

Research Article

Fault Detection of Wind Turbine Sensors Using Artificial Neural Networks

Ayşe Gökçen Kavaz  and Burak Barutcu

Energy Institute, Istanbul Technical University, 34469 Istanbul, Turkey

Correspondence should be addressed to Ayşe Gökçen Kavaz; agkavaz@itu.edu.tr

Received 15 July 2018; Accepted 9 October 2018; Published 19 December 2018

Academic Editor: Jesus Corres

Copyright © 2018 Ayşe Gökçen Kavaz and Burak Barutcu. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

This paper proposes a method for sensor validation and fault detection in wind turbines. Ensuring validity of sensor measurements is a significant part in overall condition monitoring as sensor faults lead to incorrect results in monitoring a system's state of health. Although identifying abrupt failures in sensors is relatively straightforward, calibration drifts are more difficult to detect. Therefore, a detection and isolation technique for sensor calibration drifts on the purpose of measurement validation was developed. Temperature sensor measurements from the Supervisory Control and Data Acquisition system of a wind turbine were used for this aim. Low output rate of the measurements and nonlinear characteristics of the system drive the necessity to design an advanced fault detection algorithm. Artificial neural networks were chosen for this purpose considering their high performance in nonlinear environments. The results demonstrate that the proposed method can effectively detect existence of calibration drift and isolate the exact sensor with faulty behaviour.

1. Introduction

In the last decades, both the size and capacity of wind turbines have increased by virtue of technological developments in wind energy field. This situation resulted in an increasing focus on topics such as wind turbine fault detection [1–6]. Condition monitoring and fault detection algorithms of wind turbines are important systems that contribute to reducing maintenance costs and downtime of wind power plants. Furthermore, wind power plants are generally located in distant sites which makes reliability even more important. Wind turbine faults lead to the need for repair and/or replacement actions and result in loss of energy production. Moreover, in some cases, failure in a component also affects other components and even the entire wind turbine. Considering all these facts, it is significant to detect and isolate wind turbine faults as early as possible to take required actions for preventing such undesired results. Besides, with the decrease in maintenance costs, wind energy will become commercially more competitive compared to other energy sources.

Fault detection methods can be classified into two different categories, namely, model-based and data-driven

approaches. In model-based methods, firstly a mathematical model expressing wind turbine dynamics is generated. The output of that model is compared to real measurements from wind turbine sensors, and a fault alarm is given by analyzing residuals between outputs of the real system and model. Some examples of model-based fault detection studies in wind turbines can be found in [7, 8]. Model-based methods have the advantage of not requiring high sampling rate measurements [3]. However, the success of these methods depends on the consistency of the mathematical model and the real behaviour of system, and this approach meets challenges because in addition to systematic complexities, wind turbines have complex control systems which increase the difficulty of obtaining a reasonable mathematical model of the whole system [9].

Data-driven methods do not require explicit mathematical models identifying the physical system. Instead, they are based on the analysis of measurements gathered from sensors mounted on different parts of wind power plants. Different alternatives exist in terms of choosing sensor data to be processed. The first alternative is to use sensor outputs from the Supervisory Control and Data Acquisition (SCADA) system

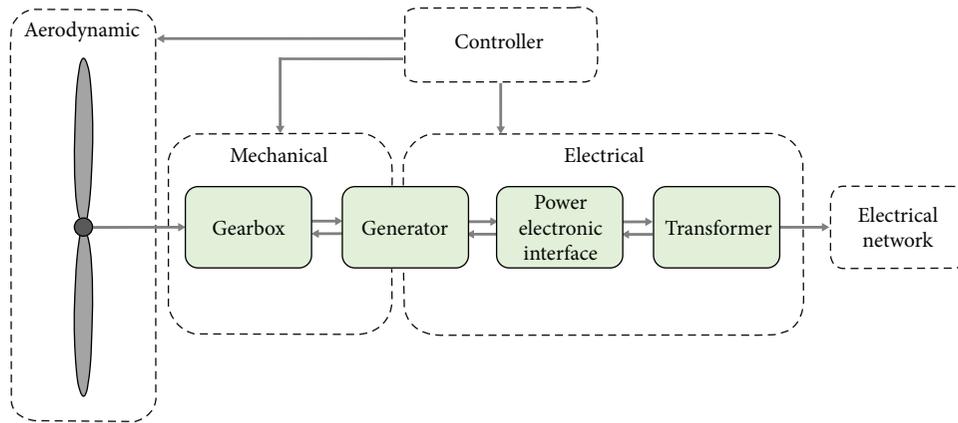


FIGURE 1: Subsystems of the wind turbine.

which is a built-in part of most modern wind turbines with no additional costs. Many attempts have been made to benefit from SCADA data for fault detection aims [9–15]. The other alternative is to use purpose-designed measurement systems having sensors specifically mounted for condition monitoring purposes. Common sensor types employed for this aim are vibration, acoustic emission, strain, torque, and bending moment sensors. Some past researches on wind turbine fault detection using specific sensors can be found in [16–24]. Both two data gathering methods have certain advantages and disadvantages. Sensor data from the SCADA system are of low frequency and generally have 10 min sampling intervals. Such a low frequency brings difficulties due to the loss of noise characteristics which carry important information on fault formation. Besides, generally there are many missing values and imperfections in data collected by the SCADA system. But as an advantage, the SCADA system is a built-in part of most of the large-scale wind turbines; therefore, no additional cost is required to reach these data. On the other hand, data output frequency for purpose-built wind turbine condition monitoring sensors can be selected high enough not to lose information in signal. However, this approach causes extra cost for maintenance and, as a result, becomes a disadvantage in terms of cost competitiveness against other energy conversion methods.

In this paper, data gathered from the temperature sensors of a wind turbine SCADA system were used to detect and isolate sensor faults. To build an effective fault detection system, it is crucial to distinguish between sensor errors from faults originated from other components of the turbine, because faulty behaviour in sensors can lead to errors in data evaluation and cause false alarms and performance degradation of fault detection system.

To compensate for the disadvantages of low-frequency sensor measurements from the SCADA system, a sophisticated signal processing method is required. Artificial neural networks (ANN) were selected for this purpose considering their high performance in modelling nonlinear systems. Different ANN architectures were built and compared to obtain the best option among different alternatives.

The layout of this paper is as follows. In Section 2, sensor selection and data collection parts of the research are

presented. In Section 3, ANN architectures designed in this research are explained, and in Sections 4 and 5, results and discussions are presented, respectively.

2. Data Collection

Wind turbines are comprised of different interconnected subsystems. The primary subsystems can be listed as aerodynamic, mechanical, electronic, and control systems (Figure 1). A sensor fault in any of these components results in a degradation in a turbine's performance. Detecting and isolating sensor faults are an important part of the overall fault detection system as distinguishing the cause of abnormality is required for repair or replacement actions.

Various types of sensor faults can be encountered in subsystems of wind turbines. Typical sensor faults can be listed as multiplicative, additive, offset faults, and faults resulting in changing dynamics in the system [25]. In this paper, multiplicative fault in temperature sensor measurements is investigated. Multiplicative faults may arise from calibration drifts and act like a scaling factor on sensor measurements. This type of sensor fault was selected to be investigated because unless scaling factor of a multiplicative fault is too big, it does not cause an easily recognizable change comparing to normal response of sensors which makes them harder to detect. SCADA measurements used in this work supply some statuses of main faults such as mains failures, feeding faults, and pitch control errors; however, the SCADA system does not provide information on sensor faults. But ensuring measurement validation is a compulsory step in the design of an overall fault detection system. Therefore, the multiplicative sensor fault to be detected was artificially created in this research. Measurement differences in a form of a multiplicative effect can be originated either from sensor calibration drifts or real temperature deviations, so it is required to distinguish these two situations. For this reason, the method designed was tested in different cases representing both these situations.

Temperature measurements from the SCADA system of a 900 kW wind turbine were collected to train ANNs designed in this work. The data have 10 min sampling intervals and were gathered between 01 November 2015 and 30 November 2015. The SCADA system includes 10

TABLE 1: Temperature sensors from the SCADA system.

Description
Generator stator temperature
Generator rotor temperature
Nacelle ambient temperature
Nacelle temperature
Front hub bearing temperature
Rear hub bearing temperature*
Nacelle control cabinet temperature
Control cabinet temperature*
Tower temperature*
Transformer temperature*

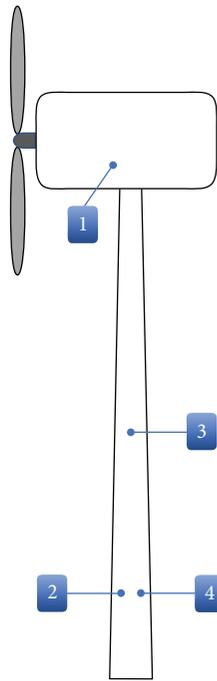


FIGURE 2: Locations of the sensors used in this research.

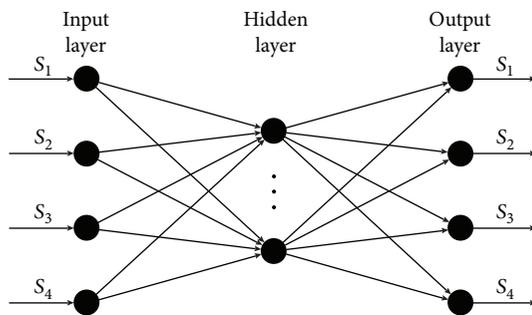


FIGURE 3: Autoassociative input-output structure.

temperature sensors mounted on different components of the turbine which are presented in Table 1.

Sensors with the * sign in Table 1 were selected for the sensor validation purpose in this work. They measure rear

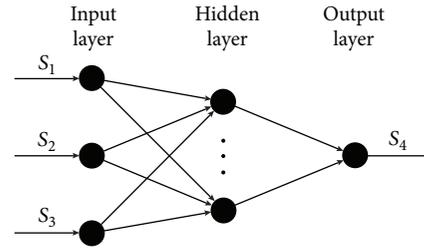


FIGURE 4: MISO input-output structure.

hub bearing temperature (S_1), control cabinet temperature (S_2), tower temperature (S_3), and transformer temperature (S_4). The algorithm proposed detects calibration drifts in one of these sensors based solely on the measurements of them without using any operational or environmental data. Therefore, this subset of sensors was selected considering the similar temperature characteristics of the areas they were installed in. Each of them is monitored by operators to find out if any abnormalities appear in the components around them. Locations of the sensors used in this research are presented in Figure 2.

Using measurements from these sensors, firstly ANNs were trained. For training purpose, 75% of the data was used. The remaining 25% of the data was used to form test data set to evaluate and compare the performance of the networks. ANN models were designed and trained in MATLAB environment.

3. Artificial Neural Network Architectures

ANNs have proven to be powerful algorithms for various aims including classification, estimation, and fault detection. Due to their capability of dealing successfully with nonlinearities and their advantages in real-time applications, it is convenient to design ANN models for detecting wind turbine faults. Several ANN models were trained and analyzed to obtain a successful network architecture that satisfies the requirements of this research.

In terms of input-output relations, 2 approaches have been used, namely, autoassociative and MISO (multiple-input single-output) structures. The autoassociative case, where the input vector is to be associated with itself is presented in Figure 3. As can be seen in the figure, all the sensors were set both as inputs and outputs of the autoassociative network. In the MISO case, 3 out of 4 selected sensors were set as inputs and the remaining sensor is set to be the output. The structure is repeated 4 times with each one of the sensors used as output separately. Figure 4 shows the MISO structure with S_4 as the output of the system. Similar networks with other sensors in the output layer were also created.

The input-output relations presented in Figures 3 and 4 were used in different neural network types, namely, back-propagation neural network (BPNN), radial basis function neural network (RBFNN), and general regression neural network (GRNN). Table 2 presents the computational principles and input-output relations of the networks designed.

TABLE 2: Neural network architectures.

Network	Computational principle	Input-output relation
Network 1	BPNN	Autoassociative
Network 2	BPNN	MISO
Network 3	RBFNN	Autoassociative
Network 4	RBFNN	MISO
Network 5	GRNN	Autoassociative
Network 6	GRNN	MISO

3.1. Backpropagation Neural Network. BPNN is a type of feedforward neural network where information flows from inputs to outputs without any feedback connections, whereas error propagates from outputs to inputs. Detailed mathematical foundations of BPNNs can be found in [26, 27]. This kind of neural network has been commonly used for fault detection purposes.

A typical BPNN consists of an input layer, one or multiple hidden layers, and an output layer. By taking requirements of problem into consideration, different activation functions can be used. With the help of nonlinear activation functions existing in hidden or output layers, neural networks become capable of learning nonlinear relationships between inputs and outputs. In this research, several BPNN architectures with one hidden layer were designed. Firstly, the type of activation functions for hidden and output layers was determined. For this reason, networks with different activation function pairs were designed and evaluated using a smaller sample data set, such as “logarithmic sigmoid-linear,” “logarithmic sigmoid-logarithmic sigmoid,” “tangent sigmoid-linear,” and “tangent sigmoid-tangent sigmoid.” The former function represents the hidden layer’s activation function, and the latter represents the output layer’s activation function. The best results were obtained by “logarithmic sigmoid-linear” activation function pair. Therefore, the detailed network developments in the following parts of the research were continued with these activation functions.

As presented in Table 2, autoassociative and MISO structures were used in the design of BPNN models. To obtain an autoassociative BPNN with high performance, different network architectures with 2 and 3 hidden neurons were designed and 10 trials were made with each architecture. The reason of multiple trials is that the performance of the network alters depending on the initial values of weights connecting different layers and to ensure the reach of a minimum point in the cost function space.

For MISO BPNNs, the number of hidden neurons was increased from 2 to 15 and again 10 trials with different random initial conditions for each architecture were held. The selection of the best network to use in the next parts of the research was made based on the comparison of R^2 goodness of fit values. Amongst the autoassociative BPNNs, the R^2 values of the best network were 0.999, 0.999, 0.996, and 0.993 for sensors 1 through 4, respectively. For MISO BPNNs, the highest scores were obtained as 0.836, 0.979, 0.984, and 0.985. Figure 5 shows the regression graphs of

the test set for autoassociative and MISO BPNN with the best performances.

3.2. Radial Basis Function Neural Network. RBFNN is a type of feedforward neural network typically having a single layer of hidden units that are connected to linear output units. They are trained in a hybrid manner. Computational parts in the hidden layer use unsupervised learning, and each of them is described by a radial basis function. The size of the units in the hidden layer is the same as the size of the training vector. The output layer is trained in a supervised manner. Mathematical details of RBFNNs can be found in [28].

The output of a neuron of RBFNN is given in the following:

$$y = \sum_{k=1}^N w_k \varphi_k(\|x - c_k\|), \quad (1)$$

where N is the number of neurons in the hidden layer, φ_k is the kernel of radial basis function for each unit, c_k is the center of radial basis function vector for neuron k , and w_k is the weight of neuron k in the output layer. The most common choice for RBFNN kernel is Gaussian activation function which was also used in this research. It is presented in the following:

$$\varphi_k(\|x - c_k\|) = \exp\left(-\frac{\|x - c_k\|^2}{2\sigma_k^2}\right), \quad (2)$$

where $\|x - c_k\|^2$ is the squared Euclidian distance between the associated center vector and the input vector and σ_k is the width factor of the k^{th} hidden unit in the hidden layer which controls the smoothness properties of interpolating function.

Several autoassociative and MISO networks with RBFNN approach were developed in this research. Width factors of the networks were selected amongst varying values between 0.1 and 100. The performances of the networks were evaluated based on the R^2 values. The best R^2 scores amongst the autoassociative RBFNNs were bigger than 0.999 for all sensors. The highest scores for MISO RBFNNs were 0.824, 0.979, 0.982, and 0.984. In the detailed analysis for the next parts of the research, the networks with the best performances were used. Regression plots for the test set of RBFNN networks with the highest performance results are presented in Figure 6.

3.3. General Regression Neural Network. GRNN computes the most probable value of an output y given only training vectors x . In this approach, a specific functional form to describe the relation between inputs and outputs is not required, instead the appropriate form is expressed as a probability density function which is determined from observed data. Since the parameters are directly calculated using examples, an iterative computation is not required [29].

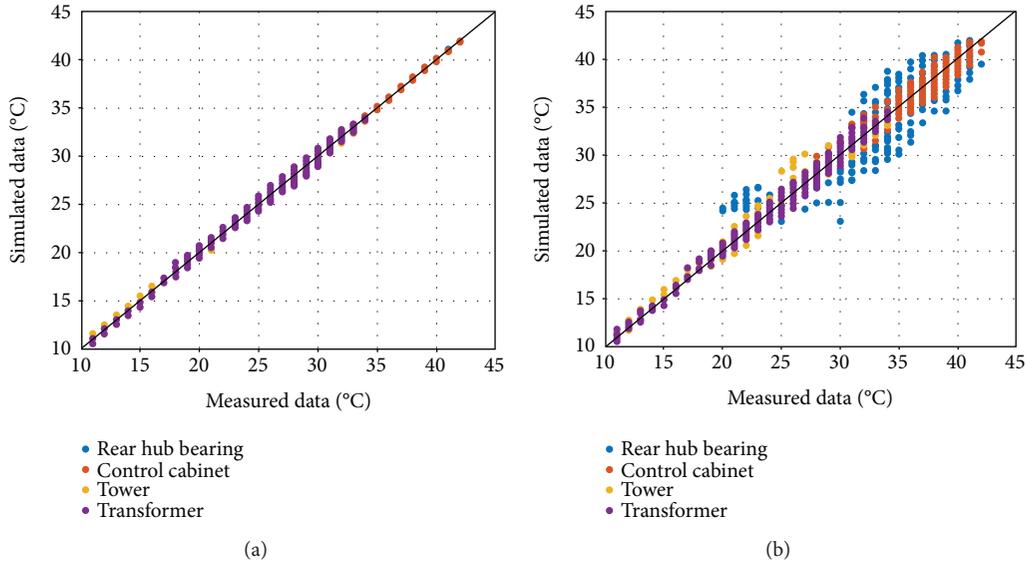


FIGURE 5: Regression plots for (a) autoassociative BPNN and (b) MISO BPNN.

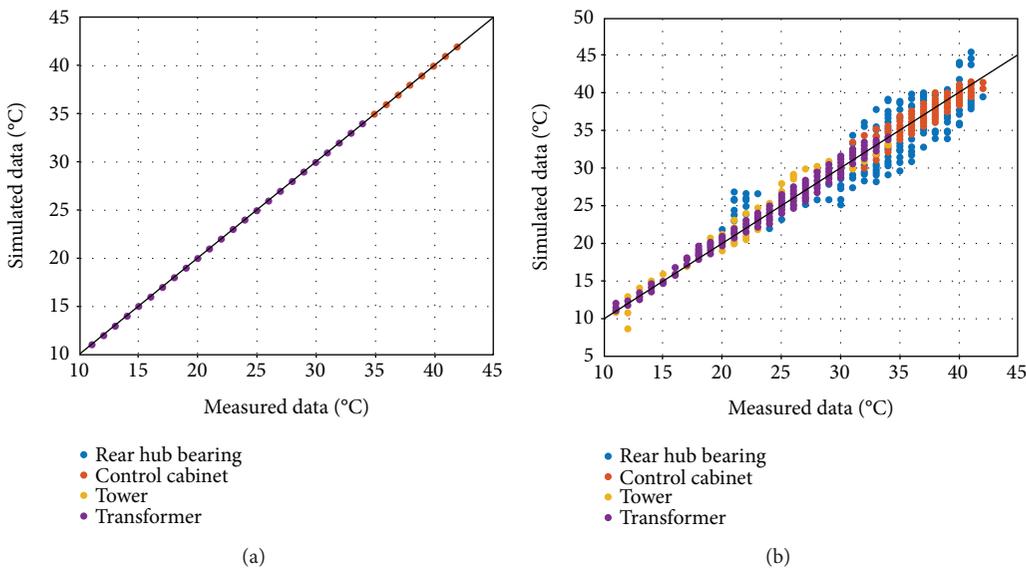


FIGURE 6: Regression plots for (a) autoassociative RBFNN and (b) MISO RBFNN.

The squared distance D_i^2 which is given in (3) is used to calculate the probable values.

$$D_i^2 = (x - x^i)^T (x - x^i). \quad (3)$$

The output is calculated by

$$\hat{y} = \frac{\sum_{i=1}^n y^i \exp(-D_i^2/2\sigma^2)}{\sum_{i=1}^n \exp(-D_i^2/2\sigma^2)}, \quad (4)$$

where \hat{y} is the estimate of the output which is a weighted average of all the observed samples y^i , x is the input vector, n is the number of sample observations, and σ is the width factor. Like the approach used in RBFNNs, GRNN models

were also designed with varying width factors from 0.1 to 100. The highest R^2 values for autoassociative GRNN networks were 0.992, 0.990, 0.990, and 0.994 for sensors 1 through 4, respectively, and the best scores for MISO GRNNs were 0.808, 0.976, 0.973, and 0.981. Figure 7 shows the regression plots for autoassociative and MISO GRNN models with the highest goodness of fit results.

4. Results and Discussion

As stated in Section 2, SCADA data do not supply information on incipient temperature sensor faults. Therefore, a multiplicative fault representing a calibration drift was artificially created in this work. The models do not evaluate single-sensor measurements independently but instead evaluate

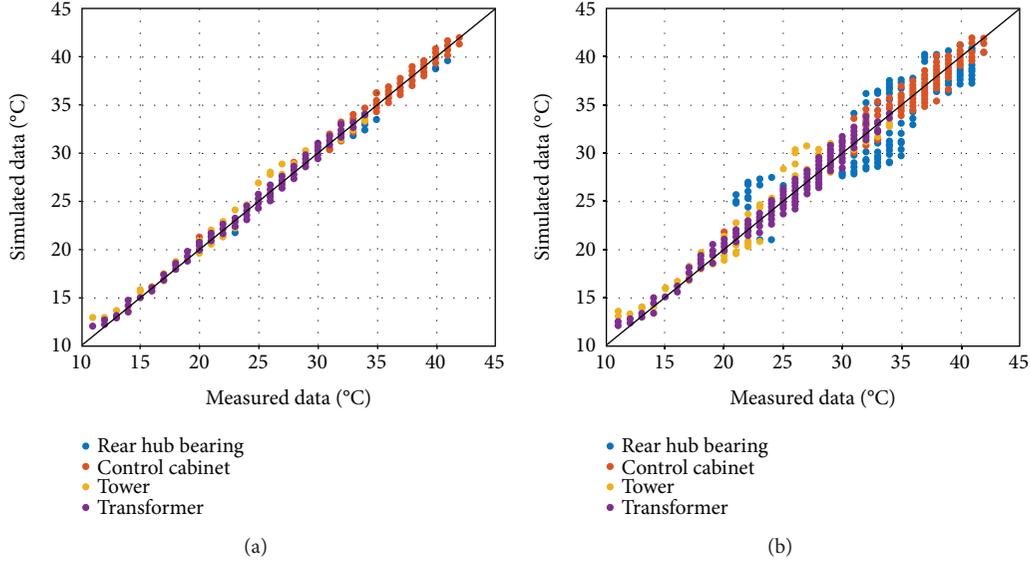


FIGURE 7: Regression plots for (a) autoassociative GRNN and (b) MISO GRNN.

TABLE 3: Root mean square error values (°C).

Sensor	Autoassoc. BPNN			Autoassoc. RBFNN			Autoassoc. GRNN		
	Case 1	Case 2	Case 3	Case 1	Case 2	Case 3	Case 1	Case 2	Case 3
S_1	0.1	0.4	0.5	$0.2e-4$	$0.3e-3$	$0.8e-3$	0.5	1.1	3.9
S_2	0.1	1.6	0.4	$0.1e-4$	$1.3e-3$	$0.5e-3$	0.4	4.2	2.3
S_3	0.3	0.3	0.7	$0.2e-4$	$0.5e-3$	$0.4e-3$	0.4	1.8	1.3
S_4	0.4	1.3	0.7	$0.1e-4$	$0.4e-3$	$0.5e-3$	0.4	1.8	1.3
Sensor	MISO BPNN			MISO RBFNN			MISO GRNN		
	Case 1	Case 2	Case 3	Case 1	Case 2	Case 3	Case 1	Case 2	Case 3
S_1	1.8	2.7	3.4	1.8	20.8	8.9	1.9	2.5	4.7
S_2	0.6	7.2	1.5	0.6	7.2	4.8	0.7	7.2	2.8
S_3	0.6	2.6	2.2	0.6	20.8	44.4	0.7	3.3	1.6
S_4	0.6	3.2	2.1	0.6	4.9	5.9	0.7	2.9	1.7

them by observing their relationship with other sensors in the group which were selected from physically relevant places, so it becomes possible to distinguish variations caused by calibration drifts from variations that appeared due to the real temperature changes. A temperature rise originated from the environmental effects would be seen in more than one sensor, whereas a calibration drift would result in a change only in the faulty sensor. To ensure that the proposed algorithm distinguishes the root reason for the temperature change, networks were tested in 3 different cases. Performances of the ANN models for the test set without any temperature drifts were observed in Case 1. In Case 2, a calibration fault was simulated by multiplying only one of the sensor's outputs with a constant factor. An overall shift in all temperature measurements was simulated in Case 3 which represents a change originated not from a fault but a real environmental temperature change.

The details and different expectations from ANN outputs for all cases are summarized as follows.

Case 1. No fault case.

In this case, temperature measurements gathered from the sensors were directly used as network inputs. 25% of the data that had been separated as the test set was used for this aim. The data presents the normal behaviour of the turbine without any faults, therefore the expectation from the networks is to produce as close output values as possible to the real sensor measurements.

Case 2. Multiplicative fault in one of the sensors.

Case 2 was designed to introduce a fault to be detected to the measurement system. A multiplicative fault was artificially created in one of the sensors. The test set for the measurements of the control cabinet temperature sensor (S_2) was multiplied by the constant term 1.2. The performances of the networks for this case were evaluated by analyzing the residuals between network outputs and sensor measurements. The expectation is to obtain greater residuals between

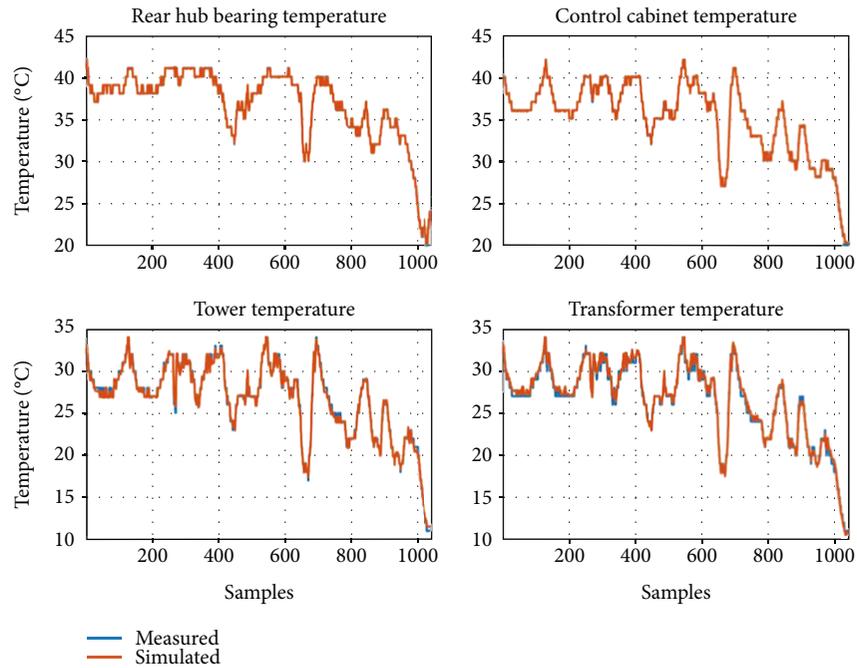


FIGURE 8: Measured and network values for Case 1 with autoassociative BPNN.

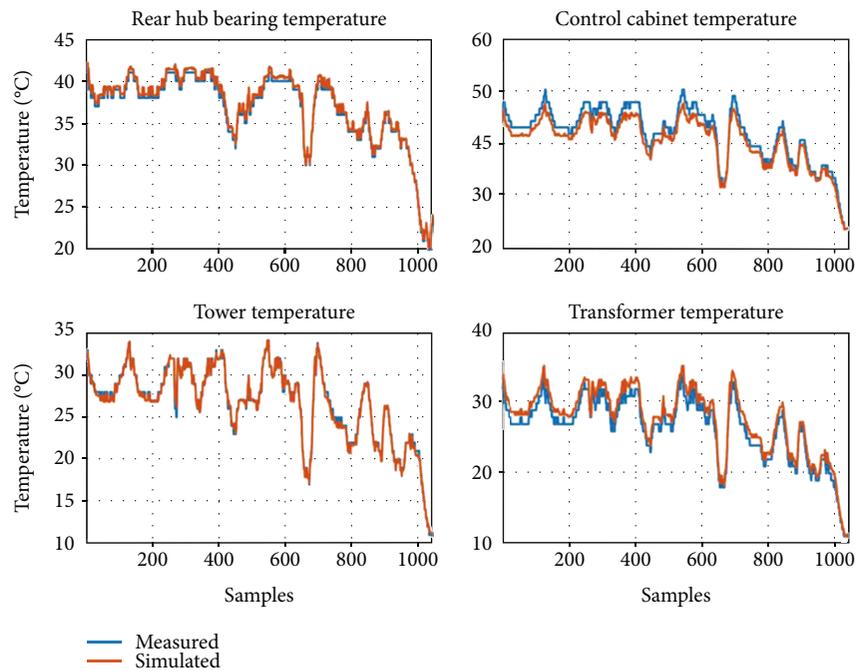


FIGURE 9: Measured and network output values for Case 2 with autoassociative BPNN.

measurement and estimation of the faulty sensor comparing to other sensors.

Case 3. No fault case. Overall shift in all measurements due to environmental temperature rise.

In this case, the measurements from all 4 sensors were shifted by multiplying by the constant term 1.2. The aim for this overall shift is to ensure that networks used for this fault detection algorithm do not produce a false fault alarm

when none of the sensors are faulty but instead a real temperature rise is recorded. The expectation from networks for this case is again to produce as close estimation values as possible to actual temperature values.

Input data for all 3 cases were implemented to the resulting 6 models with the best R^2 values. Outputs of the networks were analyzed based on different requirements for each case.

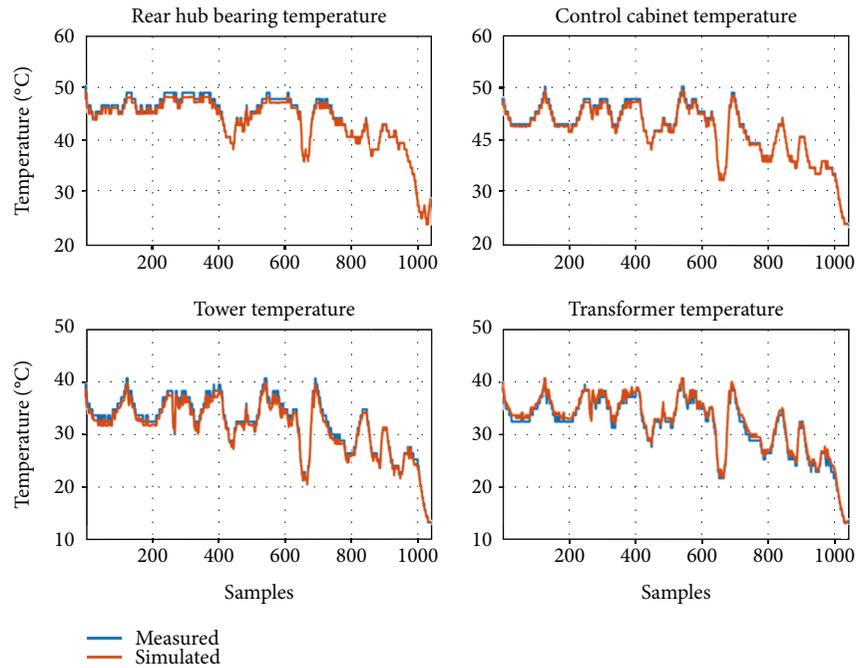


FIGURE 10: Measured and network output values for Case 3 with autoassociative BPNN.

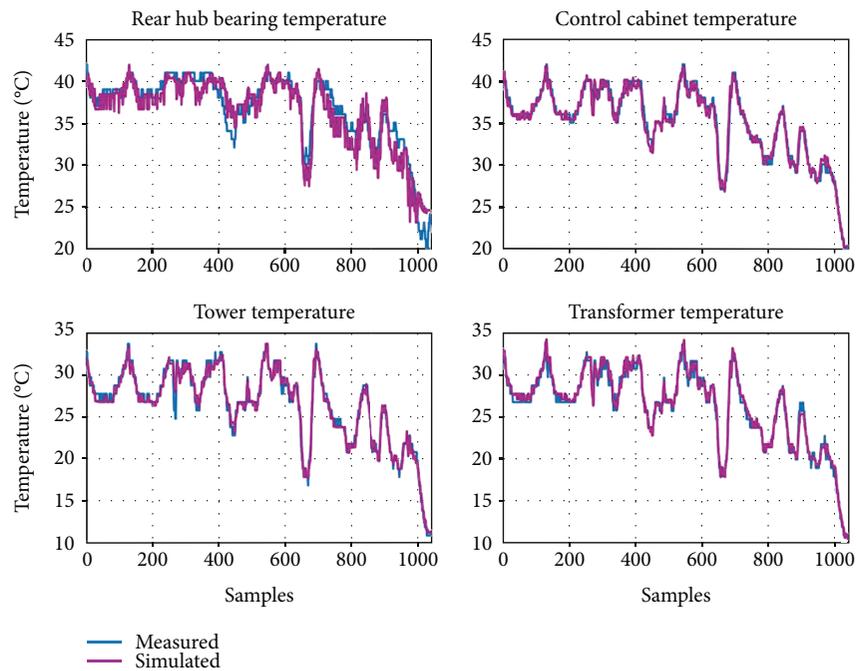


FIGURE 11: Measured and network values for Case 1 with MISO BPNN.

Table 3 presents the root mean square error (RMSE) values of ANNs for each case in °C. The results indicate that the most convenient network to distinguish the faulty case from nonfaulty ones is autoassociative BPNN. In Cases 1 and 3, the RMSE values between the real values and ANN outputs are less than 0.8°C for all sensors, whereas in Case 2, the RMSE values are 1.6°C and 1.3°C for sensors 2 and 4, respectively. Although autoassociative RBFNN outperforms the former network in Case 1 and Case 3, it was unable to

produce the required residuals in the faulty case (Case 2). Figures 8–10 presents the autoassociative BPNN results for each case. As can be seen from Figures 8 and 10, real temperature values and ANN outputs are very close in Case 1 and Case 3. Figure 9 shows that the residuals are visible in Case 2, which is a sign of fault existence. However, the magnitude of residuals for control cabinet and transformer is very close to each other. This situation prevents finding the exact location of the fault. Therefore, to isolate the fault location, the

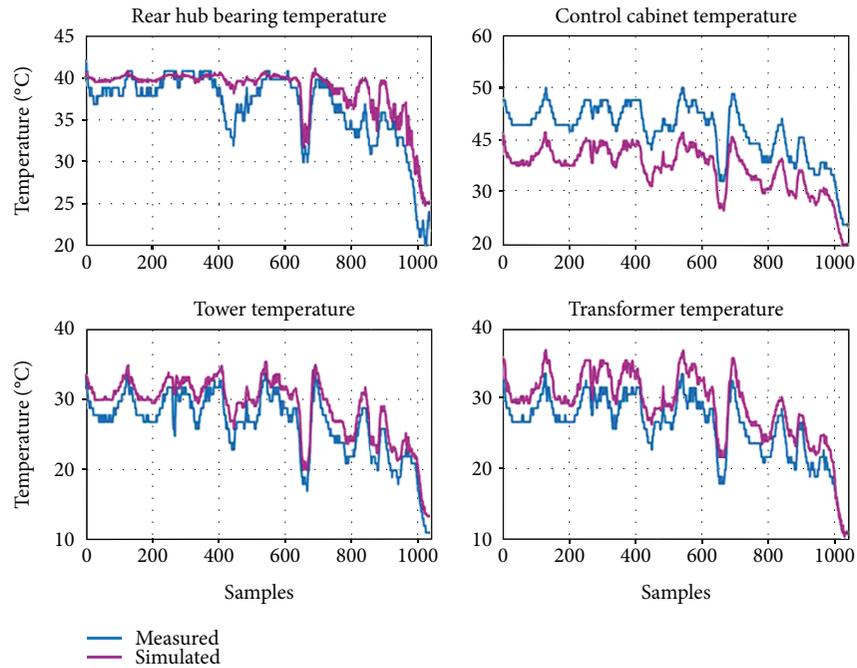


FIGURE 12: Measured and network values for Case 2 with MISO BPNN.

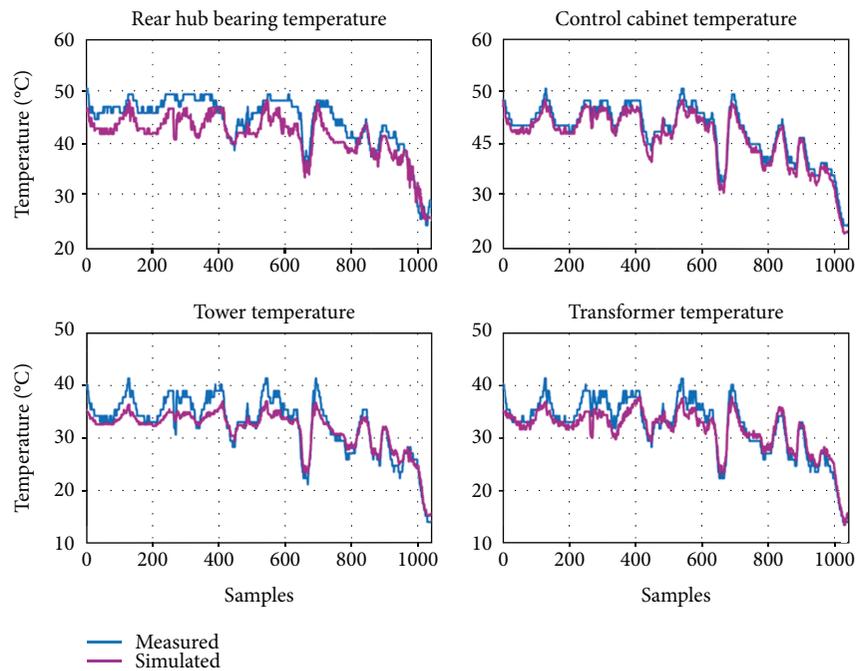


FIGURE 13: Measured and network values for Case 3 with MISO BPNN.

necessity of using different network estimations in a combined way has emerged.

The network providing the best results for fault isolation was found to be MISO BPNN, where the RMSE value of the faulty sensor (S_2) is significantly larger than other sensors in Case 2, and in other cases, the RMSE values are comparatively smaller. Figures 11–13 present the results obtained by the specified network. The corresponding figures show a combination of 4 neural networks with a different output

sensor in each subplot due to the single-output architecture of MISO networks.

As shown in Figure 11, the real data and the network outputs are consistent in the normal operation case. Figure 12 shows that in Case 2, unlike the results from the autoassociative network, this time the residuals for the control cabinet temperature are significantly bigger than the residuals for other sensors with an RMSE value of 7.2°C. Therefore, this network can be used for the isolation of fault location. For

Case 3 (as presented in Figure 13), the networks produce residuals which is undesired for this scenario; however, the decision of fault existence would be made with the support of autoassociative BPNN to prevent false alarms. By this combined decision-making algorithm using different networks, the fault detection system would be able to distinguish faulty and nonfaulty situations in a more sensitive way and give information on the exact location of faults.

5. Conclusions

A data-driven method using temperature sensors of a wind turbine SCADA system to detect and isolate sensor faults is proposed in this paper. A significant advantage of this method is the use of sensor data from the SCADA system which is a built-in part in most large-scale modern wind turbines; therefore, the data collection system does not bring any additional hardware requirements and extra costs. However, since SCADA measurements are of low frequency for fault detection, an intelligent model was needed that was provided by artificial neural networks in this research. Several artificial neural network architectures were trained for this aim. Based on the evaluation of simulation results, it is proposed to benefit from the strengths of different architectures. Autoassociative BPNN has proven to be a successful network to distinguish fault situation from nonfaulty ones, and MISO BPNN gives satisfactory results in terms of finding the exact sensor with faulty behaviour. The results demonstrate that the proposed method is feasible and effective for sensor fault detection and isolation. As a future work, the research can be extended to cover other measurements gathered from wind turbines to build an overall fault detection system.

Data Availability

The SCADA data used to support the findings of this study have not been made available because of the nondisclosure agreement signed with the wind energy company.

Conflicts of Interest

The authors declare that there is no conflict of interest regarding the publication of this paper.

Acknowledgments

This work was partly supported by the Scientific and Technological Research Council of Turkey (TUBITAK).

References

- [1] P. Tchakoua, R. Wamkeue, M. Ouhrouche, F. Slaoui-Hasnaoui, T. Tameghe, and G. Ekemb, "Wind turbine condition monitoring: state-of-the-art review, new trends, and future challenges," *Energies*, vol. 7, no. 4, pp. 2595–2630, 2014.
- [2] W. Yang, P. J. Tavner, C. J. Crabtree, Y. Feng, and Y. Qiu, "Wind turbine condition monitoring: technical and commercial challenges," *Wind Energy*, vol. 17, no. 5, pp. 673–693, 2014.
- [3] M. L. Wymore, J. E. Van Dam, H. Ceylan, and D. Qiao, "A survey of health monitoring systems for wind turbines," *Renewable and Sustainable Energy Reviews*, vol. 52, no. 1069283, pp. 976–990, 2015.
- [4] W. Qiao and D. Lu, "A survey on wind turbine condition monitoring and fault diagnosis—part I: components and sub-systems," *IEEE Transactions on Industrial Electronics*, vol. 62, no. 10, pp. 6536–6545, 2015.
- [5] W. Qiao and D. Lu, "A survey on wind turbine condition monitoring and fault diagnosis—part II: signals and signal processing methods," *IEEE Transactions on Industrial Electronics*, vol. 62, no. 10, pp. 6546–6557, 2015.
- [6] G. d. N. P. Leite, A. M. Araújo, and P. A. C. Rosas, "Prognostic techniques applied to maintenance of wind turbines: a concise and specific review," *Renewable and Sustainable Energy Reviews*, vol. 81, pp. 1917–1925, 2018.
- [7] P. F. Odgaard, J. Stoustrup, R. Nielsen, and C. Damgaard, "Observer based detection of sensor faults in wind turbines," in *Proceedings European Wind Energy Conference 2009*, pp. 1–10, Marseille, France, 2009.
- [8] X. Wei and M. Verhaegen, "Sensor and actuator fault diagnosis for wind turbine systems by using robust observer and filter," *Wind Energy*, vol. 14, no. 4, pp. 491–516, 2011.
- [9] W. Yang, R. Court, and J. Jiang, "Wind turbine condition monitoring by the approach of SCADA data analysis," *Renewable Energy*, vol. 53, pp. 365–376, 2013.
- [10] M. Schlechtingen, I. F. Santos, and S. Achiche, "Wind turbine condition monitoring based on SCADA data using normal behavior models. Part 1: system description," *Applied Soft Computing*, vol. 13, no. 1, pp. 259–270, 2013.
- [11] K. Leahy, R. Lily Hu, I. C. Konstantakopoulos, C. J. Spanos, and A. M. Agogino, "Diagnosing wind turbine faults using machine learning techniques applied to operational data," in *2016 IEEE International Conference on Prognostics and Health Management (ICPHM)*, pp. 1–8, Ottawa, ON, Canada, 2016.
- [12] Y. Zhao, D. Li, A. Dong, J. Lin, D. Kang, and L. Shang, "Fault prognosis of wind turbine generator using SCADA data," in *NAPS 2016-48th North Am. Power Symposium*, pp. 1–5, Denver, CO, USA, 2016.
- [13] P. Sun, J. Li, C. Wang, and X. Lei, "A generalized model for wind turbine anomaly identification based on SCADA data," *Applied Energy*, vol. 168, pp. 550–567, 2016.
- [14] P. B. Dao, W. J. Staszewski, T. Barszcz, and T. Uhl, "Condition monitoring and fault detection in wind turbines based on cointegration analysis of SCADA data," *Renewable Energy*, vol. 116, Part B, pp. 107–122, 2018.
- [15] L. Wang, Z. Zhang, H. Long, J. Xu, and R. Liu, "Wind turbine gearbox failure identification with deep neural networks," *IEEE Transactions on Industrial Informatics*, vol. 13, no. 3, pp. 1360–1368, 2017.
- [16] P. Caselitz and J. Giebardt, "Rotor condition monitoring for improved operational safety of offshore wind energy converters," *Journal of Solar Energy Engineering*, vol. 127, no. 2, p. 253, 2005.
- [17] M. J. Schulz and M. J. Sundaresan, *Smart Sensor System for Structural Condition Monitoring of Wind Turbines: 30 May 2002–30 April 2006*, Tech. Rep. National Renewable Energy Lab (NREL), Golden, CO, USA, 2006.
- [18] D. J. Comboni and B. Fazenda, "Acoustic condition monitoring of wind turbine blades: tip faults," *The Journal of the Acoustical Society of America*, vol. 132, p. 1956, 2012.

- [19] Y. Qu, E. Bechhoefer, D. He, and J. Zhu, "A new acoustic emission sensor based gear fault detection approach," *International Journal of Prognostics and Health Management*, vol. 4, pp. 1–14, 2013.
- [20] J. Ruan, S. C. M. Ho, D. Patil, and G. Song, "Structural health monitoring of wind turbine blade using piezoceramic based active sensing and impedance sensing," in *Proceedings of the 11th IEEE International Conference on Networking, Sensing and Control*, pp. 661–666, Miami, FL, USA, 2014.
- [21] O. M. Bouzid, G. Y. Tian, K. Cumanan, and D. Moore, "Structural health monitoring of wind turbine blades: acoustic source localization using wireless sensor networks," *Journal of Sensors*, vol. 2015, Article ID 139695, 11 pages, 2015.
- [22] Y. Qian and R. Yan, "Gearbox fault diagnosis in a wind turbine using single sensor based blind source separation," *Journal of Sensors*, vol. 2016, Article ID 6971952, 14 pages, 2016.
- [23] Y. Guo, X. Chen, S. Wang, R. Sun, and Z. Zhao, "Wind turbine diagnosis under variable speed conditions using a single sensor based on the synchrosqueezing transform method," *Sensors*, vol. 17, no. 5, p. 1149, 2017.
- [24] A. Coscetta, A. Minardo, L. Olivares et al., "Wind turbine blade monitoring with Brillouin-based fiber-optic sensors," *Journal of Sensors*, vol. 2017, Article ID 9175342, 5 pages, 2017.
- [25] P. F. Odgaard, J. Stoustrup, and M. Kinnaert, "Fault-tolerant control of wind turbines: a benchmark model," *IEEE Transactions on Control Systems Technology*, vol. 21, no. 4, pp. 1168–1182, 2013.
- [26] B. Krose and P. Van Der Smagt, *An Introduction to Neural Networks*, University of Amsterdam, 1996.
- [27] K. Patan, *Artificial Neural Networks for the Modelling and Fault Diagnosis of Technical Processes*, vol. 377, Springer, 2008.
- [28] S. Haykin, *Neural Networks and Learning Machines*, Prentice Hall, 3 edition, 2009.
- [29] D. F. Specht, "A general regression neural network," *IEEE Transactions on Neural Networks*, vol. 2, no. 6, pp. 568–576, 1991.



Hindawi

Submit your manuscripts at
www.hindawi.com

