Applied Soft Computing 13 (2013) 259–270

Contents lists available at SciVerse ScienceDirect

Applied Soft Computing

journal homepage: www.elsevier.com/locate/asoc

Wind turbine condition monitoring based on SCADA data using normal behavior models. Part 1: System description

Meik Schlechtingen[a,*], Ilmar Ferreira Santos[b], Sofiane Achiche[c]

[a] Department of Technical Operation Wind Offshore, EnBW Erneuerbare Energien GmbH, Admiralitätstr. 4, 20459 Hamburg, Germany
[b] Department of Mechanical Engineering, Section of Solid Mechanics, Technical University of Denmark, Denmark
[c] Department of Mechanical Engineering, Machine Design Section, École Polytechnique de Montréal, Canada

ARTICLE INFO

Article history:
Received 9 December 2011
Received in revised form 3 May 2012
Accepted 8 August 2012
Available online 23 August 2012

Keywords:
ANFIS models
Condition monitoring
Wind turbine
SCADA data
Normal behavior models

ABSTRACT

This paper proposes a system for wind turbine condition monitoring using Adaptive Neuro-Fuzzy Interference Systems (ANFIS). For this purpose: (1) ANFIS normal behavior models for common Supervisory Control And Data Acquisition (SCADA) data are developed in order to detect abnormal behavior of the captured signals and indicate component malfunctions or faults using the prediction error. 33 different standard SCADA signals are used and described, for which 45 normal behavior models are developed. The performance of these models is evaluated in terms of the prediction error standard deviations to show the applicability of ANFIS models for monitoring wind turbine SCADA signals. The computational time needed for model training is compared to Neural Network (NN) models showing the strength of ANFIS in training speed. (2) For automation of fault diagnosis Fuzzy Interference Systems (FIS) are used to analyze the prediction errors for fault patterns. The outputs are both the condition of the component and a possible root cause for the anomaly. The output is generated by the aid of rules that capture the existing expert knowledge linking observed prediction error patterns to specific faults. The work is based on continuously measured wind turbine SCADA data from 18 turbines of the 2 MW class covering a period of 30 months.

The system proposed in this paper shows a novelty approach with regard to the usage of ANFIS models in this context and the application of the proposed procedure to a wide range of SCADA signals. The applicability of the set up ANFIS models for anomaly detection is proved by the achieved performance of the models. In combination with the FIS the prediction errors can provide information about the condition of the monitored components.

In this paper the condition monitoring system is described. Part two will entirely focus on application examples and further efficiency evaluation of the system.

1. Introduction

Condition monitoring of wind turbines is of increasing importance as the size and remote locations of wind turbines used nowadays makes the technical availability of the turbine very crucial. Unexpected faults, especially of large components, can lead to excessive downtime offshore due to lack of suitable crane ships or other specialized vessels. However, also smaller issues and faults of auxiliary equipment like pumps or fans can cause expensive turbine downtime due to restricted turbine accessibility. From an operator's point of view it is therefore worth increasing the effort spent to monitor the turbine condition in order to reduce unscheduled downtime and thus operational costs.

The available CMS mostly require high level knowledge about the system to be monitored. However, this knowledge is difficult to access and does often not exist. Physical models of the system to monitor its condition and predict failures can thus seldom be built with high accuracy due to its complex interaction among several dynamical subsystems. Moreover the available CMS mainly focus on vibrations. Vibration analysis is by far the most prevalent method for machine condition monitoring [2]. However, vibration sensors are not installed on all turbines and components due to their high costs. This causes a large number of turbines not being condition monitored at all or vibration sensors being installed at the main components only.

On the other hand, there is a large amount of operational (SCADA) data available, which can be used to give an indication of the turbine condition. This fact is also stressed by Yang and Jiang [3] who additionally point out that these data are the cheapest resource for developing a CMS for wind turbines. The operational data can be either turbine status information or measurements of signals such

* Corresponding author. Tel.: +49 40533268134.
E-mail address: m.schlechtingen@enbw.com (M. Schlechtingen).

1568-4946/$ – see front matter © 2012 Elsevier B.V. All rights reserved.
http://dx.doi.org/10.1016/j.asoc.2012.08.033

as temperatures, currents or pressures. Using turbine status information Kusiak and Wenyan [4] and Kusiak and Verma [5] show that faults can be predicted 5–60 min in advance. In order to perform predictive maintenance this prediction period is too short, since it does not leave the operator with enough time to take maintenance actions. By applying advanced signal analysis methods focused on trends of representative signals or combination of signals, significant changes in turbine behavior can be detected at an early stage [6]. Using Neural Network (NN) model based approaches Sanz-Bobi et al. [7], Zaher et al. [1] as well as Schlechtingen and Santos [8] show that changes in signal behavior can be detected, days, weeks and in some cases months in advance. These methods are therefore better suited to allow operators to take measures to improve the condition before the component finally fails. In the model based approaches historical operational data is used to develop normal behavior models capable of predicting a certain output signal, when given one or more input signals. For wind turbine signals these attempts are well suited, since many signals can be found to be correlated to other signals measured simultaneously, e.g. the wind speed or the power output.

One advantage of using normal behavior models to monitor wind turbine signals is that no prior knowledge about the signal behavior is needed. Another important property is that with normal behavior models the possibility of monitoring the signal is widely decoupled from the operational mode as reported by Zaher et al. [1] and Sanz-Bobi et al. [7]. The normal behavior models are developed in periods where the turbine components can be considered healthy (normally operating), usually the period at the beginning of the component lifetime. Afterwards, the trained model is used to predict a specific signal where the prediction error gives an indication of changes in signal behavior and thus incipient faults.

This approach is of large interest in research. In [9] the applicability of a system identification approach using an AutoRegressive with eXogenous input (ARX) model to monitor the condition of a wind turbine generator bearing using SCADA data is shown. However, this approach requires human intervention for parameter selection to find a good performing model. Due to the large amount of signals and turbines to be monitored, human intervention should be avoided. Most activities take advantage of artificial intelligence algorithms (learning capabilities) and among the most advanced systems using this approach is SIMAP [7] and a Multi Agent System (MAS) [1,11]. Both systems use artificial NN to set up the normal behavior models of SCADA data. This line is also followed by Xiang et al. [13], where a NN model based method to monitor the condition of a wind turbine generator bearing is developed and the good performance of NN in this context shown. Additionally, earlier research activities by Schlechtingen and Santos [8] and Schlechtingen [10] proposed a condition monitoring system for wind turbine drive train components using NNs.

Instead of using NNs, ANFIS is used in this paper, which presents a novelty, in this field of application. ANFIS models can learn nonlinear signal relations by setting up a set of fuzzy rules and tuning the Membership Function (MF) parameters in a training phase. Jang [14] shows that in comparison to NN models, fewer parameters must be trained in ANFIS models, leading generally to faster training. The major drawbacks of NNs are their black-box data processing structure and slow convergence speed [15]. Furthermore, a priori knowledge about the system is difficult to be incorporated. Here ANFIS models have a major advantage, due to their output being based on linguistic rules and tuneable MFs.

The monitoring systems developed by the different researchers can furthermore be classified according to the input signals used for modeling. While Sanz-Bobi et al. [7], Zaher et al. [1] as well as Zaher and McArthur [11] use autoregressive approaches to set up the normal behavior models, the research presented by Schlechtingen and Santos [8] showed that it is advantageous using a Full Signal ReConstruction (FSRC) approach, since this approach additionally allows for monitoring the signal value magnitude. The different approaches mainly concern the question of the used input signals to model the target signals. This approach is also followed by the research presented in this paper.

To interpret the prediction error of the normal behavior models, a fuzzy expert system that outputs a diagnosis, the condition and the certainty of statement based on rules that were established by an expert is employed by Sanz-Bobi et al. [7]. With regard to the surveyed gearbox faults in their research, this approach proved to be successful, since the rules established with fault patterns from one turbine were also applicable to predict the same faults on other turbines.

The research presented in this paper is the result of two years investigations carried out to develop a CMS that uses wind turbine SCADA data available to wind turbine operators. Thereby the advances made by Sanz-Bobi et al. [7] as well as Zaher and McArthur [11] are combined and further developed.

The development of the aforementioned CMS follows the three step strategy below:

1. Normal behavior models of the relevant SCADA data were developed in order to monitor and detect anomalies by considering the prediction error.
2. Occurred anomalies within the prediction error were related to reported faults.
3. The identified relations were implemented in knowledge data bases which are used by FIS to automatically analyze these patterns and output a diagnosis.

The SCADA data used in this research is obtained from eighteen different operating onshore turbines of the 2MW class, where continuous operational data from April 2009 to March 2012 were gathered.

In Section 2 of this article the general concept of the system developed is described giving details about how the information (SCADA data) is processed to finally output condition statements. A brief description of the advantages of ANFIS models over NN models for the given application and a short description of the used model setup and structure is supplied in Section 3. Section 4 focuses on the performance of the developed normal behavior models and describes the identified input–output sets. An example for the prediction error analysis and the model interactions is given in Section 5. In Section 6 the concept of anomaly detection is shown and a definition of abnormal signal behavior is given. Section 7 clarifies how the prediction error together with the information about detected anomalies is processed by the Fuzzy Interference System (FIS) working as fuzzy expert. The output is a diagnosis about the component condition and a potential root cause of the anomaly. Results are discussed in Section 8 and conclusions are drawn in Section 9.

2. Description of the general CMS concept

The CMS developed in this research aims at detecting trends and patterns in SCADA data in order to predict possible failures, giving wind turbine operators enough time to adapt the maintenance schedule or take further measures to prevent unexpected system downtime. For this purpose 10 min averaged SCADA data are used that are commonly available to operators. The general architecture of the CMS developed is shown in Fig. 1.

In the following, the function of the different CMS modules (see Fig. 1) is briefly described.

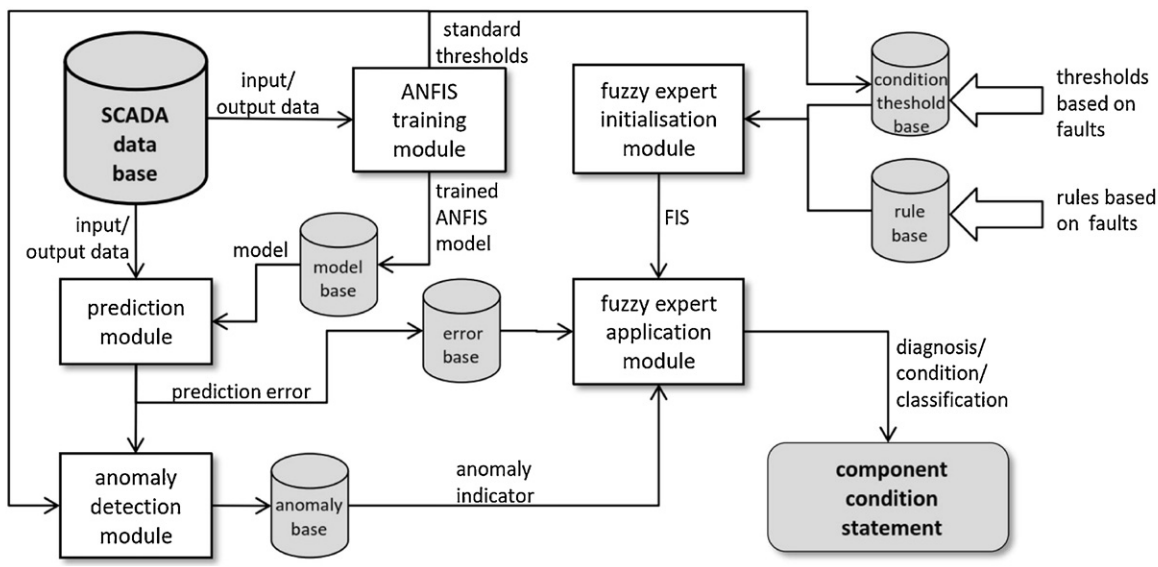

Fig. 1. CMS overview.

2.1. Training module

Within the training module the normal behavior model is trained if a model is not yet available or new training is required. The latter is true if a component is replaced and the signal relations change as a consequence. Before training the model, the data are preprocessed according to the methodology proposed in [8] which includes: (1) a validity check, (2) data range check, (3) missing data processing and (4) lag removal.

In the training module different training levels are implemented to allow early monitoring. A first model training is performed after one month of operational data collection. Further trainings are performed when three, six and nine months of data are available. The output of the training module is the trained ANFIS model and the standard thresholds marking the normal operational range of the wind turbines using the prediction error.

2.2. Prediction module

The prediction module is active once a trained model of the processed signal is available in the model base. The developed normal-behavior model is applied and the prediction error calculated and stored.

2.3. Anomaly detection module

In this module the anomalies in the prediction errors are identified. This is done on the basis of the determined normal-behavior thresholds by the training module or expert defined values. The output is an anomaly matrix containing information about the frequency and date of occurrence, as well as the duration of the current anomaly in days.

2.4. Fuzzy expert initialization module

Here the FIS structures used for diagnosis of the anomalies and component condition statements are initialized with regard to the number of inputs and outputs as well as their MFs. Each component to be monitored has its own FIS structure.

While the inputs sorely depend on the component or subsystem to be diagnosed, each FIS structure has the following outputs:

- diagnosis (information about the abnormal behaving signal)
- condition (classification in green, yellow and red color code)
- potential root cause

2.5. Fuzzy expert application module

Within this module the initialized FIS structure is evaluated, given the prediction errors and the information about present anomalies. The output is stored in text format and is visualized to give the analyst a comprehensive summery of the turbine condition.

3. Model setup and structure

3.1. Data set description

The available SCADA data sets from the operating wind turbines contain more than 150 different signals, ranging from hour counters, calculated values, digital indicators of switch positions and set points, to continuous measurements of temperatures, currents, voltages, etc. For some of the continuous measurements the mean, maximum, minimum and standard deviation of the 10 min averaging period is available. In this research only normal behavior models for mean values were considered for three reasons: (1) The peaks stored in the min.–max. values can be caused by transient situations e.g. sudden changes in wind speed that are not in the scope of this research; (2) stochastic effects in the signals are averaged out in the 10 min period, making the prediction less sensitive to stochastic variations; (3) modeling the standard deviations requires representative signals, i.e. other standard deviations or higher resolution time series, that are not accessible.

The remaining signals considered for the CMS are given in Table 1 and a schematic of the turbine and of the sensor location are illustrated in Fig. 2.

In addition to these signals, information about service and non-operational periods is extracted from the corresponding hour

Table 1
SCADA data signal list used for normal behavior model development.

Name of variable	Unit	Sensor location (Fig. 2)	Short description	Normal behavior modeled
Spinner temp.	°C	1	Spinner temperature (in hub housing)	Yes
Hub controller temp.	°C	1	Pitch controller temperature	Yes
Pitch angle	°	1	Blade pitch angle (avg. over all 3 blades)	Yes
Hydraulic oil temp.	°C	10	Hydraulic oil temperature (used for pitching blades)	Yes
Rotor speed	rpm	2	Rotor speed (low speed shaft)	Yes
Gear bearing temp. (HSS)	°C	3	High speed shaft bearing temperature	Yes
Gear oil temp	°C	3	Gearbox oil temperature	Yes
Generator speed	rpm	4	Generator speed (high speed shaft)	Yes
Generator bearing temp.1	°C	5	Generator bearing temperature gearbox end	Yes
Generator bearing temp. 2	°C	5	Generator bearing temperature transformer end	Yes
Generator slip ring temp.	°C	5	Generator slip ring temperature	Yes
Generator ph.1 temp.	°C	5	Generator stator temperature phase 1	Yes
Generator ph.2 temp.	°C	5	Generator stator temperature phase 2	Yes
Generator ph.3 temp.	°C	5	Generator stator temperature phase 3	Yes
Generator current ph.1	A	6	Generator current phase 1	Yes
Generator current ph.2	A	6	Generator current phase 2	Yes
Generator current ph.3	A	6	Generator current phase 3	Yes
Power output	kW	6	Turbine power output	Yes
Reactive power	kVAr	6	Turbine reactive power consumption	Yes
Grid inverter ph.1 temp.	°C	6	Inverter temperature grid end	Yes
Grid rotor inverter ph.1 temp.	°C	6	Inverter temperature phase 1 generator end	Yes
Grid rotor inverter ph.2 temp.	°C	6	Inverter temperature phase 2 generator end	Yes
Grid rotor inverter ph.3 temp.	°C	6	Inverter temperature phase 3 generator end	Yes
Converter cooling water temp.	°C	6	Converter cooling water temperature	Yes
Converter choke coil temp.	°C	6	Converter choke coil temperature	Yes
Converter controller temp.	°C	6	Converter controller temperature	Yes
Top controller temp.	°C	6	Turbine controller temperature	Yes
Grid busbar temp.	°C	8	Busbar temperature	Yes
HV transformer ph.1 temp.	°C	8	High voltage transformer temperature phase 1	Yes
HV transformer ph.2 temp.	°C	8	High voltage transformer temperature phase 2	Yes
HV transformer ph.3 temp.	°C	8	High voltage transformer temperature phase 3	Yes
Nacelle temp.	°C	7	Temperature in nacelle (housing) of the turbine	Yes
Wind speed	m/s	9	Wind speed	Yes
Wind direction	°	9	Wind direction	No
Ambient temp.	°C	9	Outdoor temperature	No

counters. This information is used for preprocessing the data according to the methodology presented in [8] (Fig. 3).

3.2. Model type

Several researchers applied NN to set up normal behavior models in the field of condition monitoring. Some of the key features of NN are their high processing speeds which are due to their massive parallelism, their proven ability to be trained, to produce instantaneous and correct responses from noisy or partially incomplete data, and their ability to generalize information over a wide range [16]. In earlier publications of the authors [8,10] it was shown that NN are indeed capable of accurately finding an existing input–output mapping for different wind turbine SCADA signals. However, it was found that NN have a high likelihood of becoming tapped in local minima. In Fig. 4 three examples of NN performances over the number of runs are shown. For this example two hidden neurons per input signal are used.

The variations in performances are up to 20%, stressing the risk of local minima. Furthermore the number of hidden neurons is generally unknown. Tarrassenko [17] and Rafiq et al. [18] suggest performing several runs with random weight initializations and runs with a varying number of hidden neurons to overcome this shortfall. Finally the network that performs best should be chosen. This leads to a large number of different trials to obtain an acceptable solution.

Fuzzy systems are very useful in two general contexts: (1) in situations involving highly complex systems whose behaviors are not well understood and (2) in situations where an approximate, but fast, solution is desired [19]. A further advantage of fuzzy systems is that existing expert knowledge can be implemented to improve

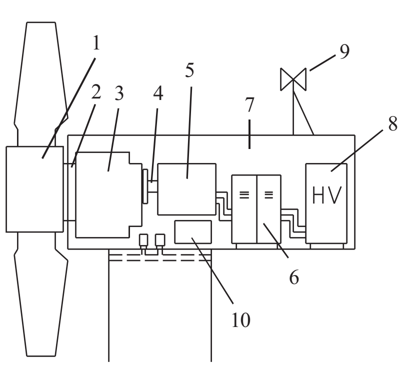

Fig. 2. Wind turbine schematic: sensor positions.

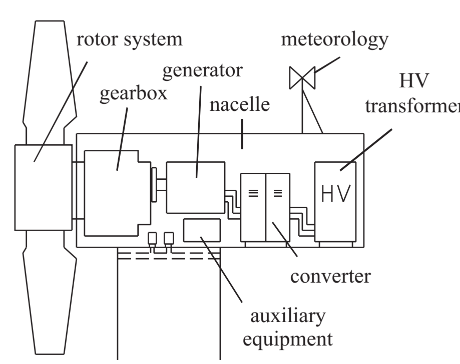

Fig. 3. Wind turbine schematic: components/subsystems in the considered wind turbines.

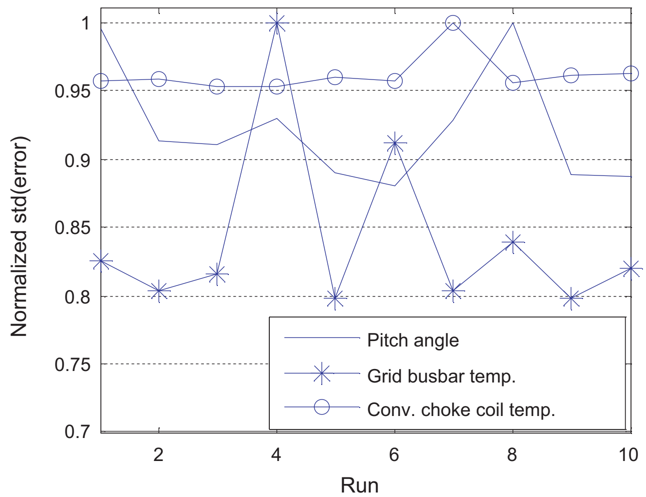

Fig. 4. Exemplary performances of trained NN's over different runs.

the approximation by tuning, removing or adding of membership functions and rules.

Considerable work has been carried out to integrate the learning capability of NN with FIS for deriving the initial rules of a fuzzy system and tuning the membership functions [20]. Fuzzy neural networks have shown to be very advantageous in dealing with real-world problems [21]. These neuro-fuzzy systems combine the benefits of these two powerful paradigms into a single capsule. This gives the ability to accommodate both data and existing expert knowledge about the problem under consideration [20].

In ANFIS the advantages of NN are combined with FIS. Thereby the FIS is used to set up a set of rules whose membership functions parameters are tuned in a training phase.

In the following sections the setup and structure of the ANFIS models are described.

3.2.1. FIS structure

There are two common types of fuzzy interference: the Mamdani [22] and Sugeno [23]. The main difference between the two types is in the consequent part, which can be a nonfuzzy equation (Sugeno) instead of a fuzzy linguistic value (Mamdani). In this work the Sugeno type is used as it has fuzzy sets only involved in the premise part. However, the consequent part can be a nonfuzzy equation. Due to the qualifiers on the premise parts, each fuzzy if–then rule can be viewed as a local description of the system under consideration [14]. Moreover the Sugeno type is known to be more computationally efficient and has guaranteed continuity of the output surface, which makes it well suited for time series prediction.

3.2.2. Membership functions

Membership functions can be represented by an arbitrary curve whose shape is defined as a function that suits the application. In most fuzzy applications straight-line membership functions are used (triangular/trapezoidal). In this paper generalized normal distribution membership functions are employed for the input space and linear MFs for the output space. The generalized normal distribution functions have the advantage of giving a broad flexibility with regard to the function shape, depending on their parameters. Furthermore these functions assure smoothness of the transitions in the input space. The free parameters are: location, scale and shape.

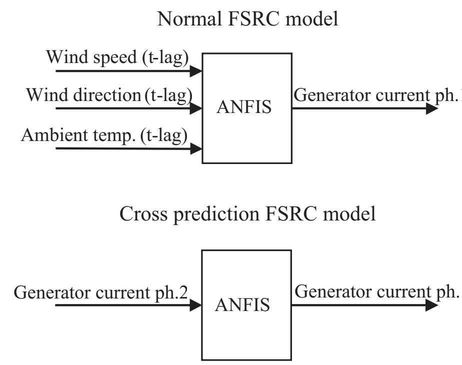

Fig. 5. Comparison between a normal FSRC model and the cross prediction FSRC model.

3.2.3. Number of membership functions/rules

In a conventional FIS, the number of rules is decided by an expert who is familiar with the system to be modeled [14]. Generally, the number of MFs can be defined for each input separately. In this research the expert for this particular task is not available, and the number of membership functions per input is fixed to two, as this, after preliminary tests, was found to be a good compromise between computational efficiency and model performance.

3.2.4. Training method

The learning algorithm used is a hybrid learning rule as proposed in [14], consisting of a combination of gradient decent and least squares estimation. Not only can this hybrid learning approach decrease the dimension of the search space in the gradient method, but in general, it will also cut down substantially the convergence time [14].

3.2.5. Input signals

The number of input signals differs for each signal to be monitored. The choice of relevant input signals is not trivial and requires a combination of physical understanding of the system to be modeled and advanced data reduction techniques. In this research a genetic algorithm combined with a partial least squares regression (GAPLS) [24] is used to detect potential input signals from the large pool of SCADA data. For this purpose the genetic algorithm toolbox supplied by Leardi [25] is applied. However, GAPLS points to the input signals containing most of the target signal features, which may not be the best ones to choose from a condition monitoring point of view if one considers the physical understanding of the phenomenon at hand. An example for possible difficulties with automatic input signal selections is the generator phase currents for which the GAPLS suggests using the generator current ph.1 as input to model generator current ph.2. A model like this is very accurate but by using this input–output set, monitoring of the absolute current level is impossible. Instead only differences between the phases can be detected.

Hence, a combination of data reduction techniques and the understanding of the physical process to select suitable input sets is the preferred strategy used in this paper.

Since both, the information about relative changes between signal readings of the same type as well as their absolute level is important, two types of models are developed. The first is a FSRC (Full Signal ReConstruction) approach, where the target signal is modeled by fully reconstructing it through correlated signals of different types and the second is by fully reconstructing it through correlated signals of the same type. The difference is illustrated in Fig. 5 for the generator phase current.

With the cross prediction FSRC models, asymmetries (for instance differences in generator phase currents or temperatures of

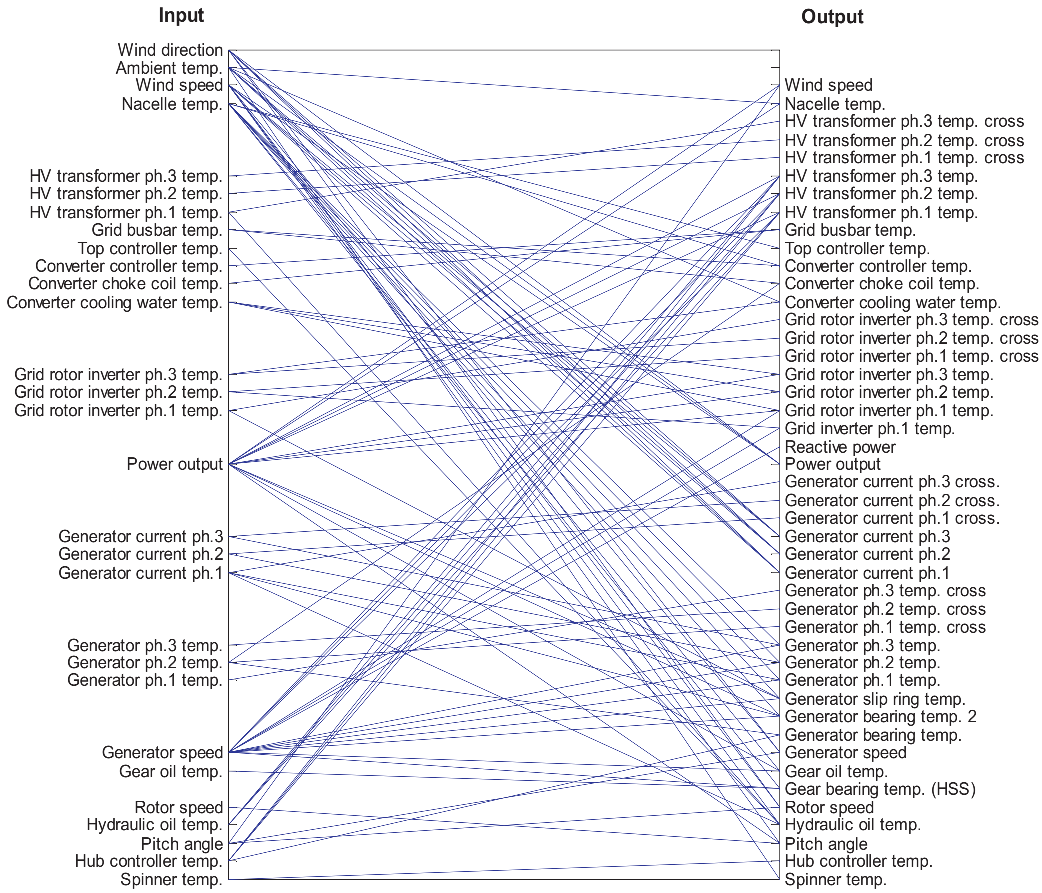

Fig. 6. Input–output sets of the developed models.

the three phases) can be identified very accurately whereas the normal FSRC model is used to monitor the absolute level and relative changes in comparison to other signals.

However, the model accuracy is not the only criterion for choosing input signals when developing normal behavior models for condition monitoring purposes. A further requisite is the fault visibility, as being the visibility of the developing fault in the prediction error. The relation between the fault visibility and the input signal choice is difficult to define and not exactly known, which is why expert knowledge is required to estimate this influence. A general requirement of normal behavior models is that they should have a low prediction error in case of normal behavior and a high prediction error otherwise. For the given problem the combination of the genetic algorithm with engineering knowledge about the component or subsystem lead to the input-output sets illustrated in Fig. 6 for the 45 models developed.

4. Model performance

A good model performance is the direct result of the correct choice of input signals and successful training. There are several ways the performance can be measured. Due to the different nature of the modeled signals (power output, currents, temperatures) a performance comparison between them is not meaningful. What is more of interest is the prediction error variation around its mean, giving an indication of how good the prediction is, by keeping the physical unit. Here the standard deviation is used. Since the prediction errors after training are close to being normal distributed and centered about zero this choice is acceptable (compare Fig. 9). The values are calculated based on test data sets that were not used for training.

$$E = x_m - x_p \tag{1}$$

$$P = \sigma(E) \tag{2}$$

E is the prediction errors; x_m is the measured values; x_p is the predicted values; P is the performance measure; σ is the standard deviation.

In order to assess the performance information about the normal operational range of the signals is essential (see Table 2). For instance a model performance of 1 °C is good when the normal operational range is −100 to 100 °C, but rather poor if the normal operational range is 25–30 °C.

4.1. Comparison between NN and ANFIS models

To stress the advances of ANFIS models the results of a brief performance and training speed comparison is shown in Table 3 for ANFIS and NN training. The chosen NN training procedure is analog to the one described in [8], but with the difference that here only five runs with random weight initializations are performed. The number of hidden neurons and the number of MFs, per input and output signal is set to two (Table 4).

The training is performed with nine month of operational data (29,513 10 min average values).

The comparison shows that the performance in terms of the standard deviation of both approaches is similar. However, the time required for training is generally smaller with ANFIS models, which is due to the necessary trial and error procedure when training NN.

Table 2
Normal operational ranges of the SCADA data.

Signal	Normal range	Signal	Normal range
Spinner temp.	−10 to 45 °C	Power output	−50 to 2100 kW
Hub controller temp.	−5 to 50 °C	Reactive power	−25 to 300 kWAr
Pitch angle (avg. over all blades)	−5° to 95°	Grid inverter ph.1 temp.	25 to 50 °C
Hydraulic oil temp.	10 to 65 °C	Grid rotor inverter ph.1 temp.	25 to 55 °C
Rotor speed	0 to 16 rpm	Grid rotor inverter ph.2 temp.	25 to 55 °C
Gear bearing temp. (HSS)	30 to 80 °C	Grid rotor inverter ph.3 temp.	25 to 55 °C
Gear oil temp.	30 to 65 °C	Converter cooling water temp.	5 to 50 °C
Generator speed	0 to 1700 rpm	Converter choke coil temp.	5 to 120 °C
Generator bearing temp. 1	10 to 85 °C	Converter controller temp.	5 to 60 °C
Generator bearing temp. 2	10 to 85 °C	Top controller temp.	10 to 50 °C
Generator slip ring temp.	5 to 45 °C	Grid busbar temp.	5 to 60 °C
Generator ph.1 temp.	10 to 140 °C	HV transformer ph.1 temp.	20 to 95 °C
Generator ph.2 temp.	10 to 140 °C	HV transformer ph.2 temp.	20 to 95 °C
Generator ph.3 temp.	10 to 140 °C	HV transformer ph.3 temp.	20 to 95 °C
Generator current ph.1	0 to 1700 A	Nacelle temp.	5 to 50 °C
Generator current ph.2	0 to 1700 A	Wind speed	0 to 35 m/s
Generator current ph.3	0 to 1700 A		

Jang [14] showed that in ANFIS models fewer parameters must be trained, leading generally to faster training.

The performance of ANFIS models does not vary among different runs, which is due to the initialization of the MFs, involving a grid partition algorithm [14,26]. What is important to notice is that with NN and the suggested number of different trials, a better performance can be achieved (see Fig. 7) potentially. However, finding an optimal solution is computational expensive. Fig. 7 further shows that with ANFIS models a good compromise between training speed and model accuracy can be found.

NN are good in identifying non-linear relations, but their output is difficult to back-track (black box model). A priori knowledge about the system is difficult to in cooperate. Here ANFIS models have a major advantage. Not only can a priori knowledge be used in terms of rules or shape of MFs, the output can also be back-tracked by using the information about the rule that fired. When using the grid portioning method, the number of rules is equal to: $\mathrm{nMF}^{\mathrm{ni}}$ with nMF being the number of MFs and ni the number of inputs. The number of inputs is smaller or equal to four and the number of MFs is equal to two for the models developed, leading to a maximum of 16 rules. Especially for small numbers of inputs this leads to very fast to train and simple to back-track models. However, the model complexity (number of rules) scales exponentially with the number of inputs, which causes training speed also increase exponentially.

Table 3
Performance and speed of ANFIS training.

SCADA signal	ANFIS	
	Elapsed time for training	*P*
Hydraulic oil temp.	125.2 s	2.00 °C
Generator bearing temp.	37.3 s	1.30 °C
Power output	93.6 s	45.78 kW
Top controller temp.	14.1 s	1.24 °C

Table 4
Performance and speed of NN training.

SCADA signal	NN	
	Elapsed time for training	*P*
Hydraulic oil temp.	452.3 s	2.06 °C
Generator bearing temp.	373.2 s	1.33 °C
Power output	284.9 s	45.99 kW
Top controller temp.	369.3 s	1.57 °C

4.2. Performance of the set up ANFIS models

Schlechtingen and Santos [8] showed that averaging the prediction error decreases the variance of the system and consequentially increases the sensitivity against anomalies. In fact it was shown that by averaging, upcoming faults and anomalies can be detected earlier and with a higher certainty (fault patterns are stronger pronounced). Generating average values can be advantageous, because daily averaged signals of several turbines can easily be compared within a single plot and turbine operators are most often interested in information that can support their day-to-day planning of necessary actions, for which not too much details (such as hourly data) are required [27]. In this paper all analysis is based on averaged prediction errors. The averaging period used is one day, i.e. the average for 144 individual prediction errors is calculated, giving one single averaged prediction error per day.

$$P_{avg} = \sigma(\bar{E}) \tag{3}$$

$\bar{E}$ is the averaged prediction errors; P_{avg} is the performance measure errors; $\sigma(\bar{E})$ is the standard deviation of $\bar{E}$.

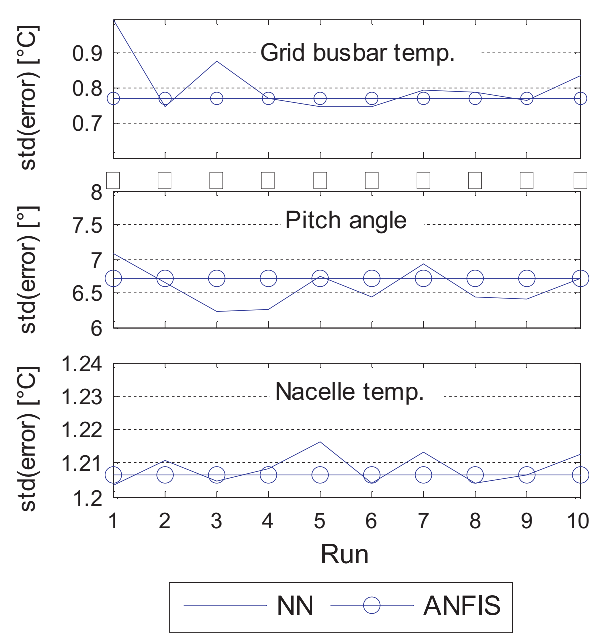

Fig. 7. Performance of different NN and ANFIS trainings over the number of runs.

Table 5
Example model performance in terms of standard deviation of the prediction error after training.

Model	P	P_{avg}	Unit	Model	P	P_{avg}	Unit
Spinner temp.	1.55	0.63	°C	Power output	47.00	16.34	kW
Hub controller temp.	0.55	0.18	°C	Reactive power	0.26	0.03	kWAr
Pitch angle (avg. over all blades)	0.44	0.08	°	Grid inverter ph.1 temp.	0.45	0.13	°C
Hydraulic oil temp.	2.59	1.35	°C	Grid rotor inverter ph.1 temp.	0.42	0.14	°C
Rotor speed	0.21	0.09	rpm	Grid rotor inverter ph.2 temp.	0.42	0.15	°C
Gear bearing temp. (HSS)	0.79	0.35	°C	Grid rotor inverter ph.3 temp.	0.41	0.13	°C
Gear oil temp.	2.09	1.02	°C	Grid rotor inverter ph.1 temp. cross.	0.39	0.07	°C
Generator speed	23.46	9.32	rpm	Grid rotor inverter ph.2 temp. cross.	0.48	0.13	°C
Generator bearing temp. 1	1.44	0.73	°C	Grid rotor inverter ph.3 temp. cross.	0.37	0.08	°C
Generator bearing temp. 2	1.86	0.75	°C	Converter cooling water temp.	0.69	0.33	°C
Generator slip ring temp.	1.30	0.46	°C	Converter choke coil temp.	2.95	1.26	°C
Generator ph.1 temp.	4.98	1.90	°C	Converter controller temp.	0.54	0.22	°C
Generator ph.2 temp.	4.92	1.87	°C	Top controller temp.	1.15	0.48	°C
Generator ph.3 temp.	4.92	1.86	°C	Grid busbar temp.	0.57	0.19	°C
Generator ph.1 temp. cross.	0.51	0.11	°C	HV transformer ph.1 temp.	3.59	1.64	°C
Generator ph.2 temp. cross.	0.51	0.09	°C	HV transformer ph.2 temp.	2.88	1.32	°C
Generator ph.3 temp. cross.	0.66	0.15	°C	HV transformer ph.3 temp.	3.25	1.47	°C
Generator current ph.1	39.07	13.62	A	HV transformer ph.1 temp. cross.	1.59	0.87	°C
Generator current ph.2	38.63	13.29	A	HV transformer ph.2 temp. cross.	1.39	0.80	°C
Generator current ph.3	39.05	13.57	A	HV transformer ph.3 temp. cross.	1.35	0.91	°C
Generator current ph.1 cross.	2.32	1.45	A	Nacelle temp.	1.21	0.60	°C
Generator current ph.2 cross.	2.37	1.72	A	Wind speed	0.20	0.09	m/s
Generator current ph.3 cross.	1.60	0.99	A				

The performances are listed in Table 5 for a random Wind Turbine Generator (WTG) of the fleet (the individual performance of each model differs from turbine to turbine).

The standard deviations decrease almost by a factor of three, when the predictions error is averaged.

In some cases it may be difficult to identify suitable inputs to achieve a good model performance. This is for instance if a signal is not found to be well correlated to any of the other signals simultaneously measured and thus an accurate FSRC model cannot be set up. Although this was not the case for the models developed, it should be mentioned that for those signals an alternative model approach is to build autoregressive models as proposed by Zaher et al. [1] and Sanz-Bobi et al. [7].

5. Prediction errors

The interpretation of the prediction errors for fault diagnosis requires some information about the validity or the trustworthiness of the input signals to the models. Abnormal behaving input signals, consequentially cause the prediction to be abnormal. Hence it is of utmost importance to take this dependency into account. Some input signals are used by a large number of models. Examples for excessively used input signals are the power output or the nacelle temperature. In turn that means that once these signals behave abnormal, a number of models will give bad predictions. This behavior is shown in Fig. 8, where a 2D waterfall plot of the normalized averaged percentage prediction errors over time is visualized. In this plot the colors indicate the prediction error amplitude. White areas mark periods where no prediction is available, e.g. due to missing data or non-operational periods. The prediction errors are normalized according to the following equation:

$$\bar{E}_{per}\quad (\%) = \frac{\bar{E}}{3\sigma(\bar{E})} \times 100 \tag{4}$$

$\bar{E}_{per}$ is the percentage error normalized by determined anomaly limit.

The color bar in Fig. 8 is set in a way that normally behaving signals are shown with a light green color. Only if the one day averaged prediction error exceeds the anomaly limit (see Section 6) other colors become visible. The color range indicates whether the prediction error is positive or negative as well as its amplitude. The prediction error of each model is shown in horizontal lines from the axis label on the left.

Fig. 8 shows that eight further models have abnormal prediction errors at the time the nacelle temperature increases. This effect influences the analysis of the root cause of the prediction error pattern. Furthermore it emphasizes the need of algorithms that take into account the context of the patterns. Here fuzzy logic is a powerful tool.

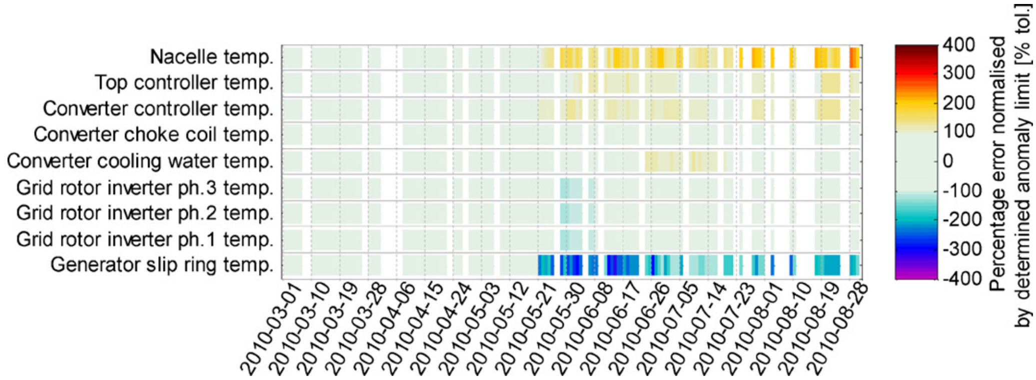

Fig. 8. 2D Waterfall plot of the normalized averaged prediction errors.

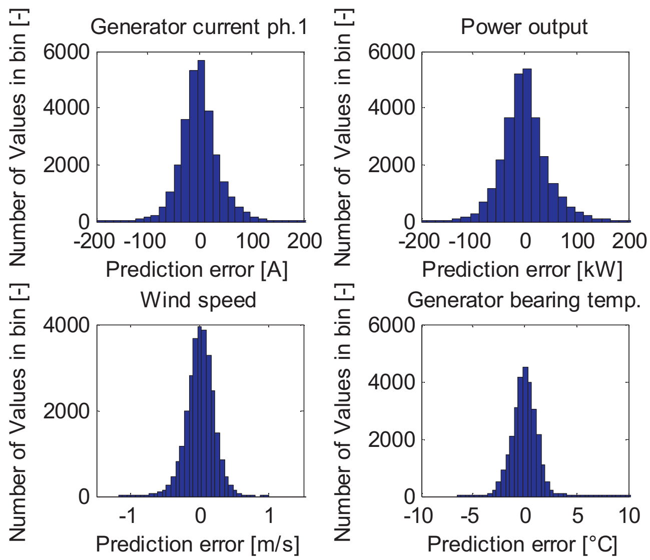

Fig. 9. Prediction error distributions after training.

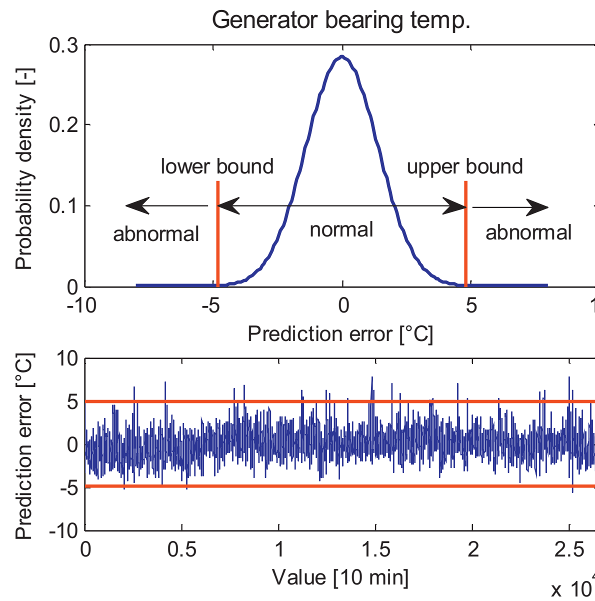

Fig. 10. Probability density distribution and evolution of the 10 min avg. prediction error (WTG 8).

6. Anomaly definition and detection

Anomaly detection refers to the problem of finding patterns in data that do not conform to expected behavior. These nonconforming patterns are among others often referred to as anomalies or outliers [28]. Detecting anomalies in the prediction error is one of the central issues of the condition monitoring approach addressed in this research. The task is to find an anomaly detection algorithm that highlights any unexpected patterns, once they exceed a certain limit. In 2009 Chandola et al. [28] reported a comprehensive review about anomaly detection algorithms. A majority of the reported techniques can be categorized into classification-based, nearest neighbor-based, clustering-based and statistical techniques [28]. In this paper the focus lies on parametric statistical techniques, because of nature of the prediction errors coming from successfully trained models that are usually normally distributed with a mean around zero. Statistical anomaly detection techniques are based on the assumption that normal data instances occur in high probability regions of a stochastic model, while anomalies occur in the low probability regions of the stochastic model [28].

Due to the availability of the prediction error in the training phase of the ANFIS models, the parameters of the underlying statistical process can be estimated. In the current CMS this is done by a maximum likelihood estimator, knowing that the parameters may be biased. Fig. 9 shows the prediction error distributions for a selection of models after training.

The estimated parameters are stored and used for estimation of the standard thresholds for anomaly detection. These thresholds form the upper and lower bounds of the normal range of the SCADA data. Instances that have a low probability of being generated from the trained model, based on the applied test statistic, are classified as anomalies [28]. For this purpose the standard thresholds are calculated according to the following probabilistic assumptions:

- Prediction errors that have a probability of being generated by the trained model larger or equal than 0.01% are considered normal
- Prediction errors that have a probability of being generated by the trained model smaller than 0.01% are considered an anomaly.

The choice of the probability influences the sensitivity of the system and can be adjusted if required. Here 0.01% was chosen to suppress false anomaly classification. Due to the large amount of turbines monitored later, the approach taken must not classify normal data as abnormal. False alarms may cause extra unscheduled maintenance visits, or cause the operator to lose faith in the system and subsequently ignore indications of real problems. Misdiagnosis can result in replacement of the wrong part and additional maintenance to correct such errors [29]. The upper and lower bound is calculated with regard to the original data resolution (10 min average) and the averaged values (one day averages). Fig. 10 shows an example of the boundary between normal and abnormal behavior, both in terms of the probability density function and the prediction error in time domain.

When averaged over one day, the predictions have a reduced variance and the upper and lower bound move closer to zero. In turn this leads to less alarm limit violations as visible in Fig. 11.

To further reduce the risk of false classifications, an additional threshold is set, that requires at least three values violating the

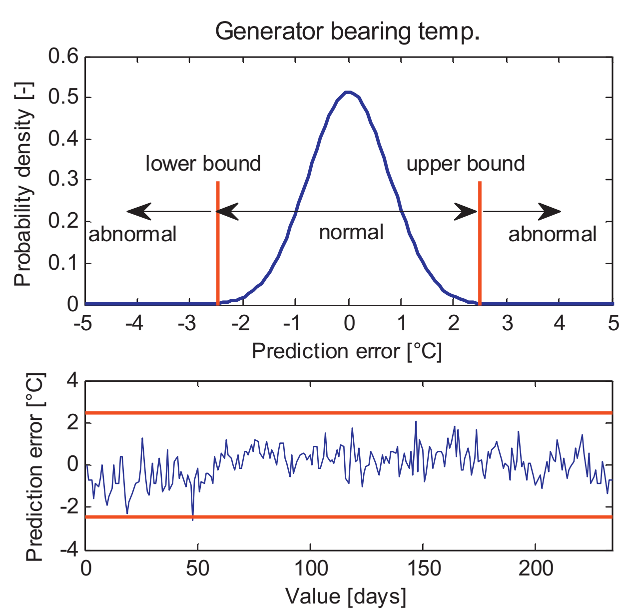

Fig. 11. Probability density distribution and evolution of the one day avg. prediction error (WTG 8).

probability threshold within a week, before the anomaly is analyzed by the fuzzy expert system.

7. Fuzzy expert system

The Fuzzy expert system consists of the fuzzy expert initialization module and the fuzzy expert application module which analyze patterns to classify the component conditions and output a potential root cause if a matching rule is found in the rule base. In total nine components or subsystems were defined, for each of which a FIS structure is developed. The components or subsystems are (see Fig. 3):

- Meteorology
- Rotor system
- Gearbox
- Generator
- Converter
- Transformer
- Auxiliary equipment
- Turbine performance
- Nacelle

7.1. FIS inputs

Only one FIS is developed per component or subsystem, hence the prediction error of several ANFIS models is processed by the FIS. The number of inputs to the FIS therefore depends on the number of models that exist to monitor the signals behavior of the component or subsystem. The relevant signals for the FIS are defined when setting up the FIS structure the first time, i.e. when the knowledge is implemented. However, the list of inputs is extendable, when rules implemented at a later stage require more inputs. The input signals are selected by considering the corresponding normal behavior model inputs and the physical understanding of the component or subsystem. For each input, a modified prediction error is passed to the input space of the FIS. The modified prediction error only contains the true prediction error value when an anomaly is present. The prediction error is set to a small number otherwise. Using the modified prediction error has the advantage that single values exceeding the anomaly limits do not cause false condition statements.

7.2. FIS membership functions

Considering which inputs are used, the fuzzy expert initialization module initializes the FIS with regard to the membership functions belonging to each modified prediction error input. Triangular and trapezoidal MFs are used in this research. A linear variation of the membership values is reasonable as long as no other dependency is known or desired. In 2008 Rodrígues and Arkkio [30] proposed a diagnosis method for detection of stator winding faults in induction motors based on fuzzy logic. The system was tested with triangular, trapezoidal and Gaussian MFs. It was found that the combination of triangular and trapezoidal MFs is the most appropriated for fault diagnosis in induction motors [30]. In comparison to trapezoidal MFs, triangular MFs require one less parameter to be defined, which is why triangular MFs are preferred in this research.

It proved useful in expert knowledge implementation to specify two separate FIS inputs per relevant prediction error. The first FIS input type is with five membership functions (MFs) allowing the classification into the categories: very low, low, normal, high and very high. The second FIS input type is with three MFs allowing classification into the categories: low, normal, high, giving the expert full freedom to implement the rules. The membership

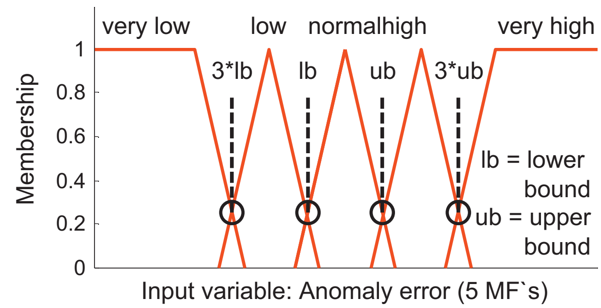

Fig. 12. Membership functions initialized for each input: FIS input type 1.

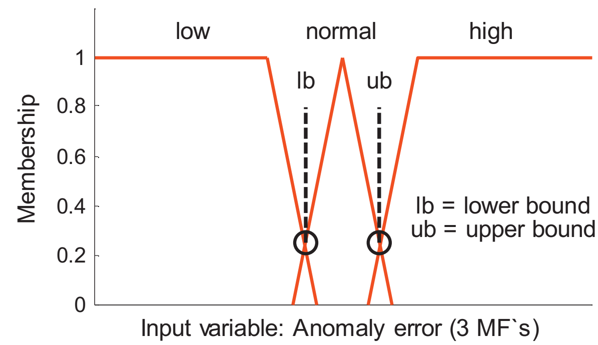

Fig. 13. Membership functions initialized for each input: FIS input type 2.

functions initialized for each of these inputs are visualized in Figs. 12 and 13.

The FIS structures are initialized each time an analysis is performed. This has the advantage that modifications of the structure inputs, rules and membership functions position and shape lead to fewer software conflicts. The definition of the statements, very low, low, normal, high and very high is given by the standard thresholds – the upper and lower bounds according to the 0.01% probability.

The standard thresholds do only serve as a first approximation for identification and classification of the component condition. These definitions are equal for all signals and turbines, but the values are individual, due to the anomaly definition given in Section 6.

In a master threshold table, the four bounds can be defined manually. This is usually done after fault occurrence, or after determination of the critical prediction error levels. Manual adoption is useful, since if for instance a model is very accurate, this consequentially also leads to tight standard upper and lower bounds. Hence the prediction error may have a high membership in the MF very high, although the prediction error level is still considered to be uncritical. In this case the thresholds can be adopted in the master condition threshold file to match the real fault progression. This procedure is required, since the real bounds are unknown for most components and signals before fault occurrence. The thresholds defined as master thresholds are valid for all turbines and replace the individual ones. It is therefore possible to identify similar patterns in the whole fleet, since it is expected that the general fault characteristics are similar for turbines of the same type.

The outputs of the FIS are diverse with regard to their MFs. One of the three outputs are initialized for each FIS equally. This concerns the initialization of the output ‘condition’, visualized in Fig. 14.

The MF gray is reserved for periods, in which no diagnosis is possible due to missing data, whereas green, yellow and red indicate the condition status.

- Green: component/subsystem working as expected; condition good; state ok

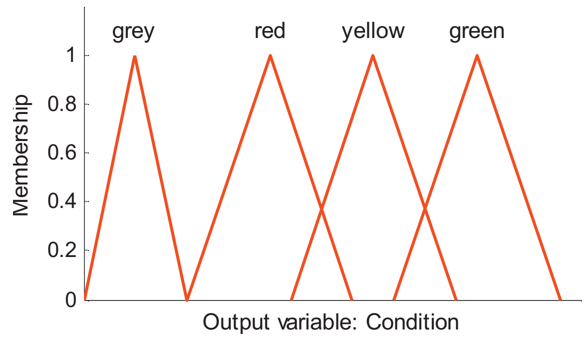

Fig. 14. Initialization of the membership functions of the variable condition.

- Yellow: service shall be scheduled for component/subsystem to improve the condition or investigate the issue; condition bad; state warning
- Red: service shall be performed at component/subsystem to improve the condition; condition very bad; state alarm

The other two outputs contain information about the diagnosis and the potential root cause. This information is individual to each FIS and is part of the knowledge that needs to be supplied by an expert. The setup is shown in Figs. 15 and 16.

7.3. FIS rules

The MFs are linked via rules that allow the FIS to output the diagnosis, condition and potential root cause. The rules are implemented by an expert manually. The concept of the rule base is that the base is constantly extended to allow the FIS to identify the various fault types accurately. The rule base is updated after fault occurrence and the validity of the already existing rules is checked. For each set of inputs, the FIS evaluates the input memberships according to their MFs and selects the rule that is most applicable. The consecutives are then given according to this rule. All rules currently implemented employ the “and method”, taking the minimum value of the MF evaluation for decision on the rule applicability. The rules can be classified into two categories:

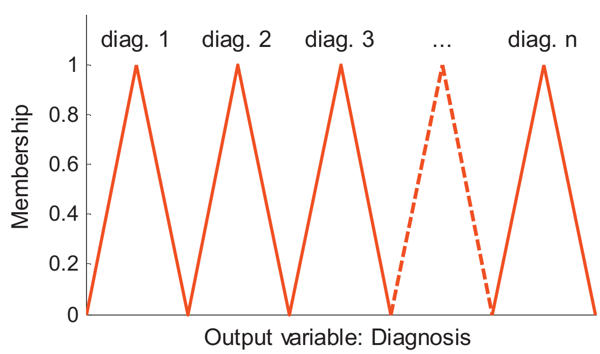

Fig. 15. Schematic of the membership function of the variables diagnosis.

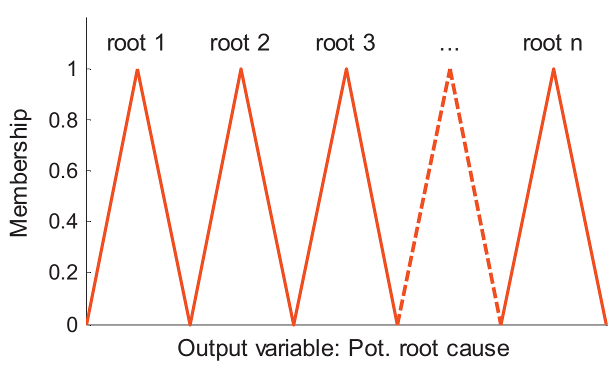

Fig. 16. Schematic of the membership function of the variables potential root cause.

- generic rules
- specific rules

The generic rules are used to highlight present anomalies, in case no specific rule applies. This rule type gives no information about the specific condition or potential root cause, but highlights to the operator that an anomaly is present in the data. The specific rules on the other hand provide this level of information. Two examples of implemented specific rules are:

(1) If (Anomaly error (5 MF’s) Spinner temp.==high) & (Anomaly error (3 MF’s) Nacelle temp.==ok) then (Diagnosis = Spinner temp. high) (Condition = yellow) (Pot. root cause = Sensor defect)
(2) If (Anomaly error (5 MF’s) Generator ph.1 temp. cross==high) & (Anomaly error (3 MF’s) Generator ph.3 temp. cross==low) then (Diagnosis = Generator Ph.1 temp. to high) (Condition = yellow) (Pot. root cause = Generator phase insulation damage)

The rules are implemented to automate the analysis of the one day average prediction errors.

As a basis for certainty statements of the diagnosis, the result of evaluating the output values through the MFs for the best applicable rule is used. This value can be seen as a matching indicator. A high value indicates that the rule represents well the input pattern, giving a high certainty in the diagnosis.

8. Results and discussion

The developed ANFIS models show good performances at a variety of different SCADA data (see Table 5), which proves their general applicability in this context. Moreover it is shown that these models are faster to train than NN models by achieving similar or better performances. The differences in training speed are significant, when taking into account that the current development counts 45 models per turbine which require training. This is, because the failure modes of the turbines are not known beforehand. An anomaly in any of the monitored SCADA signals may indicate a potential fault or malfunction. For the eighteen turbines accessible during this research this leads to 810 different ANFIS models, which require training.

The focus in this work is on one day average predictions in order to reduce the fluctuations in the prediction error. This does not only increase the fault visibility, but it also is expected to lead to fewer false alarms. False alarms are of major concern for wind turbine operators, due to the large number of operating systems. An accurate representation of the signal normal behavior builds the basis for the subsequent analysis of occurring patterns by the FIS structure set up. The overall performance levels achieved here are expected to allow early anomaly detection.

In the fuzzy expert system the overall number of rules can be drastically reduced by specifying two separate FIS inputs per relevant prediction error as proposed in this research. The FIS input type 1 is used for the signals directly linked to the component monitored, whereas type 2 may be used for all other inputs where the fine differentiation of the prediction error magnitude is not necessary. If for instance the gearbox bearing condition shall be determined by a specific rule, it is useful using type 1 for the gearbox bearing temperature to be able to classify the condition according to the prediction error magnitude (high and very high). For all other inputs, the fine differentiation may not be necessary, making FIS type 2 for those inputs more useful. This reduces the number of possible combinations within the rule.

9. Conclusions

In this paper a method to monitor wind turbine SCADA data, via normal behavior models and fuzzy logic is proposed. The methodology gives the possibility of mining large amount of SCADA data available to wind turbine operators in a systematic manner, search for anomalies and use this information for condition statements. It allows not only monitoring of large components, but also auxiliary equipment that currently available wind turbine CMS do not cover.

Using fuzzy logic the existing expert knowledge in anomaly/prediction error pattern interpretation and root cause diagnosis can be implemented in an intuitive manner. This gives the possibility for automated fault diagnosis once specific rules are implemented.

However, the applicability of the proposed method and therewith the achieved accuracies rely on the availability of a broad variety of different SCADA data to set up the ANFIS models. This criterion is usually fulfilled, which allows application of the proposed method to both existing older and new turbines. A further limitation concerns the availability of expert knowledge. In case no expert knowledge is available the methodology can only be used to highlight present anomalies in the SCADA data and only general statements about the condition and the diagnosis can be made.

Future research will focus on the implementation of other measurements, such as high resolution vibrational data or oil sample analysis in the CMS. This will open even more possibilities in implementing rules that capture the expert knowledge and will allow for larger diversity of the diagnoses. Furthermore turbines from a different type and brand will be included in the research.

In part two of the article the results of a field test will be presented by giving examples that show the applicability of the developed system based on real faults. It will be shown that the proposed method is capable of treating every turbine individually with regard to its characteristic. Additionally it will be shown that the FIS structures proposed allow generalizing rules that are valid for all turbines of the same type. This drastically reduces the development effort that is required to implement the expert knowledge.

References

[1] A. Zaher, S.D.J. McArthur, D.G. Infield, Online wind turbine fault detection through automated SCADA data analysis, Wind Energy 12 (2009) 574–593.
[2] R.B. Randall, Vibration-based Condition Monitoring, Wiley, Chichester, West Sussex, UK, 2011, ISBN 9780470747858.
[3] W. Yang, J. Jiang, Wind turbine condition monitoring and reliability analysis by SCADA information, IEEE (2011) 1872–1875.
[4] A. Kusiak, L. Wenyan, The prediction and diagnosis of wind turbine faults, Renewable Energy 36 (2011) 16–23.
[5] A. Kusiak, A. Verma, A data-driven approach for monitoring blade pitch faults in wind turbines, IEEE Transactions on Sustainable Energy 2 (1) (2011) 87–96.
[6] T.W. Verbruggen, Wind Turbine Operation & Maintenance based on Condition Monitoring, ECN-C-03-047, 2003.
[7] M.A. Sanz-Bobi, J. del Pico, M.C. Garcia, SIMAP: Intelligent System for Predictive Maintenance Application to the health condition monitoring of a windturbine gearbox, Computers in Industry 57 (2006) 552–568.
[8] M. Schlechtingen, I.F. Santos, Comparative analysis of neural network and regression based condition monitoring approaches for wind turbine fault detection, Mechanical Systems and Signal Processing 25 (5) (2011) 1849–1875.
[9] W.G. Garlick, R. Dixon, S.J. Watson, A model-based approach to wind turbine condition monitoring using SCADA data, in: 20th International Conference on Systems Engineering, 8–10 September, Coventry University, Coventry, UK, 2009.
[10] M. Süttmann (Schlechtingen), Master Thesis: Condition Monitoring in Wind Turbines – A Drive Train Monitoring System, 2010.
[11] A.S. Zaher, S.D.J. McArthur, A multi-agent fault detection system for wind turbine defect recognition and diagnosis, in: Proc. IEEE Lausanne POWERTECH, 2007, pp. 22–27.
[13] J. Xiang, S. Watson, Y. Liu, Smart monitoring of wind turbines using neural networks, Sustainability in Energy and Buildings Part 1 (2009) 1–8.
[14] J.S.R. Jang, ANFIS: adaptive-network-based fuzzy interference system, Transactions on Systems, Man and Cybernetics 23 (3) (1993) 665–685.
[15] X.Z. Gae, S.J. Ovaska, Soft computing methods in motor fault diagnosis, Applied Soft Computing 1 (2001) 73–81.
[16] K. Chandrashekhara, A. Chukwujekwu Okafor, Y.P. Jiang, Estimation of contact force on composite plates using impact-induced strain and neural networks, Composite Part B 29B (1998) 363–370.
[17] L. Tarassenko, Guide to Neural Computing Applications, Elsevier, Jordan Hill, Oxford, USA, 1998, ISBN 0340705892.
[18] M.Y. Rafiq, G. Bugmann, D.J. Easterbrook, Neural network design for engineering applications, Computers and Structures 79 (2001) 1541–1552.
[19] T.J. Ross, Fuzzy Logic with Engineering Applications, Wiley, Chichester, West Sussex, UK, 2010, ISBN 9780470743768.
[20] J. Kim, N. Kasabov, HyFIS: adaptive neuro-fuzzy interferance systems and their application to nonlinear dynamical systems, Neural Networks 12 (1999) 1301–1319.
[21] G.G. Yen, P. Meesas, An effective neuro-fuzzy paradigm for machinery condition health monitoring, IEEE Transactions on Systems, Man and Cybernetics – Part B 31 (4) (2001) 523–536.
[22] E.H. Mamdani, Application of fuzzy logic to approximate reasoning using linguistic synthesis, Transactions on Computers 23 (12) (1977) 1182–1191.
[23] M. Sugeno, Industrial Applications of Fuzzy Control, Elsevier Science Publishers, Upper Saddle River, New York, USA, 1985.
[24] R. Leardi, A.L. González, Genetic algorithms applied to feature selection in PLS regression: how and when to use them, Chemometrics and Intelligent Laboratory Systems 41 (1998) 195–207.
[25] R. Leardi The Pls-genetic Algorithm Toolbox for Matlab. http://www.models.life.ku.dk/GAPLS
[26] J.S.R. Jang, Neuro-Fuzzy and Soft Computing: A Computational Approach to Learning and Machine Intelligence, Prentice Hall, Upper Saddle River, New York, USA, 1997, ISBN 978-0132610667.
[27] E.J. Wiggelinkhuizen, et al., Conmow Final Report, ECN-E–07-044, 2007.
[28] V. Chandola, A. Banerjee, V. Kumar, Anomaly detection: a survey, ACM Computing Surveys 41 (3) (2009), 15:1–15:58.
[29] R.W. Hyers, J.G. McGrowan, K.L. Sullivan, J.F. Manwell, B.C. Syrett, Condition monitoring and prognosis of utility scale wind turbines, Energy Materials 1 (3) (2006) 187–203.
[30] R.V.J. Rodrígues, A. Arkkio, Detection of stator winding fault in induction motor using fuzzy logic, Applied Soft Computing 8 (2008) 1112–1120.