2021 60th IEEE Conference on Decision and Control (CDC)
December 13-15, 2021. Austin, Texas

Real-time Wind Direction Estimation using Machine Learning on Operational Wind Farm Data*†

Farzad Karami, Yujie Zhang, Mario A. Rotea♯, Federico Bernardoni and Stefano Leonardi

2021 60th IEEE Conference on Decision and Control (CDC) | 978-1-6654-3659-5/21/$31.00 ©2021 IEEE | DOI: 10.1109/CDC45484.2021.9683613

Abstract— **This paper presents regression and classification methods to estimate wind direction in a wind farm from operational data. Two neural network models are trained using supervised learning. The data are generated using high-fidelity large eddy simulations (LES) of a virtual wind farm with 16 turbines, which are representative of the data available in actual SCADA systems. The simulations include the high-fidelity flow physics and turbine dynamics. The LES data used for training and testing the neural network models are the rotor angular speeds of each turbine. Our neural network models use sixteen angular speeds as inputs to produce an estimate of the wind direction at each point in time. Training and testing of the neural network models are done for seven discrete wind directions, which span the most interesting cases due to symmetry of the wind farm layout. The results of this paper are indicative of the potential that existing neural network models have to obtain estimates of wind direction in real time.**

I. Introduction

Knowledge of wind direction within wind plants is critical for their operation. Real-time estimates of wind direction are required to properly orient a single wind turbine for power maximization (greedy control). Recently proposed wake steering control methods, which seek to jointly optimize the yaw angles (orientation of rotor planes) of two or more turbines also require wind direction information [1]–[3] in real time. Current technology uses meteorological towers and/or local wind turbine sensors to determine wind direction. Meteorological towers, when available, provide point measurements that may not represent the local wind direction affecting the power production of an entire wind farm. Local wind turbine sensors may suffer from yaw calibration errors, reliability and local wind flow effects reducing accuracy [4]. Thus, it is desirable to find alternative methods to identify changes in wind direction that have significant impact on the power production of an entire wind plant.

A consensus-based approach to predict local wind direction that uses the wind measurements from nearby turbines has been proposed by Annoni et al. [5] and applied to control large wind farms [6] and to forecast the short term power production [7]. The use of this consensus approach can mitigate the errors from the wind turbine vanes that measure wind directions and it can run in real time. Bernardoni et al. [8], [9] have developed a method to identify clusters of turbines in wake interaction and optimized their power production with a coordinated approach. Other researchers have applied machine learning methods to estimate wind directions [10]–[12].

Our research group is also developing machine learning tools for the estimation of wind directions in a wind farm. In this paper we present preliminary results using the simplest possible wind direction estimation problem. That is, to estimate, in real time, the direction of the wind entering the wind farm using readily available SCADA data only.

Machine learning concepts have been around for decades. The availability of significant amounts of data, inexpensive computing power, and tools to develop and train machine learning models are making their application widespread. From a theoretical point of view, it has been shown that machine learning can be used to estimate non-linear functions [13], [14]. In essence, our problem is to find a model that represents the mapping between appropriate and available data from wind turbines and the direction of the wind in real time. Two major machine learning models are classification (e.g., when a label, amongst a discrete set of choices, needs to be estimated from data) and regression (e.g., when a continuous variable needs to be estimated from data). These two approaches require supervised learning to train the machine learning models [15]. Artificial neural networks [16], [17] provide a useful representation to implement regression and classification models for non-linear functions. In addition, there is a plethora of tools available to develop and train neural networks [18], [19].

The objective of this work is to find a data-driven method for wind direction estimation using data typically available in SCADA systems. *Our hypothesis is that the direction of the wind flow traveling across a wind farm can be determined using the spatial distribution of the rotor angular velocity of the turbines.* The intuition behind this hypothesis is that when the wind speed is below the rated values for the turbines, the spatial distribution of the rotor angular velocity may contain enough information to infer the direction of the wind at one or more locations in the wind farm. In this exploratory study, we focus on estimating the direction of the flow at the inlet of a wind farm (incoming wind direction).

The incoming wind direction changes during the day due to variations in atmospheric conditions. Thus, wind direction is a continuous variable. As a result, it seems natural to infer its value (i.e., heading angle) using a regression neural network with the rotor angular velocities as the input data. In practice, it may suffice to know the incoming wind direction at discrete values within a specified range. In this case, it

* This work has been supported by the National Science Foundation, award no. 1839733.

† UTD Center for Wind Energy (https://wind.utdallas.edu/) and Department of Mechanical Engineering, University of Texas at Dallas, Richardson, TX 75080.

♯Corresponding author; Email: rotea@utdallas.edu

is also possible to use a classification neural network to estimate the wind direction among a discrete and finite set of labels (predefined wind directions).

This paper presents regression and classification methods to estimate wind direction in a wind farm from operational data. Two neural network models are trained using supervised learning. The data are generated using high-fidelity large eddy simulations (LES) of a wind farm with 16 turbines. These simulations include the high-fidelity flow physics and turbine dynamics, as such they are representative of the data available in actual SCADA systems [8]. The LES data used for training and testing the neural network models are the rotor angular speeds of each turbine. Our neural network models use sixteen angular speeds as the input vector to produce an estimate of the wind direction at each point in time. Training and testing of the neural network models are done for seven wind directions, which span the most interesting cases due to symmetry of the farm layout.

This paper provides evidence that both regression and classification neural networks can provide real-time estimates of the incoming wind direction using the rotor speeds on the turbines as the input data, which is readily available from a SCADA system.

The paper is organized as follows. Section II introduces the wind farm layout, visualizations of the flow for various wind directions, and the LES data used for training and testing the neural networks. Sections III and IV present the regression and classifier models respectively. The resulting estimates of incoming wind direction using the test data are presented in Section V, and conclusions are given in Section VI.

II. Case Study

The wind turbine data to train and test the neural networks, is obtained from Large Eddy Simulations (LES). The data used is from [8]. The simulated wind farm consists of 16 NREL-5MW reference turbines [20], arranged in a 4 × 4 layout. The turbines are all identical with rotor diameter $D = 126$ m, rated wind speed $U_{rated} = 11.4$ m/s, rated rotor speed $\Omega_{rated} = 12.1$ RPM, and rated power $P_{rated} = 5$ MW.

The wind farm layout is depicted in Figure 1a. The turbine spacing in the transversal direction (West-East) is $3D$, while in the longitudinal direction (South-North) the spacing is $5D$. The color-coded plot in Figure 1b depicts the time-averaged hub-height wind speed as a function of spatial coordinates. This flow visualization corresponds to an assumed prevailing wind direction (South-North), which we take as the reference angle for the wind direction $\theta = 0°$. The wind turbines are controlled in a classical fashion, where the rotor-plane normal is aligned with the local wind direction and the rotor angular speed is proportional to the local wind speed in order to track the optimal tip-speed ratio for power maximization at the turbine level [21]. The spatial and temporal average of the hub-height wind speed at the inlet of the wind farm is set at $U = 0.8U_{rated}$, with 11% turbulence intensity, which implies (in theory) that all turbines operate below-rated wind speeds, where wind farm power maximization is a key goal and there is variability in the power and rotor angular speed of each turbine to potentially infer wind direction. Testing this loosely-stated hypothesis is the main objective of this conference paper.

Due to the symmetry of the wind farm layout, wind directions between $\theta = 0°$ and $\theta = 90°$ suffice for this study. For several wind directions in this range, LES simulations were conducted for 1600 seconds. The LES simulation data include the generated power and rotor angular speed of each turbine. A full account of the LES simulations and their theoretical background can be found in [8], [9].

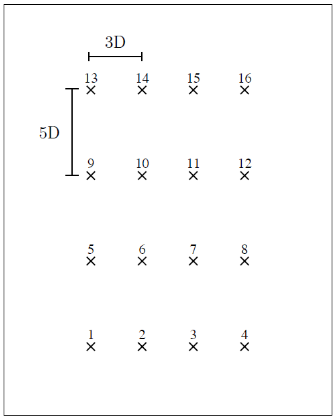
In this paper, the neural networks are trained and tested using angular rotor speeds operating below rated wind speeds. The motivation for this choice is the proportionality between wind speed and rotor angular speed, coupled with the fact that reductions in wind speed (i.e,, rotor speed) are indicative of downstream turbines operating in the wake of upstream turbines, which may help establish the wake direction and hence the wind direction (when rotor-plane normal directions align with the local wind). In Figure 2 and Figure 3, the time series of the rotor angular speed for the first and third rows of the wind farm are shown when the wind direction is $\theta = 0°$ (Figure 1b). As expected, higher rotor speeds occur in the upstream turbines (Figure 2), which produce more power than the third row (Figure 3) due to wake interaction that reduces the wind speed in front of the trailing turbine rotors.

In order to provide more information on the role of wakes (reduced velocity regions) for other wind directions, the color contours of the time-averaged wind velocity at the hub height are shown in Figure 4 through Figure 6 for prevailing wind directions $\theta = 30°$, $\theta = 60°$ and $\theta = 90°$, respectively. The instantaneous values of the rotor angular speeds for a discrete number of prevailing wind directions shall be used to train and test the neural network models explained subsequently.

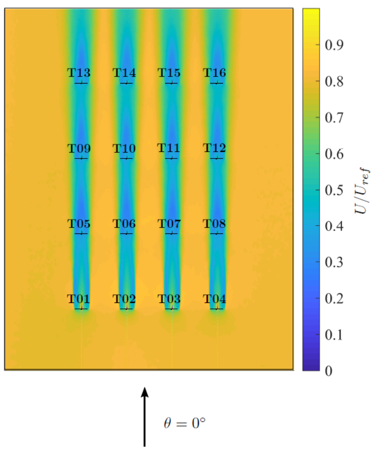
III. Regression model

We built a neural network for the regression problem of estimating wind direction using the spatial relationships between the rotor angular speeds of the wind turbines at each point in time. That is, the input of the regression model is a vector with 16 scalar components containing the angular rotor speed of each turbine at each instant. The output of the network is a wind direction estimate at each point in time. The model consists of an input layer, three fully connected hidden layers, and an output layer. The hidden layer 1, 2, and 3 have 64, 32, and 16 neurons respectively. The actuation function for hidden layers is $tanh(\cdot)$. The output layer with a single neuron have a linear activation function. The architecture of the developed network is shown in Figure 7.

The dataset is split randomly between a training dataset consisting of 41,804 samples, corresponding to 80% of the entire dataset, and 17,917 test samples representing 20% of the dataset. This split helps us mitigate overfitting by assesing the performance of the model over an unseen dataset. We considered 7 wind directions, from $\theta = 0°$ to $\theta = 90°$ with 15° increments, with equal numbers of samples. Tensorflow is used for implementing the model



(a)



(b)

Fig. 1: a) Wind farm layout with sixteen 5MW wind turbines. b) Color contour of the time-averaged wind velocity at the hub-height; $\theta = 0°$ prevailing wind direction.

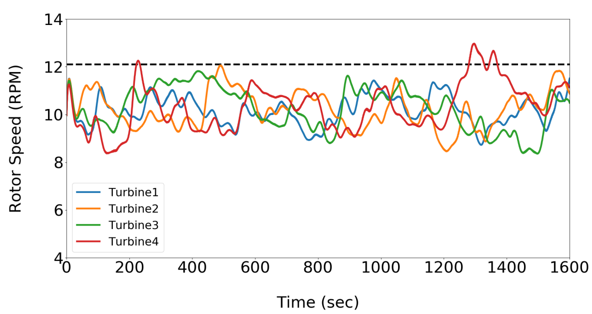


Fig. 2: Time series of rotor angular speed for turbines 1, 2, 3 and 4 with wind direction $\theta = 0°$. The horizontal dash line indicates the rated rotor speed.

in Python. For training the model, Adam optimizer with a learning rate of 0.001 is used [22]. Training is done by minimizing the mean squared error (MSE) loss function. That is, the mean of squared difference between the predicted and true values on the training samples given by [15]

$$MSE = \frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y}_i)^2 \quad (1)$$

where N is the number of training samples, y_i is the true

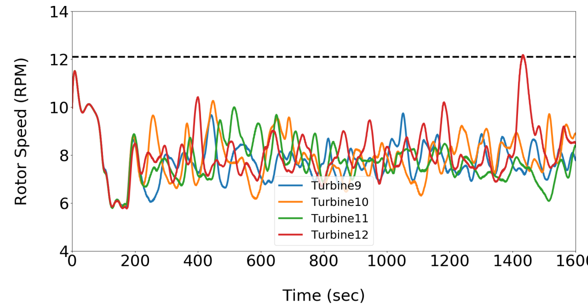


Fig. 3: Time series of rotor angular speed for turbines 9, 10, 11 and 12 with wind direction $\theta = 0°$. The horizontal dash line indicates the rated rotor speed.

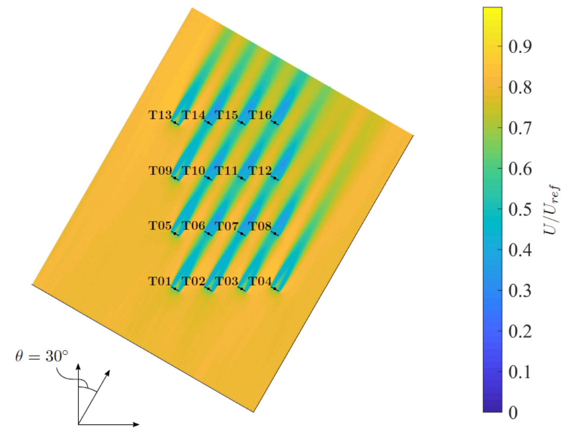


Fig. 4: Color contour of the time-averaged wind velocity at the hub-height; $\theta = 30°$ prevailing wind direction.

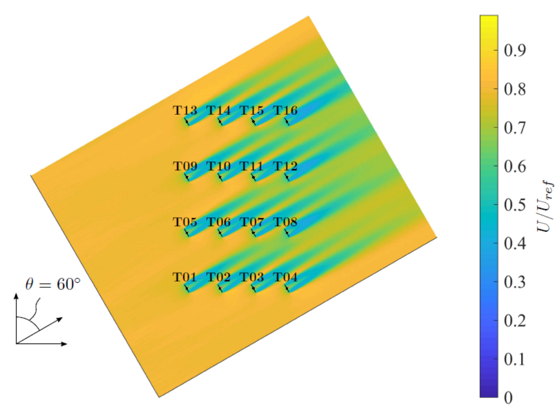


Fig. 5: Color contour of the time-averaged wind velocity at the hub-height; $\theta = 60°$ prevailing wind direction.

output of the i^{th} sample and $\hat{y}_i$ is the estimated value for the same sample. The model is trained in 100 epochs with a batch size of 64.

IV. Classification Model

The problem of estimating the wind direction from the instantaneous rotor angular velocities can also be posed as a classification problem, where the labels are the discrete wind directions we wish to estimate [23]. This approach requires that we select the labels apriori. For consistency

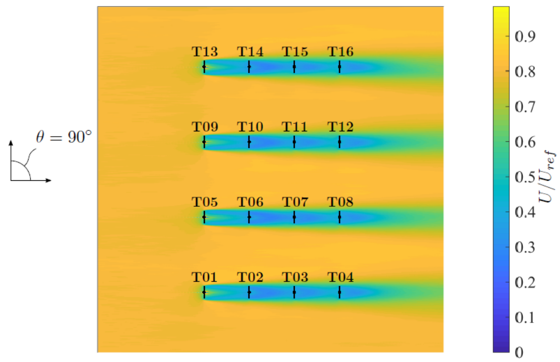


Fig. 6: Color contour of the time-averaged wind velocity at the hub-height; $\theta = 90°$ prevailing wind direction.

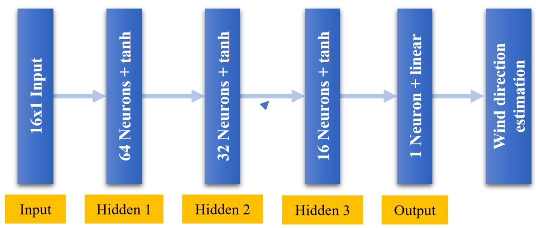


Fig. 7: Regression model neural network architecture.

with the regression model, the labels are the seven wind directions, from $\theta = 0°$ to $\theta = 90°$, defined in the previous section. We selected the LSTM model as our classifier. The LSTM model is an advanced neural network architecture. For consistency with the training of the regression neural network, the training of the classifier is done by splitting randomly between a training dataset consisting of 80% of the entire data and the remaining 20% assigned to the test dataset. The classifier architecture is shown in Figure 8. There is one single hidden layer containing 512 neurons and one output layer estimating the wind direction. The output label is the estimated wind direction, whose value can only be one of the wind direction labels used for training (seven directions in this case).

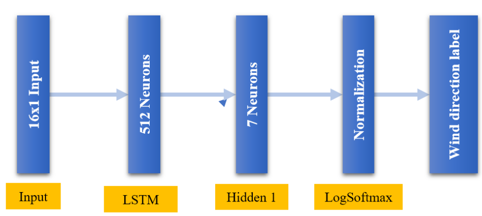


Fig. 8: Classification model neural network architecture.

Pytorch library is used for implementing the model. The Stochastic Gradient Decent (SGD) optimizer [24] is used with a learning rate of 0.001 for minimizing the loss function, which is defined as negative log-likelihood loss. The following expression shows the definition of this loss function [15]

$$C_{\text{LL}} = -\frac{1}{N}\sum_{i=1}^{N}\sum_{k=1}^{K} p_k \ln(\hat{p}_k) \tag{2}$$

where p_k is the true probability of sample i belonging to class k, $\hat{p}_k$ is the probability predicted by the model, N is the number of training samples, and K is the number of classes.

V. Results

In this section, the results from the developed models are presented and discussed.

A. Regression results

To evaluate the performance of the regression model, we calculate the mean absolute error (MAE) on the test dataset. The following expression shows its mathematical expression [15].

$$MAE\,(S_t) = \frac{1}{N_t}\sum_{i=1}^{N_t} |y_i - \hat{y}_i| \tag{3}$$

where N_t is the number of test samples, and S_t is the test dataset, y_i is the true output of the i^{th} sample and $\hat{y}_i$ is the model output for the same sample.

A summary of the results, including the MAE, and the mean and standard deviation of the estimation error, is shown in Table I. The table shows these error metrics for each wind direction angle. Note that the maximum MAE is $0.26°$ (for $\theta = 90°$), while the minimum MAE is $0.08°$ (for $\theta = 15°$). The achieved average MAE on the test dataset (all seven wind directions) is $0.1709°$.

Wind Direction	MAE (deg)	Error Mean (deg)	Error STD (deg)
0°	0.1641	-0.0774	1.6385
15°	0.0752	-0.0241	0.1193
30°	0.1573	-0.1136	0.7543
45°	0.0901	0.0319	0.1627
60°	0.2582	0.1011	1.0020
75°	0.1869	0.0084	0.2657
90°	0.2643	0.2330	2.4420

TABLE I: Accuracy and error statistics of regression model for each wind direction.

To gain further insight into the performance of the regression neural network, we study the distribution of the wind direction estimates using the test data for the best and worst cases according to the MAE values reported in Table I. These empirical distributions are shown in Figure 9 and Figure 10 for the best and worst case, respectively. For each wind direction, the regression model gives estimations well-centered around the ground truth. The distributions of the estimation error for these cases are shown in Figure 11 for the best case ($\theta = 15°$) and in Figure 12 for the worst case MAE ($\theta = 90°$). These plots exhibit a tight and symmetric distribution around zero indicating a good performance of the regression model. From these preliminary results, we expect that using instantaneous values of rotor angular speed, processing these data to obtain an instantaneous estimate of wind direction and taking an appropriate time average (as commonly reported in wind plant SCADA systems) would provide a more accurate indicator of the incoming wind direction to a wind farm.

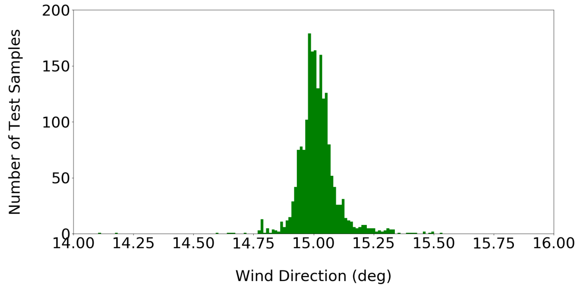


Fig. 9: Distribution of wind direction estimation with the regression neural network for test data corresponding to $\theta = 15°$

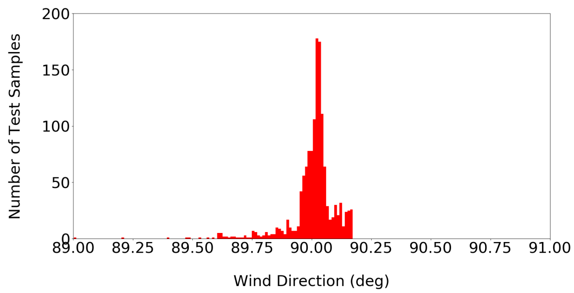


Fig. 10: Distribution of wind direction estimation with the regression neural network for test data corresponding to $\theta = 90°$

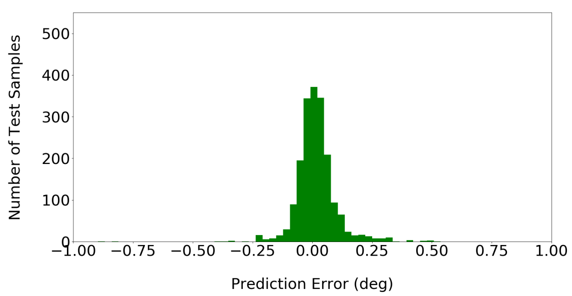


Fig. 11: Distribution of wind direction estimation error with the regression neural network for test data corresponding to $\theta = 15°$

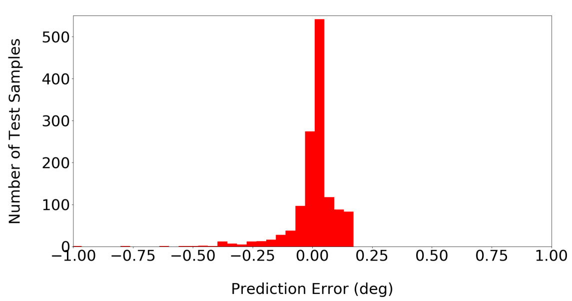


Fig. 12: Distribution of wind direction estimation error with the regression neural network for test data corresponding to $\theta = 90°$

B. Classification results

In this section, we present test accuracy results of the classification model. We use accuracy score as our metric for this model. The accuracy score is defined as follows: [15]

$$Accuracy = \sum_{i=1}^{N_t} \frac{e_i}{N_t} \qquad \begin{cases} e_i = 1 \text{ if } \hat{y}_i = y_i \\ e_i = 0 \text{ otherwise} \end{cases} \tag{4}$$

where N_t represents the number of samples in the test dataset, $\hat{y}_i$ is the predicted label (wind direction angle), and y_i is the true label. The numerator in this definition denotes the number of instances where $\hat{y}_i = y_i$. Note that $Accuracy = 1$ means a perfect classification. The accuracy of estimation results with the classification model are shown in Table II. The distribution of estimated wind directions for the best and worst accuracy values from Table II are shown in Figure 13 and Figure 14, respectively.

The test results suggest that classification neural networks can be used to estimate discrete wind directions given instantaneous rotor angular speed values of the wind turbines. Note that the output of the classifier is a discrete wind direction that has already been seen during the training phase.

Wind Direction	Accuracy
0°	0.9527
15°	0.9779
30°	0.9162
45°	0.8382
60°	0.7423
75°	0.9020
90°	0.9756

TABLE II: Accuracy of classification model for each wind direction

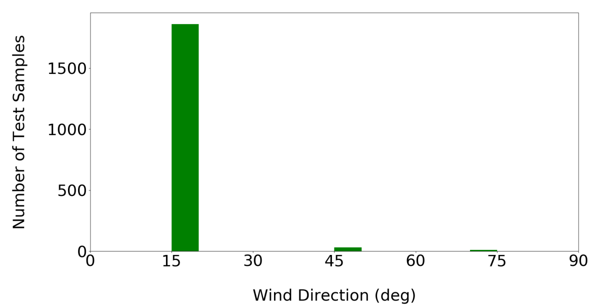


Fig. 13: Distribution of estimation results in the classification model with 15° wind direction

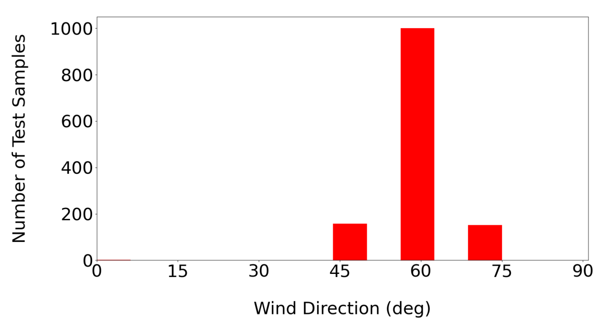


Fig. 14: Distribution of estimation results in the classification model with 60° wind direction

C. Inference Speed

Tables III and IV show the response time for both regression and classification models. All cases are run on a laptop with Intel Core i7-3840QM @ 2.80GHz and 32 GB RAM. Due to the simplicity of our model networks, the calculations are very fast, which enable the real-time application. The regression model runs around 10x faster than the classification model because of the network structure differences between two models. Our classification model implements the LSTM architecture and it has heavier operations than the multi-layer neuron network used in the regression model.

Wind dir. (deg)	Samples	Total time (sec)	msec/sample
0	1504	0.15	9.90e-2
15	1904	0.10	5.36e-2
30	1338	0.06	4.86e-2
45	1948	0.07	3.70e-2
60	1313	0.05	3.50e-2
75	2575	0.08	3.28e-2
90	1359	0.06	4.27e-2
Total	11941	0.57	4.77e-2

TABLE III: Response time for the regression model

Wind dir. (deg)	Samples	Total time (sec)	msec/sample
0	1504	0.76	0.50
15	1904	0.98	0.52
30	1338	0.70	0.52
45	1948	1.03	0.53
60	1313	0.84	0.64
75	2575	1.39	0.54
90	1359	0.74	0.55
Total	11941	6.44	0.54

TABLE IV: Response time for the classification model

VI. Conclusion

This exploratory study has evaluated the possibility of estimating incoming wind direction into a wind farm from the time samples of rotor speed of the turbines. Two neural network models (regression and classification) have been trained and tested using data from high-fidelity large eddy simulations. These simulation data is representative of the data available in actual SCADA systems. The numerical results provide evidence on the ability of these neural network models to estimate wind direction from the information contained in the spatial distribution of the rotor speeds across the wind farm (the time samples of a 16×1 vector for our case study). The neural network models can run very fast; thus, enabling real-time estimation and control.

Future work will consider developing a statistical framework to gain further insight into the relation between rotor speed distributions and wind directions within a wind farm. We shall also use data from actual wind farms and plan on customizing these results to neural networks that can estimate local incoming wind speeds entering clusters of turbines in a wind farm. A problem of importance to power optimization via wake steering (a.k.a. yaw control).

References

[1] P. Fleming, J. Annoni, J. Shah, L. Wang, S. Ananthan, Z. Zhang, K. Hutchings, P. Wang, W. Chen, and L. Chen, "Field test of wake steering at an offshore wind farm," *Wind Energy Science*, vol. 2, no. 1, pp. 229–239, 2017.

[2] J. Bartl, F. Mühle, and L. Sætran, "Wind tunnel study on power output and yaw moments for two yaw-controlled model wind turbines," *Wind Energy Science*, vol. 3, no. 2, pp. 489–502, 2018.

[3] M. F. Howland, S. K. Lele, and J. O. Dabiri, "Wind farm power optimization through wake steering," *Proceedings of the National Academy of Sciences*, vol. 116, no. 29, pp. 14 495–14 500, 2019.

[4] G. Cortina, V. Sharma, and M. Calaf, "Investigation of the incoming wind vector for improved wind turbine yaw-adjustment under different atmospheric and wind farm conditions," *Renewable Energy*, vol. 101, pp. 376–386, 2017.

[5] J. Annoni, C. Bay, K. Johnson, E. Dall'Anese, E. Quon, T. Kemper, and P. Fleming, "Wind direction estimation using scada data with consensus-based optimization," *Wind Energy Science*, vol. 4, no. 2, pp. 355–368, 2019.

[6] J. Annoni, C. Bay, T. Taylor, L. Pao, P. Fleming, and K. Johnson, "Efficient optimization of large wind farms for real-time control," in *2018 Annual American Control Conference (ACC)*. IEEE, 2018, pp. 6200–6205.

[7] J. Annoni, C. Bay, K. Johnson, and P. Fleming, "Short-term forecasting across a network for the autonomous wind farm," in *2019 American Control Conference (ACC)*. IEEE, 2019, pp. 2837–2842.

[8] F. Bernardoni, U. Ciri, M. Rotea, and S. Leonardi, "Real-time identification of clusters of turbines," in *Journal of Physics: Conference Series*, vol. 1618, no. 2. IOP Publishing, 2020, p. 022032.

[9] F. Bernardoni, M. Ciri, Rotea, and S. Leonardi, "Identification of wind turbine clusters for effective real time yaw control optimization," *Journal of Renewable and Sustainable Energy*, vol. 13, no. 4, 2021.

[10] A. Khosravi, R. Koury, L. Machado, and J. Pabon, "Prediction of wind speed and wind direction using artificial neural network, support vector regression and adaptive neuro-fuzzy inference system," *Sustainable Energy Technologies and Assessments*, vol. 25, pp. 146–160, 2018.

[11] Z. Dzulfikri, N. Nuryanti, and Y. Erdani, "Design and implementation of artificial neural networks to predict wind directions on controlling yaw of wind turbine prototype," *Journal of Robotics and Control (JRC)*, vol. 1, no. 1, pp. 20–26, 2019.

[12] F. Tagliaferri, I. Viola, and R. Flay, "Wind direction forecasting with artificial neural networks and support vector machines," *Ocean Engineering*, vol. 97, pp. 65–73, 2015.

[13] J. G. Carbonell, R. S. Michalski, and T. M. Mitchell, "An overview of machine learning," *Machine learning*, pp. 3–23, 1983.

[14] R. E. Schapire, "The boosting approach to machine learning: An overview," *Nonlinear estimation and classification*, pp. 149–171, 2003.

[15] J. Friedman, T. Hastie, R. Tibshirani *et al.*, *The elements of statistical learning*. Springer series in statistics New York, 2001, vol. 1, no. 10.

[16] M. H. Hassoun *et al.*, *Fundamentals of artificial neural networks*. MIT press, 1995.

[17] B. Yegnanarayana, *Artificial neural networks*. PHI Learning Pvt. Ltd., 2009.

[18] F. Chollet *et al.*, *Deep learning with Python*. Manning New York, 2018, vol. 361.

[19] A. P. Marugán, F. P. G. Márquez, J. M. P. Perez, and D. Ruiz-Hernández, "A survey of artificial neural network in wind energy systems," *Applied energy*, vol. 228, pp. 1822–1836, 2018.

[20] J. Jonkman, S. Butterfield, W. Musial, and G. Scott, "Definition of a 5-mw reference wind turbine for offshore system development," National Renewable Energy Lab.(NREL), Golden, CO (United States), Tech. Rep., 2009.

[21] L. Y. Pao and K. E. Johnson, "Control of wind turbines," *IEEE Control systems magazine*, vol. 31, no. 2, pp. 44–62, 2011.

[22] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[23] S. B. Kotsiantis, I. Zaharakis, and P. Pintelas, "Supervised machine learning: A review of classification techniques," *Emerging artificial intelligence applications in computer engineering*, vol. 160, no. 1, pp. 3–24, 2007.

[24] L. Bottou, "Large-scale machine learning with stochastic gradient descent," in *Proceedings of COMPSTAT'2010*. Springer, 2010, pp. 177–186.