Condition Monitoring Of Wind Turbines and Extraction Of Healthy Training Data Using an Ensemble Of Advanced Statistical Anomaly Detection Models

Xavier Chesterman[1], Timothy Verstraeten[2], Pieter-Jan Daems[3], Ann Nowé[4] and Jan Helsen[5]

[1,2,4] *Artificial Intelligence Laboratory, Vrije Universiteit Brussel, Brussels, Belgium*

xavier.chesterman@vub.be
timothy.verstraeten@vub.be
ann.nowe@vub.be

[1,3,5] *Acoustics and Vibrations Research Lab/OWI-lab, Vrije Universiteit Brussel, Brussels, Belgium*

pieter-jan.daems@vub.be
jan.helsen@vub.be

ABSTRACT

Premature failures caused by excessive wear are responsible for a large fraction of the maintenance costs of wind turbines. Therefore, it is crucial to be able to identify the formation of these failures as early as possible. To this end, a novel condition monitoring method is proposed that uses univariate and multivariate statistical data analysis techniques to construct an anomaly detection framework based on temperature SCADA data from wind turbines. The purpose of this framework is twofold. On the one hand it should give early warnings for failures, and on the other hand it should be able to extract healthy training data from unverified data for more advanced machine learning models. A large limitation of the latter models is that they require at least one year of training data. This is necessary to avoid seasonal dependence in the sensitivity of the models. The framework developed in this research contains multiple steps. First, there is a preprocessing step in which feature engineering and data transformation happens. The second step entails anomaly detection on the temperature time series data. This method uses fleet information to filter out common factors like wind speed and environmental temperature. Multiple models are combined to get more stable and robust anomaly detections. By combining them the weaknesses of the individual models are alleviated resulting in a better overall performance. To validate the model, temperature and failure data of a real operational wind farm is used. Although the methodology is general in its scope, the validation case focusses specifically on generator bearing failures.

1. INTRODUCTION

Under the impulse of a global shift towards renewable energy production, there are currently large investments happening in the wind turbine industry. According to the Global Wind Report 2021 of the Global Wind Energy Council, more than 90 GW of new wind power installations were installed. This brings the total installed energy production capacity of wind turbines to 743 GW, which is a growth of 14 % compared to the year before (GWEC, 2021). However, to keep the investments flowing into the sector, their profitability needs to be guaranteed. One of the main factors that influence the profitability are the costs, of which maintenance costs are an important part. Recent studies have shown that the operation and maintenance of wind turbines accounts for 25-40% of the levelized cost of energy (Pfaffel, Faulstich & Rohrig, 2017). These costs are driven in part by premature failures caused by excessive wear on components. These are, among other things, the result of high loads caused by environmental conditions and aggressive control actions (Verstraeten, Nowé, Keller, Guo, Sheng & Helsen, 2019), (Tazi, Châtelet & Bouzidi, 2017), (Greco, Sheng, Keller & Erdemir, 2013). Being able to predict the failure of a component plays an important part in reducing the operational costs of wind turbines, since it can avoid unnecessary downtimes. This in turn can give a boost to the profitability of the industry.

This research is a contribution towards the early identification of bearing failures. The value of this lies in the fact that it should allow the wind farm operators to optimize their maintenance schedule and better steer the control of the turbines towards an optimal balance between production and durability. Next to that this research also focusses on the extraction of “healthy” or nominal training data from

unverified data. Advanced anomaly detection algorithms require such a dataset to learn the normal relation between predictors and target. Distortions in the training data, due to anomalies of any kind, can lead to errors in the modelling of the normal behavior, which in turn lead to poor anomaly detection capabilities. The use of simple algorithms that are computationally inexpensive alleviates the need for large clean datasets.

In this research we present a framework that is based on more traditional statistical techniques like Autoregressive Integrated Moving Average (ARIMA), Ordinary Least Squares (OLS), cumulative sum control (CUSUM) charts, etc… These are well-studied methods that have proven their worth in the past. The novelty of this research lies in the combination of these relative simple techniques with fleet information. This makes anomaly detection on a complex system possible with relative simple models. As a validation step the framework will be applied on 82 generator bearing failure cases coming from a real operational wind farm. The general idea is to identify as early as possible bearing replacements by attaching an "anomaly" flag to the observations. This flag warns the operator that something is going wrong. It is also indicative for unhealthy behavior. This is important when a training dataset is constructed for more advanced machine learning models. All observations labelled with the anomaly flag should be excluded from the training.

The data used to validate the methodology is Supervisory Control and Data Acquisition (SCADA) data with a resolution of 10 minutes. The observations stretch over a period of more than 9 years, during which 82 replacements happened that are useful for the validation of the models. The results show that the statistical anomaly detector is able to predict most replacements months before they actually happen.

2. RELATED WORK

Much research is done on detecting anomalies in wind turbine signals. This has been driven by the fact that more and more sensors are being installed on them. More data is a blessing for engineers and scientists who depend on it for the performance analysis of certain components or the wind turbine as a whole. However, too much data can also be a curse. It resulted in a drive to automate the analysis of sensor data. Many different methods have been and are being developed to detect the anomalies. According to Helbing and Ritter (2018) a distinction can be made between model-based, signal processing and data-driven methods. The data-driven models train a normal behavior model (NBM) that predicts state values given that the wind turbine is in a normal operational state (a normal state is one in which the wind turbine has no defects of any kind). This implies that the NBM is used for the prediction of the normal behavior. In the next step the difference between the observed and predicted state values are analyzed. If the difference or deviation surpasses a certain threshold it is considered as evidence for an anomaly. In practice it means that linear regression, machine learning or deep learning models are used to predict the normal wind turbine state and that anomaly detection methods coming from statistical process control (SPC) like for example the exponentially weighted moving average (EWMA) chart or another type of statistic are used to assess the size of the deviation between the prediction and the actual value. The data-driven method is the method that is used in this research. In what follows a short overview will be given of papers that also use the data-driven method to detect anomalies.

Zhao, Liu, Hu and Yan (2018) use deep autoencoders (DAE) for anomaly detection and failure analysis on wind turbines. They train a DAE on healthy data, and test it on the remaining data. The reconstruction error is used as a measure to identify anomalies. To decide whether or not an observation is an anomaly, they designed an adaptive threshold based on Extreme Value Theory. They apply their methodology on several cases like a.o. the identification of anomalies in generator rear bearing temperatures. Kusiak and Vera (2012) predict bearing faults using SCADA data with a resolution of 10 seconds. For this they use neural networks to model the normal behavior. The predictors for these models were selected using a combination of domain knowledge and feature selection techniques. A moving average on the difference between the observed values and the predictions is then used to predict abnormal high temperatures. They succeed in predicting errors 1.5 hours before they effectively appear.

Bangalore and Tjernberg (2015) focus on gearbox bearings. The authors use an artificial neural network as the basis for a condition monitoring system. For the analysis they make use of SCADA data. The neural network is used to estimate the average 10-minute temperature. This estimation is compared to the observed temperature. They use the Mahalanobis distance to determine whether the bearings show anomalous behavior or not. Kusiak and Li (2010) use SCADA data with a 5 minute resolution together with status and fault data. Their methodology contains three levels: 1) the identification of the existence of a status/fault, 2) predicting the severity of the failure, and 3) predicting a specific error. In most cases they could predict the error with reasonable accuracy. Papatheou, Dervilis and Maguire (2014) also make use of SCADA data. The authors use neural networks and Gaussian processes to create reference power curves for each turbine in the farm. In the next step they investigate how well each power curve fits to the other turbines in the fleet. They use a confusion matrix of the mean squared errors (MSE) of the predictions for each reference power curve on each wind turbine. These MSE values are then used to identify anomalies.

Saputra and Marhadi (2020) develop a condition monitoring system (CMS) called Automatic Diagnosis that is based on vibration analysis. This system has four steps: angular

resampling, identification of peaks, priority labeling and frequency tracking. They obtain good results with this methodology: a sensitivity equal to 97.62% and a specificity of 99.21%. In Meyer and Brodbeck (2020) a machine-learning based anomaly detection method is developed for the detection and quantification of power generation anomalies. They do this by combining the predictions of several machine learning algorithms, and comparing the expected turbine output with the true output. Zgraggen, Ulmer, Jarlskog, Pizza and Huber (2021) use a combination of convolutional neural networks and linear regression for fault detection on wind turbines. As input they use SCADA data with a 10-minute resolution. They show that transfer learning by transferring a model trained on one turbine to other turbines and even fleets is possible. This has great advantages when the amount of healthy data is limited. Beretta, Cárdenas, Koch and Cusidó (2020) detect generator faults using a combination of an autoencoder and alarm log data of several fleets of wind turbines. They find that combining the two methods increases the performance drastically. Hendrickx, Meert, Cornelis, Gryllias and Davis (2020) develop a technique that identifies faulty turbines by looking at the whole fleet. The anomalies are identified by searching for turbines with signatures that deviate from the signature of the fleet. Hierarchical clustering and a pairwise similarity-based anomaly score are used.

The detection of anomalies in process data is by no means new. At least since the 30s techniques have been developed to detect anomalies in processes. For example, the Shewhart chart method was published in (Shewhart, 1931). Over two decades later the CUSUM chart method was developed in (Page, 1955). Specific to this method is that the cumulative sum of the process is used to detect changes in the mean of the underlying population. This makes this chart much more sensitive to small process changes than the Shewhart chart. A couple of years later the EWMA chart was described in (Roberts, 1959). These methods have however some data assumptions that need to be fulfilled. One of those assumptions is that the observations are independent of each other (no autocorrelation). This is an important assumption since it can have a profound impact on the accuracy of the control chart if it is ignored. It will also play an important role in this research. Bagshaw and Johnson (1974) show that CUSUM is not robust for deviations of the independence assumption and that it has a large impact on the Average Run Length (ARL). Lu and Reynolds (2001) pointed out that tight control limits and the presence of autocorrelation can result in a much higher false alarm rate. Unfortunately, many data signals in the industry are time series that are sampled at a high frequency. This increases the probability of having autocorrelation in the data. This is also the case for the data used in this research. Several solutions have been suggested to cope with autocorrelation.

On the one hand there are model-based (not to be confused with the definition of model-based methods in Helbing and Ritter (2018)) solutions that focus on modelling the autocorrelation by using ARIMA models. The SPC techniques are subsequently used on the residuals or the out-of-sample predictions. This is a method that is used in for example (Alwan & Roberts, 1988), (Kawod & Abbasi, 2016). They fit first a Box-Jenkins ARIMA model on the process. Next they use SPC techniques on the out-of-sample predictions. However, the model-based solutions also have their challenges. Kovarik, Sarga and Klimek (2015) point out that they require an in-control dataset which is not a straightforward requirement in many industrial contexts. On the other hand there are model-free solutions that do not depend on time series models. Apley and Tsung (2002) developed an autoregressive T^2 chart that can be used on autocorrelated data. This technique uses a rolling window on top of which a Hoteling's T^2 chart is modelled. In contrast to (Alwan & Alwan, 1994) no delays are used between the samples to break the statistical dependence. Another important assumption is that the process is identically distributed over time.

For the interested reader, Helbing and Ritter (2018) provides an overview on deep learning-based anomaly detection for wind turbines, while Montgomery (2009) provides an extensive overview of the statistical process control domain.

3. METHODOLOGY

The case that is studied in this paper focusses on generator bearings. For this, SCADA data with a 10-minute resolution is used. The data contains multiple temperature signals related to generator bearings. The study focusses on the generator rotor side bearing temperature (T_{rotor}), the generator rear bearing temperature (T_{rear}) and the generator cooling water temperature ($T_{cooling\_water}$). These are used as the targets for the NBMs. Trends in the temperatures are considered useful information for the detection of wear, which means they are interesting to monitor.

To detect abnormal behavior, an NBM needs to be designed. As discussed in the related work section, NBMs are in general created using advanced machine learning and deep learning models. The strength of these models is that they are capable of modelling complex relations. A disadvantage however is that they in general require a large dataset. Since the goal of this research is to develop a detector that has low computational requirements and can be used for the extraction of healthy data that can serve as input for the training of advanced machine learning models, an alternative type of models is needed. An alternative methodology is used that combines fleet information with more traditional statistical techniques. The anomaly detector uses a fleet aggregated signal temperature to normalize the signal temperatures of the individual wind turbines. Also, because this research is about bearing wear patterns that form over many months, the data is aggregated to a resolution of 1 day.

This reduces the amount of noise in the data, which should improve the anomaly detection accuracy.

$$y_{i,t} = f_t + \epsilon_{i,t} \quad (1)$$

- i = the wind turbine index
- t = time step
- f_t = common component
- $\epsilon_{i,t}$ = idiosyncratic component
- $y_{i,t}$ = raw temperature signal

The bearing temperature of a specific wind turbine can be decomposed in a common and an idiosyncratic part (Eq. 1). The common part is composed of factors that affect the whole fleet, such as environmental conditions at the site (e.g., ambient temperature and wind speed). This common part can be modelled by normalizing the wind turbine specific temperatures using the fleet behavior. It is a statistical calculation, based on the data of the whole fleet, of the most likely temperature value. It is important however that the fleet is sufficiently large to get a reliable aggregated fleet temperature. If the fleet is too small, it is more likely that a large percentage of the fleet is in a non-standard state like for example a cool-down due to a scheduled maintenance. Fortunately, the SCADA data available for this research comes from a sufficiently large wind farm.

Figures 1-3 (plots at the top) show the temperatures for the three signals for all the wind turbines of the wind farm. Non-standard states, such as cool-downs, are also clearly visible in the plots. These are the low temperatures that are sporadically visible. What is also apparent is that some turbines tend to have higher bearing temperatures than most other turbines during certain periods. This observation on its own is insufficient to conclude that these turbines are experiencing an anomaly.

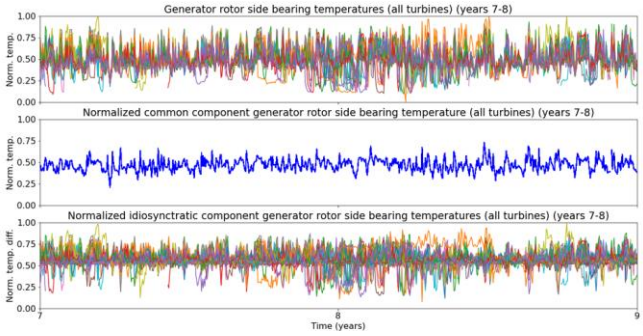

Figure 1: The plot gives an overview of the common and idiosyncratic component of the normalized T_{rotor} signal for all wind turbines during a period of two years. The first plot (top) shows the normalized temperature signal, the second plot (middle) shows the normalized common component and the third plot (bottom) shows the normalized idiosyncratic component.

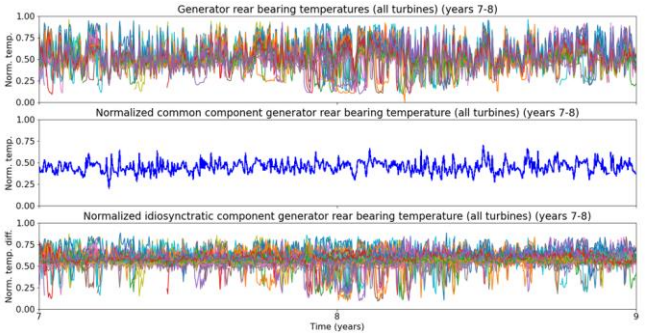

Figure 2: The plot gives an overview of the common and idiosyncratic component of the T_{rear} signal for all wind turbines during a period of two years. The first plot (top) shows the normalized temperature signal, the second plot (middle) shows the normalized common component and the third plot (bottom) shows the normalized idiosyncratic component.

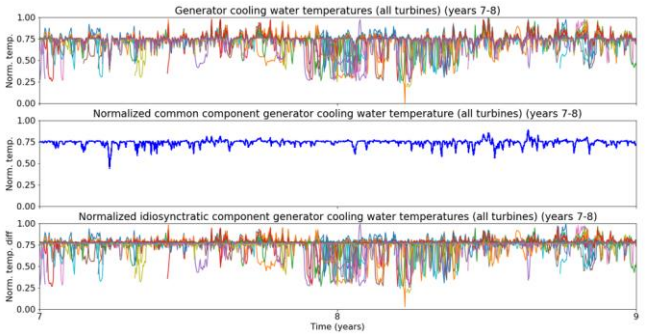

Figure 3: The plot gives an overview of the common and idiosyncratic component of the $T_{cooling\_water}$ signal for all wind turbines during a period of two years. The first plot (top) shows the normalized temperature signal, the second plot (middle) shows the normalized common component and the third plot (bottom) shows the normalized idiosyncratic component.

The middle plots of figures 1-3 show the fleet bearing temperatures for T_{rotor} , T_{rear} and $T_{cooling\_water}$. The extremely low and high temperatures that are visible in figure 1 are now not visible anymore. This indicates that the fleet aggregated temperature is an appropriate measure to model the common component. Temperature evolutions that can be attributed to specific wind turbine conditions are filtered out. For example around the end of year 7 and the start of year 8, there are multiple turbines that show large dips in the temperatures. This evolution is not visible in the fleet aggregated temperature.

The bottom plots of figures 1-3 show the idiosyncratic component for each wind turbine. This component contains turbine-specific factors of the bearing temperature. The idiosyncratic component is calculated by subtracting the fleet aggregated temperature from the wind turbine bearing temperature. The plots show again clearly the extremely low

temperatures that can be associated with cool-downs. Also the very high temperatures of certain turbines are again visible. If there are wind turbines that have temperature anomalies then they can be found in the idiosyncratic component, not in the common component. This means that the statistical process control algorithms that are used to find anomalies need to be used on the time series shown in the bottom plots of figures 1-3.

For the detection of anomalies in the idiosyncratic component multiple SPC techniques were tried. The most performant univariate technique for this research was the CUSUM. This method is simple. Nonetheless it is able to detect even small drifts and changes in the data. CUSUM is traditionally used to check whether a process is in-control (nominal), or not. Examples of application are chemical and industrial production processes. However, the technique has also been applied to detect anomalies in wind turbines like for example in (Xu, Shixiang, Zhongping & Cuixia, 2020) and (Dao, 2021).

$$\widehat{\epsilon_{i,t}} = \beta_0 + \sum_{j=1}^{p} \varphi_{j,i} X_{t-j,i} + \sum_{j=1}^{q} \theta_{j,i} \varepsilon_{t-j,i} \quad (2)$$

$$error_{ARIMA,i,t+1} = \epsilon_{i,t+1} - \widehat{\epsilon_{i,t+1}} \quad (3)$$

$$\widehat{error_{ARIMA,i,t+1}} = \beta_{0,i} + \beta_{1,i}\Delta_{active\ power,i,t+1} + \beta_{2,i}\Delta_{rotor\ speed,i,t+1} + \beta_{3,i}\Delta_{nacelle\ temp,i,t+1} \quad (4)$$

$$error_{OLS,i,t+1} = error_{OLS,i,t+1} - \widehat{error_{OLS,i,t+1}} \quad (5)$$

- i = wind turbine index
- t = time step
- ϵ = raw time series – fleet aggregate
- $\widehat{\epsilon_{i,t+1}}$ = one-step ahead ARIMA prediction
- $\beta_0, \varphi_{j,i}, \theta_{j,i}$ = parameters of ARIMA model
- X = lagged terms time series
- ε = lagged error terms
- $error_{ARIMA}$= the error made by the ARIMA model.
- $\widehat{error_{ARIMA,}}$ = estimate of ARIMA error by OLS
- $error_{OLS}$ = the error made by the OLS model.
- $\Delta_{active\ power}$= 'wind turbine active power' – 'fleet aggregated active power'
- $\Delta_{rotor\ speed}$ = 'wind turbine rotor speed' – 'fleet aggregated rotor speed'
- $\Delta_{nacelle\ temp}$ = 'wind turbine nacelle temperature' – 'fleet aggregated nacelle temperature'

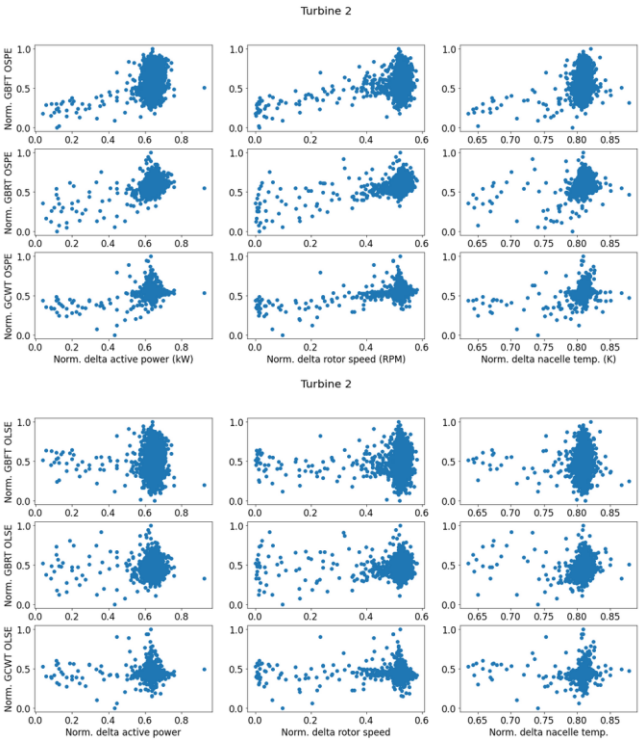

Figure 4: The plots show the relation between the idiosyncratic time series and the predictors before and after the OLS is applied for turbine 2. The top plot shows the situation before the OLS, the bottom plot after. On the x-axis stands the normalized delta active power, the normalized delta rotor speed and the normalized delta nacelle temperature. The top plot has on the y-axis the normalized T_{rotor} out-of-sample prediction (Norm. GBFT OSPE), the normalized T_{rear} out-of-sample prediction error (Norm. GBRT OSPE), and the normalized $T_{cooling\_water}$ out-of-sample prediction error (Norm. GCWT OSPE). The bottom plot has on the y-axis the normalized T_{rotor} OLS error (Norm. GBFT OLSE), the normalized T_{rear} OLS error (Norm. GBRT OLSE) and the normalized $T_{cooling\_water}$ OLS error (Norm. GCWT OLSE).

An important assumption of CUSUM is that the data samples must be independent and normally distributed (Cheng & Thaga, 2005). This is unfortunately not the case for the idiosyncratic time series, which is characterized by a strong autocorrelation between multiple lags. This property cannot be ignored when applying CUSUM since it results in unreliable anomaly detections. Several methods have been devised to handle autocorrelation in the data: 1) modified CUSUM techniques that are able to cope with the autocorrelation, 2) batching of the data and leaving gaps between the successive observations, and 3) fitting an ARIMA model to the data that models out the autocorrelation. In this research the last solution is

implemented. An ARIMA model is fit on the idiosyncratic time series (Eq. 2). Once a suitable model is found, the one-step ahead out-of-sample predictions are calculated. The difference between the predictions and the observations is now the time series of interest (Eq. 3). A check of the autocorrelation showed that the issue was resolved. There were however some deviations from the normal distribution. Still, in general they were relatively small.

A further examination showed that the one-step ahead forecast errors exhibit some (weak) correlation with the variables $\Delta_{active\ power}$, $\Delta_{rotor\ speed}$ and $\Delta_{nacelle\ temp}$. This can be expected, as for example, the active power of a wind turbine fluctuates in general around the fleet aggregate, which has an influence on the bearing temperatures. If the ARIMA one-step ahead prediction error shows a positive or negative value that can be explained by certain variables, then that error should not be considered an anomaly. If it cannot be explained it should be flagged as an anomaly. To cope with this a least squares (OLS) model was fit on the one-step ahead prediction errors with $\Delta_{active\ power}$, $\Delta_{rotor\ speed}$ and the $\Delta_{nacelle\ temp}$ as predictors (Eq. 4). By using the OLS errors the relation with the predictors is removed (Eq. 5). Figure 4 shows the end result for turbine 2. The clear positive relation that is visible in the top plot is gone after applying the OLS.

The CUSUM method that is used in this research is called the 'Tabular CUSUM'. It requires only a small number of parameters to be set: K and H. K is usually called the reference value, the allowance or the slack value (Montgomery, 2009). A general guideline to set K is to equal it to half the absolute deviation of the in-control mean and the out-of-control mean. H is called the decision interval. If C_i^+ (see Eq. (6)) or C_i^- (see Eq. (7)) exceed H then the process is considered out-of-control. A common value for H is around five times the process standard deviation σ. For each target signal, a separate univariate CUSUM was calculated.

$$C_i^+ = \max\left[0, x_i - (\mu_0 + K) + C_{i-1}^+\right] \quad (6)$$

$$C_i^- = \max\left[0, (\mu_0 - K) - x_i + C_{i-1}^-\right] \quad (7)$$

$$\text{With } C_0^+ = C_0^- = 0$$

$$K = \frac{|\mu_1 - \mu_0|}{2} \quad (8)$$

To smooth out the anomaly detections, a moving average filter is applied to the anomaly detections. If the concentration of detected anomalies surpasses a certain threshold, an anomaly flag is assigned to the observation. If this is not the case a no-anomaly or normal flag is assigned to it. The threshold used in this research is 0.06. A last step in the anomaly detection process is combining the results for the three signals. If at least one signal flags an anomaly, meaning the anomaly concentration surpasses the threshold for that specific signal, then a general anomaly is raised. The general anomaly flag gives the operator a first idea whether or not something went wrong.

To validate the methodology, the accuracy is determined by calculating the ratio of the number of replacements that are immediately preceded by an anomaly flag versus the total number of replacements. Furthermore we also calculated how much time there on average is between the raising of the flag and the replacement. This measure is different from the more standard Average Run Length (ARL) metric that is often used in statistical process control. The reason for this is that we do not exactly know when the problem first appears. This makes it impossible to calculate the ARL. The models also flag observations as anomalies that are not connected to a replacement. These flags are not necessarily false positives, as they may be caused by other issues or failures of which we have no knowledge. This means that a false positive rate cannot be calculated. The best that can be done in this situation is assuming that a turbine cannot be all the time in an anomalous state, and that anomalies should be clustered in time.

The use of fleet information is not new, e.g. Beretta et al., (2020), Hendrickx et al. (2020). The latter is most similar to our work. They assume that most wind turbines are healthy. Furthermore they assume that the healthy wind turbines are grouped in a single cluster, and that this cluster is the largest in size. The faulty wind turbines are clustered in smaller separate clusters. The anomaly score is then calculated as the ratio of the median of the similarities between a specific wind turbine and all other wind turbines in the fleet vs. the median of the similarities of the wind turbines in the largest cluster. In our research no clustering techniques are used. Instead a fleet aggregated temperature is calculated based on the information in the fleet. This metric is robust for anomalies, but it also assumes that the majority of the wind turbines are healthy. The anomaly scores are not based on pairwise distances like in Hendrickx et al. (2020), instead they are based on the cumulative difference between the fleet aggregate temperature and the wind turbine temperature. This has several advantages. 1) Calculating pairwise distances becomes prohibitively expensive when the fleet is very large. This is not the case with our methodology. 2) By using CUSUM advantage is taken from the fact that the data are time series. This is not the case with the methodology in Hendrickx et al. (2020). 3) The methodology presented here does not depend on a guess of the number of wind turbine clusters.

4. RESULTS

For this research SCADA data was available from a large operational wind farm. The data stretches over a period of more than nine years. The resolution of the data is 10 minutes.

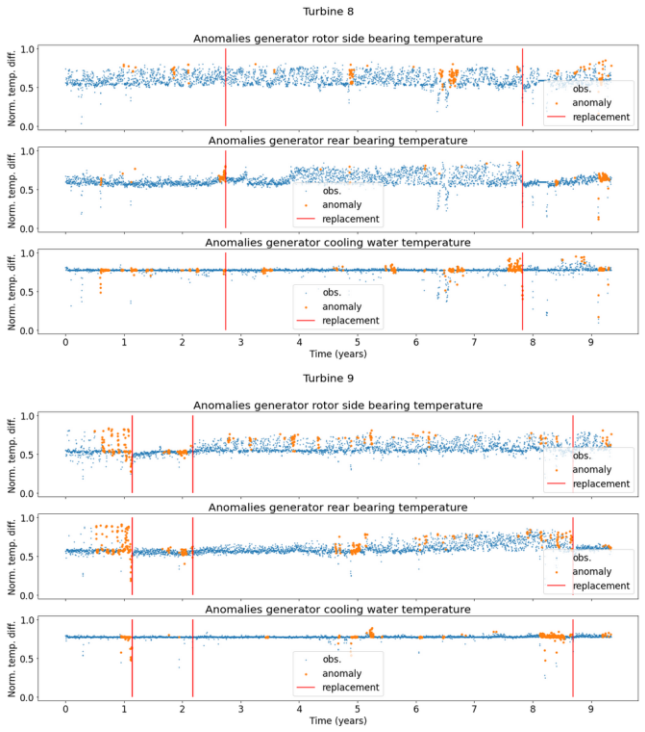

Figure 5: Detected anomalies on delta target signals (i.e. difference between wind turbine and fleet aggregated target signals) for two randomly selected wind turbines. The orange points are the identified anomalies. The red vertical bars indicate the moment a bearing was replaced. "Norm. temp. diff." stands for "Normalized temperature difference" or the normalized difference between the wind turbine temperature and the fleet aggregated temperature.

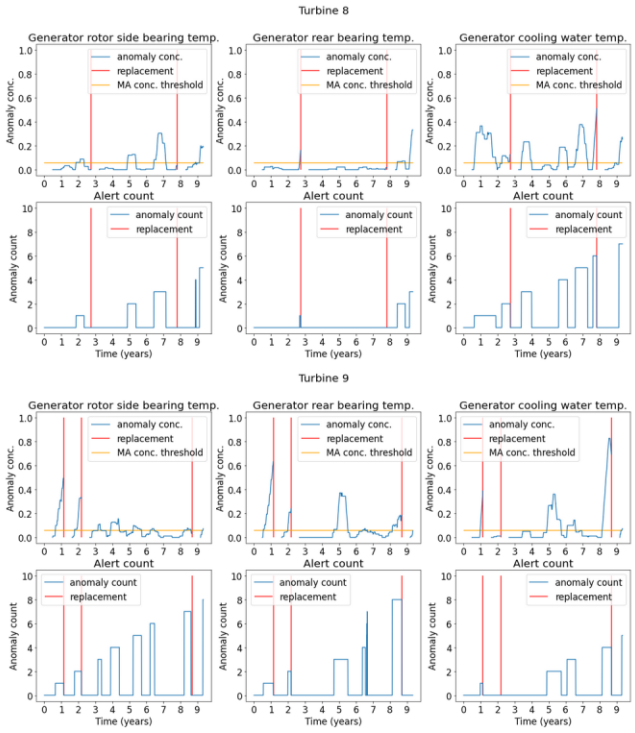

Figure 6: Frequency of anomalies in a rolling window with a length of 6 months. The plots on the first and third row show the anomaly frequency. The orange horizontal line is the threshold. The red vertical bars are the replacements. The plots on the second and fourth row show us the regions that got an "anomaly" flag (height not equal to 0). The height of the line gives the cumulative flag count.

Figure 5 shows the CUSUM anomaly detection results (that is before the moving average filter has been used) for two representative examples. It shows that the model is able to identify problems well before the actual replacement. The anomalies are also clustered in time which indicates that the detections are not just random. Moreover, they are generated more frequently toward the replacement, signifying wear progression. Each time series is divided into several runs. A run is defined as the observations between two replacements. The first run is the time series between the first observation in the dataset and the first replacement, while the last run contains the observations between the last replacement and the last observation in the data. This means that turbine 8 has three runs and turbine 9 four. Because a replacement changes something structurally to the data the model is reset at the beginning of each run. This implies that a new ARIMA model is learned based on 6 months of information, a new OLS is fit and the CUSUM is reset to 0.

Although the anomalies are clustered in time, which suggests that the methodology is quite stable in nature, there are still observations in those clusters that are not signaled as anomalies. This is not surprising since the CUSUM gives for each single day an indication of whether the temperature values seen that day are anomalous or not. This phenomenon can be attributed to variability in the data and to a limited extent to variability in the methodology. The former is quite substantial. The average coefficient of variance for T_{rotor} , T_{rear} and $T_{cooling\_water}$ is respectively 3.06%, 3.46% and 0.85%. The values for T_{rotor} and T_{rear} are quite high. The variability in the methodology follows from the fact that multiple preprocessing steps are taken: ARIMA and OLS model fitting, and then the CUSUM. However, these techniques have relative few parameters and as such have also a relative limited inherent instability compared to more complex models such as neural networks. Nevertheless, the instability in the results might be confusing for the user or operator. The goal of this research is to present the user with an unmistakable clear assessment of the health. To this end the CUSUM results will be smoothed, and "anomaly flags" will be assigned to the smoothed regions. These flags

are based on the concentration of anomalies in a certain time window. To accomplish this the moving average over a period of 6 months (rolling window) was taken. If the number of anomalies in that period surpassed the threshold, then the observation that is associated with the rolling window is flagged as an anomaly.

Figure 6 shows the results of taking the moving average. It is apparent that larger homogenous blocks are formed, which is more representative for slow and continuous wear progression. The threshold to consider a concentration as significant is set on 0.06 for all the turbines. This threshold balances out the detection accuracy versus the number of flagged regions that are not associated with a replacement. A lower threshold implies more correctly predicted replacements, but also implies more observations that are identified as anomalies but are not related to a replacement.

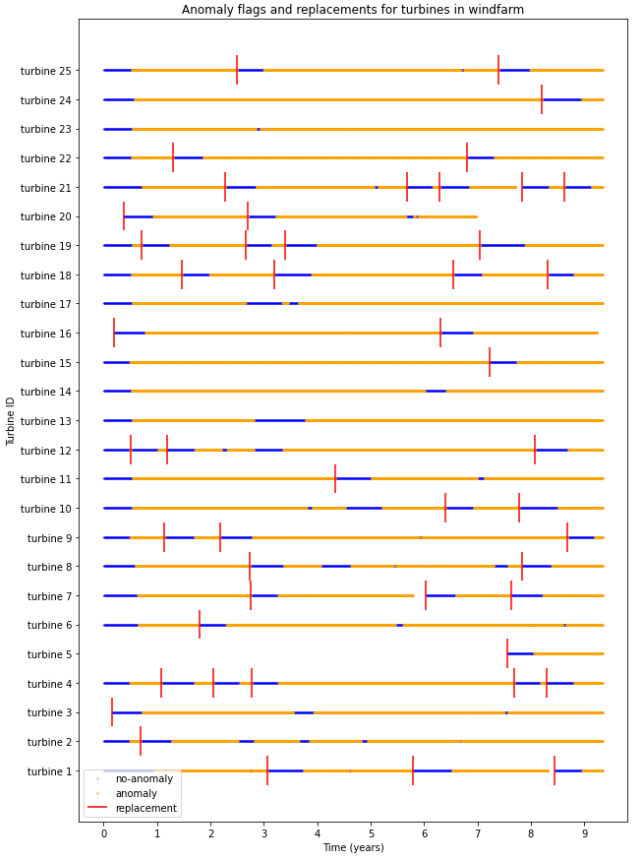

Figure 7: Anomaly flags and replacements for subset of 25 turbines of the windfarm. Orange are the observations identified with an anomaly flag. Blue are the observations identified with a no-anomaly flag. The red vertical bars are the replacements. White spaces indicate missing values.

Figure 7 gives an overview of the replacements and the anomaly flags for all the turbines in the windfarm. The threshold is here set to 0.01, which is a low (aggressive) setting. It is clear from this plot that the large majority of the replacements are predicted well before the replacement (>92%). A replacement is correctly predicted if it is immediately preceded by an anomaly flag (orange bars). In some cases there was data missing just before the replacement. If there is no data, it is obviously not possible to detect anomalies. So these cases were discarded in the accuracy calculation. In some cases the replacement came very early in the observation period (e.g. the first replacement for turbines 3, 16, 20). Also for these replacements it isn't possible to identify anomalies. So they were also discarded.

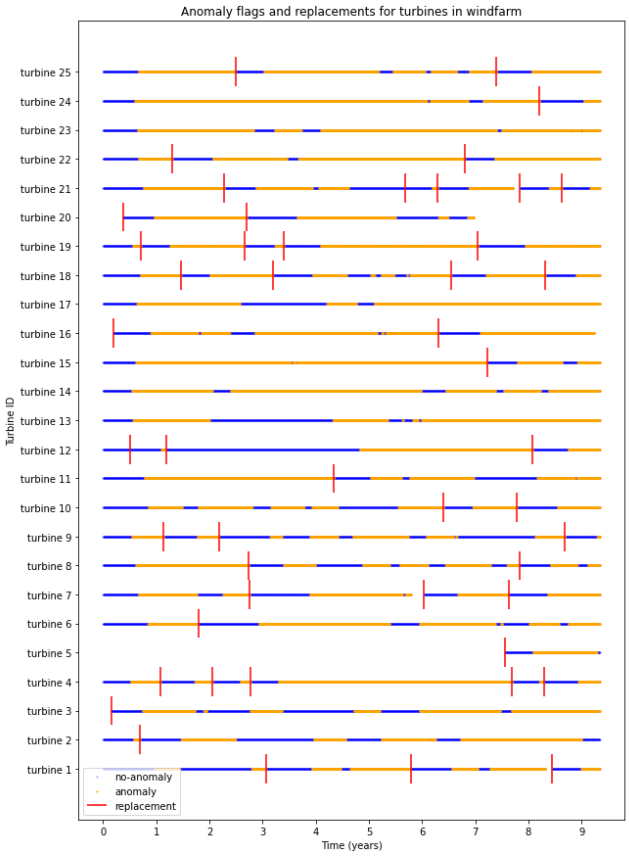

Figure 8: Anomaly flags and replacements for a subset of the turbines of the windfarm. Orange are the observations identified with an anomaly flag. Blue are the observations identified with a no-anomaly flag. The red vertical bars are the replacements. White spaces indicate missing values.

Figure 7 also shows that many observations that cannot be associated with a replacement are identified as anomalies. It is possible that these anomalies are justified, and caused by unknown factors. Therefore, they cannot be dismissed as false positives. Still, further analysis of these anomalies is required to classify them. Moreover, a trade-off exists between the number of correctly predicted replacements and the overall number of anomalies found. In figure 7 the moving average anomaly concentration threshold was set to 0.01. If the threshold is increased to 0.06 (see figure 8), then

we see that the overall number of anomalies decreases drastically. The percentage of correct replacement predictions unfortunately also decreases (>85%).

Table 1: Replacement prediction accuracy and the ratio of flagged anomalies vs. the no-anomalies. The threshold is the moving average anomaly concentration threshold.

Threshold (# anomalies / 180 day window)	Accuracy (%)	Ratio anomalies vs. no-anomalies
0.01	92.68	4.01
0.02	90.24	3.17
0.03	87.80	2.63
0.04	86.85	2.24
0.05	85.37	1.94
0.06	85.37	1.80
0.07	82.93	1.57
0.08	81.71	1.39
0.09	79.27	1.25
0.10	76.83	1.14
0.11	76.83	1.09
0.12	76.83	1.01
0.13	75.61	0.93
0.14	75.61	0.86
0.15	74.39	0.82
0.16	73.17	0.80
0.17	70.73	0.75
0.18	68.29	0.70
0.19	67.07	0.66
0.20	65.85	0.62

Table 1 gives an overview of how the accuracy and the ratio of anomalies / no-anomalies (= the ratio of the number of observations that are assigned to the "anomaly" category vs. the number of observations that are assigned to the "not an anomaly" category) evolves if the threshold is changed. If the threshold is set to 0.01 an accuracy of 92.68% is reached. The anomalies / no-anomalies ratio (i.e., = 4.01) is however also high. An increase of the threshold quickly reduces this ratio, while the accuracy decrease more slowly. This is an interesting observation since it allows us to optimize the threshold a bit. Depending on the relative importance of the accuracy and the ratio of the anomalies vs. no-anomalies, the threshold can be decreased or increased. This of course depends on the use case and the preferences of the user. However, the fact that for accuracies between 75% and 85%, the ratio of anomalies / no-anomalies evolves from 0.86 to 1.94, makes that the user has quite some options to optimize the results to his or her needs. It is however fair to say that setting the threshold to 0.01 (which results in an anomalies / no-anomalies ratio of 4) has relatively little economic value.

Table 2: Mean duration of an anomaly for different threshold values. The threshold is the moving average anomaly concentration threshold.

Threshold (# anomalies / 180 days window)	TTR (days)
0.01	828
0.02	662
0.03	581
0.04	513
0.05	463
0.06	434
0.07	368
0.08	337
0.09	320
0.10	314
0.11	312
0.12	298
0.13	291
0.14	284
0.15	275
0.16	271
0.17	268
0.18	261
0.19	250
0.20	243

Another element of interest is how fast a replacement is detected. Since we have no information on the true moment of damage initiation, we can only calculate how long in advance the anomaly detector signals that something is going wrong. We will call this measure Time Till Replacement (TTR). TTR depends also on the threshold. A lower threshold results in a higher TTR, and vice versa. Table 2 shows the evolution of the mean TTR. For a threshold equal to 0.01 the mean TTR is 828 days. At a threshold value of 0.06 the TTR has already nearly halved. The mean TTR hides however a large variance.

Table 3: Overview of the number of replacements that are detected by the anomaly detector 6 months, 3 months and 1 month before the replacement itself. The values have been given for different threshold values.

Threshold (# anomalies / 180 day window)	6 months (%)	3 months (%)	1 month (%)
0.01	74.87	85.35	95.83
0.02	74.21	82.28	95.61
0.03	71.93	80.53	96.49
0.04	71.93	80.96	96.49
0.05	72.37	81.40	94.21
0.06	69.74	78.77	92.89
0.07	59.86	74.82	89.32
0.08	61.53	75.97	91.81
0.09	56.29	73.14	90.86
0.10	57.94	74.71	93.28
0.11	56.23	74.71	91.81
0.12	54.75	72.50	88.87
0.13	50.83	69.56	84.46
0.14	47.89	68.97	84.46
0.15	47.89	68.97	85.93
0.16	45.44	70.44	87.40
0.17	45.30	69.55	85.51
0.18	44.64	68.59	85.05
0.19	45.16	69.11	85.83
0.20	43.39	69.73	84.30

Table 3 gives us an overview of how often a replacement is detected 6 months, 3 months and 1 month in advance given different threshold values. As expected the low threshold values give more often a long early warning then the high thresholds. When the threshold is set to 0.01, 74.87 % of the correctly predicted replacements are found at least 6 months in advance. For the more conservative threshold of 0.06, this percentage is still 69.74%. More than 92% is found at least one month in advance. It's up for debate whether detecting the replacement 6 months before the actual replacement is valuable since the bearing is still usable for 6 months. However, the purpose of this early warning system is to indicate to the operator that substantial wear is forming on the bearing and that in the near future actions will have to be taken. To get more precise predictions of when the bearing will fail, the methodology presented here can be used in conjunction with other models that allow for more precise end-of-life predictions. The advantage is that the areas where those other models need to look are fewer since a preliminary selection has already happened. Furthermore, the findings are also interesting if the methodology presented here is used as a preprocessing step (the selection of healthy data) for more complex machine learning and deep learning algorithms. By flagging the wear on the bearings early on, the healthy data, which serves as the basis for training the more complex models, is cleaned more properly. This should improve their accuracy.

5. CONCLUSION

The purpose of this research was to develop an ensemble of advanced statistical anomaly detection methods that can detect bearing failures months before they actually happen. To that end SCADA data with a resolution of 10 minutes was used of a large operational wind farm, for which we had over 9 years of data. The method is based on traditional statistical methods because those require less data and are less computationally intensive. The disadvantage is however that they require more preprocessing and are potentially not as performant as machine learning and deep learning techniques. The end goal was to develop a method that can be used to extract healthy data, which can then be used as input to the more advanced models.

The methodology assumes that the bearing temperature time series can be decomposed in a common and an idiosyncratic component. The common component is modelled using an aggregate of the temperatures of all the wind turbines in the fleet. This fleet aggregate can be seen as a model for the normal temperatures that are caused by general, meaning not wind turbine specific, factors. The fleet aggregate is here used instead of an advanced machine learning or deep learning model. The reason for this decision is that it isn't entirely clear which factors all have an impact on the bearing temperatures of real wind turbines. We also don't have data on all the relevant factors. So using the fleet aggregate circumvents this problem. By normalizing the wind turbine signal temperatures (by subtracting the fleet aggregate from them), we find the wind turbine specific deviations from the fleet normal (the idiosyncratic component). Part of it is caused by normal turbine characteristics that make the turbine deviate slightly from its peers. This can be due to e.g. higher active power, a higher rotor speed. These turbine characteristics were modelled out using OLS. The rest of the idiosyncratic component can be attributed to noise and abnormal deviations. To distinguish abnormal deviations or anomalies from noise the CUSUM method is used which is able to detect even small shifts in the mean temperatures.

The end result is an anomaly detector that is able to detect most replacements long before they actually happen. However, depending on certain parameters, more or less

anomalies are detected that are not related to a replacement. There is a trade-off between the accuracy of the model in predicting replacements and the number of anomalies that are found that are unrelated to replacements. The latter anomalies can't just be called false positives since they might very well correspond to real issues of which we don't have information.

6. ACKNOWLEDGMENTS

Xavier Chesterman, Timothy Verstraeten, Pieter-Jan Daems, Ann Nowé and Jan Helsen received funding from the Flemish Government (AI Research Program). The research presented in this paper is partly financed by the European Union (H2020 PLATOON, Pr. No: 872592). The authors would also like to acknowledge FWO for the support through the SBO Robustify project (S006119N), and Blue Cluster ICON project Supersized 4.0.

7. NOMENCLATURE

ARIMA	Autoregressive Integrated Moving Average
OLS	Ordinary Least Squares
CUSUM	cumulative sum control
SCADA	Supervisory Control and Data Acquisition
NBM	normal behavior model
SPC	statistical process control
EWMA	exponentially weighted moving average
DAE	deep autoencoders
MSE	mean squared errors
CMS	condition monitoring system
T_{rotor}	generator rotor side bearing temperature
T_{rear}	generator rear bearing temperature
$T_{cooling\_water}$	generator cooling water temperature
GBFT OSPE	T_{rotor} out-of-sample prediction error (ARIMA)
GBRT OSPE	T_{rear} out-of-sample prediction error (ARIMA)
GCWT OSPE	$T_{cooling\_water}$ out-of-sample prediction error (ARIMA)
GBFT OLSE	T_{rotor} ordinary least squares error
GBRT OLSE	T_{rear} ordinary least squares error
GCWT OLSE	$T_{cooling\_water}$ ordinary least squares error
$error_{ARIMA}$	The error made by the ARIMA model.
$error_{OLS}$	The error made by the OLS model.
$\Delta_{active\ power}$	wind turbine active power - fleet aggregated active power
$\Delta_{rotor\ speed}$	wind turbine rotor speed – fleet aggregated rotor speed
$\Delta_{nacelle\ temp}$	wind turbine nacelle temperature – fleet aggregated nacelle temperature
OLS	Ordinary Least Squares
K	reference value CUSUM
H	allowance or slack value CUSUM
ARL	Average Run Length
TTR	Time Till Replacement

8. REFERENCES

Alwan, L.C., & Roberts, H.V. (1988). Time-Series Modeling for Statistical Process Control. *Journal of Business & Economic Statistics*, 6 (1), 87-95.

Bagshaw, M., & Johnson, R.A. (1974). The Effect of Serial Correlation on the Performance of CUSUM Tests. *Technometrics*, 16 (1), 103-112.

Bangalore, P., & Tjernberg, L.B. (2015). An Artificial Neural Network Approach for Early Fault Detection of Gearbox Bearings. *IEEE Transactions on Smart Grid*, 6 (2), 980-987.

Beretta, M., Cárdenas, J.J., Koch, C., and Cusidó J. (2020). Wind Fleet Generator Fault Detection via SCADA Alarms and Autoencoders. *Applied Sciences*, 10 (23), 8649. https://doi.org/10.3390/app10238649.

Cheng, S.W., & Thaga, K. (2005). Max-CUSUM chart for autocorrelated processes. *Statistica Sinica*, 15, 527-546.

Dao, P.B. (2021). A CUSUM-Based Approach for Condition Monitoring and Fault Diagnosis of Wind Turbines. *Energies*, 14 (11), 3236. doi:10.3390/en14113236.

Dawod, A.B.A., & Abbasi, S.A. (2016). On Model Selection for Autocorrelated Processes in Statistical Process Control. *Quality and Reliability Engineering*, 33, 867-882. doi:10.1002/qre.2063.

Greco, A., Sheng, S., Keller, J.A., & Erdemir A. (2013). Material wear and fatigue in wind turbine systems. *Wear*, 302 (1-2), 1583-1591.

Hendrickx, K., Meert, W., Cornelis, B., Gryllias, K., & Davis, J. (2020). Similarity-based anomaly score for fleet-based condition monitoring. *Annual Conference of the PHM Society*, *12* (1), 9. https://doi.org/10.36001/phmconf.2020.v12i1.1178

Kovarik, M., Sarga, L., & Klimek, P. (2015). Usage of control charts for time series analysis in financial management, *Journal of Business Economics and Management*, 16 (1), 138-158.

Kusiak, A., & Li, W. (2011). The prediction and diagnosis of wind turbine faults. *Renewable Energy*, 36, 16-23.

Kusiak, A., & Verma, A. (2012). Analyzing bearing faults in wind turbines: A data-mining approach. *Renewable Energy*, 2012, 48, 110-116.

Lee J., & Zhao F. (Eds.). (2021). *Global wind report 2021*. Brussels, Belgium: Global Wind Energy Council.

Lu, C.W., & Reynolds, JR. M.R. (2001). CUSUM charts for monitoring an autocorrelated process. *Journal of Quality Technology*, 33, 316-334.

Meyer, A., & Brodbeck, B. (2020). Data-driven Performance Fault Detection in Commercial Wind Turbines. *PHM Society European Conference*, *5*(1), 7. https://doi.org/10.36001/phme.2020.v5i1.1276

Montgomery, D.C. (2009). *Introduction to statistical quality control (6th Edition)*. USA: John Wiley & Sons, Inc.

Page E.S. (1955). A Test for a Change in a Parameter Occurring at an Unknown Point. *Biometrika*, 42 (3/4), 523-527.

Papatheou, E., Dervilis, N., & Maguire, A.E. (2014). Wind Turbine Structural Health Monitoring: A Short Investigation Based on SCADA Data. *7th European Workshop on Structural Health Monitoring (EWSHM 2014)* (512-519), July 8-11, Nantes, France.

Pfaffel, S., Faulstich, S., & Rohrig, K. (2017). Performance and Reliability of Wind Turbines: A Review. *Energies*, 10, 1904, doi:10.3390/en10111904.

Roberts, S.W. (1959). Control Chart Tests Based on Geometric Moving Averages. *Technometrics*, 1 (3), 239-250.

Saputra, D., & Marhadi, K. (2020). On Automatic Fault Diagnosis in Wind Turbine Condition Monitoring. *PHM Society European Conference*, *5*(1), 8. https://doi.org/10.36001/phme.2020.v5i1.1251

Schlechtingen, M., & Santos, I.F. (2011). Comparative analysis of neural network and regression based condition monitoring approaches for wind turbine fault detection. *Mechanical Systems and Signal Processing*, 25, 1849-1875.

Shewhart, W.A. (1931). *Economic control of quality of manufactured product*. England: MacMillan and Co., Limited.

Tazi, N., Châtelet, E., & Bouzidi, Y. (2017). Wear Analysis of Wind Turbine Bearings. *International Journal of Renewable Energy Research*, 7 (4), 2120-2129.

Verstraeten, T., Nowé, A., Keller, J., Guo, Y., Sheng, S., & Helsen, J. (2019). Fleetwide data-enabled reliability improvement of wind turbines. *Renewable and Sustainable Energy Reviews*, 109, 428-437.

Xiao, C., Liu, Z., Zhang, T., & Zhang, X. (2021). Deep Learning Method for Fault Detection of Wind Turbine Converter, *Applied Sciences*, 11, 1280. doi:10.3390/app11031280.

Xu, Q., Shixiang, L., Zhongping, Z., & Cuixia, J. (2020). Adaptive fault detection in wind turbine via RF and CUSUM. *IET Renewable Power Generation*, 14 (10), 1789-1796.

Zgraggen, J., Ulmer, M., Jarlskog, E., Pizza, G., & Huber L.G. (2021). Transfer Learning Approaches for Wind Turbine Fault Detection using Deep Learning. PHM Society European conference, 6 (1), 12. https://doi.org/10.36001/phme.2021.v6i1.2835

Zhao, H., Liu, H., Hu, W., & Yan, X. (2018). Anomaly detection and fault analysis of wind turbine components based on deep learning network. *Renewable Energy*, 127, 825-834.