



HAL
open science

A combined mono- and multi-turbine approach for fault indicator synthesis and wind turbine monitoring using SCADA data

Alexis Lebranchu, Sylvie Charbonnier, Christophe Bérenguer, Frédéric Prevost

► **To cite this version:**

Alexis Lebranchu, Sylvie Charbonnier, Christophe Bérenguer, Frédéric Prevost. A combined mono- and multi-turbine approach for fault indicator synthesis and wind turbine monitoring using SCADA data. *ISA Transactions*, 2019, 87, pp.272-281. 10.1016/j.isatra.2018.11.041 . hal-01965733

HAL Id: hal-01965733

<https://hal.science/hal-01965733v1>

Submitted on 26 Dec 2018

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Distributed under a Creative Commons Attribution - NonCommercial - NoDerivatives 4.0 International License

A Combined Mono- and Multi-Turbine Approach for Fault Indicator Synthesis and Wind Turbine Monitoring Using SCADA Data

Alexis Lebranchu^{a,b}, Sylvie Charbonnier^{b,*}, Christophe Bérenguer^b, Frédéric Prevost^a

^a*Univ. Grenoble Alpes, CNRS, Grenoble INP, GIPSA-lab, F-38000 Grenoble, France*

^b*Valemo S.A.S, F-33323, Bègles, France*

Abstract

The monitoring of wind turbines using SCADA data has received lately a growing interest from the fault diagnosis community because of the very low cost of these data, which are available in number without the need for any additional sensor. Yet, these data are highly variable due to the turbine constantly changing its operating conditions and to the rapid fluctuations of the environmental conditions (wind speed and direction, air density, turbulence, ...). This makes the occurrence of a fault difficult to detect. To address this problem, we propose a multi-level (turbine and farm level) strategy combining a mono- and a multi-turbine approach to create fault indicators insensitive to both operating and environmental conditions. At the turbine level, mono-turbine residuals (i.e. a difference between an actual monitored value and the predicted one) obtained with a normal behavior model expressing the causal relations between variables from the same single turbine and learnt during a normal condition period are calculated for each turbine, so as to get rid of the influence of the operating conditions. At the farm level, the residuals are then compared to a wind farm reference in a multi-turbine approach to obtain fault indicators insensitive to environmental conditions. Indicators for the objective performance evaluation are also proposed to compare wind turbine fault detection methods, which aim at evaluating the cost/benefit of the methods from a production manager's point of view. The performance of the proposed combined mono- and multi-turbine method is evaluated and compared to more classical methods proposed in the literature on a large real data set made of SCADA data recorded on a French wind farm during four years : it is shown that it can improve the fault detection performance when compared to a residual analysis limited at the turbine level only

Keywords: Wind turbine monitoring, Wind farm monitoring, SCADA data, Fault detection, Condition monitoring, Performance evaluation

1. Introduction

Renewable energy has been a growing sector for several years, because of the necessity to reduce CO2 emission in the near future. The electrical power produced by wind turbines has been multiplied by 10 in the last 10 years. A significant amount of the electricity consumed in the world now relies on the electric power produced by wind farms, which have to be operational all along the year. Failures may cause important production losses, mainly due to the damages they cause and the time it takes to repair, which are no longer acceptable. This calls for a drastic change in maintenance solutions, which must switch from periodic and corrective to condition-based. One of the motivations, and a perspective, for the work presented in this paper is to develop a fault detection procedure that can deliver information on a developing fault early enough so that it can be used for condition-based or predictive maintenance decision making : indeed an in-advance detection allows the maintenance decision-maker to better plan the maintenance operations.

From the sensor technology point of view, 3 technologies are possible to monitor a wind turbine, [1]:

- Using high rate data from a Condition Monitoring System (CMS) monitoring e.g. vibration: the use of these data for fault detection usually relies on signal processing methods aiming at identifying the signature of a fault, e.g. in the signal spectrum;
- Medium rate 1-second SCADA data: these data can be used for fault detection using for example model-based fault detection and isolation (FDI) methods, [2];
- Slow rate 10-minutes averaged SCADA data: in this case, the measured quantities are averaged on 10-minutes length windows.

Several condition monitoring systems (CMS) are available on the market [3, 4, 5, 1]. They are based on vibration analysis for the most part. They require additional sensors to be placed on each nacelle to measure the vibrations of several components of the drive train. The highly sampled

*Corresponding author - Univ. Grenoble Alpes, CNRS, Grenoble INP, GIPSA-lab, F-38000 Grenoble, France - sylvie.charbonnier@gipsa-lab.grenoble-inp.fr

Email addresses: alexis.lebranchu@valemo.fr (Alexis Lebranchu),
sylvie.charbonnier@gipsa-lab.grenoble-inp.fr (Sylvie Charbonnier),
christophe.berenguer@grenoble-inp.fr (Christophe Béranger), frederic.prevast@valemo.fr (Frédéric Prevost)

acceleration signals they produce must be analyzed by an expert using signal processing methods, so as to detect faults in progress. Indeed, the automatic decision systems developed still generate a high number of false alarms. These make them a costly solution to implement [6, 7].

On the opposite, SCADA systems have been integrated in wind farms since the emergence of wind energy. They generate loads of cheap data containing useful information on the turbines state but the data low sampling rate, one average measure every 10 minutes, is not adapted to an accurate in-depth monitoring of wind turbines. However, their availability for free and the breadth of SCADA data scope have encouraged researchers to propose solutions to create SCADA based fault indicators [8, 1]. The work presented in this paper is developed for 10-minutes averaged SCADA, classically recorded for production supervision purposes.

SCADA data monitoring relies on the numerous variables measured mainly for production supervision purposes. Fault monitoring approaches based on SCADA data differ in the way the data can be merged to synthesize a fault indicator residual, and can be categorized in “mono-turbine methods” or “multi-turbine methods”, as proposed in [9]. Mono-turbine methods combine variables recorded on a unique wind turbine while multi-turbine methods combine variables coming from all the turbines within a wind farm or subset of a wind farm. In this work, it is proposed to use 10-minutes SCADA data in an innovative way, combining a mono-turbine (or turbine-level) and a multi-turbine (or farm-level) information processing.

Wind turbines SCADA variables are highly non-stationary because of the frequent changes in operational conditions and of the variations in external conditions. Following a classical diagnosis approach [2], mono-turbine methods aim at explaining a SCADA variable evolution as a function of other SCADA variables recorded on the same turbine and thus generate residuals (unexplained part) for diagnosis purposes. Variables may be linked by causal relations - a change in some variables induces a change in other variables- or by similarity relations –two variables evolve in the same way because they are submitted to the same excitations, [9]. The most common causal model in wind energy is the power curve [1], which expresses the link between the wind speed and the active power produced by the turbine. It can be used as a visualization tool by comparing the power curve built with data gathered during the current period with the reference curve provided by the constructor [10]. It can also be used to create fault indicators by measuring the difference between the produced power and its value predicted by a model using the wind speed. In order to reduce the dispersion of

the power data, additional input variables may be added such as the air density, the wind direction [11, 12, 8], the rotational speed or the pitch angle [13]. The model may be a simple polynomial approximation, a Gaussian process model or an artificial neural network [14]. The fault indicators may be the difference at each sampling time between the measured power and the expected power or it can be a distance between a reference power curve and an on line curve built with the most current measures [8]. The limitation of this approach comes mainly from (i) the length of the training period, which has to be large in order to cover all the wind speed and temperature ranges, and (ii) the difficulty to localize the fault once it is detected with the power curve. To address the localization issue, one solution is to split the wind turbine into smaller independent systems, such as the rotor, the gearbox, the generator, the transformer and the convertor and to build models of these reduced systems, [15, 16]. One popular variable to be modeled is the temperature of some components, whose variations can be explained by changes in the operational conditions or in the outside temperature. Models explaining the temperature variations use at least the produced power, the nacelle temperature and the train rotation speed as input variables. They are learnt on data measured when the turbine operates in normal conditions. The fault indicator is defined as the difference between the actual measured temperature and the expected temperature, named residual. Models may be static, i.e. they use data measured at one sampling time [17, 18] or dynamic, such as ARX models [19, 20]. They can be simple linear regression models or more complex artificial neural networks [21, 22, 23, 24, 25]. Let note at this point that several published works follow a classical model-based fault detection and isolation approach (model-based FDI) to build fault indicators sensitive to faults, but robust to disturbances, for wind turbine monitoring, [26, 27, 28, 29, 30]. In these works, the residuals are built using state observers or parity equations, [2]. The proposed methods are usually implemented on wind turbine benchmarks, emulated by differential equations models where different faults can be simulated. They use high frequency SCADA data, recorded every second. Though the results obtained are very interesting, the problem addressed in these works is very different in nature from the one addressed in the present paper where 10-minutes SCADA data are used and the methods are validated on real data.

Another approach is to model the temperature of one component as a function of the temperature of some other reference components that should evolve in a similar way, such as the temperatures of two bearings but also the bearing temperature and the stator temperature [18]

or the temperature of the hydraulic break and the bearing temperature [31]. Such models using similarity may be simpler than models using causal relations but they rely on two temperatures evolving in the same way.

Mono-turbine approaches merge variables from the same wind turbine to generate fault indicator residuals that are insensitive to changes in its operational conditions. However, these residuals remain sensitive to the variations in the external environment such as the wind direction, air humidity and so on. On the opposite, multi-turbines approaches merge variables recorded from different turbines of the same wind farm in order to reduce the influence of the environmental conditions. Indeed, turbines of the same farm are submitted to the same weather conditions so variables should evolve in a similar way, somehow. [32] compares the behavior of different turbines in the same farm using curves displaying the temperature of a drive train bearing as a function of the produced power. The temperatures and powers are measured during a period of time and the curves from all the turbines are plot on the same graph. Faults can be identified visually when one of the curves deviates from the others. [10] measures the difference between the power curve given by the manufacturer and an actual power curve built using data gathered during a current period of time. The difference between the two curves can exhibit a loss in performance. The differences measured for all the turbines in the farm are compared one with each other to detect a turbine with a larger loss of performance, which can be due to a fault. [33] compares the evolution in time of the temperature of a component normalized by the external temperature for turbines from the same farm. The deviation of the temperature of one turbine from the others can be the symptom of a fault. [17] builds residuals from the differences in the normalized temperatures and concludes that changes in operational conditions can create themselves fluctuations that are too large to allow for reliable fault detection.

The literature review shows that most authors proposed methods to build fault indicators using mono-turbine approaches and so get rid of the influence of the operating conditions on the fault indicators. Few authors adopt a multi-turbine approach that allows getting rid of the influence of the external conditions, and mostly as a visualization tool. A method to synthesize a fault indicator for each turbine in a farm by comparing the temperature measured on a turbine to a farm reference (average or median of temperatures measured on all the turbines within the farm) has been proposed in [34] and used in [35] on other types of measured SCADA variables. It has

been shown that such an indicator remains sensitive to the operating conditions, which can be different from one turbine to another. Hence, no solution able to get rid of the influence of both the operating conditions and external environment has been proposed thus far. Moreover, as stated by [36], in their extensive review on wind turbines condition monitoring using SCADA data, there is a lack of published performance metrics to properly evaluate the advantage of one method from the others in terms of false alarm, true failure prediction and normal behavior prediction. To address these issues and fill this gap, we propose in this paper a hybrid multi-level synthesis method to take benefits of both approaches -mono- and multi-turbine - and to build fault indicators combining the two approaches. At the turbine level, residuals obtained with a mono-turbine model learnt during a normal condition period are first calculated for each turbine. At the farm level, these residuals are compared to a wind farm residual reference, in a multi-turbine approach. The use of mono-turbine residuals enables the influence of the operating conditions to be reduced while the use of a wind farm reference enables the changes in the environmental conditions to be accounted for. The performance of the method proposed is evaluated and compared to methods proposed in the literature on a large data set made of SCADA data recorded on a French wind farm during four years. Objective performance evaluation metrics are also proposed to compare the methods, which aim at evaluating the cost/benefit of the methods from a production manager’s point of view. Fault indicators should warn of a progressing fault early enough for a maintenance operation to be scheduled. However, they should not generate false alarms with the extra cost of sending a maintenance operator team on a remote site for no reason.

The contribution of the work presented in this paper is then twofold:

- The first and main contribution of the hybrid multi-level approach proposed in this paper lies in the online real-time comparison of the prediction residual computed on one turbine (i.e. at the turbine level, using a turbine normal behavior model) to a farm-level reference prediction residual, computed using the prediction residuals from the neighbor wind turbines. This approach is said “hybrid multi-level” because it combines a mono-turbine (or turbine-level) step with a multi-turbine (or farm-level) step. To the best of the authors’ knowledge, no other existing and published method proposes to compare online in real-time the prediction residual generated on a given turbine to a farm reference synthesized from the prediction residuals generated on the neighbor turbines.

- Second, the other contributions of the work presented in this paper are (i) the definition of objective performance metrics for the fault detection consistent with the applicative requirements of a wind farm manager and (ii) using these performance metrics, the evaluation of the proposed fault detection approach on a 4 years real data base from wind farm with 6 turbines, which shows that it has better performance than a residual analysis limited at the turbine level only.

The outline of this paper is as follows. Section 2 first presents different possible methods for the generation of fault indicators and focuses on the proposed combined mono- and multi-turbine approach developed in this work. The objective performance metrics considered for the evaluation of the fault detection performance are also detailed in this section. The dataset used to analyze the performances is described in Section 3 followed by a presentation and a discussion of the results obtained.

2. Fault detection combining turbine-level and farm-level information

This section presents the proposed two-levels methodology to synthesize fault indicators by combining both a mono-turbine and a multi-turbine approach. Each of the two levels is described : synthesis of the fleet reference at the farm-level, and elaboration of a relevant turbine-level variable for comparison to the fleet reference. Finally, the performance evaluation metrics, which are used latter in Section 3 to compare the obtained fault detection results with different fault indicators, are introduced.

2.1. Fault indicators synthesis

2.1.1. Farm-level : comparison of the turbine variables to a fleet reference

In wind farms, turbines are part of a fleet : they are of the same make and are subject to the same environmental conditions (wind speed, external temperatures, ...). Thus, the SCADA variables monitored on turbines from the same farm should evolve in a similar way when the turbines operate under normal conditions. The rationale for the method proposed in this paper relies on this assumption. SCADA variables from different turbines are compared on line. A deviation between a variable computed from a turbine and a farm reference is an indicator of an abnormal behavior. Its general concept is presented in Figure 1.

[Figure 1 about here.]

SCADA variables or model residuals computed from SCADA variables are recorded on line, averaged over a given time period and compared to a fleet reference. The distance between the variable monitored on a wind turbine and the fleet reference serves as a fault indicator for the turbine.

Let V be a variable measured on a wind turbine or a model residual synthesized on a wind turbine. V is assumed to carry information on the turbine deterioration, and several ways to choose this variable V are considered in Section 2.1.2. $V_j(k)$ is the value of the variable V from turbine j at time k . Let N_T be the number of turbines in the wind farm. Let Win be an analysis window of size W . For each turbine j , with j varying from 1 to N_T , a fault indicator $F_j(k)$ is built at time k , as follows :

1. For each turbine j in the farm, an **averaged value of the turbine-level variable** V_j is calculated over the N_{Wj} samples present in the analysis window Win :

$$\bar{V}_j(k) = \frac{1}{N_{Wj}} \sum_{t=k-W}^k V_j(t) \quad (1)$$

For the practical implementation, it has to be noticed that, for some reasons that will be explained further, some samples may be lacking and N_{Wj} may be strictly inferior to W . If the number of samples N_{Wj} is smaller than N_{WL} , $\bar{V}_j(k)$ is not computed as it would not be representative enough.

2. The **fleet reference** $\bar{V}_{fleet}(k)$ is calculated using the averaged values $\bar{V}_j(k)$ at the turbine level, as follows :

$$\bar{V}_{fleet}(k) = \text{median}_{(l=1, \dots, N_T)}(\bar{V}_l(k)) \quad (2)$$

The fleet reference is calculated using the median and not the mean, so as to make it insensitive to abnormal values generated by faulty turbines. For the practical implementation, the fleet reference is computed if the number of averaged values $\bar{V}_j(k)$ computed at time k is higher than a limit number N_{TL} (i.e. information is available from enough turbines) and the reference should be representative of the farm normal conditions as long as more than half of the turbines used to calculate the reference operate in normal conditions.

3. A **fault indicator** $F_j(k)$ is then calculated for each turbine j as the distance between the monitored variable $V_j(k)$ and the fleet reference $\bar{V}_{fleet}(k)$:

$$F_j(k) = |\bar{V}_j(k) - \bar{V}_{fleet}(k)| \quad (3)$$

In practice, N_{WL} is set to $\frac{W}{2} + 1$, which means that the variable $\bar{V}_j(k)$ should be available during at least half the analysis window W . N_{TL} is set to $\frac{N_T}{2} + 1$, which means that $\bar{V}_j(k)$ should be available for more than half the turbines of the farm for the fleet reference to be computed.

2.1.2. Synthesis of a turbine-level variable for comparison to the fleet reference

At the turbine level, the variable V used to create the fault indicator may be of two kinds : it can be either a residual generated as the difference between a measured value of SCADA variable and its prediction by a normal behavior model, or in a more straightforward way, it can be directly a measured SCADA variable, Figure 1.

- The variable V may be a residual, computed as the difference between a SCADA variable and its prediction by a normal behavior model. In this paper, in the same logic as using a component temperature as a fault indicator, the difference between the measured temperature of a component and its temperature predicted by a physics-related model is used.

In the generator, the active power produced by the turbine, the rotation speed of the drive shaft, and the nacelle temperature can account for a change in a component temperature. So a linear model relating these variables to the component temperature is proposed, Equation (4)

$$\hat{T}_j(k) = a.P_j(k) + b.\Omega_j(k) + c.T_{nacelle_j}(k) + d \quad (4)$$

with $\hat{T}_j(k)$ the component predicted temperature, $P_j(k)$ the active power produced, $\Omega_j(k)$ the shaft rotation speed, $T_{nacelle_j}(k)$ the nacelle temperature, recorded by the SCADA system and a , b , c and d constant parameters. To built a normal behavior model, the parameters a , b , c and d of the model are regression coefficients learnt using a classical least squares algorithm [2], on data gathered during periods when the turbine is operating under normal conditions. The variable V is then taken equal to the temperature residual Rs taken as the difference between the actual measured temperature and its predicted value, i.e. for turbine j at time k :

$$V_j(k) := Rs_j(k) = T_j(k) - \hat{T}_j(k) \quad (5)$$

The normal behavior model (4) expresses normal temperature variations due to the turbine producing electric power. Thus, the model is valid only when the turbine is in operating conditions and, for the practical implementation, all the samples gathered when the power produced by the turbine is below a production threshold Th_p , ie when the turbine is stopped, are removed from the analysis window Win .

This hybrid “mono-multi-turbine” approach consisting in generating a fault indicator by comparing the residual generated for each (“mono”) turbine to a (“multi-turbine”) farm reference is the original fault indicator synthesis proposed and defended in this paper for its good performance and lower sensitivity to both internal (operating conditions) and external (environmental variations) influences. The main steps for the implementation of this fault indicator synthesis are sketched in Algorithm 1.

- In a more direct way, the variable V may be a SCADA variable, relevant in itself as a fault indicator, such as a component temperature. As seen in the literature -e.g. [31, 18]-, component temperatures are commonly used as fault indicators. An overheating component can be the sign of a mechanical problem or of a cooling system failure. In this case, the variable V is taken as the monitored temperature T of a considered component, i.e. for turbine j at time k :

$$V_j(k) := T_j(k) \tag{6}$$

Reading or sensors errors may affect SCADA measures. Thus, in the practical implementation of this fault indicator, any temperature value outside an acceptable range is removed from the analysis window Win .

This approach based on the direct comparison of SCADA data of each individual turbine to a farm reference has already been proposed and assessed in [34] for temperature measurements, and extended to other kinds of measurements in [35] : it has been proved too sensitive to the influence of possibly different operating conditions between each individual turbine. It is considered in this paper for comparison purposes with the proposed combined mono- and multi-turbine method.

Algorihm 1 General algorithm for the combined mono- and multi-turbine fault indicator generation

Phase 1 - Offline learning phase

For each turbine j, with j=1 from Nt

Learn (by a least squares algorithm) the model parameters a, b, c d
of the normal behavior model in Equation(5)

EndFor

Phase 2 - Online implementation phase

For each sampling period k,

Step 1 : calculation of the averaged mono-turbine residuals

For each turbine j, with j=1 from Nt

Calculate the predicted temperature $\hat{T}_j(k)$ - Equation (5)

Calculate the corresponding residual $Rs_j(k)$ - Equation (6)

Calculate the averaged residual $\overline{Rs}_j(k)$ on the analysis window Win - Equation (1)

EndFor

Step 2 : calculation of the farm reference for residuals comparison

Calculate the farm reference $\overline{Rs}_{fleet}(k)$ - Equation (2)

Step 3 : calculation of the fault indicator

For each turbine j, with j=1 from Nt

Calculate the multi-turbine fault indicator - Equation (3)

EndFor

Step 4 :Alarm set off

For each turbine j, with j=1 from Nt

If the fault indicator is higher than the detection threshold

Set an alarm off for the turbine i

EndIf

EndFor

EndFor

2.1.3. Fault detection based on the generated fault indicators

The occurrence of a fault on wind turbine j leads to a change in the corresponding fault indicator F_j . Ideally, a fault indicator should be sensitive to a fault occurrence, but insensitive to the variations of the farm environment and of the wind turbine operating conditions : building such a fault indicator was the motivation to propose a combined “mono-multi-turbine” approach. Under these assumptions, the statistical properties, in particular the expected value, of the fault indicator F_j changes with a fault occurrence, and the fault detection can be performed using statistical hypothesis testing tools (eg testing hypothesis $\mathcal{H}_0 : \mathbb{E}(F_j) = \mu_{0j}$ vs $\mathcal{H}_1 : \mathbb{E}(F_j) \neq \mu_{0j}$), and can be implemented by setting a threshold on F_j . In theory, in this classical setting, when no information is available on the properties of the fault indicator F_j in presence of a fault, the value of this threshold can be determined using the estimated or assumed statistical properties of F_j when no fault is present in order to guarantee, for example, a false alarm rate. In our approach, we want to avoid any additional hypothesis on the distribution of F_j under the “no fault ” assumption, which could be different for each turbine and could possibly lead to a different threshold for each turbine. Consequently, in the following, our approach is to set the detection threshold for the whole wind farm, on the basis of the whole available historical 4-years database for the farm in order to get a given false alarm level.

2.2. Performance evaluation indicators

Fault indicators are to be used in wind farms to assist maintenance operators. They should allow the detection of faults occurring on a wind turbine without generating too many false alarms. However, in the literature on wind turbine fault indicators, no paper addresses the issue of the performance evaluation of the indicators proposed. Most of them merely report the ability of their indicator to detect a particular fault. In this paper, we chose to evaluate the fault indicators proposed from a wind farm production manager’s point of view. Fault indicators should definitely be able to detect faults on wind turbines, to avoid a major degradation and then a costly repair. However, they should be able to do it early enough for a maintenance operation to be scheduled. Another major issue is the cost of false alarms. A false alarm results in a useless maintenance action: a maintenance team has to be sent to the production site to inspect the supposedly faulty turbine. Because wind turbines are usually located in remote sites, the inspection takes time and consequently costs a lot of money. So, in this paper, we propose 3 performance evaluation indicators,

able to make a fair and objective assessment of the indicator usefulness: the detection time before failure, the useless maintenance action number and the indicator persistence, which gives to the operator some indications about the relevance and performance of the fault detection process based on the proposed indicator.

Performance evaluation indicators are computed from a data base gathered on a wind farm during a period of length D , where faults occurred. The data base is previously split into normal conditions periods and faulty periods. A normal conditions period is a period of time where no fault occurred on a wind turbine. A faulty period is a period of time where a fault was continuously present on a turbine. It ends when the failure occurs, ie when the turbine stops functioning.

Let Th be a detection threshold, set on the fault indicators $F_j(k)$. When the indicator exceeds Th , an alarm is raised. Th is set for the whole wind farm, ie the same value is set for all the turbines fault indicators, $F_j(k)$.

2.2.1. Detection time before failure

The detection time before failure measures the time separating the first time the fault indicator exceeded the detection threshold Th during a faulty period from the failure time, ie the end of the faulty period. It gives an estimation of the time left to the maintenance team to repair the fault before the failure.

2.2.2. Number of useless maintenance actions

A false alarm occurs when the fault indicator exceeds the detection threshold Th during a normal conditions period. The false alarm rate is converted into an equivalent “number of useless maintenance actions” so as to better consider its cost. A “false alarm day” is a day when at least one false alarm occurred on at least one of the turbines. A “false alarm period” is a period made of consecutive false alarm days. The shorter possible length for a false alarm period is one day. During a false alarm period, a maintenance team has to be sent once to the wind farm location. So each false alarm period results in a useless maintenance action. The useless maintenance action number is the number of times a maintenance team has to be sent to the wind farm during the period D ; it is equal to the number of false alarm periods.

2.2.3. Indicator persistence

The persistence measures the percentage of time during which the fault indicator remains above the detection threshold Th during a faulty period. A persisting indicator makes the maintenance operator more confident in the occurrence of the fault than an indicator constantly being set on and off.

3. Results and discussion

3.1. Data base

The methods presented in Section 2 are applied on a set of real data, gathered on a French wind farm located in the south of France. The 6 wind turbines forming the farm are identical. They are of the same make, conceived to produce 2 MW, with a horizontal axis. SCADA data were recorded every 10 minutes during 4 years, from November 2009 till December 2013.

During this 4 years, 6 single faults affecting the generator occurred on different turbines at different times, some of them generating a major failure and, consequently, the machine shutdown for several weeks, see Figure(2) :

- Fault on two bearings : 2 bearings broke down on 2 generators, because of a lack of lubrication. One of the generators had to be replaced. These 2 episodes are named “faulty bearing WT6 (FB_WT6)” and “faulty bearing WT9 (FB_WT9)”, for faulty bearing on wind turbines 6 and 9.
- Faults on two stator windings : two generators were stopped because of a short-cut on the stator winding, induced by an over-heating period which damaged the winding insulating material. The generators had to be replaced. These episodes are named “faulty winding WT9 (FW_WT9)” and “faulty winding WT11 (FW_WT11)”.
- Faults on two cooling systems : a fault occurred on the cooling system of two machines, due to a fixation screw getting loose. These episodes are named “faulty cooling system WT7 (FCS_WT7)” and “faulty cooling system WT10 (FCS_WT10)”.

[Figure 2 about here.]

The SCADA data used to implement and compute the fault indicators are the bearing temperatures, the stator temperatures, the nacelle temperatures, the active power produced and the shaft rotational speed on the 6 turbines.

3.2. Implementation of fault indicators

Four fault indicators are implemented and compared using the performance evaluation indicators presented in Section 2.2 calculated on the 4 years of data : the first considered fault indicator corresponds to the proposed combined mono- and multi-turbine approach, the three others are more classical and are implemented for comparison purposes. Each fault indicator is calculated every 10 minutes, using a sliding window of size $W = 144$ samples, ie 24 hours. N_{WL} is thus set to 72 and N_{TL} to 4. When relevant, the limit production threshold Th_p , above which the temperature can be predicted, is set to 50 kW, mainly because production levels below 50 kW correspond to starting or shutting down transient behaviors. In the following, depending on the considered fault to monitor, T_j can be either the bearing temperature or the stator temperature.

3.2.1. Implementation of the proposed combined mono- and multi-turbine approach

The indicator $RES_{multi_j}(k) = \overline{Rs_j}(k) - \overline{Rs_{fleet}}(k)$ is computed for each turbine j , following the proposed combined mono- and multi-turbine approach, see Algorithm 1 :

- The residual $Rs_j(k)$ is generated using Equation (5), with \widehat{T}_j the predicted bearing or stator temperature using the normal behavior model in Equation (4). The temperature variations due to changes in the power production are accounted for by the model. The model is learnt for each turbine, during a normal conditions period of two months at the beginning of the 4 years recording. Whenever a fault occurs on a turbine, the turbine model is learnt just after the necessary repair is made, on a fixed period of 2 months.
- This model residual $Rs_j(k)$ is then averaged to obtained $\overline{Rs_j}(k)$, using Equations (1) with $V_j = Rs_j$;
- The average residual $\overline{Rs_j}(k)$ is finally compared to the fleet reference $\overline{Rs_{fleet}}(k)$ computed using Equation (2) with the averaged residuals from all the turbines, to obtain the fault indicator $RES_{multi_j}(k)$.

3.2.2. Implementation of classical fault detection indicators for comparison

For each turbine j , three other fault indicators are considered : two of them are generated using a mono-turbine approach, the third one is generated by a simple multi-turbine comparison of the raw measurements (no residual generation at the turbine level).

- The two indicators T_{mono_j} (directly based on raw measurements) and RES_{mono_j} (based on residuals) are implemented following a classical mono-turbine approach, ie using SCADA data recorded from the turbine j alone :

- $T_{mono_j}(k) = \bar{T}_j(k) = \frac{1}{N_{W_j}} \sum_{t=k-W}^k T_j(t)$ where T_j can be either the bearing temperature or the stator temperature.

In this case, the SCADA temperature is averaged over 24 hours and thresholded, in a way very similar to the traditional SCADA detection system.

- $RES_{mono_j}(k) = \overline{Rs}_j(k) = \frac{1}{N_{W_j}} \sum_{t=k-W}^k Rs_j(t)$ with $Rs_j(k) = T_j(k) - \hat{T}_j(k)$ where T_j can be either the bearing temperature or the stator temperature.

In this case, the residual generated using Equation (5) and the model in Equation (4) is averaged and thresholded. The temperature variations due to changes in the power production are accounted for by the model. The model is learnt for each turbine, during a normal conditions period of two months at the beginning of the 4 years recording. Whenever a fault occurs on a turbine, the turbine model is learnt just after the necessary repair is made, on a fixed period of 2 months.

- An indicator T_{multi_j} (directly based on measurements) is implemented using a multi-turbine approach, using a fleet reference

- $T_{multi_j}(k) = \bar{T}_j(k) - \bar{T}_{fleet}(k)$, using Equations (1) and (2) with $V_j = T_j$, where T_j is the bearing temperature or the stator temperature. The component temperature is averaged and compared to the fleet averaged temperature reference.

3.3. Performance results

[Figure 3 about here.]

Figure (3) shows the evolution of the 4 fault indicators during the four years when the SCADA data were recorded on the 6 turbines of the wind farm. The upper figures display the evolution of

the mono-turbine indicators (temperatures on the left, residuals on the right), the lower figures the multi-turbine indicators (temperatures on the left, residuals on the right). The detection thresholds corresponding to a useless maintenance action number of 10, 20 and 30 (ie a total of 10, 20 or 30 useless maintenance actions will be generated on the whole farm during 4 years) are also displayed. The faulty periods are delimited by vertical lines. The five faulty periods correspond to the following faults, in chronological order : FB_WT6, FB_WT9, FCS_WT7, FW_WT11 and FCS_WT10 and FW_WT9 during the same faulty period. One can see that the seasonal variations inducing a change in the external temperatures are visible on the mono-turbine fault indicators. Both the temperatures and residuals are higher in summer and lower in winter. The fault indicators monitoring the 6 turbines follow the same pattern. During faulty periods, abnormal increases in temperatures can be seen on the faulty turbines. However, the temperature increases are somehow buried in the variations observed in the temperatures during normal conditions periods. The faulty indicator to normal indicator contrast ratio (defined in a similar way as a signal to noise ratio: the energy of the indicator during a faulty period on the energy of the indicator during a normal condition period) is rather low, which makes the detection of the fault with a minimum amount of false alarms using T_{mono_j} quite difficult. The evolution of RES_{mono_j} is rather similar. Since it is a residual, its evolution varies around 0. Its standard deviation is reduced compared to T_{mono_j} since the variations in temperature due to the change in power production are accounted for by the model and thus removed but its mean remains affected by seasonal temperature variations. Abnormal changes due to faults are more visible on the indicators but the fault detection using a fixed detection threshold seems still difficult to do.

The use of a multi-turbine approach makes it possible to remove the external temperature variations (seasonal effects) affecting the mono turbine indicators since the temperature variations are also present in the fleet reference and consequently subtracted from the residuals. The faulty indicator to normal indicator contrast ratio seems to be increased and it is all the more so when RES_{multi_j} is used, since the residual is rid of variations due to external temperature changes and variations due to changes in power production. Whenever a fault occurs on a turbine, it is preceded by an increase in the corresponding turbine residual, which clearly stands out from the other residuals. A fault detection strategy using a fixed detection threshold seems much more feasible.

To compare the four indicators performances, two performance curves plotting the persistence

or the detection time before failure as a function of the useless maintenance action numbers are built for each of the 6 faults. The curves are similar to traditional ROC curves, [37]. For a given value of the detection threshold, the detection time before failure (advance detection time), the persistence and the useless maintenance action number are calculated and form a point in the performance curves.

[Figure 4 about here.]

[Figure 5 about here.]

The two performances curves are displayed in Figures (4) and (5). The performance curves of multi-turbine indicators based detection are displayed in plain lines and mono-turbine ones in dotted lines. One can see that for the faults FB_WT6, FB_WT9, FW_WT11 and FW_WT9, the detection implemented with multi-turbine indicators outperforms the detection based on the mono-turbine ones, as could be expected from Figure 3. Using these multi-turbine indicators allows detecting the faults earlier with a reduced number of useless maintenance actions. If the detection threshold is set at its highest value, ie set to obtain zero useless maintenance actions, with RES_{multi_j} it is still possible to detect all the faults at least 500 minutes ahead of time ie about 4 days. If the threshold is decreased to allow 3 useless maintenance actions, the faults can be detected as early as 1200 minutes ahead of time ie about 8 days, which is sufficient to plan a maintenance intervention. When mono-turbine indicators are used, FB_WT6 and FW_WT11, two major faults, cannot be detected. The detection time before failure remains close to zero whatever the value of the detection threshold. FB_WT9 and FW_WT9 can be detected two weeks ahead of time, with RES_{mono} or T_{mono} , but at the very high cost of more than 15 useless maintenance actions. The two faults where the mono-turbine indicators performances equal the multi-turbine indicators are the faults on the cooling systems, FCS_WT7 and FCS_WT10, which occurred suddenly and generated a sudden and large increase in the recorded temperature.

The detection time before failure as a function of the useless maintenance action number curves show that RES_{multi_j} performs slightly better than T_{multi_j} . The performances are comparable on FB_WT6, FW_WT9, FCS_WT11, FCS_WT7 and FCS_WT10 but improved with RES_{multi_j} on FB_WT9. However, the persistence as a function of the useless maintenance action numbers curves shows the superiority of RES_{multi_j} . Indeed, even if the detection time before failure is equivalent,

the persistence is globally higher when RES_{multi_j} is used. This is an important criterion because the higher the persistence is, the higher the confidence the operator may have in the indicator. Based on this criterion, RES_{multi_j} clearly outperforms all the other fault indicators on the considered real data set, especially the mono-turbine ones. This is illustrated in Figure (6) where a zoom in time is made on the period preceding FB_WT9. Indicators are presented in the same order as in Figure (3). The colored lines are the detection thresholds which set the number of useless maintenance actions at 10, 20 or 30. Their value is adapted to each indicator.. One can see that, though a slight increase in T_{mono_j} can be visually observed during the period preceding the fault, this increase cannot be detected with a fixed threshold. RES_{mono_j} can detect the fault but only because it exceeds the detection threshold for a very short period of time. This could be figured out from the persistence curve whose value remained close to zero. On the contrary, the multi-turbines indicators show an obvious increase, which can be easily detected. The fault symptom appears earlier in RES_{multi_j} and the indicator remains more often above the detection threshold, which results in a higher persistence.

[Figure 6 about here.]

4. Conclusion

In this paper, we propose a new hybrid multi-level strategy to build fault indicators to monitor turbines within a wind farm, following a combined mono- and multi-turbine approach. The approach proposed reduces the effect of changes in the operational and environmental conditions on the fault indicators. Residuals computed from a mono-turbine model are compared to a fleet reference residual to form a multi-turbine fault indicator. The mono-turbine residuals are less sensitive to the internal variations because the SCADA temperatures are first processed using a model explaining the variations of the temperature in function of the operational conditions. Rises in the temperature due to increases in the power produced or in the rotational speed are accounted for by the model and removed from the residuals. The mono-turbine residuals are further post processed in a multi-turbine approach, which enables the impact of the variations in the environmental conditions to be reduced. Indeed, turbines in a same wind farm are part of a fleet. They are of the same make and model, and their variables evolve in a similar way since they are submitted to the same environmental conditions. New performance evaluation criteria were proposed to analyze

the indicator performances for fault detection purpose, keeping in mind the cost of any useless maintenance action. The performance of the combined mono- and multi-turbine fault indicators were compared to more traditional mono-turbine indicators on a data base made of SCADA data recorded every ten minutes from a real wind farm on a four year period. As far as we know, this is the first time a paper presents such an extensive analysis of fault indicators performance for fault detection, on real data with a wind farm production manager point of view. The results clearly showed that the effects of changes in both environmental and operational conditions on the combined mono- and multi-turbine residuals were reduced during the fault-free periods. This results in the combined mono- and multi-turbine fault indicators being able to detect all the faults occurring on the farm with a limited number of useless maintenance interventions and with an increased persistence.

5. References

- [1] P. Tavner. *Offshore Wind Turbines - Reliability, availability and maintenance*. The Institution of Engineering and Technology, 2012.
- [2] R. Iserman. *Fault Diagnosis Systems - An Introduction from Fault Detection to Fault Tolerance*. Springer, 2006.
- [3] P. Tchakoua, R. Wamkeue, M. Ouhrouche, F. Slaoui-Hasnaoui, T.A. Tameghe, and G. Ekemb. Wind Turbine Condition Monitoring: State-of-the-Art Review, New Trends, and Future Challenges. *Energies*, 7, 2014.
- [4] W. Qiao and D. Lu. A Survey on Wind Turbine Condition Monitoring and Fault Diagnosis - Part I: Components and Subsystems. *IEEE Transactions on Industrial Electronics*, 62(10):6536–6545, October 2015.
- [5] W. Qiao and D. Lu. A Survey on Wind Turbine Condition Monitoring and Fault Diagnosis - Part II: Signals and Signal Processing Methods. *IEEE Transactions on Industrial Electronics*, 62(10):6546–6557, October 2015.
- [6] W. Yang, P.J. Tavner, C.J. Crabtree, Y. Feng, and Y. Qiu. Wind turbine condition monitoring: technical and commercial challenges. *Wind Energy*, 17(5):673–693, May 2014.
- [7] D. McMillan and G.W. Ault. Quantification of condition monitoring benefit for offshore wind turbines. *Wind Engineering*, 31(4):267–285, 2007.
- [8] W. Yang, R. Court, and J. Jiang. Wind turbine condition monitoring by the approach of SCADA data analysis. *Renewable Energy*, 53:365–376, May 2013.
- [9] A. Lebranchu, S. Charbonnier, C. Bérenguer, and F. Prevost. Review and analysis of SCADA data-based methods for health monitoring of wind turbines. In Podofillini et al, editor, *Safety and Reliability of Complex Engineered Systems - Proc. of the 25th European Safety and Reliability Conference (ESREL 2015)*, pages 2413–2421, Zürich, Switzerland, September 2015. Taylor & Francis.
- [10] D. McLaughlin, P.J.M. Clive, and J.H. McKenzie. Wind farm performance assessment in the real world. In *Proceedings of the 2009 European Wind Energy Conference & Exhibition*, pages 4520–4528, 2009.

- [11] P. Cambron, R. Lepvrier, C. Masson, A. Tahan, and F. Pelletier. Power curve monitoring using weighted moving average control charts. *Renewable Energy*, 94:126–135, August 2016.
- [12] S. Butler, J. Ringwood, and F. O’Connor. Exploiting SCADA system data for wind turbine performance monitoring. In *2013 Conference on Control and Fault-Tolerant Systems (SysTol)*, pages 389–394, October 2013.
- [13] F. Eljorje, S. Kim, and J. Lee. A Wind Turbine Fault Detection Approach Based on Cluster Analysis and Frequent Pattern Mining. *KSII Transactions on Internet and Information Systems*, 8(2):664–677, February 2014.
- [14] S. Li, D. C. Wunsch, E. A. O’Hair, and M. G. Giesselmann. Using neural networks to estimate wind turbine power generation. *IEEE Transactions on Energy Conversion*, 16(3):276–282, September 2001.
- [15] M. Schlechtingen, I.F. Santos, and S. Achiche. Wind turbine condition monitoring based on SCADA data using normal behavior models. Part 1: System description. *Applied Soft Computing*, 13(1):259–270, January 2013.
- [16] M. Schlechtingen and I.F. Santos. Wind turbine condition monitoring based on SCADA data using normal behavior models. Part 2: Application examples. *Applied Soft Computing*, 14:447–460, January 2014.
- [17] M. Wilkinson, K. Harman, T. van Delft, and B. Darnell. Comparison of methods for wind turbine condition monitoring with SCADA data. *IET Renewable Power Generation*, 8(4):390–397, May 2014.
- [18] M. Schlechtingen and 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):1849–1875, July 2011.
- [19] W.G. Garlick, R. Dixon, and S.J. Watson. A model-based approach to wind turbine condition monitoring using SCADA data. In *0th Int. Conf. System Engineering, Coventry, 2009*.
- [20] P. Cross and X. Ma. Model-based and fuzzy logic approaches to condition monitoring of operational wind turbines. *International Journal of Automation and Computing*, 12(1):25–34, February 2015.
- [21] M.C. Garcia, M.A. Sanz-Bobi, and J. del Pico. SIMAP: Intelligent System for Predictive Maintenance: Application to the health condition monitoring of a windturbine gearbox. *Computers in Industry*, 57(6):552–568, August 2006.
- [22] A. S. Zaher and S. D. J. McArthur. A multi-agent fault detection system for wind turbine defect recognition and diagnosis. In *2007 IEEE Lausanne PowerTech Proceedings*, pages 22–27, July 2007.
- [23] A. Zaher, S.D.J. McArthur, D.G. Infield, and Y. Patel. Online wind turbine fault detection through automated SCADA data analysis. *Wind Energy*, 12(6):574–593, 2009.
- [24] J. Li, X. Lei, H. Li, and L. Ran. Normal Behavior Models for the Condition Assessment of Wind Turbine Generator Systems. *Electric Power Components and Systems*, 42(11):1201–1212, August 2014.
- [25] Z.-Y. Zhang. Comparison of data-driven and model-based methodologies of wind turbine fault detection with SCADA data. In *Proceedings of the 2014 European Wind Energy Conference - EWEA 2014*, 2014.
- [26] Shen Yin, Guang Wang, and Hamid Reza Karimi. Data-driven design of robust fault detection system for wind turbines. *Mechatronics*, 24(4):298 – 306, 2014.
- [27] W. Chen, S.X. Ding, A. Haghani, A. Naik, A.Q. Khan, and S. Yin. Observer-based fdi schemes for wind turbine benchmark. *IFAC Proceedings Volumes*, 44(1):7073 – 7078, 2011. 18th IFAC World Congress.
- [28] L. Wang and C. Wen. Fault diagnosis of wind turbine blade based on robust residual error design. In *2016*

- Chinese Control and Decision Conference (CCDC)*, pages 574–577, May 2016.
- [29] Y. Liu and D. L. Yu. Robust fault detection for wind turbine systems. In *2014 20th International Conference on Automation and Computing*, pages 38–42, Sept 2014.
- [30] A. B. Borchersen and M. Kinnaert. Model-based fault detection for generator cooling system in wind turbines using SCADA data. *Wind Energy*, 19(4):593–606, 2016. we.1852.
- [31] S. Butler. *Prognostic Algorithms for Condition Monitoring and Remaining Useful Life Estimation*. PhD thesis, National University of Ireland Maynooth, 2012.
- [32] D. Astolfi, F. Castellani, and L. Terzi. Fault prevention and diagnosis through SCADA temperature data analysis of an onshore wind farm. *Diagnostyka*, 15, 2014.
- [33] B. Boucher. Lowering the cost of project using simple analysis of SCADA data – a real case example. In *Annual Conference of the Prognostics and Health Management Society 2013 - New Orleans, LA - USA, October 14-17, 2013*, 2013.
- [34] A. Lebranchu, S. Charbonnier, C. Bérenguer, and F. Prevost. Using SCADA data for fault detection in wind turbines: local internal model vs distance to a wind farm reference. In *Proc. of the 4th International Conference on Condition Monitoring of Machinery in Non-Stationary Operations (CMMNO 2014)*, Lyon, France, December 2014.
- [35] P. Cambron, C. Masson, A. Tahan, and F. Pelletier. Control chart monitoring of wind turbine generators using the statistical inertia of a wind farm average. *Renewable Energy*, 116:88–98, February 2018.
- [36] J. Tautz-Weinert and S. J. Watson. Using SCADA data for wind turbine condition monitoring - a review. *IET Renewable Power Generation*, 11(4):382–394, 2017.
- [37] H.L. Van Trees. *Detection, Estimation, and Modulation Theory, Part I*. Wiley-Interscience, 2001.

List of Figures

1	Multi-turbine approach : generic principle for the synthesis of fault indicator F_j for wind turbine j . The choice of the variable V as a mono-turbine residual corresponds to the combined mono- and multi-turbine method proposed in this paper.	24
2	Occurrence of failures and “failure periods” on the 6 wind turbines of the considered farm during the 4 years recorded database - Bearing failures in red, cooling system failures in green and windings failures in blue	25
3	Evolution of the 4 indicators for the 6 wind turbines over the 4-years period. The coloured shaded areas correspond to the failure periodes shown in Figure (2)	26
4	Performance characteristics curves : advance detection time as a function of the number of useless maintenance interventions for 6 different faults : FB_WT6, FB_WT9, FCS_WT7, FW_WT11, FCS_WT10, FW_WT9	27
5	Performance characteristics curves : fault indicator persistence as a function of the number of useless maintenance interventions for 6 different faults : FB_WT6, FB_WT9, FCS_WT7, FW_WT11, FCS_WT10, FW_WT9	28
6	Evolution of the 4 indicators over the period preceding FB_WT9	29

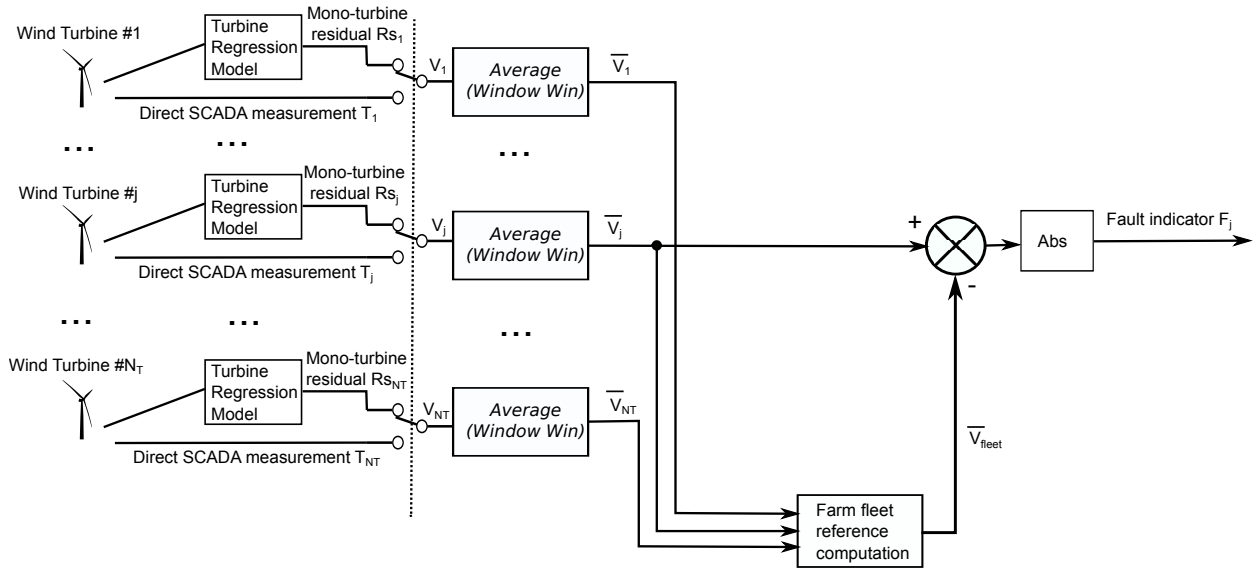


Figure 1: Multi-turbine approach : generic principle for the synthesis of fault indicator F_j for wind turbine j . The choice of the variable V as a mono-turbine residual corresponds to the combined mono- and multi-turbine method proposed in this paper.

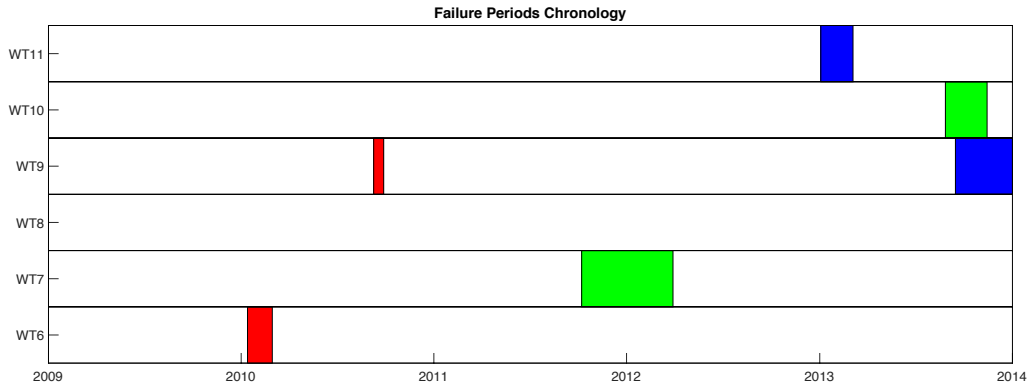


Figure 2: Occurrence of failures and “failure periods” on the 6 wind turbines of the considered farm during the 4 years recorded database - Bearing failures in red, cooling system failures in green and windings failures in blue

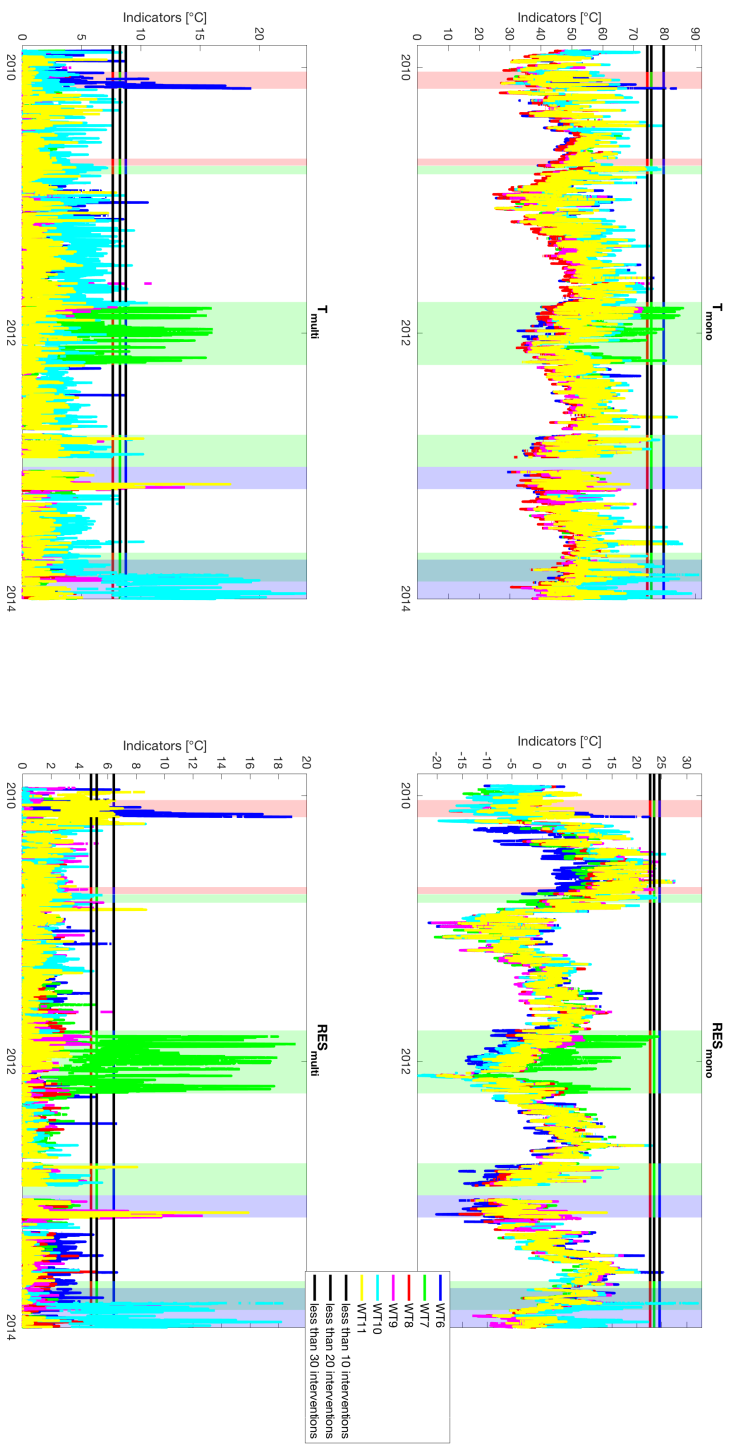


Figure 3: Evolution of the 4 indicators for the 6 wind turbines over the 4-years period. The coloured shaded areas correspond to the failure periodes shown in Figure (2)

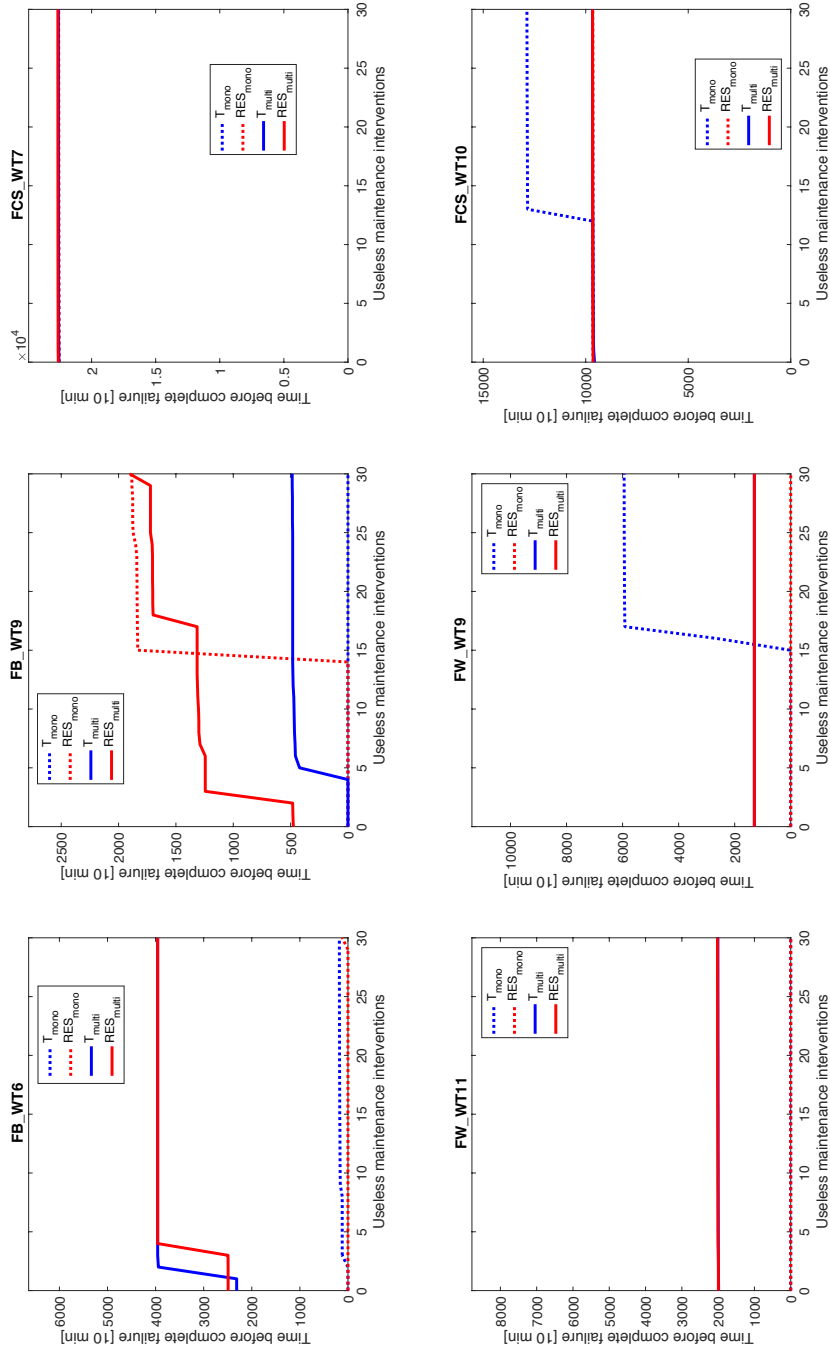


Figure 4: Performance characteristics curves : advance detection time as a function of the number of useless maintenance interventions for 6 different faults : FB_WT6, FB_WT9, FCS_WT7, FW_WT11, FCS_WT10, FW_WT9

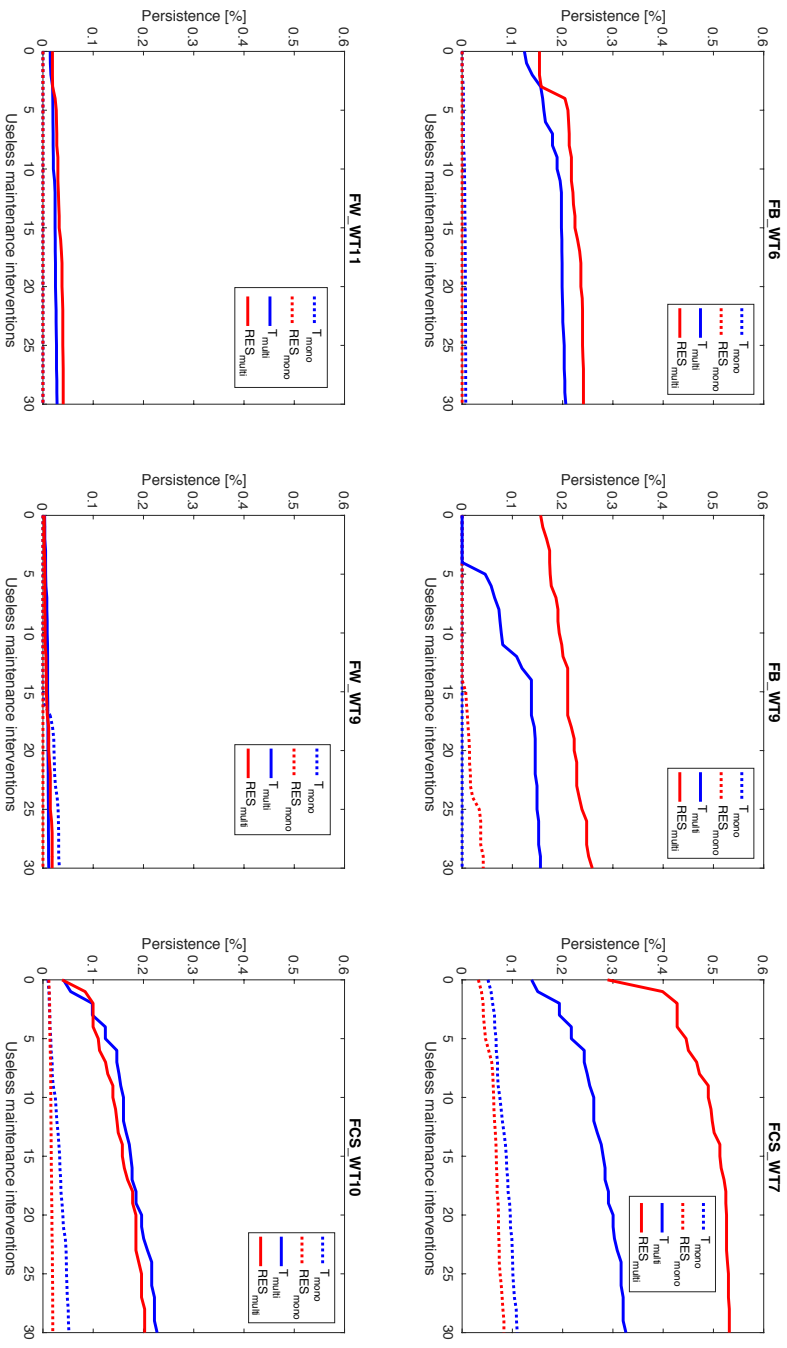


Figure 5: Performance characteristics curves : fault indicator persistence as a function of the number of useless maintenance interventions for 6 different faults : FB_WT6, FB_WT9, FCS_WT7, FW_WT11, FCS_WT10, FW_WT9

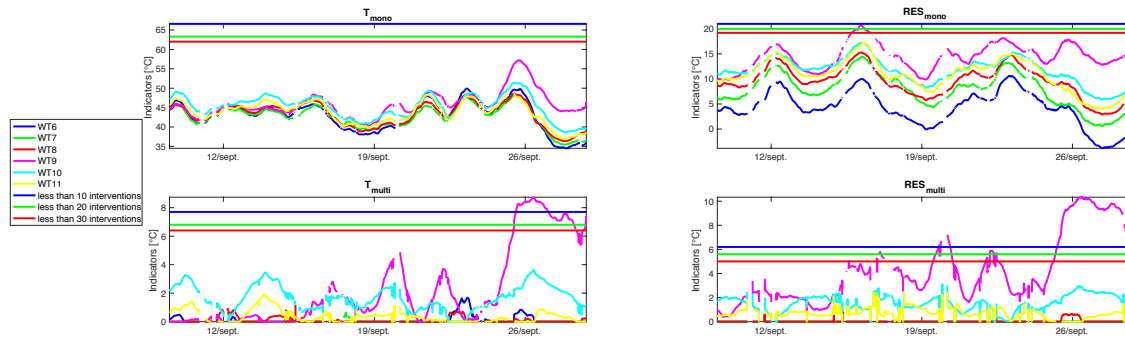


Figure 6: Evolution of the 4 indicators over the period preceding FB_WT9