Model-Based Fault Detection in Wind Turbines

Basheer Wasef Fawzi SHAHEEN

Master thesis submitted under the supervision of

Prof. Ahmed Abuhanieh, BZU

Prof. Michel Kinnaert, ULB

In order to be awarded the Master's Degree in

Sustainable Engineering in Production

SAAS: Service d'Automatique et d'Analyse des Systèmes, ULB

Academic year

2016-2017

Abstract

An early Model-based fault detection was developed and presented in the basis of WT's power curve to detect the degradation (faults) in gear box efficiency, resulted from the existing mechanical losses (torque losses) through the low-speed shaft(LSS) and the high-speed shaft (HSS), then to assist in implementing predictive maintenance. The detection was performed on two levels; the first level represents a slight and progressive degradation in the gear box efficiency, and the other one represents a radical (abrupt) degradation in the efficiency. Artificial SCADA data for different measurements (wind speed and active power) in both, fault free and faulty operating modes were generated using FAST-NREL simulator. Two WT power curves' parameters were estimated; the first one through Least Squares algorithm, and the second one using non-linear optimization through unconstrained function minimization, then power residuals were generated from each power point. Finally, on-line CUSUM statistical change detection algorithm was used to evaluate and detect small changes in power residuals generated from the first model. The presented fault detection system successfully detected faults in both detection levels under realistic wind turbulence, and with fault magnitude of 2% efficiency degradation for the progressive degradation level.

Key Words:

Wind Turbine, Fault detection, SCADA data, Condition Monitoring, Wind Turbine Power Curve, System Identification, Wind Turbine Simulator.

Acknowledgment

I would like to express my deepest thanks to everyone who pushed me further and kept me from giving up; to all people who helped me during my master's study period.

At the first place, I would like to thank my mother and my beloved family who were available whenever I needed them, pushing me even further and keeping me from running away when it got difficult.

Secondly I would like to express my sincere gratitude to my supervisors Prof.Michel Kinnaert (ULB) and Dr.Ahmad Abuhanieh (BZU) who reflected devotion, commitment and valuable insights while supervising my MSc. thesis, the entire academic staff at Faculty of Engineering and Technology at Birzeit University (BZU) and An-Najah National University (ANU), and my friends who helped and supported me through every bit of my study journey.

Last but not least, I would like to say a big "thank you" to all researchers at Service d'Automatique et d'Analyse des Systèmes (SAAS), Department of Control Engineering and System Analysis at Brussels School of Engineering and the entire staff of SAAS for their help, support and guidance throughout my study period at Université libre de Bruxelles (ULB).

Abbreviations

CM: Condition Monitoring

CMS: Condition Monitoring System

WT: Wind Turbine

WTCM: Wind Turbine Condition Monitoring

VAWT: Vertical Axis Wind Turbine

HAWT: Horizontal Axis Wind Turbine

O&M: Operations and Maintenance

CBM: Condition Based Maintenance

FDI: Fault Detection and Isolation

SCADA: Supervisory Control and Data Acquisition system

PCM: Process Condition Monitoring

CAE: Computer-Aided Engineering

FAST: Fatigue, Aerodynamics, Structures, and Turbulence

NREL: National Renewable Energy Lab

LSS: Low Speed Shaft

HSS: High Speed Shaft

IEC: International Electro-Technical Commission

PDF: Probability Density Function

CDF: Cumulative Distribution Function

TI: Turbulence Intensity

HH: Hub Height

3D: Three Dimensional

PC: Power Curve

GBoxEff: Gear Box Efficiency

ANN: Artificial Neural Network

LS: Least Squares

MSE: Mean Squared Error

MPPT: Maximum Power Point Tracking

CUSUM: Cumulative Summation

Table of Contents

Table of Figures

Chapter 1

Introduction

1.1 Background

Currently, wind energy is one of the main renewable energy sources that are used to generate electrical power. This energy source plays a key role of reducing the harmful effects of electrical power generation traditional methods such as, using fossil fuels, coal and natural gas which are the main responsible of global warming and the radical increase of the atmospheric carbon dioxide (CO2) levels [1].

Electricity generation from the wind using wind turbines is considered as one of the cleanest, environmentally friendly electricity generation methods, accepted by the society, and has economical competitive advantages. The efficiency of the generated power from wind turbine (WT) could be increased through controlling the wind turbine's operations according to the information related to wind state changes and the turbine location [2].

1.2 Wind Turbines (Wind Energy Conversion Systems)

Since the wind energy is considered as the most preferred renewable and clean energy source, this encourages and boosts the growth of wind farms especially for electricity generation [3]. The wind energy conversion system is known as Wind Turbine. A wind turbine can be defined as "a rotating mechanical device that converts wind kinetic energy to practical mechanical energy, resulting in electricity production" [4].

Wind turbine has two common types; horizontal axis wind turbines (HAWT) and vertical axis wind turbines (VAWT) [5,6].

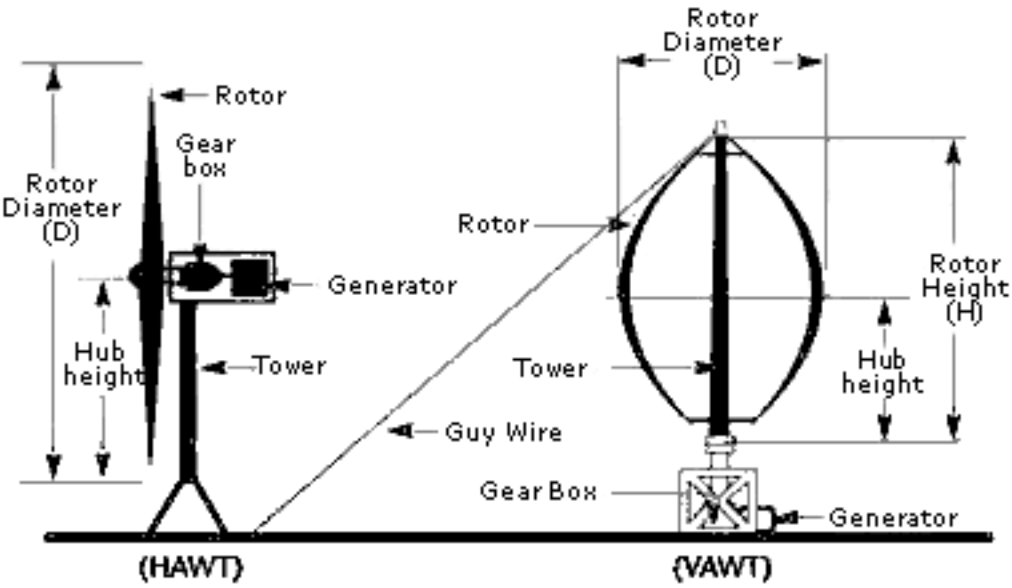


Figure 1: HAWT & VAWT [6]

Nowadays, the most common used design is HAWT for which the axis of rotation is parallel to ground surface. As shown in figure above, the main components of the wind turbine consist of rotor, drive train, nacelle, blades, yaw actuator, shafts, generator, tower, gearbox and control system. Main components and their functions are defined briefly below [7,8,9].

- **Rotor**

  The rotor includes the hub and blades of WT. Following aerodynamic principles; the blades coverts the kinetic energy of the wind into mechanical rotational energy.

- **Drive Train**

  The drive train consists of all rotating parts of the WT behind the rotor, such as, low speed shaft, gearbox and high speed shaft. Also, it has other components such as, the support bearings, braking system and other rotating parts in the generator.

- **Gearbox**

  Located in the drive train and used to transform the low speed-high torque rotational energy from the hub into high speed – low torque to the generator.

- **Generator**

  The generator of the WT is situated after the gearbox; it transforms high speed rotational energy coming from the high-speed shaft of the gearbox into electrical energy to be used then directly or dispatched to the central grid. Different types of generators are used in WTs; the common used types are synchronous generators and single or double fed asynchronous generators.

- **Nacelle**

  The WT's nacelle is the cover (main frame) which houses all components in a wind turbine, including the generator, drive train, gearbox and the brakes.

- **Yaw system**

  Also, called yaw orientation system; used to keep rotor shaft properly aligned with the variable direction of winds.

- **Tower and foundation**

  The common used types of WT's tower design are "the free-standing types using steel tubes, lattice towers, and concrete towers". The most important system dynamics factor which needs to be considered in the tower design is stiffness of the tower due to the possibility of occurring coupled vibrations between the tower body and the WT's rotor.

- **Control system**

  The main function of the WT's control system is to supervise and support the control of WT in all operating modes. For instance, it controls the pitch system to maximize the generated power and many other control processes.

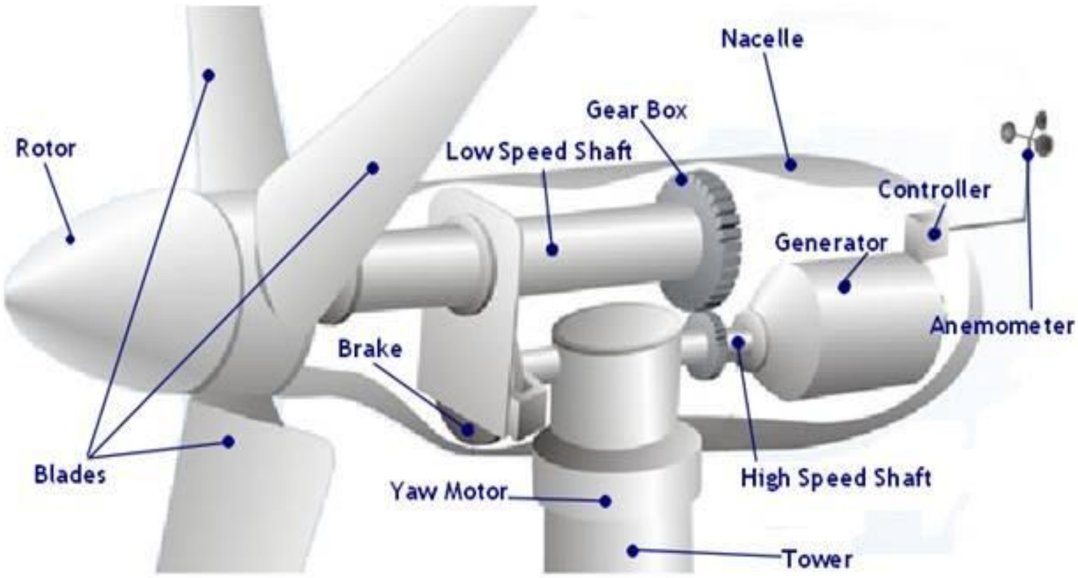


Figure 2.HAWT Main Components [8].

To sum up the wind energy conversion process, it could be illustrated as the following; wind energy transformed into mechanical energy through WT rotor, then, the mechanical energy will be transmitted to the generator through the drive train (shafts, gearbox ...) to be transformed into electrical energy [10].

1.3 Condition Monitoring of WTs

The possible failures which could happen to wind turbines, depend on either momentary events or the age of wind turbine components and their related failures. These failures lead to system interruption and huge economical losses as a result [7].

The rapid growth of wind energy market in the last few years has drawn attention of researchers to focus on operations and maintenance (O&M) costs of wind turbine, especially off-shore wind turbines, which account for approximately 25-30% of the overall cost of energy generations [11].

Condition monitoring systems or "health monitoring systems" play a key role in condition-based maintenance (CBM), also called predictive maintenance, which leads to reduce the costs of faults correction and increase the work performance of any device [10].

Fault Detection and Isolation (FDI) is an important topic in industrial processes because of the high competitiveness between the industrial companies and the increased demand of reliable and safe products along with high quality characteristics [12]. FDI systems provides early warnings when the small faults occur to prevent each single fault to make a failure on the entire system level. Moreover, the coherent system of fault diagnosis must include not only the detection of faults, but also isolate and identify the type of these faults and its severity levels on the whole system [10].

1.4 Condition Monitoring of WTs using SCADA data

The supervisory control and data acquisition (SCADA) system, is an important part of wind turbines process condition monitoring systems, which can provide a wide scale of measurements such as, temperatures, wind speed, wind directions (as wind parameters), rotor speed, pitch angle and output power These parameters are widely used to monitor the health conditions of wind turbine farms. SCADA data is very preferable to be used in many researches to forecast the wind speed and wind power due to its high availability. Furthermore, the SCADA system records have comprehensive process condition parameters of wind turbines which could be considered as fault informative parameters. Thus, fault detection in wind turbines using SCADA data is a cost-effective approach which leads to improve the reliability of wind turbines and assisting in reducing the maintenance cost of wind turbine farms [11].

1.5 Motivation

Today WTs are being used widely in on-shore and off-shore areas where the wind sources are available. There are too many challenges facing the access of WTs site, especially in offshore areas due to the lack of transportations and installation vehicles needed for the operation and maintenance processes of these giant WTs at these areas.

Most of the maintenance operations implemented on offshore wind turbines are unplanned; mostly corrective maintenance operations. The costs of this type of maintenance are typically high.

Due to these high unplanned maintenance costs, CBM or predictive maintenance, is needed to be implemented to reduce all associated costs of any failure occurs, in addition to reduce the large down times associated with the unplanned maintenance operations compared with predictive maintenance operations. This could help in lowering the costs of wind energy, which is the main challenge of this industry.

One of the best solutions to enable predictive maintenance system in wind turbines is through an early FDI system using Supervisory Control and Data Acquisition (SCADA). Due to the high availability of these data, based fault detection processes would be applicable to be implemented on a large number of wind turbines.

1.6 Key Objectives

The main goal of this thesis is to develop an early model-based fault detection and to monitor the performance degradation of WT (faults through the Gear Box) in the basis of the WT's power curve, using artificial (simulated) SCADA data (typically sampled at low frequency: from 30 s to 10 minutes).This performance degradation (power loss) is typically due to the degradation in Gear Box Efficiency (GBoxEff) resulted from the existing mechanical losses (torque losses) through the low speed shaft(LSS) and the high speed shaft (HSS).

It's worth mentioning that, the power loss or performance degradation could be affected by other factors rather than the GBoxEff, but in this research we focused on study the effect of degradation in the GBoxEff on the performance of the WT, in addition to the ease of simulating the degradation in GBoxEff.

The specific objectives of the research work are described as :

- To generate realistic "artificial SCADA Data" for measurements (i.e. wind speed and active power) in both fault free (GBoxEff=100%) and faulty (GBoxEff=99,98,97 & 90%) operating modes using the wind turbine simulator based on Aerodyn-FAST software from NREL,
- To estimate system's model and parameters using systematic mathematical and statistical procedures.
- To apply an on-line CUSUM statistical change detection algorithm to detect and localize small changes (degradation) in performance on two levels; the first level represents a slight and progressive degradation in the gear box efficiency (100-99-98-97%), and the other one represents a radical (abrupt) degradation in the efficiency (100-90%).

Chapter 2

Simulation of Wind Turbines

2.1 Introduction

Constructing condition monitoring system based on WT's power curve requires SCADA data measurements supported with well-documented maintenance information for both fault free and faulty operational conditions.

Real SCADA data measurements is not easy to be obtained; the wind turbine simulator based on Aerodyn-FAST software from NREL was used to generate artificial SCADA data for different measurements (i.e. wind speed and active power). The inputs of the wind simulator "TurbSim" are the wind environment, which is used as inputs for Aerodyn-FAST, and the final outputs are the "Artificial SCADA data" (wind speed and active power) in both fault free and faulty operating modes, which will be used later to construct the wind turbine power curve model and to generate power residuals to detects small changes (faults / performance degradation) using an appropriate algorithm.

It's worth mentioning that, some adjustments on TurbSim and Aerodyn-FAST MATLAB codes were applied in order to simulate and generate more than one single SCADA data point (i.e. 1000 point per single GBoxEff percentage).

2.2 Wind Turbine Aerodynamic (Wind Environment)

2.2.1 Energy in Wind

Understanding the wind environment is a very important factor to analyze WTs, since the wind is considered the main energy source for the WTs [13].

This could be explained by the relationship derived from the kinetic energy between wind speed u and the energy E existing in wind.

$$E = \frac{1}{2} At\rho u^3 \tag{1}$$

Where in our case; A is the perpendicular area to the wind direction, ρ is air density (1.225 kg/m3) and t is time interval. Thus, the importance of understanding wind characteristics resulted from the cubic relationship between wind speed u and wind energy E. Such knowledge in those characteristics are important for many relevant topics such as, WT's design, WT's performance analysis, the work mechanism and operations of WTs [14].

2.2.2 Wind Speed Patterns

Wind speed is typically difficult to be predicted due to the high time and space variability of wind, thus wind is analyzed by an appropriate probabilistic tool. Wind speed spectrum is the best tool to depict wind speed patterns (see figure below).

The high values of the spectrum (the peaks) represent a significant change in wind speed over the corresponding time period.

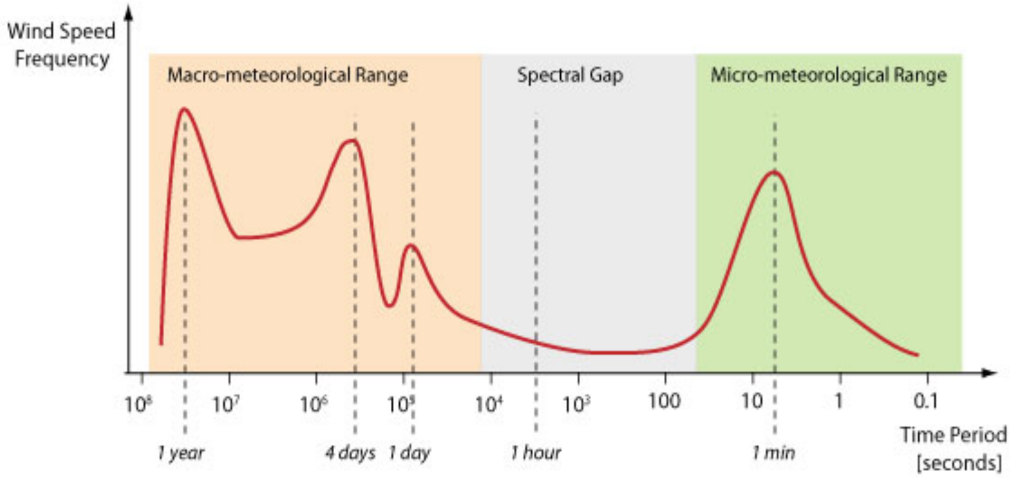


Figure 3: Wind Speed Spectrum [16]

The spectrum is mainly divided into two peak regions, the first one is the low-frequency peak which depicts the variation of the mean wind speed, and the second one is the high-frequency peak which depicts the turbulence [15]. These patterns (regions) are important for yield estimations, and for forecasting of wind power output.

2.2.3 Wind Speed Variations

2.2.3.1 Long-term wind variations

Due to the difficulty of predicting the annual wind speed, variations during the year can be characterized using probability density function. According to [16], it has been proven that the Weibull Probability Density Function (PDF) is effective to represent the mean wind speed for a full year.

p(U) is the Weibull probability density function with shape factor k and scale factor c. This gives the probability of occurrence of wind speed (U).

$$p(U) = \left(\frac{k}{c}\right)\left(\frac{U}{c}\right)^{(k-1)} exp\left[-\left(\frac{U}{c}\right)^{k}\right] \tag{2}$$

The dimensionless shape factor reflects the influence of the topography on wind speeds and ranges between 1.2 (mountains) to 4.0 (monsoon regions). The scale factor A is roughly 125% of the average annual wind speed [16].

The Cumulative Distribution Function(CDF) F(U) can be calculated from p(U) to define the probability of the wind speed lower than U.

$$F(U) = 1 - exp\left[-\left(\frac{U}{c}\right)^{k}\right] \tag{3}$$

According to the International Electrotechnical Commission (IEC) standards; for normal wind condition and the shape factor of the k=2, Rayleigh distribution is to be assumed here which is a particular case of Weibull distribution, thus, the scale factor $c = \frac{2U_{avg}}{\sqrt{\pi}}$ where U_{avg} is the expected value of the wind speed for the distribution.

$$U_{avg} = \int_0^{\infty} Up(U)dU \tag{4}$$

Thus, the PDF p(U) and the CDF F(U) become:

$$p(U) = \frac{\pi}{2}\left(\frac{U}{U_{avg}^2}\right) exp\left(-\frac{\pi}{4}\left(\frac{U}{U_{avg}}\right)^2\right) \tag{5}$$

$$F(U) = 1 - exp\left(-\frac{\pi}{4}\left(\frac{U}{U_{avg}}\right)^2\right) \tag{6}$$

2.2.3.2 Short-term wind variations

The instantaneous wind speed u(t) is characterized by a short-term mean wind speed U_{st} and its turbulence fluctuation value with a zero mean $\bar{u}(t)$

$$u(t) = U_{st} + \bar{u}(t) \tag{7}$$

U_{st} is the mean wind speed averaged over a time period longer than the characteristic time of the turbulence, the time period T is usually chosen to be 10 minutes [14].

$$U_{st} = \frac{1}{T}\int_{0}^{T} u(t)dt \tag{8}$$

Turbulence could be defined as a fluctuation in the wind speed within a very short time scale and caused from the friction with the earth's surface and the thermal effects acting on the air. Turbulence cannot be represented in terms of deterministic equations. Thus, it could be measured by turbulence intensity ‘‘I’’, which expressed in percent, (I) could be defined directly using standard IEC categories of turbulence characteristics. It's also possible to specify the turbulence intensity in percent instead of choosing the turbulence categories.

2.2.4 Normal Wind Profile Model

Variation of wind speed u is depicted through the wind profile U (z) as a function of height z. Following the IEC standard for WT design; the variation of wind speed u with the height z assumed to follow a power law model as the following:

$$U(z) = U(z_{hub}) \left(\frac{z}{U(z_{huv})}\right)^{\alpha} \tag{9}$$

Where α= Power Law Exp (PLExp) is assumed to be 0.3 according to IEC standard and z is the height above ground level (reference height) [17].

2.2.5 IEC Wind Turbine Classes

WT's site location plays a major role in the design process. WT needs to be designed for optimal performance and reliability through different weather conditions.

Four different classes of wind turbines are defined in IEC (61400-1: 2005) standard to suit different weather conditions including, high, medium, low and very low wind according to the reference wind speed U_{ref}. Also, wind turbulence is another parameter used to define wind classes. The combination of the two aforementioned parameters defines the class of the WT [17].

These input parameters are used only if the spectral model is IECKAI (Kaimal) or IECVKM (Von Karmal).

Table 1: IEC Wind Classes and turbulence intensity parameters [17]

Wind Speed Classes	I (High Wind)	II (Medium Wind)	III (Low Wind)	IV (Very Low Wind)
U_{ref} (m/s)	50	42.5	37.5	30
U_{avg} (m/s)	10	8.5	7.5	6
I_{ref}	A	0.16		
	B	0.14		
	C	0.12		

Where:

Rayleigh distribution is assumed, i.e. k = 2.

U_{avg} is the annual mean wind speed at hub height; U_{ref} is the 50-year extreme wind speed over 10 minutes; V50, I_{ref} is the mean turbulence intensity at 15 m/s.

A, B and C are the categories of higher, medium and lower Turbulence Intensity (I) characteristics respectively.

To sum up, the pre-illustrated wind environment theories represent the basis of the WT simulation process, since the wind environment is the main input parameters of the wind simulator "TurbSim" to generate the hub height files to be used then to conduct the mechanical behavior simulation of the WT using AeroDyn and FAST software as discussed below.

2.3 The Wind Turbine Simulator

The wind turbine simulator used in this research consisted of a set of codes which were developed by NREL. For example, the AeroDyn code conducts aerodynamic calculations while FAST code conducting the aeroelastic simulation. Both together simulate the wind turbine's mechanical behavior.

The control scheme of all operations of wind turbine has been added to the aforementioned two codes through SIMULINK in MATLAB (see appendix I).

The needed wind input files to AeroDyn and FAST codes were modeled and generated through TurbSim software depending on the required input parameters such as, mean wind speed ad turbulence intensity.

All the above-mentioned codes are illustrated below briefly.

2.3.1 NREL Design Codes

2.3.1.1 TurbSim

TurbSim is a" stochastic, full-field, turbulent-wind simulator". TurbSim simulates numerically time series of 3D wind velocity vectors through different points located in a vertical rectangular grid. The resulted files with an extension of ".hh" (see appendix II) can be used then as input files for other codes such as FAST – AeroDyn simulator to model the behavior of the wind turbine for turbulent winds [18] .

2.3.1.2 AeroDyn

AeroDyn is "a time-domain wind turbine aerodynamics module that has been coupled into the FAST to enable aero-elastic simulation of horizontal axis wind turbines" [19]. It can compute the forces acting on the different elements of the WT's blades for each step time after defining these elements along with their geometries.

2.3.1.3 FAST

Fatigue, Aerodynamics, Structures, and Turbulence (FAST) is "An aero-elastic computer-aided engineering tool (CAE) for horizontal axis wind turbines" [20]. FAST was developed by the researchers of the National Renewable Energy Lab (NREL) at the USA, and it's considered as the primary CAE tools in the lab, which used for simulating the coupled dynamic response of WTs. Using FAST, we can conduct analysis for wide range of WTs' configurations such as, 3 blades HAWT, pitch or stall regulation, up-wind or downwind rotor, and lattice or tubular tower. Also, we can model on-shore or off-shore WTs. It's worth mentioning that FAST relies on advanced engineering models; these models were derived from fundamental laws, but with appropriate simplifications and assumptions [20].

Refer to appendices III and IV to see samples from FAST input files.

Detailed information about TurbSim, FAST and Aerodyn is available on: http://www.nrel.gov

2.4 Simulation Basic Characteristics

In order to perform the simulation process, meteorological and control characteristics were set as illustrated in sections 2.4.1 and 2.4.2 below.

2.4.1 Simulation Meteorological Conditions

Table below shows the used simulation meteorological conditions.

Table 2: Simulation Meteorological Conditions

Meteorological Conditions	
Turbulence Model	IECVKM=Kaimal
IEC standard	1-ED2
Turbulence intensity percent	10-20 %
Wind profile type	Normal
Height of reference wind speed	84.672 m
Mean of the wind speed at the reference height	2-18 m/s
Power law exponent	0.3

2.4.2 Simulation Control Characteristics

Table below shows the main simulation control characteristics used in this research.

Table 3: Simulation Control Main Characteristics

Simulation Control	
Total Run time	12 minutes
Module step time	0.005 s
Sampling Frequency	200 Hz

Chapter 3

Artificial SCADA Data Generation

3.1 Introduction

In this chapter, the characteristics of the simulated DeWindD6 WT will be presented, and the process of generating SCADA data (measurements) required to construct the WT's power curve in fault free and faulty operating modes will be presented as well. WT's fault free (healthy) operating mode in this research means that the WT is assumed to be operated with no mechanical losses (torque losses) being transmitted through the gearbox, thus the GBoxEff (gearbox efficiency) was set to 100%, and to simulate losses (the faulty operating mode) the WT assumed to be operated with mechanical losses (torque losses) being transmitted through the gearbox if the GBoxEff is less than 100%, thus the GBoxEff was set to 99,98,97 & 90% in order to detect slight and progressive degradation in efficiency (100-99-98-97% respectively) and radical degradation in efficiency (100-90% at once).

When generating power, FAST will multiply the LSS torque by the efficiency and divide by the gearbox ratio to determine HSS torque. When motoring, FAST will multiply the HSS torque by the efficiency and gearbox ratio to compute the torque on the LSS [20].

3.2 Wind Turbine Characteristics

3.2.1 DeWindD6 Technical Characteristics

The simulated wind turbine in this research is from the type of "DeWindD6" with 1250 kW rated electrical power. Table below shows the main characteristics of this WT [21].

Table 4: Main Characteristics of DeWindD6 WT [21]

DeWindD6 Wind Turbine Model Specifications	
Hub Height	84.672 m (Typically 68-91.5 m)
Rotor Diameter	64 m
Grid height	80 m
Grid width	80 m
Total height	123.5 m
No. Of blades	3 blades
Blade length	31 m
Swept area	3217 m2
Cut-in wind speed	2.8 m/s
Nominal wind speed	12.5 m/s
Cut-out wind speed	23 m/s
Nominal rotational speed	~ 21.1 rpm
Rotational speed range	~ 13.2 – 24.5 m/s
Generator	Induction, doubly fed
Grid frequencies	50 Hz
Rated voltage	690 V
Nominal Current	1046 A

Full technical characteristics of DeWindD6 wind turbine available at:

http://www.mywindpowersystem.com/usedwindturbines/wp-content/up-loads/2015/06/DeWind_D6_Brochure.pdf

3.2.2 DeWindD6 Power Curve Characteristics

Power output of the WT can be depicted through the power curve as a function of wind speed.

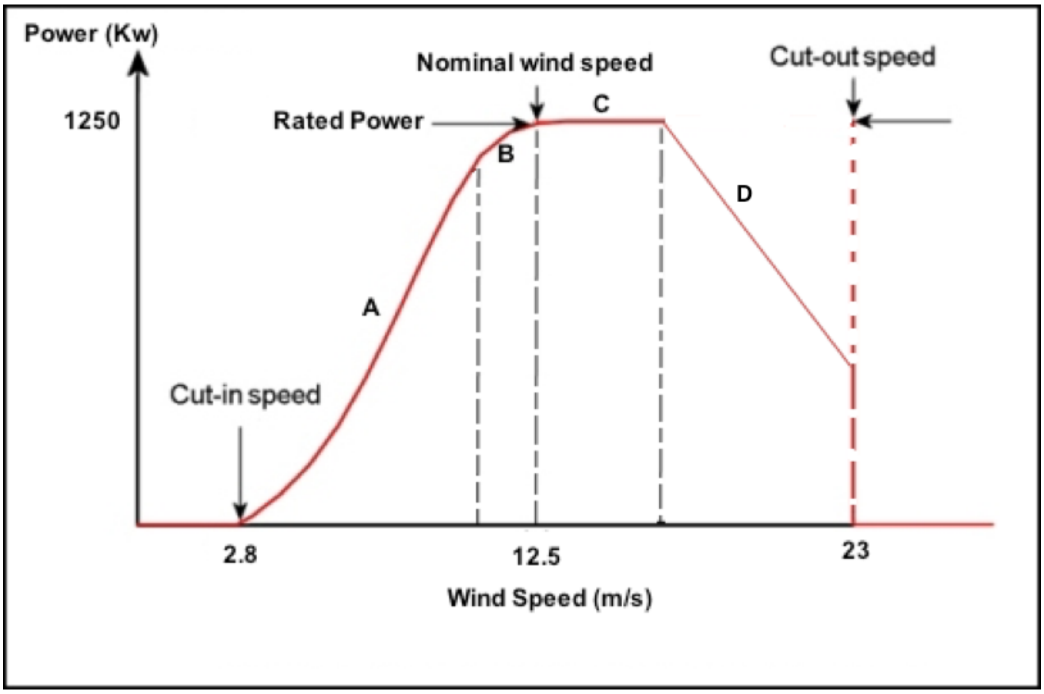


Figure 4: Nominal Power Curve of DeWindD6 WT [21]

The power curve of DeWindD6 WT is illustrated in the figure above which depicts the power curve of DeWindD6 WT which feature three key wind speeds [21]:

1. Cut-in wind speed: The wind speed at which the WT starts generating power.
2. Nominal wind speed: The wind speed at which the WT reaches the nominal power output, knowing that it is possible to generate higher power output above the nominal wind speed, but a control system is exist to maintain a constant power in order to limit loads and stresses on WT's blades.
3. Cut-out wind speed: The highest wind speed which the WT can operate at. The WT will stop if the wind speed exceeds the cut-out wind speed to prevent damage to WT's blades.

The different operating regions are, the maximum power point tracking (MPPT) region (A), transition region (B), power limitation or constant power region (C) and the soft storm transition region (D).

When the WT operates under variable speed, a control unit will control the rotor speed in order to maximize the output power and minimize the torque of loads. To reach the maximum level of electrical power production, the best way is to change the turbine speed with respect to wind speed to produce a maintained speed ratio to keep power at maximum. When the wind speed is in regions (A & B), to reach the maximum power, the rotor speed should be adjusted and maintained by the control unit. While when the wind speed is in region (C), the power output is maintained constant by controlling the bitch angle of the WT's blades [22].

3.3 Electrical Power Generation from WTs

The power generated from WTs is often represented by the WT's power curve. The relationship between the wind speed and the generated power is given by [23].

$$P = \frac{1}{2} A \rho C_P w^3 \tag{10}$$

where p is the power, ρ is the air density in kg/m3, A is the rotor area in m2, w is the wind speed and C_p is the power coefficient. The maximum value of power coefficient is known as Betz limit = 0.593, but in real WTs this value is not achievable and the maximum value is normally around 0.5. Moreover, the power coefficient can be obtained from the WT's manufacturer data.

The power curve of WT could be presented as the following:

$$\hat{P}(w) = \begin{cases} 0, & w < w_o \\ \hat{p}(w), & w_o \leq w \leq w_r \\ P_r, & w_r < w \leq w_1 \\ 0, & w > w_1 \end{cases} \tag{11}$$

where w_0 is the cut-in wind speed =2.8 m/s and w_1 is the cut-out wind speed=23 m/s, respectively. Also, w_r is the nominal wind speed=12.5 m/s, P_r is the rated power, and $\hat{p}(w)$ is the non-linear relationship between power and wind speed, see figure 4 above.

The shape of the non-linear region is related to the control strategy of extracting as much as possible power from the subjected wind.

Equation (11) was presented using two different estimated models as illustrated in sections 4.3.5 and 4.3.6 below.

3.4 Power Curve Construction

For each simulated measurement; the scalar average wind speed (w) and the output power (p) were calculated according to equation (14) below.

$$w_j = \frac{1}{n}\sum_{i=1}^{n} w_i \tag{12}$$

Where (w_i) is the wind speed generated by the simulator with step time 0.005 seconds. Same procedures were used to calculate the corresponding output power (p).

The resulted mean wind speed (w_j) is a 10 minutes' average wind speed out of 12 minutes' total run time, where the first two minutes were assumed representing the transient time of the generated measurements; corresponding (suspicious) data were eliminated.

The power curve consists of primarily two input variables: wind speed and power output. Using FAST-AeroDyn simulator from NREL, and following the simulation characteristics of DeWindD6 WT presented in section 2.4, raw artificial SCADA data measurements (active power), typically sampled at low frequency: from 30 s to 10 minutes were generated along with wind speed required to construct the power curve.

According to IEC 61400-12-1 standard, the wind speed of the power curve is "the undisturbed free-stream wind speed at hub height, normalized for a certain air density" [17]. Once the simulation process was complete, 10 minutes' averages of wind speed and active power were generated for, j: number of measured artificial SCADA data =1000 points per measurment for each single GBoxEff percentage.

Figure 5 below depicts a sample scatter plot of wind speed characterized by mean wind speed and turbulence intensity against active power output (i.e., the power curve) for both fault free (GBoxEff=100%) and faulty operating modes (i.e., GBoxEff=90%).

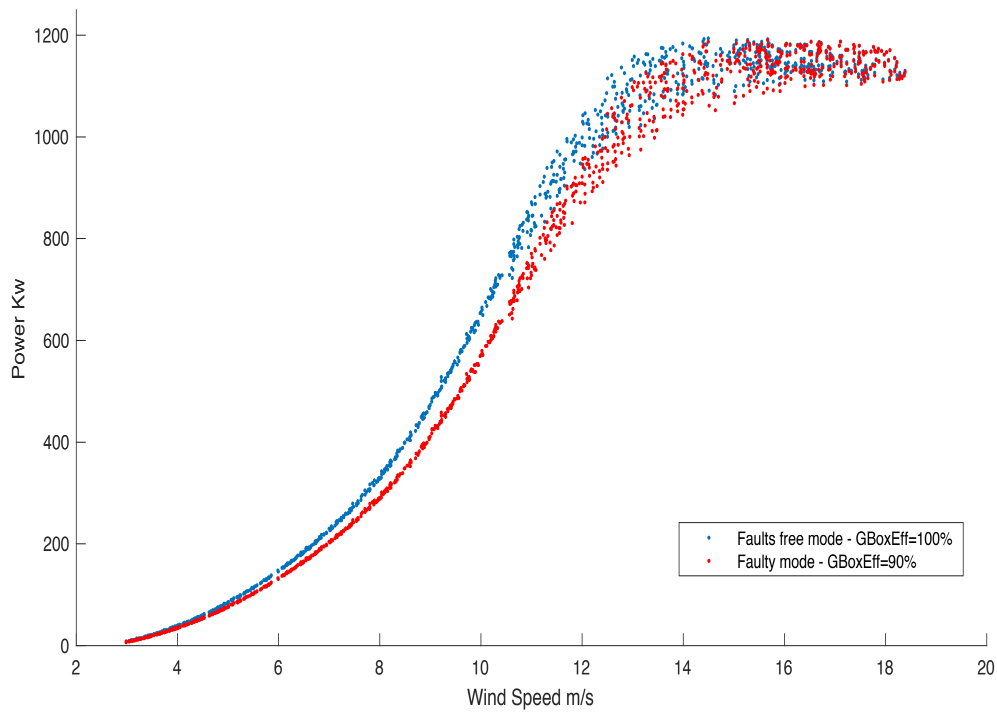


Figure 5: Sample of Power Curve's scatter plot from raw SCADA data for fault free and faulty operating modes

Chapter 4

Fault Detection in WTs: Model and Analysis

4.1 Introduction

Most of research papers focused on Condition Monitoring Systems (CMS) tools to diagnose faults and monitor the health of wind turbines, which includes a "sensors, signal acquisition and processing software, cabling and installations that gives continuous information about the monitored component condition" [4]. CMS is used in off-shore wind turbines especially, to monitor the components which are the most critical in the WT system such as, gear box, generator, rotor blades, yaw actuator ...etc. [4]. One of the most commonly used CM techniques is Operations and Maintenance (O&M) techniques for the turbine [24].

To determine the portion of each WT's components out of the total number of failures occurred; Hahn et al. [25],reported a survey of 1500 WTs over a period of 15 years indicated that five component groups, "i.e., electrical system, control system, hydraulic system, sensors, and rotor blades" are responsible for 67.0 % of failures occur in wind turbines; figure below depicts the share of main components of the total number of failures.

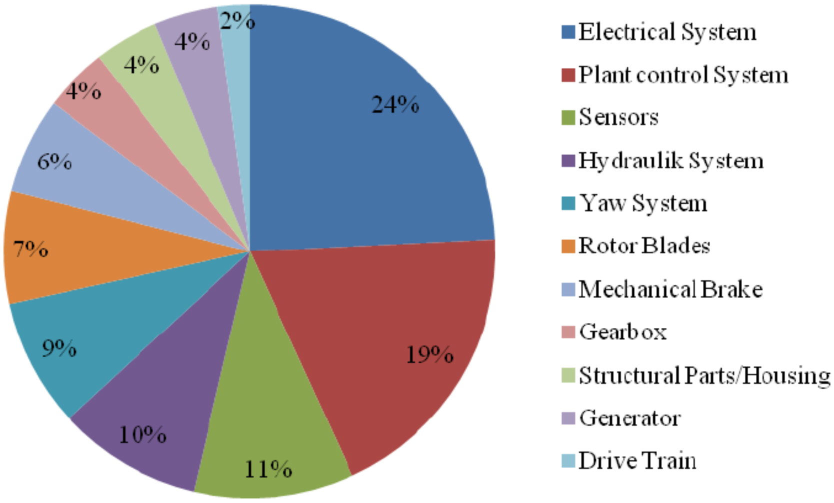


Figure 6:Portion of WT main components out of the total number of failures [25]

In order to illustrate the main CMS techniques of WTs existing in the literature, the main approaches of maintenance of WTs should be illustrated first. Hence, Tchakoua et al. [24] illustrated that maintenance approaches in the WT industry can be mainly classified into three groups:

- Reactive or corrective maintenance (run to failure);
- Preventive maintenance (time-based);
- Predictive maintenance (condition-based).

The costs associated with each type of traditional maintenance are presented in Figure 8.

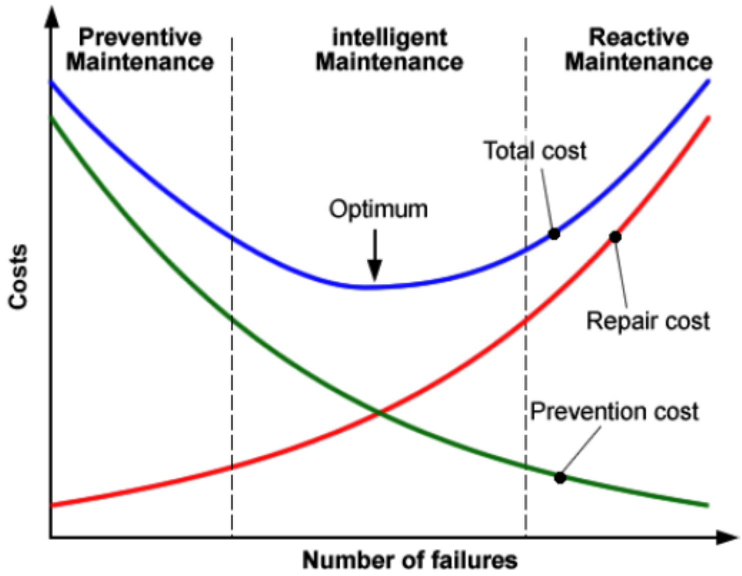


Figure 7:Maintenance Strategies' Costs [24]

In preventive maintenance, the prevention cost is high, while the repair cost is low due to the limited number of failures occurred. In reactive (corrective) maintenance, a larger number of faults will occur; this leads to a high cost of repair and low cost of prevention.

Predictive maintenance is a combination of preventive and reactive (corrective) maintenance which also called (intelligent maintenance) can improve the reliability, availability, and maintainability of wind turbines and reduce the maintenance costs as a result.

Also through Tchakoua et al. [24], a description of and models for CMSs which address maintenance techniques and methods was developed depending on other research works. Figure 3 below illustrates this description which indicated that, condition monitoring is performed through 3 steps:

1- Data acquisition using sensors.

2- Signal processing using various data processing techniques.

3- Feature extraction via the retrieval of parameters.

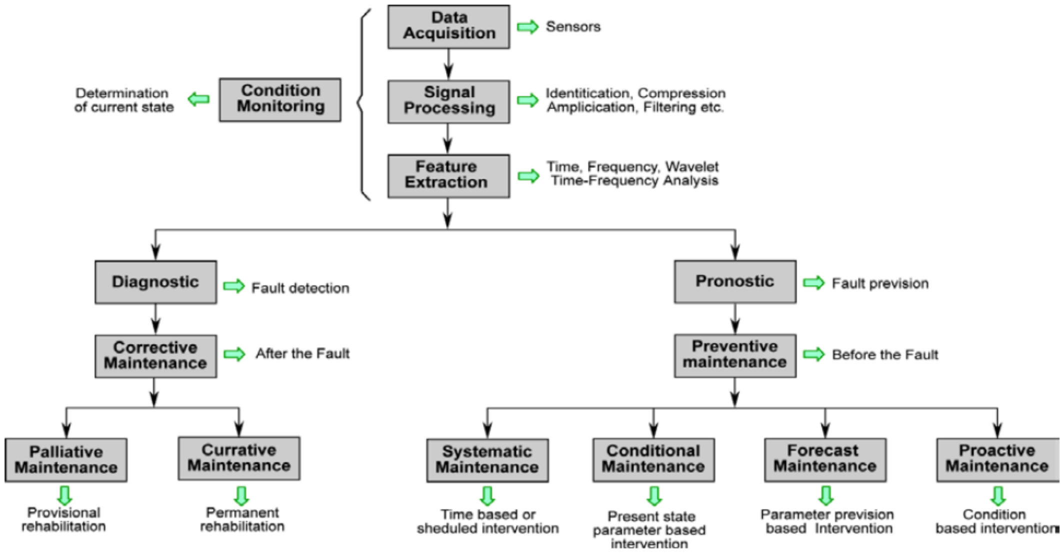


Figure 8:CM and Maintenance in WTs [24]

As shown in Figure 3above, CM was applied through 3 steps; using both: (i) current available information sources; and (ii) historical data obtained from a data base, these data indicate the failures occurred or predicted over a specific period of time. After the diagnostic process of the faults corrective maintenance is applied.

If a fault is predicted, preventive maintenance is applied before the occurrence of the fault. In this case, four approaches can be used: time-based or scheduled maintenance, current-state based or conditional maintenance, parameter-projection-based or forecasting maintenance, and status-based or proactive (Predictive) maintenance.

4.2 Fault Detection in Wind Turbines Using SCADA data – Literature Review

Many research works related to WTs' Condition Monitoring (CM) systems and Fault Detection and Isolation (FDI) can be found in the literature using many data analysis and data mining techniques (models) such as, Fuzzy logic, Artificial Neural Network and many other techniques, some of these research works will be discussed below.

In order to develop a WT's CMS we need data to validate the model; in modern WTs, SCADA data systems are commonly used. SCADA systems for data analysis of WTs CM are cost-effective, reliable and practical [26]. The principle of SCADA system is based on collecting extensive information from key WT subassemblies using sensors installed on the WT [24].The operational data of WT usually indicate either the WT status or measurements of signals, such as wind speed, temperature, power and current which reflect real time condition of the components of WT. By analyzing the SCADA data these signals and the different relationships between them can be observed and the condition (health) of the WT can be concluded [27].

In Schlechtingen et al. [28][, [29], WTCM system based on SCADA data, using normal behavior models and adaptive neuro-fuzzy inference systems (ANFIS) was presented. The developed CMS was designed to detect patterns in SCADA data, in addition to detect the potential failures, this system showed good performances at a variety of different SCADA signals (set of 45 signals).

It’s worth mentioning that, it is difficult to detect a fault from raw SCADA data without using an appropriate data analysis tool [29].

Another research work by Yang et al. [30] focused on developing a cost-effective and reliable CM (technique) for WT blades and drivers through interpretation of SCADA data collecting from a farm of WTs. Instead of using Instantaneous responses; a mathematical model was developed based on collecting responses through reviewing the general performance of the WT working in a range of operating conditions. The results indicated that the developed technique has powerful capabilities to detect the initial faults in WT blades and drivers, in addition to its ability in tracing the other deteriorations occur in WT blades and drivers.

In Zaher et al. [31] and Zhang et al. [32], techniques for anomaly detection in WTs based on exiting SCADA data using an artificial neural network (ANN)were used. Different parts of the WT were considered for the modeling and the analysis of potential failures; such as, cooling system and the bearing systems’ parts of WT. The results indicated that the proposed techniques for SCADA data interpretation can identify the early faults and give a WT performance assessment in order give system’s operator a sufficient time to make the needed decisions concerning machine maintenance process.

Similarly, In Godwin et al. [33], a data driven system was proposed to classify the faults associated to WT pitch through SCADA data. Data were collected from 8 WTs every 10 minutes consequently over a period of 28 months. The results culminated in developing a set of instructions/ rules which are easy to be read by the system operator; these rules can help the operator for better maintenance decisions and diagnosis for WT pitch faults.

The proposed pitch fault diagnostic system was highly accurate (87.05%) with (42.12%) reduction in WT pitch alarms.

Another research work of Kusiak et al. [34], used many data mining algorithms to develop FDI system including many WT's components; fault data were obtained by SCADA data system and fault prediction was applied through 3 levels, which include fault-no fault prediction, fault classification into categories and prediction of specific types of faults. Faults were predicted 60 minutes before their occurrence. Fault prediction model was developed using different types of data mining algorithms including, the Neural Network (NN), the Standard Classification and Regression Tree (CART), the Boosting Tree Algorithm(BTA), and the Support Vector Machine (SVM). The resulted prediction accuracy from this model was somewhat acceptable. The major constraint during the development process was with the low frequency data and the detailed description of each fault was not obvious.

Finally, in a recent research work; Borchersen and Kinnaert [35], an early model-based fault detection for the cooling system of WT's generator was proposed and tested. Model parameters were estimated on-line by Extended Kalman Filter then residuals were evaluated by CUSUM statistical change detection algorithm in order to detect the faults. The model was tested using real historical data from 43 WTs collected over a period of 3 years. The fault detection results during the test were as the following: one false alarm, 16 detections, and two missed alarms, the test results showed improvements compared to the current system but in some circumstances, the model based fault detection warns of the presence of a fault much earlier than the current alarm system. Furthermore, proposed model can be used to validate if the alarms issued from the system are false alarms or correct alarms.

In the above-mentioned literatures; different fault detection models associated with residuals extracted from SCADA data systems for different parts (components) of the WTs were proposed, then they were tested / validated to illustrate their efficiency in detecting corresponding faults.

4.3 System Identification

In order to study any system's behavior, we need to generate system's residuals. In the literature, there are many methods for generating residuals from the system being studied such as;

- Parity relations.
- State observers.
- System Identification.

In this research, system identification method along with Least Squares (LS) algorithm were used to estimate the model that describes the system (wind turbine power curve) and to generate system's residual in addition to study the behavior of the system according to the obtained artificial SCADA data from FAST-AeroDyn Simulator (experimental data).

Then, according to the experimental data, a model which describes the system behavior was formulated to explain the experimental data and allowed to make predictions of the future responses of the system (WT power curve).

Then, systematic procedures were applied to generate the system's residuals preparing for fault detection test. Figure below depicts the overall process for system identification [36], each step is illustrated in detail below.

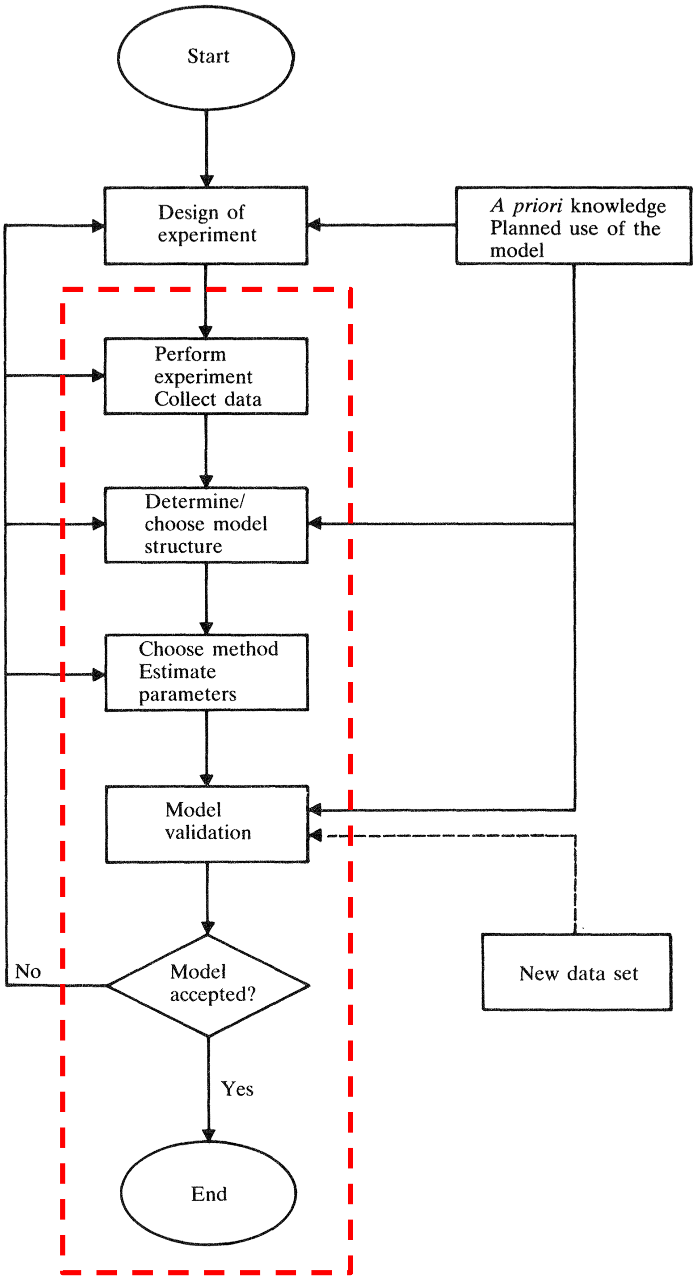


Figure 9: Schematic flowchart of system identification method, adopted from [36]

4.3.1 WT's Power Curve Model Formulation

A parametric model of WT's power curve was chosen to describe the system. Parametric models assume some finite set of parameters (θ), given the parameters, future predictions ($\hat{p}$) which are independent of the experimental data(p). These parameters are usually collected together to form a single parameter vector $\theta = [\theta_1\ \theta_2 \dots \theta_n]$ [37].

4.3.2 WT Power Curve Model Structure and Parameters Estimation

A Polynomial model was used to describe the WT's power curve as a simple empirical model, where WT's power curve modeling using polynomial expressions of different orders (degrees) is widely used in many literatures [38].

Given the power curve of m data pairs of power (p1, p2, ….., p_m) versus wind speed (w_1,w_2,…..,w_m) , an n^{th} order polynomial fitting by:

$$\hat{p}(w) = a_1 w^n + a_2 w^{n-1} + \cdots + a_n w + a_{n+1} \tag{13}$$

It's worth mentioning that the fitted model is linear in its parameters.

The main objective is to minimize the least square error between the fitted value and the actual value as:

$$\{a_i\} = \arg\min \sum_{j=1}^{m} \left\{ \left(p_j - \hat{p}(w_j)\right)^2 \right\}, \tag{14}$$

where $\{a_i\}$ is the set of polynomial coefficients, i $\in$ [1; n+1] .

4.3.3 Modeling assumptions and parameter estimation

To fulfill the abovementioned objective, the least squares algorithm (LS) was used to fit the power curve data and then to generate residual from the fitted polynomial model (residual generator).

Least squares algorithm estimates the coefficients of the model (parameters) by minimizing the summation of squares of the residuals.

Problem Statement:

- Regression model (linear in the parameters)

$$\hat{p}(k) = \varphi 1(k)\theta_1 + \varphi 2(k)\theta_2 + \ldots + \varphi n(k)\theta_n \tag{15}$$

Where:

$\theta_1\ \theta_2 \ldots \theta_n$: unknown parameters (model coefficients $a_1, a_2 \ldots . a_n$)

$\varphi 1\ \ \varphi 2 \ldots \varphi n$: known functions of known variables (w_1, w_2...w_n)

$\hat{p}$: observed (or measured) power.

- Vector form:

$$\hat{p}(k) = \varphi^T(k)\theta \tag{16}$$

Where:

$\varphi^T(k) = [\varphi 1(k)\ \varphi 2(k) \ldots \varphi n(k)]$
$\theta^T = [\theta_1\ \theta_2 \ldots \theta_n]$

- Power measured from

$$\hat{p}(k) = \varphi^T(k)\theta^0 + \xi(k) \tag{17}$$

Where:

θ^0:" true" value of the parameter vector {ξ(k)}: white noise sequence with zero mean and variance σ^2

We have a set of actual (experimental) data.

$$\{(P_m(\mathrm{k}),\varphi(\mathrm{k})),\ \mathrm{k} = 1\ .\ .\ .\ ,\mathrm{N}\}$$

In order to determine the parameters', set $\theta_1\ \theta_2 \ldots \theta_n$, in a way that model outputs have the best fit to the measurement data $\{P_m(\mathrm{k})\}$ in the least squares sense.

$$\hat{\theta} = \min \theta \sum_{i=1}^{N} (p_m(i) - \varphi(\mathrm{i})^T \theta)^2, \tag{18}$$

- Measurements vector

$$\hat{P}_m(N) = \left[\hat{P}_m(1)\ \hat{P}_m(2) \ldots \hat{P}_m(N)\right]^T \tag{19}$$

- Error vector

$$E(N) = [\xi(1)\ \xi(2) \ldots \xi(N)]^T \tag{20}$$

Where:

$$\xi(i) = p_m(i) - \hat{p}\,(i) = p_m(i) - \varphi(\mathrm{i})^T \theta \tag{21}$$

- In a matrix notation

$$\Phi(N) = \begin{bmatrix} \varphi^T(1) \\ \vdots \\ \varphi^T(N) \end{bmatrix}; P(N) = (\Phi^T(N)\Phi(\mathrm{N}))^{-1} = \left(\sum_{i=1}^{N} \varphi(\mathrm{i})\varphi(\mathrm{i})^T \right)^{-1} \tag{22}$$

- From equations (18) and (21); cost function $\left(J_N(\theta)\right)$ is calculated as the following:

$$J_N(\theta) = \sum_{i=1}^{N} \xi(i)^2 = \sum_{i=1}^{N} (p_m(i) - \varphi(\mathrm{i})^T\theta)^2 = E^T(N)E(N) \tag{23}$$

Where:

$$E(N) = P_m(N) - \Phi(\mathrm{N})\theta \tag{24}$$

Thus, value of θ which achieves the minimum cost function, is denoted by $\hat{\theta}(\mathrm{N})$, fulfils the so called "normal equation" as the following:

$$\Phi^T(N)\Phi(\mathrm{N})\hat{\theta}(N) = \Phi^T(N)P_m(N) \tag{25}$$

If $\Phi^{\mathrm{T}}(\mathrm{N})\Phi$ is non-singular, then the only value for the minimum is:

$$\hat{\theta}(N) = (\Phi^T(N)\Phi(\mathrm{N}))^{-1}\Phi^T(N)P_m(N) \tag{26}$$

Accordingly;

$$\theta = \hat{\theta} = (\Phi^T\Phi)^{-1}\Phi^T P_m \tag{27}$$

4.3.4 Power Residual Generation - Model no.1

Residuals can be calculated as the following:

$$r_i = p_i - \hat{p}_i \tag{28}$$

Where r is the power residual, p is the actual power (experimental) and $\hat{p}$ is the estimated power based on the power curve model.

Figure below depicts a sample of power residuals generated from the estimated model no.1 for fault free mode (i.e. before k=1000 where wind speed order is from 2-18 m/s and after that it's reversed from 18-2 m/s; to avoid the abrupt degradation in wind speed) and faulty operating mode with GBoxEff=90% (i.e. after k=1000 where wind speed order is from 18-2 m/s).

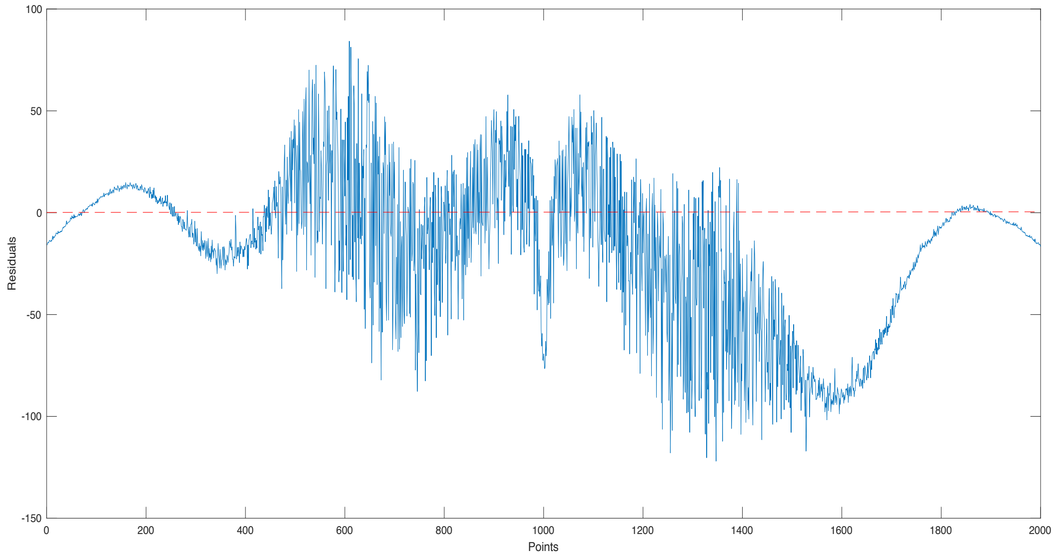


Figure 10:Sample from the Generated Residuals - Model no.1 - Fault Free and 90% GBoxEff Faulty Modes

For k below 1000; its noted that the variance of abut the first 400 points is smaller than the rest; referring to figure 13 below, this set of points located within low wind speed area in which the residuals has a small magnitude (superimposed). Reversely for k above 1000.

Table below summarizes the calculated means and standard deviations of power residuals generated from different GBoxEff percentages from model no.1.

Table 5: Means and standard deviations of power residuals generated from different GBoxEff percentages - Model no.1

GBoxEff %	Mean	Standard Deviation
100% (fault free)	0.0043	25.5353
99%	-3.1109	25.5462
98%	-6.2845	25.8824
97%	-9.5213	26.5451
90%	-34.1460	28.8795

From the above-mentioned data, it’s noticed that the shifted mean of the residuals to the negative side is increased (in value) progressively (slightly) for the GBoxEff from 100-97%, and radically for the GBoxEff from 100-90% at once.

4.3.5 Model no.1 Validation

The proposed model no.1 was validated (limited validation within the same set of data) through analyzing the goodness of fit of the regression by calculating the mean squared error (MSE) between the actual (experimental) data and the fitted model (measured).

The MSE is a measure of the quality of an estimator, it is always positive and the less value is better.

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(p_i - \hat{p}_i)^2 \qquad (29)$$

Where, n is number of data points of power, p_i is the actual power (experimental) and $\hat{p}_i$ is the estimated (measured) power.

The MSE of the fitted polynomial models with order (degree) 3 up to order 6 are illustrated in table 6 below, the minimum MSE between the experimental data and the fitting was selected. Hence 6th degree polynomial was selected.

Table 6: Best fit - polynomial degree - model no.1

	Polynomial degree			
Fault free operating mode GBoxEff 100%	3^{rd}	4^{th}	5^{th}	6^{th}
MSE	1657.3	1339.5	765.1	651.7

Thus, 6th degree polynomial model is the best fit to the experimental data and could be considered as an accepted as model no.1.

To check the normality of model no.1 residuals; the histogram below shows the distribution of the power residuals of the fitted model no.1 combined with the normal distribution overlay, and figure 12 below shows the normal probability plot of the residuals of the fitted model no.1.

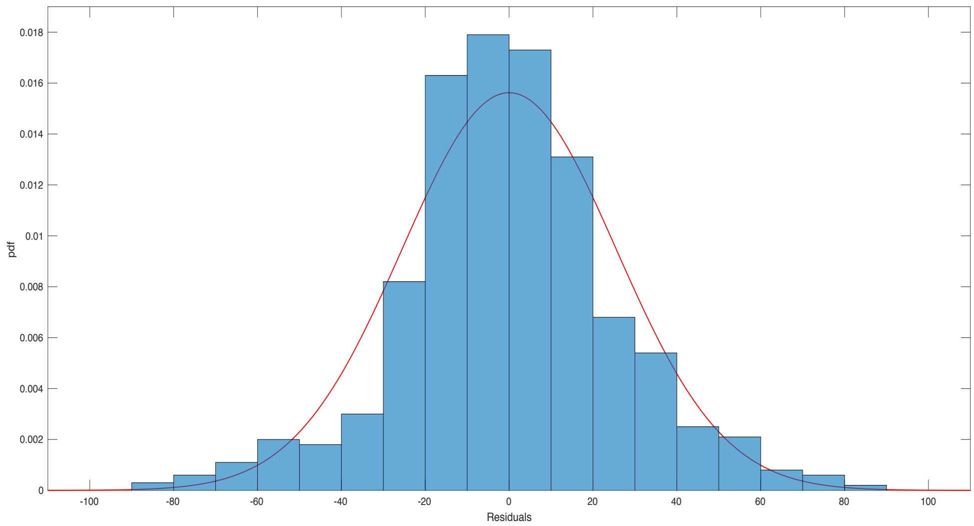


Figure 11: Residuals Normal Distribution - Model no.1

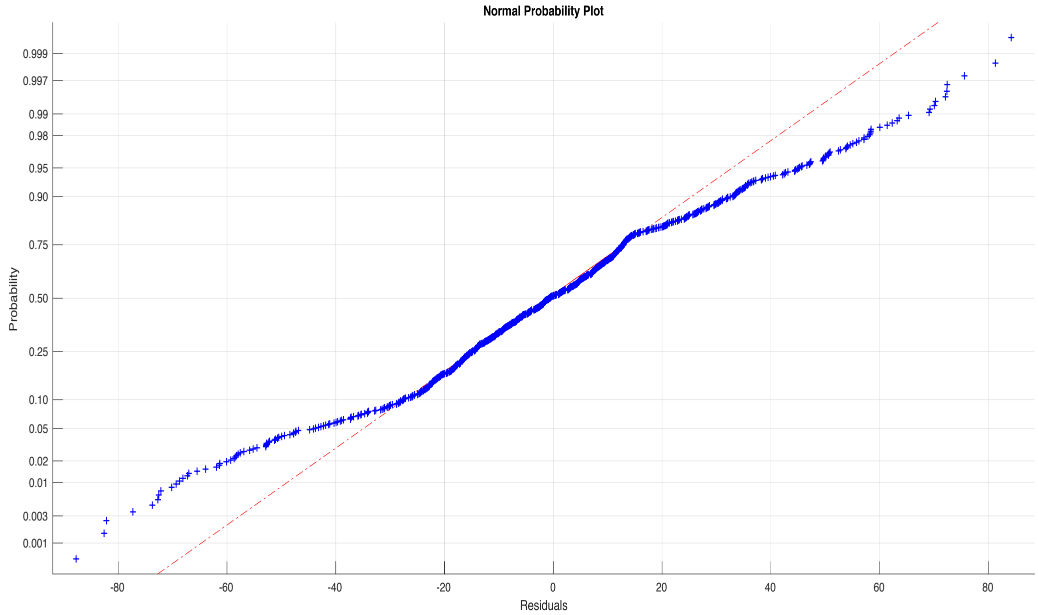


Figure 12: Normal Probability Plot - Model no.1

From the histogram and figure 12 above, most of power residuals are concentrated (superimposed) on or close to the straight line and the deviations here are mostly at the “tails” the top and bottom (highest and lowest) points and such small deviations are expected in data-based residuals.

Power residuals are assumed “approximately” to be normally distributed, with mean, $\mu = 0.0043$ and standard deviation, $\sigma = 25.5353$.

4.3.6 Mathematical Model no.1 of the identified system

Referring to equation (13), the 6^{th} degree polynomial model that fit the experimental data of the fault free operational mode of the WT is considered as the first developed model (Model 1) in this research, and described below as a function of wind speed:

$$\hat{p}(w_i) = 0.0012{w_i}^6 - 0.0523{w_i}^5 + 0.7154{w_i}^4 - 3.0313{w_i}^3 + 5.3390{w_i}^2 + 3.2694w_i + 1.2002 \quad (30)$$

Where:

$\hat{p}(w_i)$ is the estimated (measured) power, and w_i is the wind speed.

Figure below shows the estimated 6th degree polynomial model.

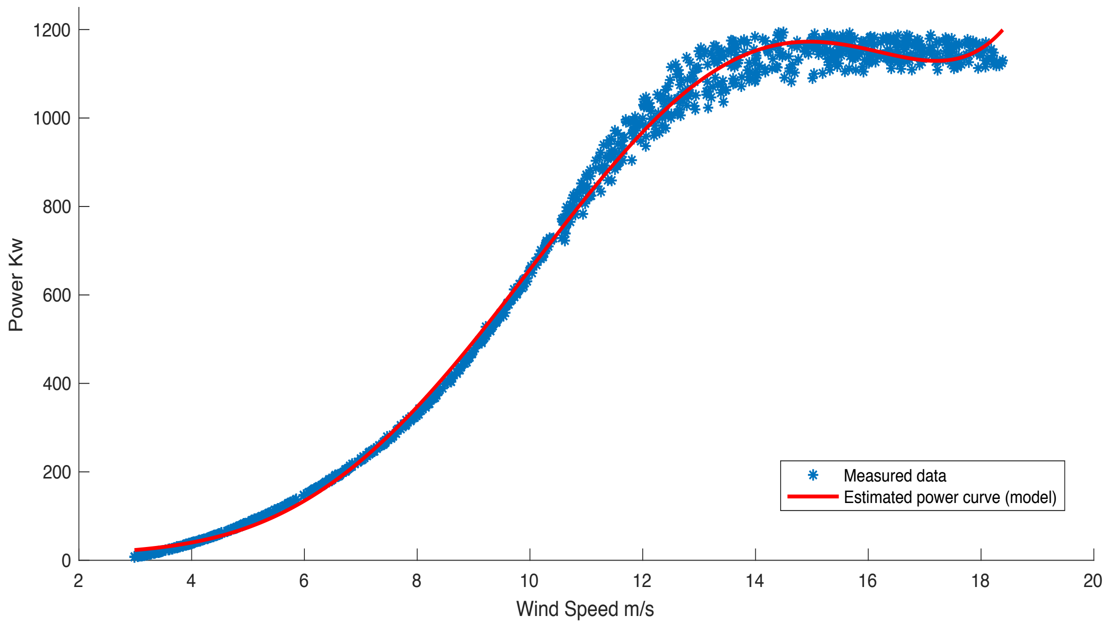


Figure 13: Power Curve Estimated Model no.1:

Fault free mode-GBoxEff=100%

4.3.7 Model no.2 - A tuned model toward the nominal power curve

Referring to the manufacturer nominal power curve (figure 4) of DeWind D6 WT, the rated power at the nominal wind speed (12.5 m/s) should be a constant to satisfy equation (11).

Thus, a new (tuned) power curve model was estimated to be more consistent with the manufacturer nominal power curve at power limitation or constant power region (C).

The actual (experimental) data for fault free mode were separated into two sets; the first set contains the active power measurements below the nominal wind speed 12.5 m/s, which includes number of power measurements i=608 data points, and the other one contains the active power measurements above the nominal wind speed and includes number of power measurements i=392 data points.

A zero-degree polynomial was fitted to power data points above nominal wind speed; these data points were averaged to get an estimation of the nominal power (rated power) as;

$$P_r = 1136.8\, Kw \tag{31}$$

For power data below the nominal wind speed, the model was estimated as the best polynomial fit with nominal wind speed as discussed below.

To achieve this, a non-linear optimization fit was performed through constrained function minimization;

$$\{a_i\} = \min \sum_{i=1}^{608} \left(p_i - \hat{p}(w_i)\right)^2 \tag{32}$$

Subjected to

$$\hat{p}(12.5) = P_r$$

where $\{a_i\}$ is the set of polynomial coefficients.

The optimization problem in (32) was simplified using penalty function to convert the constrained problem into an un-constrained problem. The general technique is to add a term to the objective function that produces a high cost for violation of constraints (i.e. $\lambda|P_r - \hat{p}(12.5)|$) as in (33);

$$\{a_i\} = \min \sum_{i=1}^{608} \left(p_i - \hat{p}(w_i)\right)^2 + \lambda|P_r - \hat{p}(12.5)| \tag{33}$$

where $\{a_i\}$ is the set of polynomial coefficients, λ is the penalty coefficient which must be much larger than the first part of the function (i.e. λ =10000).

The un-constrained problem was solved using 'fminsearch' solver in MATLAB, assuming a 6th degree polynomial model fit the power data; this assumption was based on the estimated model no.1 in (30). Moreover, the estimated coefficients of model no.1 were used to initialize the minimization of (33).

The new estimated model is a 6'th degree polynomial for data points below the nominal wind speed, and a zero-degree polynomial for data points above the nominal wind speed as illustrated below in (34) and depicted in figure14.

$$\hat{p}(w_i) = \begin{cases} 0.0015{w_i}^6 - 0.0561{w_i}^5 + 0.6970{w_i}^4 \\ - 2.8663{w_i}^3 + 6.0594{w_i}^2 - 0.0682w_i + 0.9346, & 0 \leq i \leq 608 \\ 1136.8, & 608 < i \leq 1000 \end{cases} \quad (34)$$

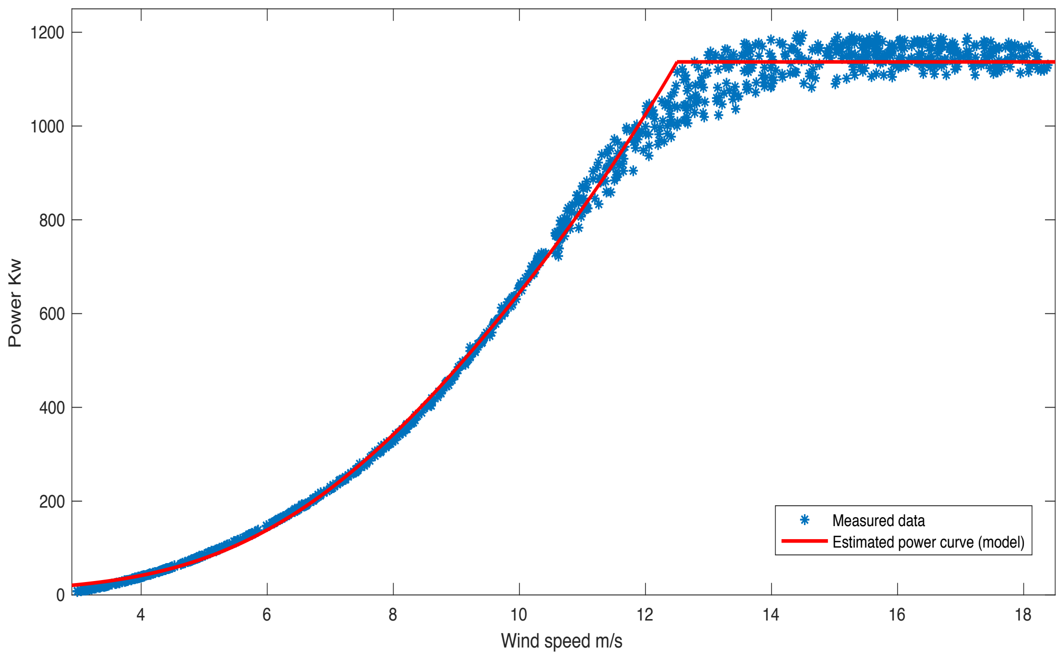


Figure 14: Power Curve Estimated Model.2:

Fault free mode-GBoxEff=100%

4.3.8 Power Residual Generation - Model no.2

Power residuals from model no.2 were calculated according to equation (28). Figure below depicts a sample of power residuals generated from the estimated model no.2 for fault free mode (i.e. before k=1000) and faulty operating mode with GBoxEff=90% (i.e. after k=1000).

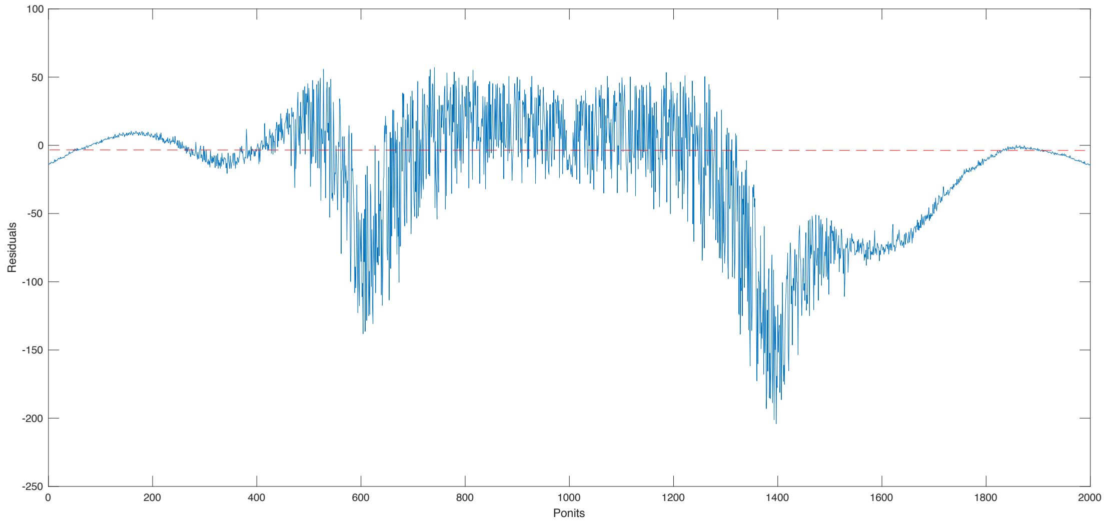


Figure 15:Sample from the Generated Residuals - Model no.2 - Fault Free and 90% GBoxEff Faulty Modes

Table below summarizes the calculated means and standard deviations of power residuals generated from different GBoxEff percentages from model no.2.

Table 7: Means and standard deviations of power residuals generated from different GBoxEff percentages - Model no.2

GBoxEff %	Mean	Standard Deviation
100% (fault free)	-1.9877	29.9365
99%	-5.1028	30.6752
98%	-8.2765	31.7026
97%	-11.5132	33.0044
90%	-36.1279	48.2066

From the above-mentioned data, it's noticed that the shifted mean of the residuals to the negative side is increased (in value) progressively for the GBoxEff from 100-97%, and radically for the GBoxEff from 100-90% at once.

4.3.9 Model no.2 Validation

The polynomial model for data below nominal wind speed was validated (limited validation within the same set of data) by using the MSE. The MSE of the fitted polynomial models with order (degree) 4 up to order 6 were calculated according to equation (29) and are illustrated in table below, the minimum MSE between the experimental data and the fitting was selected. Hence 6th degree polynomial was selected and met our assumption.

Table 8: Best fit (data point below nominal wind speed) - polynomial degree - model no.2

	Polynomial degree		
Fault free operating mode GBoxEff 100%	4^{th}	5^{th}	6^{th}
MSE	1542.04	540.2	531.1

Thus, 6^{th} degree polynomial for data below nominal wind speed and zero-degree for data above the nominal wind speed model is the best fit to the experimental data and could be considered as an accepted as model no.2.

MSE of model no.2 for the whole set of data (below and above nominal wind speed) is equal to 899.25.

To check the normality of model no.1 residuals; the histogram below shows the distribution of the power residuals of the fitted model no.2 combined with the normal distribution overlay, and figure 17 below shows the normal probability plot of the residuals of the whole fitted model no.1.

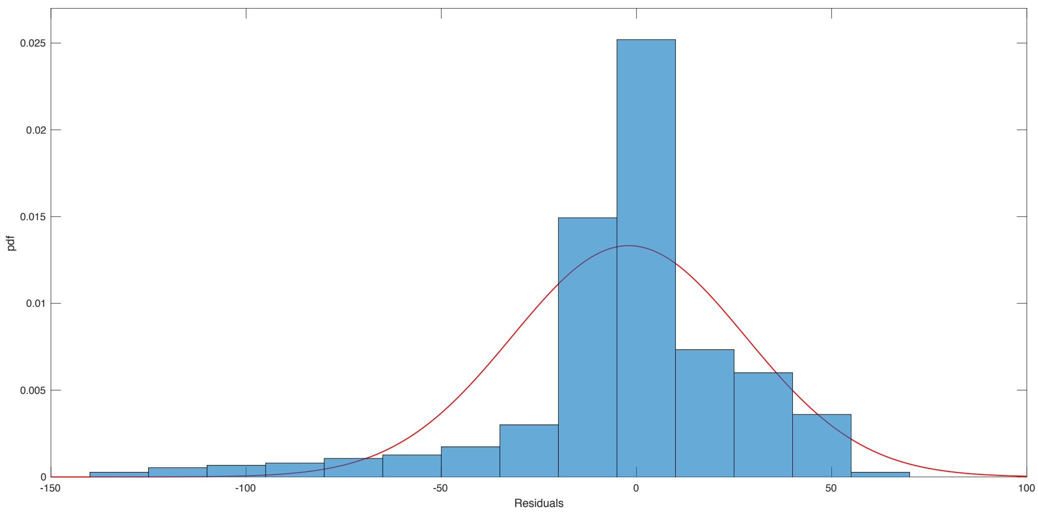


Figure 16: Residuals Normal Distribution - Model no.2

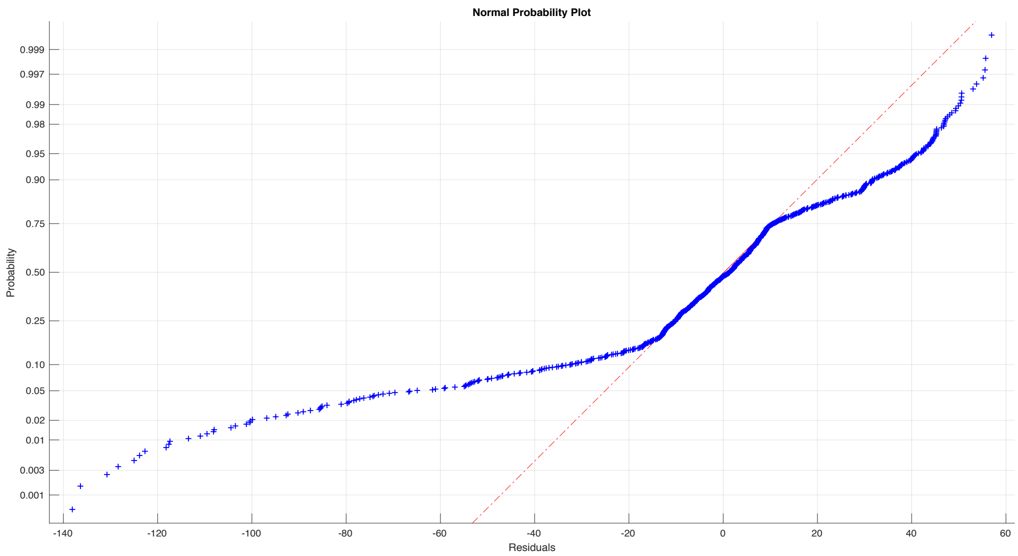


Figure 17: Normal Probability Plot - Model no.2

From the histogram and figure 17 above, power residuals (for the whole data below and above nominal wind speed) could not be assumed as normally distributed, with mean, $\mu = -1.9877$ and standard deviation, $\sigma = 29.9365$; there are too many extreme positive and negative residuals (heavy-tailed).

4.4 Fault Detection

In this section, the generated residuals from model no.1 will be evaluated using an on-line CUSUM statistical change detection algorithm to perform the fault detection (performance degradation).

4.4.1 Residual Evaluation using CUSUM Statistical Change Detection Algorithm

The general purpose of CUSUM test is to test two hypotheses $\mathcal{H}_0$ and $\mathcal{H}_1$ against each other to determine which of them describes the data, where $\mathcal{H}_0$ and $\mathcal{H}_1$ represent the fault free and faulty operating modes, respectively.

In order to detect small changes in the power residuals, the Cumulative Summation (CUSUM) statistical change detection algorithm was used, where the detection was for each single power residual point.

The power residuals vector was not evaluated at once due to the changes off-diagonal values of the residual covariance matrix which are changing significantly over time, thus, the matrix should be updated continuously. To avoid this; a positive change detection in the residuals mean was applied using CUSUM algorithm for each power residual component [35].

We have a scalar set of power residuals {r(1),...., r(k)}:

Assuming that the power residuals approximately followed a Gaussian (normal) distribution and the PFD is:

$$p_{\mu}(z) = \frac{1}{\sigma\sqrt{2\pi}} exp^{-\frac{(r-\mu)^2}{2\sigma^2}} \tag{35}$$

The hypotheses are as the following:

$\mathcal{H}_0: r(i) \sim \mathcal{N}(\mu_0, \sigma^2)\ for\ i = (1, \dots, k)$

$\mathcal{H}_1: r(i) \sim \mathcal{N}(\mu_0, \sigma^2)\ for\ i = (1, \dots, k_0)\ , r(i) \sim \mathcal{N}(\mu_1, \sigma^2)\ for\ i = (k_0, \dots, k)$

Where:

k_0 is the unknown change time.

where μ_0 and μ_1 are, the residual means before and after the possible change.

The CUSUM decision function are as the following:

$$g(k) = S(k) - m(k) \tag{36}$$

Where:

$$S(k) = \sum_{i=1}^{k} s(i) \tag{37}$$

$$s(i) = ln \frac{p_{\mu_1}(r(i))}{p_{\mu_0}(r(i))} \tag{38}$$

$$m(k) = \min_{1 \le j \le k} S(j) \tag{39}$$

From equations (38) and (41); the corresponding log-likelihood ratio $s(i)$ for detecting a change in the residual mean from μ_0 and μ_1 can be calculated as the following;

$$s(i) = \frac{\mu_1 - \mu_0}{\sigma^2}\left(r(i) - \frac{\mu_1 + \mu_0}{2}\right) \quad (40)$$

$$s(i) = \frac{b}{\sigma}\left(r(i) - \mu_0 - \frac{\beta}{2}\right) \quad (41)$$

Where:

Thus; $\beta = \mu_1 - \mu_0$ is the change in the mean and $b = \frac{\mu_1 - \mu_0}{\sigma}$ is the signal to noise ratio.

Figure below depicts (radical degradation at gearbox efficiency from model 1) the corresponding log-likelihood ratio $s(i)$ for the residuals with $\mu_0 = 0.0043$ before k=1000, μ_1=-34.146 after k=1000, $\sigma = 25.5353$ calculated from equation (40).

Noted that, the typical behavior of the log-likelihood ratio $s(i)$ shows a negative drift before change (before k=1000), and a positive drift after change (after k=1000) as depicted in figure below.

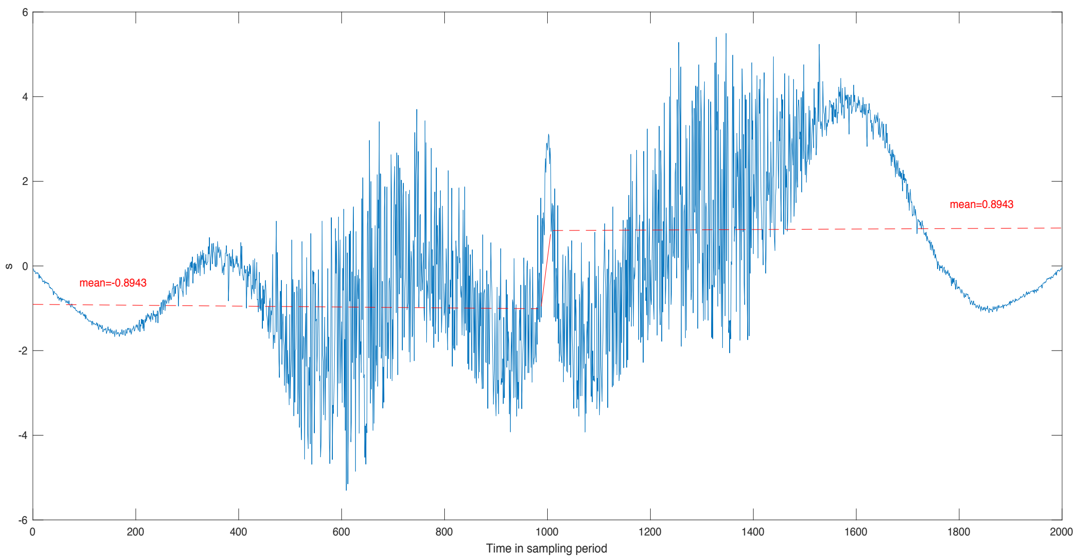


Figure 18:Log-Likelihood ratio of the Residuals (Residuals Realization), Model no.1/radical GBoxEff degradation, time on the x-axis is expressed as the number of samples

The recursive form of CUSUM algorithm is an efficient and practical way to implement the CUSUM algorithm. Depending on the fact that, the threshold 'h' is always positive; only the contributions to the cumulative sum that add up to a positive number must be considered to determine the decision function [39].

The recursive calculation of the decision function is as the following:

$$g(k) = max\,(0;\ g(k-1) + s(k)) \tag{42}$$

And the alarm function is:

$$d(k) = \begin{cases} 1, & if\ g(k) > h \\ 0, & else \end{cases} \tag{43}$$

The recursive CUSUM test was implemented on the residuals' sets generated from model no.1 only and on two levels; level one represents the progressive (slight) degradation in gear box efficiency (GBoxEff: 100-99-98-97% respectively) and the other level represents the radical degradation in gear box efficiency (GBoxEff: 100-90% at once).

To set the user-defined threshold in a way to avoid or reduce the false alarm and missed detection due to the parameters variations the user-defined threshold needs to consider the maximum magnitudes of residuals under the fault-free test, let the threshold;

$$h = 1.5 * (max\ g(k)\ before\ k = 1000) \tag{44}$$

Figure below depicts the evolution of the recursive CUSUM decision function and the user-defined threshold 'h' for the residuals of radical degradation case with μ_0 before k=1000, and with μ_1 after k=1000, calculated from (43).

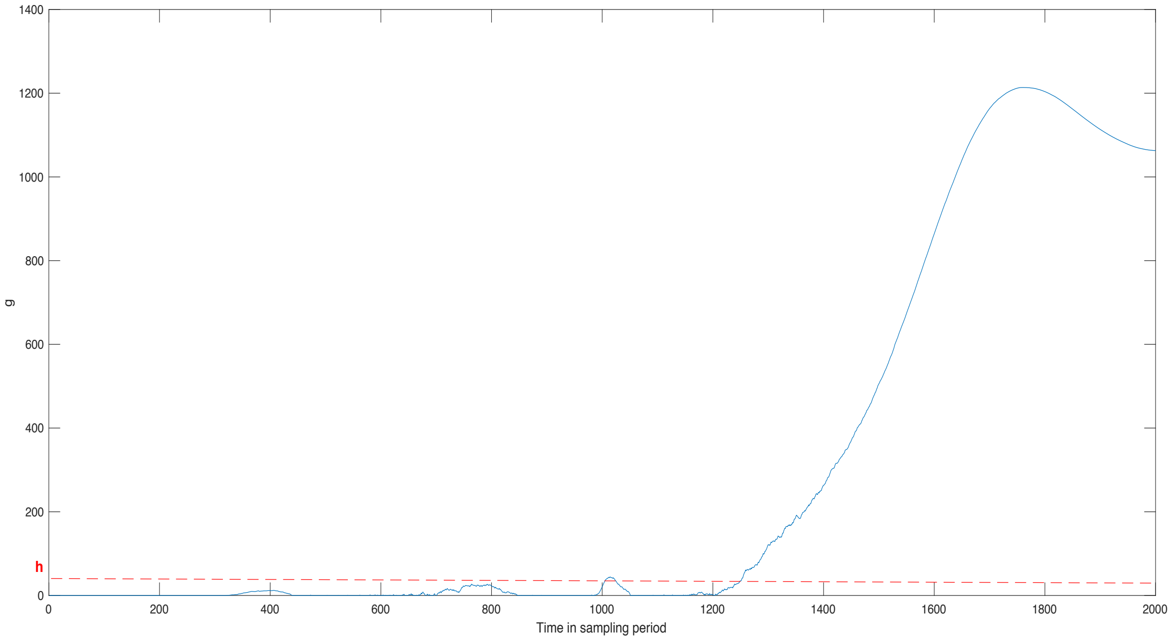


Figure 19: Evolution of the recursive CUSUM decision functions (Radical degradation, GBoxEff=100-90%), Model no.1 with μ_0=0.0043, μ_1=-34.146, σ=25.5353 and h=40.92

For the residuals of the progressive degradation; figure below depicts the evolution of the recursive CUSUM decision function and the user-defined threshold 'h' with μ_0 before k=1000, and with μ_1 for k=1001 to 4000.

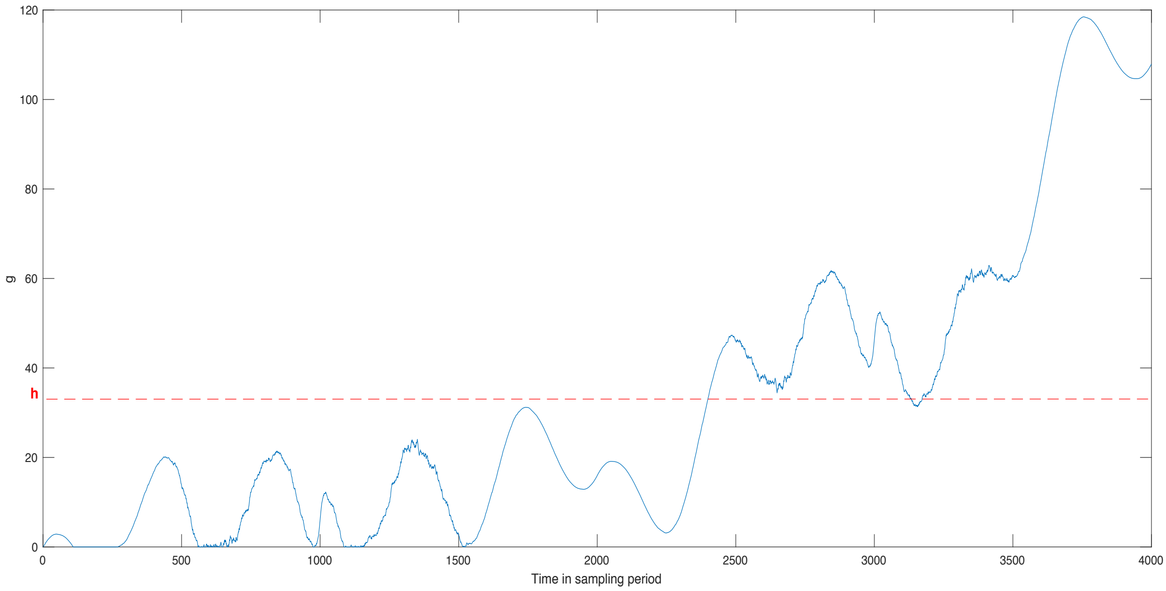


Figure 20:Evolution of the recursive CUSUM decision functions (Progressive degradation, GBoxEff=100-99-98-97%), Model no.1 with $\mu_0 = 0.0043, \mu_1 = -6.3056, \sigma = 25.5353$ and $h = 32.172$

- Where μ_1 is the average of the means for faulty modes (99,98 and 97%).

The stopping time (also called alarm time) k_a, is the time instant at which g(k) crosses the user-defined positive threshold h;

$$k_a = \min\{k: g(k) \geq h\} \tag{45}$$

The fault occurrence time k_0, can be estimated as the time instant $\hat{k}_0$ at which S(k) has changed from negative to positive slope. It is formally expressed by;

$$\hat{k}_0 = k_a - N(k_a) \tag{46}$$

Where N(k) is the number of successive observations for which the decision function remains strictly positive.

$$N(k) = N(k-1)1_{\{g(k-1)>0\}} + 1 \tag{47}$$

where 1{x} is the indicator of event x, namely, 1{x} = 1 when x is true, and 1{x} = 0 otherwise.

From (43); if g(k) > h, an alarm will be issued, an estimate of the change occurrence time $\hat{k}_0$ will be provided by (46) and the decision function will be re-initialized to 0.

The re-initialization after an alarm allowed us to check whether the change in the mean persists as time elapses or not. The result is a sequence of alarm time instants k_a and estimated change occurrence times $\hat{k}_0$ for increasing time horizon k.

Figures below depict the evolution of the recursive CUSUM decision function with re-initialization when an alarm has been issued for the radical and progressive degradation with μ_0 before k=1000, and with μ_1 after k=1000.

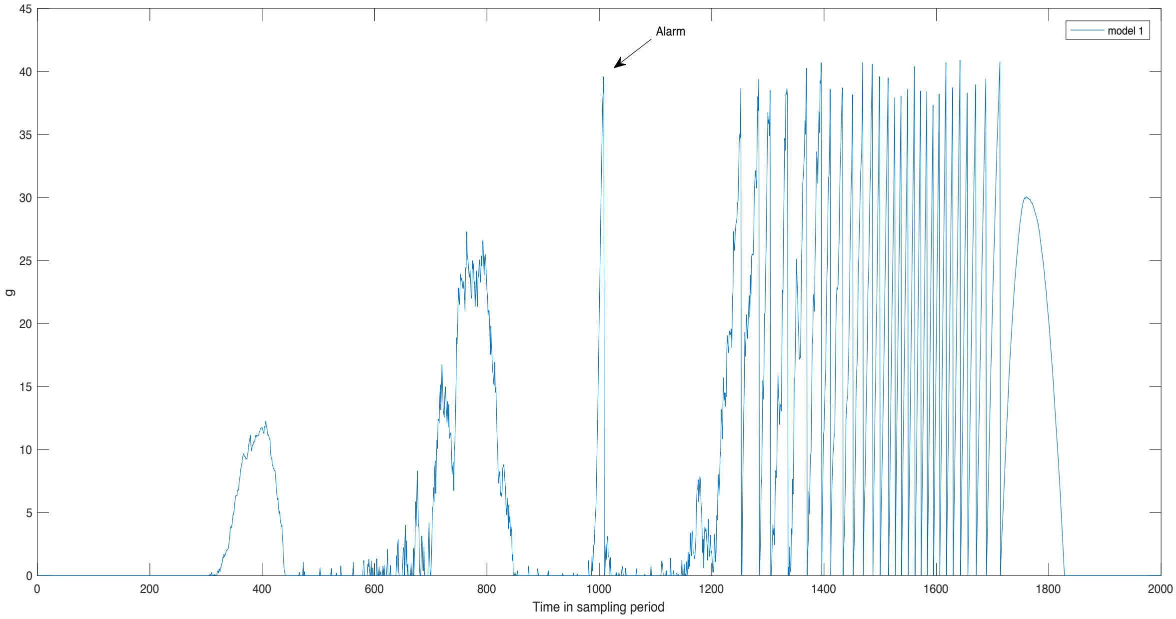


Figure 21: **Evolution of the recursive CUSUM decision functions with re-initialization (Radical degradation, GBoxEff=100-90%), Model no.1 with** $\boldsymbol{\mu_0 = 0.0043, \mu_1 = -34.146, \sigma = 25.5353}$ and $h = 40.92$

For the case of radical gearbox efficiency degradation; from equation (45), the stop alarm $k_a = 1009$, while from equations (46) and (47) the fault occurrence time estimate $\hat{k}_0 = 986$.

From figure 21 above, it's noticed that there is a regular cross of the threshold started from k=1235 due to the relative large signal to noise ratio; this indicates that there is a permanent failure occurred.

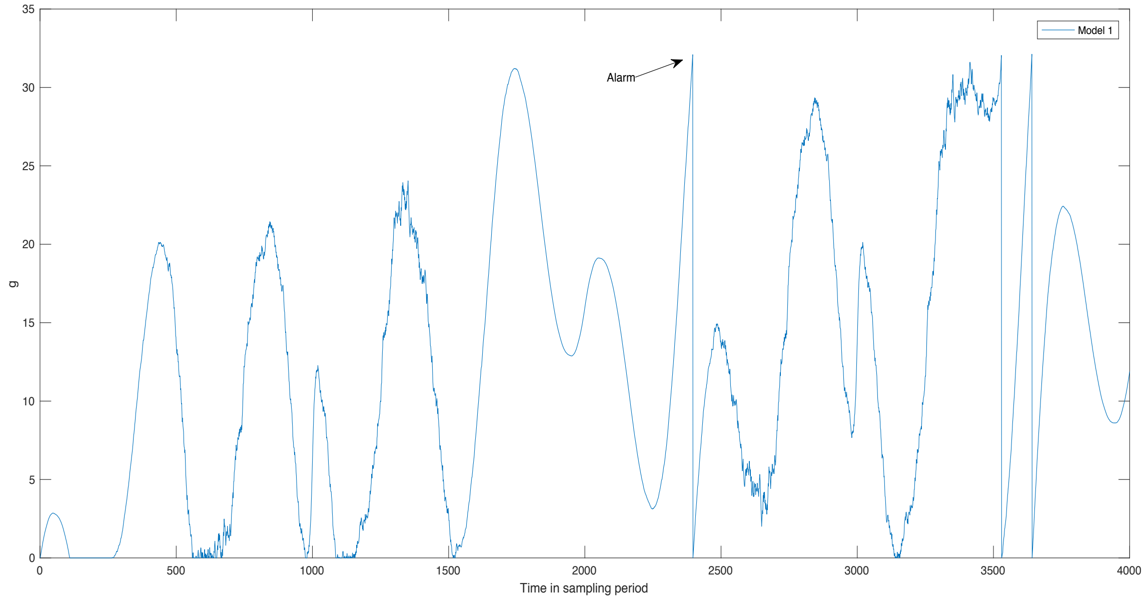


Figure 22: Evolution of the recursive CUSUM decision functions with re-initialization (Progressive degradation, GBoxEff=100-99-98-97%), Model no.1 with $\mathbf{\mu_0 = 0.0043, \mu_1 = -6.3056, \sigma = 25.5353}$ and $\mathbf{h = 32.172}$

The threshold values 'h' were calculated for each case according to equation (44).

Similarly, for the level of progressive gearbox efficiency degradation; the stop alarm $k_a = 2397$, while the fault occurrence time estimate $\hat{k}_0 = 1524$, there is no regular cross of the threshold because the signal to noise ratio is small, accordingly the algorithm need more time for detection.

As shown in figures 19 and 20, the CUSUM of the log likelihood ratio will be increasing during the fault occurrence. The more increment (higher slope), indicates the larger fault magnitude presence (larger percent of degradation in GBoxEff). In reference to the time at which the GBoxEff changed from fault free mode to faulty mode, i.e. k=1000; there is a detection time delay which illustrated in table below, this delay may be occurred due to the value of threshold h;

Table 9: Detection Time Delay of CUSUM - Model no.1

Detection time delay (time is expressed as the number of sample)	
Radical GBoxEff Degradation	Progressive GBoxEff Degradation
9	1397

This could be interpreted by the little high value of threshold h, in the same time if we set the threshold to a lower value there may be issues of false alarms. This tells us that there is a trade-off between the false alarms and detection time. From figure 21, 22 and table 9 above; It can be concluded that the larger the fault (larger percent of degradation in GBoxEff), the shorter the detection time.

The pattern of the CUSUM of the log likelihood ratio in figure 20 can be interpreted as the following;

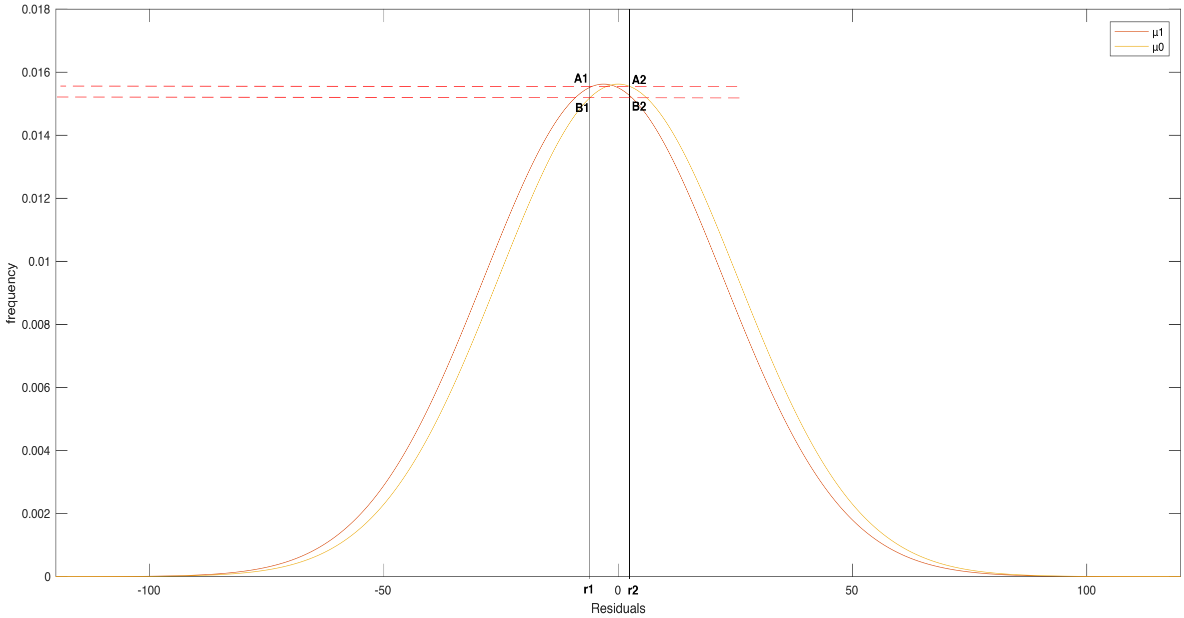


Figure 23: **Log likelihood ratio -progressive GBoxEff degradation with $\mu_0 = 0.0043, \mu_1 = -3.1109$ and $\sigma\mu_2 = -6.3056$**

The log likelihood ratio $s(i) = \frac{p_{\mu_1}(r(i))}{p_{\mu_0}(r(i))}$; for residual point (r_i) at which the CUSUM of the log likelihood ratio S(k) is increasing, the log likelihood ratio has a value $s(i) \geq 1$, and when S(k) is decreasing the log likelihood ratio has a value $1 > s(i) > 0$, since the log has a negative value for $i \in]0,1[$ and a positive value for $i \geq 1$.

A simple example depicted in figure above, for $s(r_1) = \frac{p_{\mu_1}}{p_{\mu_0}} = \frac{A1}{B1} > 1$, log sign is positive (increasing), and for $s(r_2) = \frac{p_{\mu_1}}{p_{\mu_0}} = \frac{B2}{A2} < 1$, log sign is negative (decreasing).

Chapter 5

Conclusion

Through this research work, the methodologies which were used to estimate two related models (power curves) using Artificial (simulated) SCADA data in both, fault free and faulty operating modes and to perform a model-based fault detection were presented. The main method used to estimate the first model's parameters is system identification method; polynomial model as a model structure and least squares algorithm for polynomial's parameters estimation. For the second model which is more consistent with the nominal power curve of the simulated wind turbine, non-linear optimization through unconstrained function minimization combined with penalty function is used to estimate the model's parameter, and to generate power residuals then.

Model-based fault detection was performed for the power residual generated from model no.1 and using residuals' evaluation modules based on CUSUM statistical change detection algorithm and for two different levels of gearbox efficiency degradation; the first level was for a progressive (slight) degradation in GBoxEff and the other level represented a radical or abrupt degradation in the efficiency.

The power residuals of model no.1 were assumed to be normally distributed; the small deviations residuals (see figure 12) should not affect the detection; as found in the literature, CUSUM algorithm has a high robustness to non-normality and it's effective in detect changes in all sizes, even in highly skewed and heavy-tailed process distributions.

The results of power residuals' evaluation using CUSUM algorithm have been observed for the estimated model no.1 and for all degradation levels; for progressive GBoxEff degradation level, the developed fault detection system was able to detect a fault magnitude of 2% under realistic wind turbulence. For the radical degradation level, the fault detection time was shorter than the progressive degradation; the larger the fault, the shortest detection time. Model no.1 was chosen to perform the detection due to the less MSE compared with model no.2; hence it showed the best fit of the simulated data.

Thus, it can be concluded that the artificial "simulated by FAST_NREL" SCADA data could be used as an efficient source of measurements with the absence of well-documented real SCADA data to study the behavior of a specific component of WT and perform a model-based fault detection then to help understanding the fault effects. This allows a better planning for maintenance activities in addition to give a better opportunity to implement predictive maintenance, which resulting in reduce all associated maintenance cost and increase the reliability of the WT.

As mentioned before, this work was validated for one type of fault and analyzed one fault indicator in the gearbox based on the power curve, this can be used as a framework for future to develop a model-based fault detection in WT's gearbox and to include another fault indicators which could affect the WT's performance. Further work can be done by considering full fault diagnosis study includes fault detection, isolation and estimation in addition to validate and test the efficiency of the proposed model on real SCADA data.

Chapter 6

Bibliography

[1] IEA. (2011) World Energy Outlook. [Online]. https://www.iea.org/publications/freepublications/publication/WEO2011_WEB.pdf

[2] M. Monfared, H. Rastegar, and H. Kojabadi, "A new strategy for wind speed forecasting using artificial intelligent methods," *Renewable Energy*, vol. 34, no. 7, pp. 845-848, 2009.

[3] M. Mabel and E. Fernandez, "Analysis of wind power generation and prediction using ANN: A case study’," *Renewable Energy*, vol. 33, no. 5, pp. 986-992, 2008.

[4] B. Abichou, D. Flórez , M. Sayed-Mouchaweh, and H. Toubakh, "Fault Diagnosis Methods for Wind Turbines Health Monitoring: a Review," in *The European Conference of the prognostic and health management society*, Lille, France, 2014.

[5] J. Herbert, S. Iniyan, E. Sreevalsan, and S. Rajapandian, "A review of wind energy technologies," *Renewable and Sustainable Energy Reviews*, vol. 11, no. 6, pp. 1117-1145, 2007.

[6] M. Carlberg, "Quantify Change in Wind Turbine Power Performance Using Only SCADA Data," KTH School of Industrial Engineering and Management, KTH, Sweden, MSc. Thesis 2015.

[7] S. Mohsen, M. Aval, and A. Ahadi, "Wind Turbine Fault Diagnosis Techniques and Related Algorithms," *International Journal for Renewable Energy Research*, vol. 6, no. 1, pp. 669-688, 2016.

[8] I. Munteanu, A. Bratcu, N. Cutulis, and E. Ceanga, "Optimal Control of Wind Energy Systems," *Springer-Verlag*, vol. 6, no. 1, pp. 199-221, 2008.

[9] J. Manwell, J. Mcgowan, and A. Rogers, "Wind Energy Explained. Theroy, design and application," *Wiley*, vol. 7, no. 2, pp. 211-236, 2009.

[10] M. CARRILLO, "Sensor fault diagnosis for wind-driven doubly-fed induction generators," Brussels, Belgium, PhD. Thesis 2011.

[11] P. Sun, J. Li, C. Wang, and X. Lei, "A generalized model for wind turbine anomaly identification based on SCADA data," *Applied Energy*, vol. 168, no. 2, pp. 550–567, 2016.

[12] S. Donders, "Fault Detection and Identification for Wind Turbine Systems: a closed-loop analysis," Netherlands, 2002.

[13] D. Farnoud, "Fatigue Load Control in Wind Turbines," ULB, Belgium, Master thesis 2013.

[14] J. Manwell, J. McGrowan, and A. Rogers, "Wind Energy Explained: Theory, Design and Applications," *Wiley*, vol. 8, no. 1, pp. 91-118, 2010.

[15] Green Rhino Wind Energy. (2017) Wind Characteristics. [Online]. http://www.greenrhinoenergy.com/renewable/wind/wind_characteristics.php

[16] T. Burton, D. Sharper, N. Jenkins, and E. Bossanyi, *Wind Energy Handbook,chapter 2.*: Wiley, 2001.

[17] International Electrotechnical commission (IEC), "IEC 61400-1:2005, international standard – wind turbine," 2005.

[18] B. Jonkman and M. Buhl. (2006) TurbSim User's Guide'. [Online]. http://www.nrel.gov/docs/fy06osti/39797.pdf

[19] J. Jonkman, G. Hayman, B. Jonkman, and R. Damiani. (2016) AeroDyn user guide and theory manual. [Online]. https://wind.nrel.gov/nwtc/docs/AeroDyn_Manual.pdf

[20] J. Jonkman and L. Buhl Jr. (2005, August) FAST User's Guide. [Online]. http://wind.nrel.gov/public/bjonkman/TestPage/FAST.pdf

[21] Composite Technology corporation. (2007) DeWindD6 technical characteristics brochure. [Online]. http://www.mywindpowersystem.com/usedwindturbines/wp-content/uploads/2015/06/DeWind_D6_Brochure.pdf

[22] U. Eminoglu and S. Ayasun, "Modeling and Design Optimization of Variable-Speed Wind Turbine Systems," *Energies*, vol. 7, no. 1, pp. 402-419, 2014.

[23] C. Carrillo, A. Montano, J. Cidra, and E. Diaz-Dorado, "Review of Power Curve Modelling for Wind Turbines," *Renew. Sustain. Energy Rev.*, vol. 21, no. 2, pp. 572–581, 2013.

[24] P. Tchakoua et al., "Wind Turbine Condition Monitoring: State-of-the-Art Review, New Trends, and Future Challenges," *Energies*, vol. 7, pp. 2595-2630, 2014.

[25] B. Hahn, M. Durstewitz, and K. Rohrig, "Reliability of Wind Turbines'," *In Wind Energy. Springer: Berlin/Heidelberg. Germany*, pp. 329–332, 2007.

[26] S. Butler, F. O'Connor, D. Farren, and J. Ringwood, "A Feasibility Study into Prognostics for the Main Bearing of a Wind Turbine," in *IEEE International Conference on Control Applications (CCA)*, Dubrovnik, Croatia, 2012, pp. 1092–1097.

[27] B. Chen, D. Zappalá, C. Crabtree, and P. Tavner, "Survey of Commercially Available SCADA Data Analysis Tools for Wind Turbine Health Monitoring," School of Engineering and Computing Sciences, Durham University, Durham, UK., Survey 2010.

[28] M. Schlechtingen, I. Ferreira Santos, and S. Achiche, "Wind turbine condition monitoring based on SCADA data using normal behavior models: Part 1: System description," *Applied Soft Computing*, vol. 13, pp. 259–270, 2013.

[29] M. Schlechtingen, I. Ferreira Santos, and S. Achiche, "Wind turbine condition monitoring based on SCADA data using normal behavior models: Part 2: Application examples'," *Applied Soft Computing*, vol. 14, pp. 447–260, 2014.

[30] W. Yang, R. Court, and J. Jiang, "Wind turbine condition monitoring by the approach of SCADA data analysis," *Renewable Energy*, vol. 53, pp. 365–376, 2013.

[31] A. Zaher, S. McArthur, and D. Infield, "Online wind turbine fault detection through automated SCADA data analysis’," *Wind Energy*, vol. 12, no. 6, pp. 574-593, 2009.

[32] Z. Zhang and K. Wang, "Wind turbine fault detection based on SCADA data analysis using ANN," *International Workshop of Advanced Manufacturing and Automation*, vol. 2, pp. 70–78, 2014.

[33] J. Godwin and P. Matthews, "Classification and detection of wind turbine pitch faults through SCADA data analysis," *International Journal of Prognostics and Health Management; Special Issue Wind Turbine PHM*, vol. 4, 2013.

[34] A. Kusiak and W. Li, "The prediction and diagnosis of wind turbine faults," *Renewable Energy*, vol. 36, no. 1, pp. 16-23, 2011.

[35] A. Borchersen and M. Kinnaert, "Model-based fault detection for generator cooling system in wind turbines using SCADA data," *Wind Energy*, vol. 19, no. 4, pp. 593-606, 2016.

[36] T. Soderstorm and P. Stoica, *Systems Identification’*, 2nd ed. UK: Prentice Hall International, UK, 2001.

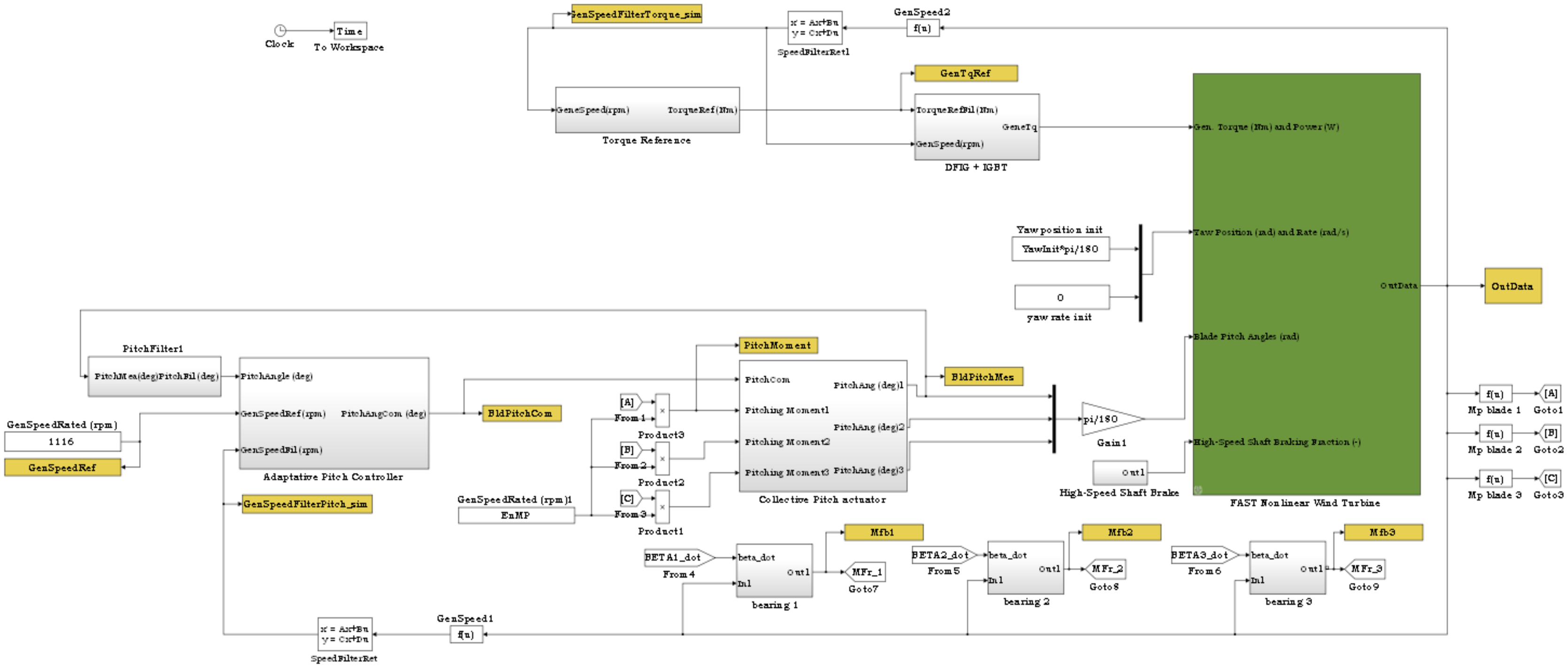
[37] O. Uluyol, G. Parthasarathy, W. Foslien, and K. Kim, "Power Curve Analytic for Wind Turbine Performance Monitoring and Prognostics," in *the Annual Conference of the Prognostics and Health Management Society, Golden Valley, USA*, 2011.

[38] M. Lydia, S. Kumar, A. Selvakumar, and G. Kumar, "A comprehensive review on wind turbine power curve modeling techniques," *Renewable and Sustainable Energy Reviews*, vol. 30, pp. 452–460, 2014.

[39] M. Blanke, M. Kinnaert, J. Lunze, and M. Staroswiecki, *Diagnosis and Fault-Tolerant Control*, 3rd ed.: Springer, 2015.

Appendix I

DeWind D6 WT's FAST-Control Scheme



Appendix II

Sample of TurbSim Hub-Height File (.hh)

```
! This hub-height wind-speed file was generated by TurbSim (v1.50, 25-Sep-2009) on 20-Mar-2017 at 10:02:41.
!
! The requested statistics for this data were:
!    Mean Total Wind Speed =   14.170 m/s
!
!  Time  HorSpd  WndDir  VerSpd  HorShr  VerShr  LnVShr  GstSpd
!  (sec)  (m/s)  (deg)  (m/s)  (-)  (-)  (-)  (m/s)
  0.000   14.17   0.00   0.00  0.000  0.300  0.000   0.00
  0.050   14.17   0.00   0.00  0.000  0.300  0.000   0.00
  0.100   14.17   0.00   0.00  0.000  0.300  0.000   0.00
  0.150   14.17   0.00   0.00  0.000  0.300  0.000   0.00
  0.200   14.17   0.00   0.00  0.000  0.300  0.000   0.00
  0.250   14.17   0.00   0.00  0.000  0.300  0.000   0.00
  0.300   14.17   0.00   0.00  0.000  0.300  0.000   0.00
  0.350   14.17   0.00   0.00  0.000  0.300  0.000   0.00
  0.400   14.17   0.00   0.00  0.000  0.300  0.000   0.00
  0.450   14.17   0.00   0.00  0.000  0.300  0.000   0.00
  0.500   14.17   0.00   0.00  0.000  0.300  0.000   0.00
  0.550   14.17   0.00   0.00  0.000  0.300  0.000   0.00
  0.600   14.17   0.00   0.00  0.000  0.300  0.000   0.00
  0.650   14.17   0.00   0.00  0.000  0.300  0.000   0.00
  0.700   14.17   0.00   0.00  0.000  0.300  0.000   0.00
  0.750   14.17   0.00   0.00  0.000  0.300  0.000   0.00
  0.800   14.17   0.00   0.00  0.000  0.300  0.000   0.00
  0.850   14.17   0.00   0.00  0.000  0.300  0.000   0.00
  0.900   14.17   0.00   0.00  0.000  0.300  0.000   0.00
  0.950   14.17   0.00   0.00  0.000  0.300  0.000   0.00
…….  …….  ……..  ……..  ……..  ……..  ………  ……
```

Appendix III

Sample of FAST-NREL main input file

------------------------- FAST v8.12.* INPUT FILE --------------------------------

FAST for Dewind D6 1.250 MW Model properties

---------------------- SIMULATION CONTROL ------------------------------------

False Echo - Echo input data to <RootName>.ech (flag)

"FATAL" AbortLevel - Error level when simulation should abort (string) {"WARNING", "SEVERE", "FATAL"}

120 TMax - Total run time (s)

0.005 DT - Recommended module time step (s)

2 InterpOrder - Interpolation order for input/output time history (-) {1=linear, 2=quadratic}

0 NumCrctn - Number of correction iterations (-) {0=explicit calculation, i.e., no corrections}

99999 DT_UJac - Time between calls to get Jacobians (s)

1E+06 UJacSclFact - Scaling factor used in Jacobians (-)

---------------------- FEATURE SWITCHES AND FLAGS ---------------------

1 CompElast - Compute structural dynamics (switch) {1=ElastoDyn; 2=ElastoDyn + BeamDyn for blades}

1 CompInflow - Compute inflow wind velocities (switch) {0=still air; 1=InflowWind; 2=external from OpenFOAM}

1 CompAero - Compute aerodynamic loads (switch) {0=None; 1=AeroDyn v14; 2=AeroDyn v15}

1 CompServo - Compute control and electrical-drive dynamics (switch) {0=None; 1=ServoDyn}

0 CompHydro - Compute hydrodynamic loads (switch) {0=None; 1=HydroDyn}

0 CompSub - Compute sub-structural dynamics (switch) {0=None; 1=SubDyn}

0 CompMooring - Compute mooring system (switch) {0=None; 1=MAP++; 2=FEAMooring; 3=MoorDyn; 4=OrcaFlex}

0 CompIce - Compute ice loads (switch) {0=None; 1=IceFloe; 2=IceDyn}

-------------------------------- INPUT FILES ---

"D6_ElastoDyn.dat" EDFile - Name of file containing ElastoDyn input parameters (quoted string)

"unused" BDBldFile(1) - Name of file containing BeamDyn input parameters for blade 1 (quoted string)

"unused" BDBldFile(2) - Name of file containing BeamDyn input parameters for blade 2 (quoted string)

"unused" BDBldFile(3) - Name of file containing BeamDyn input parameters for blade 3 (quoted string)

"Aerodyn\AD_D6_InflowWind2.ipt" InflowFile - Name of file containing inflow wind input parameters (quoted string)

"Aerodyn\AD_D6.ipt" AeroFile - Name of file containing aerodynamic input parameters (quoted string)

"D6_ServoDyn.dat" ServoFile - Name of file containing control and electrical-drive input parameters (quoted string)

"unused" HydroFile - Name of file containing hydrodynamic input parameters (quoted string)

"unused" SubFile - Name of file containing sub-structural input parameters (quoted string)

"unused" MooringFile - Name of file containing mooring system input parameters (quoted string)

"unused" IceFile - Name of file containing ice input parameters (quoted string)

------------------------------------ OUTPUT ---

True SumPrint - Print summary data to "<RootName>.sum" (flag)

1 SttsTime - Amount of time between screen status messages (s)

99999 ChkptTime - Amount of time between creating checkpoint files for potential restart (s)

"default" DT_Out - Time step for tabular output (s) (or "default")

0 TStart - Time to begin tabular output (s)

1 OutFileFmt - Format for tabular (time-marching) output file (switch) {1: text file [<RootName>.out], 2: binary file [<RootName>.outb], 3: both}

True TabDelim - Use tab delimiters in text tabular output file? (flag) {uses spaces if false}

"ES10.3E2" OutFmt - Format used for text tabular output, excluding the time channel. Resulting field should be 10 characters. (quoted string)

Appendix IV

Sample of FAST-NREL main Inflow Wind input file

---------------------- InflowWind v3.01.* INPUT FILE --------------------------

Sample InflowWind input file.

False Echo - Echo input data to <RootName>.ech (flag)

2 WindType - switch for wind file type (1=steady; 2=uniform; 3=binary TurbSim FF; 4=binary Bladed-style FF; 5=HAWC format; 6=User defined)

0 PropogationDir - Direction of wind propogation (meteoroligical rotation from aligned with X (positive rotates towards -Y) -- degrees)

1 NWindVel - Number of points to output the wind velocity (0 to 9)

0 WindVxiList - List of coordinates in the inertial X direction (m)

0 WindVyiList - List of coordinates in the inertial Y direction (m)

91 WindVziList - List of coordinates in the inertial Z direction (m)

==== Parameters for Steady Wind Conditions [used only for WindType = 1] =======

0 HWindSpeed - Horizontal windspeed

91 RefHt - Reference height for horizontal wind speed

0.3 PLexp - Power law exponent

====== Parameters for Uniform wind file [used only for WindType = 2] ========

"Wind\myTurbSimNew1.hh" Filename - Filename of time series data for uniform wind field.

91 RefHt - Reference height for horizontal wind speed

64 RefLength - Reference length for linear horizontal and vertical sheer

== Parameters for Binary TurbSim Full-Field files [used only for WindType = 3] ====

"Wind\myTurbSimNew1.hh" Filename - Name of the Full field wind file to use (.bts)

= Parameters for Binary Bladed-style Full-Field files [used only for WindType = 4] =

"Wind\myTurbSimNew1.hh" FilenameRoot - Rootname of the full-field wind file to use (.wnd, .sum)

False TowerFile - Have tower file (.twr) [flag]

=== Parameters for HAWC-format binary files [Only used with WindType = 5] ===

"wasp\Output\basic_5u.bin" FileName_u - name of the file containing the u-component fluctuating wind

"wasp\Output\basic_5v.bin" FileName_v - name of the file containing the v-component fluctuating wind "wasp\Output\basic_5w.bin" FileName_w - name of the file containing the w-component fluctuating wind

64 nx - number of grids in the x direction (in the 3 files above)

32 ny - number of grids in the y direction (in the 3 files above)

32 nz - number of grids in the z direction (in the 3 files above)

16 dx - distance (in meters) between points in the x direction

3 dy - distance (in meters) between points in the y direction

3 dz - distance (in meters) between points in the z direction

91 RefHt - reference height; the height (in meters) of the vertical center of the grid

---------------------- Scaling parameters for turbulence -------------------------------------

1 ScaleMethod - Turbulence scaling method [0 = none, 1 = direct scaling, 2 = calculate scaling factor based on a desired standard deviation]

1.0 SFx - Turbulence scaling factor for the x direction (-) [ScaleMethod=1]

1.0 SFy - Turbulence scaling factor for the y direction (-) [ScaleMethod=1]

1.0 SFz - Turbulence scaling factor for the z direction (-) [ScaleMethod=1]

12.0 SigmaFx - Turbulence standard deviation to calculate scaling from in x direction (m/s) [ScaleMethod=2]

8.0 SigmaFy - Turbulence standard deviation to calculate scaling from in y direction (m/s) [ScaleMethod=2]

2.0 SigmaFz - Turbulence standard deviation to calculate scaling from in z direction (m/s) [ScaleMethod=2]

------------- Mean wind profile parameters (added to HAWC-format files) -----------------

5.0 URef - Mean u-component wind speed at the reference height [m/s]

2 WindProfile - Wind profile type (0=constant;1=logarithmic,2=power law)

0.2 PLExp - Power law exponent [-] (used only when WindProfile=2)

0.03 Z0 - Surface roughness length [m] (used only when WindProfile=1)

================OUTPUT ================

True SumPrint - Print summary data to <RootName>.IfW.sum (flag)

OutList - The next line(s) contains a list of output parameters. See OutListParameters.xlsx for a listing of available output channels, (-)

"Wind1VelX" X-direction wind velocity at point WindList(1)

"Wind1VelY" Y-direction wind velocity at point WindList(1)

"Wind1VelZ" Z-direction wind velocity at point WindList(1)

END of input file (the word "END" must appear in the first 3 columns of this last OutList line

----