonlinear, this can result in an oversimplification. Data-driven surrogate models (SMs) present an interesting alternative for this purpose: once trained, they are computationally inexpensive and so can quickly generate finite time-horizon predictions of the variables required by the controller. These

models also require less domain knowledge, and the framework for producing the models is easily transferable between turbines, given that data is available or easily generated.

The ability of data-driven surrogate models to accurately emulate physics-based models while having a greatly reduced computational cost has seen them used in various ways in wind energy. This includes speeding up wind turbine modelling through reducing the expense of computational fluid dynamics for wake modelling [5, 6] and characterizing hydrodynamic response [7], bypassing blade element momentum theory for aerodynamic load estimation [8], and to predict the load statistics on wind turbines [9, 10, 11]. For this reason, there has been significant work in producing SMs for wind farm design and planning, fatigue monitoring and design load assessment.

However, there has been very limited work in data-driven time series forecasting of loads, as the majority of work in time series forecasting has been focused on producing forecasts of wind power and speed [12, 13, 14]. Dimitrov and Göçmen [15] used LSTMs as a virtual sensor for predicting blade root flapwise bending moment estimation, blade tip deflection forecasting, and wake center position detection. In this study, the SM inputs are obtained from Supervisory Control and Data Acquisition (SCADA), lidars, and met-mast measurements, whilst in the aforementioned studies, the SMs take SCADA measurements or metocean conditions as input. To be implemented in a model predictive control strategy, the SM would have to be trained on measurement data available during turbine operation as inputs, which is the goal of this work.

In this paper, we present a methodology to produce surrogate models that generate time series forecasts of edgewise, flapwise and pitching moments for both fixed and floating offshore wind turbines with long short-term memory networks (LSTM). The SMs are trained from data generated by the aero-hydro-servo-elastic code, OpenFAST [16, 17], with turbine measurement data as inputs and finite time horizon predictions of the blade root moments as outputs.

The paper is structured as follows. Section 2 describes the overall methodology in producing the SMs, with section 2.1 explaining the generation of the training dataset and section 2.2 going over the SM structure, inputs and outputs, and training. The performance of the SMs for the different cases is discussed in section 3. The main findings are summarised in the conclusions in section 4.

## 2. Methodology

### *2.1. Dataset generation*

*2.1.1. Simulation setup* The training datasets are generated by performing OpenFAST simulations on the IEA-15MW offshore reference wind turbine with frozen turbulence field generated by TurbSim as input [18]. For the fixed cases, the turbine is fixed on a monopile support structure, whilst for the floating cases, the semisubmersible UMaine VolturnUS-S reference platform is used [19]. The aerodynamics calculations are performed using Blade Element Momentum Theory (BEM). First-order waves are modelled with the JONSWAP Spectrum in HydroDyn. All simulations are run for 900s each, with the first 300s removed to avoid initial transient. A total of 500 cases are simulated for the fixed case and 2048 for the floating case.

*2.1.2. Metocean conditions* The OpenFAST inputs varied in the simulations are:

- 3 wind variables (2 for fixed case):
  - Wind speed $u$
  - Turbulence intensity $TI$
  - Wind shear exponent $\alpha$ (only for the floating case)
- 3 wave variables:
  - Significant wave height $H_s$

  - Wave direction $W_{dir}$
  - Peak spectral wave period $T_p$
- 3 random seeds (2 wave seeds and 1 turbulence seed)

In the fixed case, the variable bounds are defined through site-specific data located 30km off the west coast of Ireland, obtained from [20]. In the floating case, the bounds are given in Table 1, and are referenced from [9, 10]. The inputs are generated within the bounds through quasi-random low discrepancy sequencing.

Table 1: Input variable ranges for the floating turbine dataset

| Variable | Lower Bound | Upper Bound |
|---|---|---|
| Wind speed $u$ ($\mathrm{ms}^{-1}$) | 3 | 25 |
| Turbulence Intensity $TI$ (%) | 2.5 | $\frac{18}{u}\left(6.8+0.75u+3(\frac{10}{u}^2)\right)$ |
| Wind shear exponent $\alpha$ (-) | $0.15-0.23\left(\frac{u_{max}}{u}\right)$ $\left(1-\left(0.4log\frac{R}{z}\right)^2\right)$ | $0.22+0.4\left(\frac{R}{z}\right)\left(\frac{u_{max}}{u}\right)$ |
| Significant wave height $H_s$ (m) | 0 | 6 |
| Wave direction $W_{dir}$ (deg) | -180 | 180 |
| Peak spectral wave period $T_p$ (s) | 1 | 21 |
| Turbulence seed (—) | -50000 | 50000 |
| Wave seed 1 (—) | -50000 | 50000 |
| Wave seed 2 (—) | -50000 | 50000 |

*2.2. Surrogate Model Training*

*2.2.1. Model Inputs* To investigate the impact of the SM inputs on the prediction accuracy, two cases are considered:

- Case 1, with as SM inputs: (1) the TurbSim reference wind speed URef, (2) the estimated wind speed time series WindVel, (3) the wave elevation time series WaveElev, (4) the in-plane deflection IPDefl (for Edgewise and Pitching Moments), (5) the out of plane deflection OoPDefl (for Flapwise Moment) and (6) the endogenous feature (the forecasted blade load).
- Case 2, with as SM inputs: (1) the estimated wind speed time series WindVel, (2) the rotor speed RotSpeed and (6) the endogenous feature (the forecasted blade load).

These two cases are considered to investigate the differences made by the additional available inputs to those used by typical PI controllers to the SM predictions. Figure 1 shows a heatmap of the mean cross-correlation coefficients between the loads and the exogenous input features (other than reference wind speed, which is constant over a simulation). It is clear that the in-plane and out-of-plane deflections have the highest cross-correlation with the blade loads; however, the cross-correlation coefficient does not tell the full story as it only identifies linear relationships. For example, we know that the value of the rotor speed determines the 1P frequency, which impacts the frequency of oscillation of the blade loads. As such, they are still included as input features.

*2.2.2. Surrogate Model structure* Long short-term memory networks (LSTMs) are a type of Recurrent neural network (RNN) designed to solve the vanishing/exploding gradient problem encountered in training traditional RNNs [21]. As opposed to more typical feedforward neural networks, RNNs are bi-directional, thus resulting in a form of memory of the previous process,

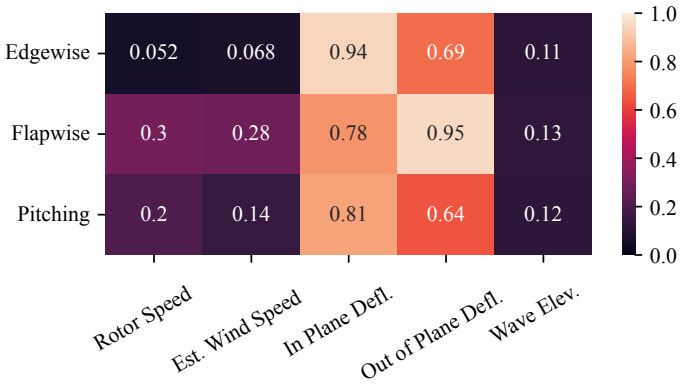


Figure 1: Heatmap of mean cross-correlation coefficients between loads and exogenous features.

which can be used for subsequent predictions. This characteristic makes RNNs a popular choice for time-series data. Due to the vanishing/exploding gradient problem, traditional RNNs can typically only look back in time for about 10 timesteps, whilst LSTMs can learn from more than 1000 timesteps [21], which makes them a suitable choice of surrogate model for the purposes of this work.

The SMs used in this work use an encoder-decoder architecture. This model consists of an encoder layer which maps the input sequence into a fixed-length internal representation that captures the sequential dependencies [22], and an output layer which maps this to the output sequence recurrently.

The general structure of all SMs in this work are the same, consisting of an encoder layer and a decoder layer with 30 nodes each, chosen as a compromise between training time and model accuracy after a trial-and-error tuning process. For the fixed cases, 20 previous timesteps are used as inputs to the SM, whilst for the floating cases, this is increased to 40 (corresponding to 30 s). This is to account for the lower frequency dynamics present in floating wind turbines; the longest peak spectral wave period in the dataset being 21 s and the natural period of the floater in pitch being 27.8 s [19]. Finally, the output of the SMs consists of a one-dimensional vector made of 40 values (representing the 40 forecasted timesteps). The SM input and output sequences are illustrated in Figure 2.

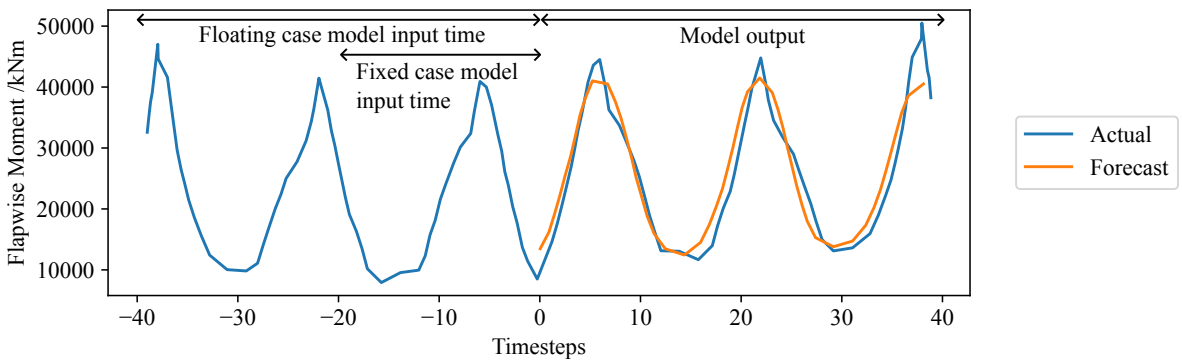


Figure 2: Illustration of model input and output, where t=0 is the current time, the actual value of the blade load is in blue and the surrogate forecast is in orange.

*2.2.3. Model Training* The OpenFAST simulations generated results with a time–step of 0.005s, corresponding to a frequency of 200Hz. Averages over every 150 values are performed, resulting in a time-step of 0.75 s, which is sufficient to capture the frequency of blade loads. Out of the 500 simulations for the fixed case, the 200 most significant are deemed sufficient to represent the sampled area . For the floating case, all 2048 simulations in the dataset are used. For both cases, the data is split into two subsets: 80% for training and 20 % for validation.

The training input-output pairs are formed using a sliding window process — for each 10 minute (800 timesteps) simulation, 19 overlapping sub-sequences are obtained, containing the input-output sequences (20 or 40 timesteps as input and 40 timesteps as output), as illustrated in Figure 3. The use of this sliding window augments the dataset, increasing its size and diversity.

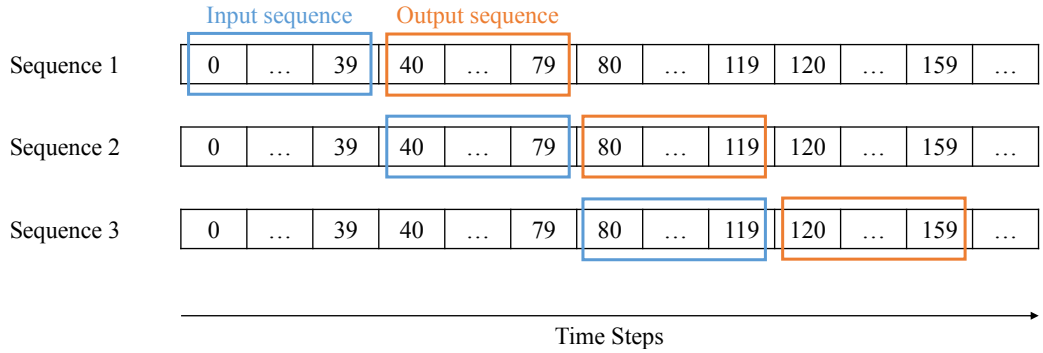


Figure 3: Diagram illustrating the sliding window of the floating case. with the blue boxes representing the input sequence and the orange boxes representing the output sequence

Taking into account training time and model accuracy, the training hyper-parameters selected are summarised in Table 2.

Table 2: Summary of hyper-parameters used for training the LSTMs of both the fixed and floating cases

| Hyper-parameter | Fixed Case | Floating Case |
|---|---|---|
| Batch size | 16 | 32 |
| Number of epochs | 120 | 60 |
| Learning rate | 0.0005 | 0.0001 |
| Number of nodes per layer | 30 | 30 |
| Number of input timesteps | 20 | 40 |
| Number of output timesteps | 40 | 40 |

## 3. Results and Discussion

The results from testing the SMs for the different cases are summarised in Figure 4, which depicts the Root Mean Squared Error (RMSE) of each respective blade load against the forecasted timestep. The RMSE is given by:

$$\text{RMSE} = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y}_i)^2} \tag{1}$$

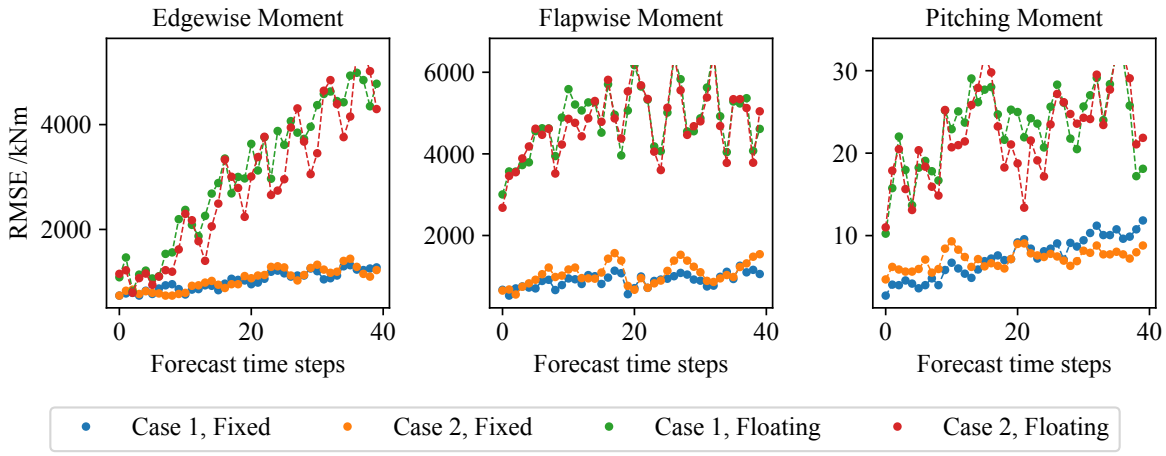


Figure 4: RMSE against forecasted time-steps for different cases

where $N$ is the number of considered data points, $y_i$ is the actual value of the data point and $\hat{y}_i$ is the predicted value.

For the fixed turbine, both Case 1 and Case 2 SMs performed well, generating accurate forecasts of the blade loads. The average range in blade load over 40 timesteps is about 380000 kNm for the edgewise and flapwise moments, and 330 kNm for the pitching moment. From Figure 4, it is evident that the RMSE values are a small fraction of this, demonstrating the accuracy of the SMs for the fixed cases. It is also evident that SMs in both cases performed very similarly, which is a very promising result as it demonstrates that including time histories of the endogenous feature with the information used by current PI controllers is sufficient for accurate forecasting of blade loads. Due to the model structure, there is no significant difference in forecasting CPU time between the two input cases — Case 1 takes an average of 88.2ms per forecast while Case 2 takes an average of 83.3ms. However, a lower number of input features means reduced uncertainty (due to uncertainty in measurements), which means that there is greater confidence in SM predictions.

For the floating turbine, the RMSE values are significantly higher than the fixed cases. While the errors are still reasonable when compared to the typical load ranges, it is clear that the SMs do not perform as well as the fixed cases. Looking at the example forecasts in Figure 5, it can be seen that there are several ways in which the SMs can lead to larger values of RMSE. In Figure 5, three examples of poor predictions are shown — in the case of edgewise moment, it is due to the SM forecasting a different frequency to the actual signal. In the case of the pitching moment, it is due to the forecast signal having a different amplitude to the actual signal. In the case of the flapwise moment, the SM fails to capture the high frequency variations in the reference signal. Despite the relatively inaccurate predictions, the SM output in those cases would still be useful for control purposes. We can observe that the SMs struggle the most when the actual signals do not have any apparent seasonality — this can be due to the environmental conditions (highly turbulent, wind-dominant dynamics) or limited information in the input features.

Figure 5 also reveals the limitations of using RMSE as a performance metric for our purposes. Cases with mismatched frequencies or amplitudes may result in very high RMSE despite being comparatively good forecasts. For this reason, to evaluate the performance of the floating case and the reason behind the higher errors, the Variance Accounted For (VAF) is used. The VAF is a performance metric which measures the percentage of the total variance in the actual signal that is accounted for by the variance in the forecast signal. VAF values range from 0 to 100%, with larger values signifying better performance. It is expressed as

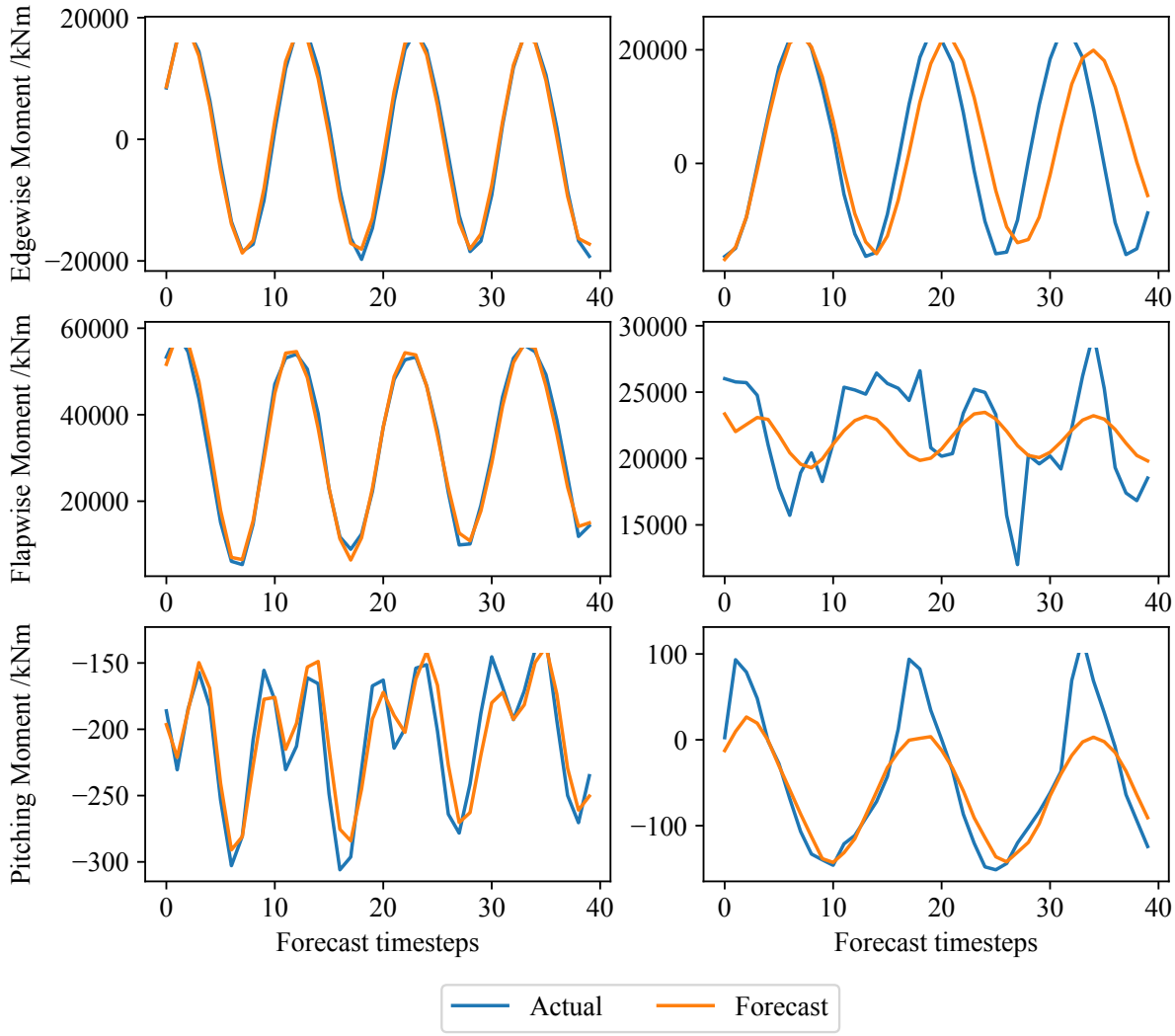


Figure 5: Examples of time series forecasts of blade loads for the floating turbine (case 2). Plots on the left show examples of cases with good predictions (low RMSE values), while plots on the right show examples of cases with poor predictions (high RMSE values).

$$\mathrm{VAF}(y, \hat{y}) = 100\% \times \max\left(\frac{\sum_{i=1}^{n}(y_i - \bar{y})(\hat{y}_i - \bar{\hat{y}})}{\sum_{i=1}^{n}(y_i - \bar{y})^2}, 0\right) \tag{2}$$

where $y$ is the actual signal and $\hat{y}$ is the forecast signal. Since it is found that the SM performance does not vary significantly over time (i.e. SM forecast performance is similar over the 20 sub-sequences within each 10 minute simulation), the mean VAF is taken, resulting in one value for each set of input conditions in the test data. To look at the impact of floating platform motion on SM performance, we take the approximate average absolute speed of the hub $s$ in the surge direction, given by

$$s = \mathrm{mean}\left(\left|\frac{d}{dt}\left(H_{hub}\sin(\theta) + x\right)\right|\right) \tag{3}$$

where $H_{hub}$ is the hub height, $\theta$ is the platform pitch and $x$ is the platform surge. Figure 6 shows scatter plots of the mean VAF against $s$ for each of the three blade loads.

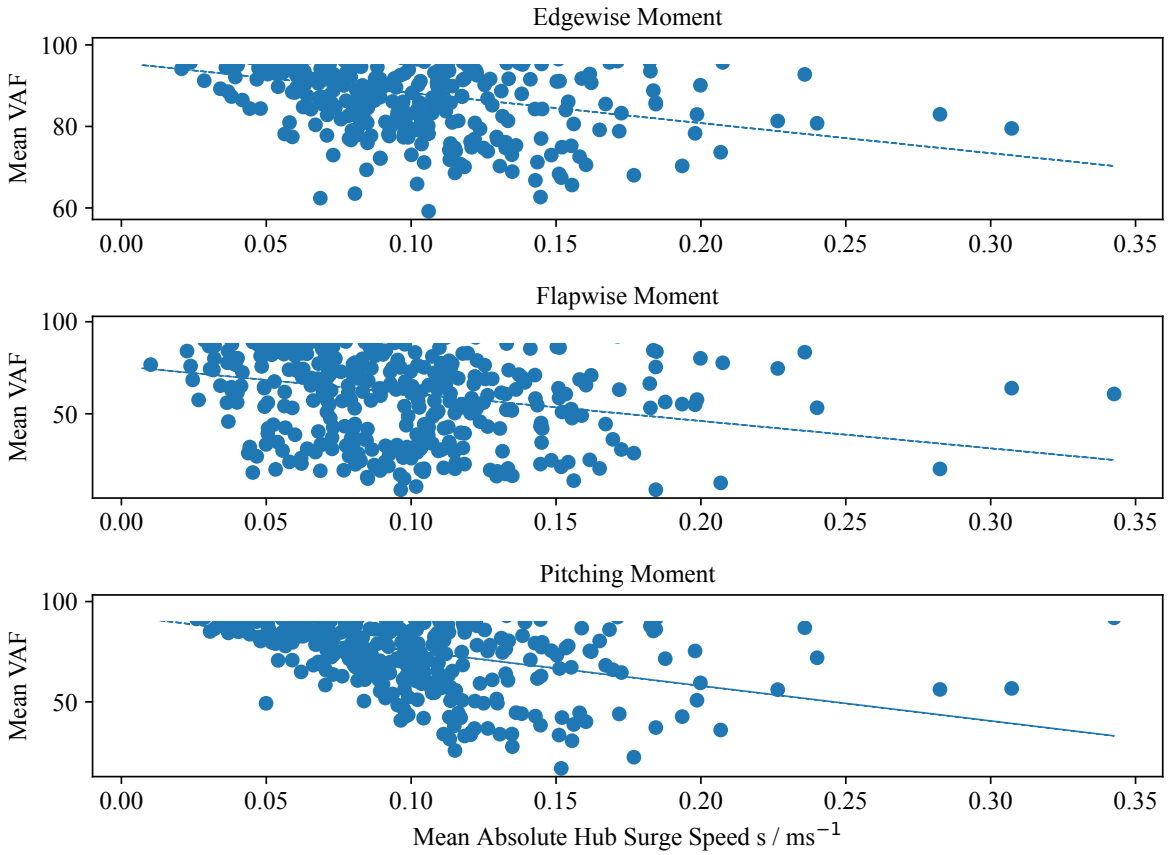


Figure 6: Mean VAF (over 19 sub-sequences within a simulation) against Mean Absolute Hub Surge Speed for all 3 blade loads for the floating turbine (case 2).

It is clear that there is a negative correlation between the VAF and the rotor hub motion. When values of $s$ tend to zero, the values of VAF are close to 100%, while variation in VAF increases with $s$. This corresponds to the low RMSE values of the fixed cases shown in Figure 4, as stationary (or very slow moving) floating turbines would behave similarly to a fixed turbine. Out of the three loads shown here, the SM for flapwise moment has the greatest sensitivity to blade loads, having cases with low VAF even at small values of hub speeds which is not the case for the edgewise or pitching moments. This is expected, as the turbine motion is in the same direction as the flapwise moment.

Figure 7 shows the cumulative distribution of Mean VAF for the 3 blade loads. Values of VAF $> 60\%$ indicates that the model performs reasonably well for the given signal. From Figure 7, it can be concluded that the SMs for edgewise and pitching moments perform well. For edgewise moment, 99.75% of the test data has a VAF $> 60\%$, while for the pitching moment, 82.5% lies above this threshold. For flapwise moment, however, the SM did not perform satisfactorily, with only 58% of its forecasts having VAF $> 60\%$.

There are several options to improve the modelling performance of flapwise moment for floating wind turbines. Further tuning (i.e. increasing number of epochs, nodes or layers at the expense of training time) may yield higher accuracy, although this is unlikely to work for the cases with no apparent trend or seasonality, such as that shown in Figure 5. It is possible that the reason behind this signal behaviour is due to the presence of high frequency dynamics not

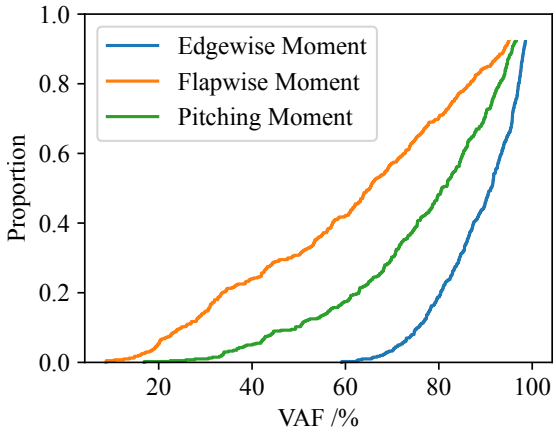


Figure 7: Cumulative distribution of Mean VAF for all 3 blade loads for the floating turbine (case 2).

captured by the sampling rate. If this is the case, it would add an additional complexity to the SM, as the number of input timesteps would need to be higher leading to higher computational costs during training, which might make it impractical.

## 4. Conclusions

This paper presented the use of LSTMs for the time series forecasting of edgewise moment, flapwise moment and pitching moment on fixed and floating offshore wind turbines. The surrogate models were trained on datasets generated with OpenFAST on the IEA-15MW offshore reference wind turbine (with the UMaine VolturnUS-S semisubmersible for the floating case). Two cases were investigated: Case 1 where the SM receives as inputs the reference wind speed, the estimated wind speed, the wave elevation, the in-plane deflection, the out of plane deflection and the respective blade load history (the endogenous feature), and Case 2 where the SM receives only inputs used by typical PI controllers: the estimated wind speed and the rotor speed and the respective blade load history.

The RMSE results show that LSTM-based SMs can accurately forecast time series of all 3 loads for the fixed turbine. Both Case 1 and Case 2 SMs demonstrate similarly low RMSE, revealing that LSTMs can model fixed turbine blade loads by including the load histories with the limited information available to current PI controllers. For the floating turbine, both Case 1 and Case 2 show similar RMSE, however, they are significantly higher than those of the fixed turbine. As both Case 1 and Case 2 SMs demonstrate similar RMSE and similar forecasting time, future work should build on Case 2 as it has the advantage of lower input uncertainties (due to having less features) and requiring a smaller dataset.

From the time series forecasts of the loads, it was determined that there are 3 main patterns behind cases with high RMSE: (1) Frequency mismatch, (2) amplitude mismatch and (3) lack of seasonality in actual signal. By looking at the Mean VAF of the blade load forecasts, it was found that there is a negative correlation between the rotor motion due to platform pitch and surge, and the SM performance. This explains why the floating cases had higher RMSE than the fixed ones. It was found that while the SMs for forecasting edgewise and pitching moments had good performance, flapwise moment was not modelled sufficiently well, with only 58% of test cases above the 60% VAF threshold. Future work will focus on further investigating how the predictions of the flapwise moment can be improved for the floating turbine. The training dataset will also be enriched with higher-fidelity models, which can increase the accuracy of the

dataset if the floating turbine experiences large unsteady aerodynamics.

**Acknowledgements**
This research has received funding by the Dutch National Research Council (NWO) under the Talent Programme Vidi scheme (project number 19675).

## References

[1] Dong J and Viré A 2021 *Renewable Energy* **163** 882-909
[2] Sebastian T and Lackner M A 2012 *Wind Energy* **16** 339-352
[3] Lio W H, Rossiter J A and Jones B L 2014 *2014 UKACC International Conference on Control (CONTROL)* 673-678
[4] Schlipf D, Schlipf D J and Kühn M 2013 *Wind Energy* **16** 1107-1129
[5] Zhou L, Wen J, Wang Z, Deng P and Zhang H 2023 *Energy* **275** 127525
[6] Wilson B, Wakes S and Mayo M 2017 *2017 IEEE Symposium Series on Computational Intelligence (SSCI)* 1-8
[7] Ilardi D, Kalikatzarakis M, Oneto L, Collu M and Coraddu A 2024 *IEEE Access* **12** 6494-6517
[8] Baisthakur S and Fitzgerald B 2024 *Renewable Energy* **224** 120122
[9] Dimitrov N, Kelly M C, Vignaroli A and Berg J 2018 *Wind Energy Science* **3** 767-790
[10] Singh D, Dwight R P, Laugesen K, Beaudet L and Viré A 2022 *J. Phys.: Conf. Ser.* **2265** 032070
[11] Schröder L, Dimitrov N K and Sørensen J A 2020 *J. Phys.: Conf. Ser.* **1618** 042040
[12] Liu X, Lin Z and Feng Z 2021 *Energy* **227** 120492
[13] Zhang W, Lin Z and Liu X 2022 *Renewable Energy* **185** 611-628
[14] Panapakidis I P, Michailides C, Angelides D C 2019 *Electronics* **8** 420
[15] Dimitrov N and Göçmen T 2022 *Wind Energy* **25(9)** 1626-1645
[16] OpenFAST https://github.com/openfast accessed: 2023/09/11
[17] Jonkman J 2013 *51st AIAA Aerospaces Sciences Meeting including the New Horizons Forum and Aerospace Exposition*
[18] Gaertner E, Rinker J, Sethuraman L, Zahle F, Anderson B, Barter G, Abbas N, Meng F, Bortolotti P, Skrzypinski W, Scott G, Feil R, Bredmose H, Dykes K, Shields M, Allen C and Viselli A 2020 *IEA Wind TCP Task 37*
[19] Allen C, Viselli A, Dagher H, Goupee A, Gaertner E, Abbas N, Hall M, Barter G 2020 *IEA Wind TCP Task 37*
[20] WaveClimate http://waveclimate.com accessed:2023/09/12
[21] Staudemeyer R C and Morris E R 2019 *arXiv preprint arXiv:1909.09586.*
[22] Du S, Li T and Horng S -J 2018 *2018 9th International Symposium on Parallel Architectures, Algorithms and Programming (PAAP)* 171-176