N° d'ordre : 41829

THESE

présentée en vue
d'obtenir le grade de

DOCTEUR

en

Spécialité : Automatique et Informatique Industrielle

Par

TOUBAKH Houari

DOCTORAT DELIVRE CONJOINTEMENT
PAR MINES DOUAI ET L'UNIVERSITE DE LILLE 1

Titre de la thèse :

Automated On-line Early Fault Diagnosis of Wind Turbines Based on Hybrid Dynamic Classifier

Soutenu le 27 novembre 2015 devant le jury:

Rapporteur	*Sylvie CHARBONNIER*	*Maître de conférences (HDR) à l'Université de Joseph Fourier Grenoble*
Rapporteur	*Hamid BOUCHACHIA*	*Associate Professor (HDR) at Bournemouth University, United Kingdom*
Examinateur	*Patrice BILLAUDEL*	*Professeur à l'Université de Reims Champagne-Ardenne*
Examinateur	*Bruno FRANCOIS*	*Professeur à l'Ecole Centrale de Lille*
Président	*Denis HAMAD*	*Professeur à l'Université de Littoral Côte d'Opale*
Membre invité	*Nicolas GIRARD*	*Responsable expertise à Maia Eolis*
Directeur de thèse	*Moamar SAYED-MOUCHAWEH*	*Professeur à l'Ecole des Mines de Douai*

Laboratoire(s) d'accueil : Département Informatique Automatique, Mines Douai
Ecole Doctorale SPI 072 (Lille I, Lille III, Artois, ULCO, UVHC, Centrale Lille)

ACKNOWLEDGEMENTS

Je remercie le Conseil Régional Nord Pas de Calais et l'Ecole des Mines de Douai pour leur participation à mes travaux de recherche, participation sans laquelle mon doctorat n'aurait pu être mis en place

Contents

List of Tables

List of Figures

Chapter 1

General introduction

Contents

1.1 Context and motivations

The number and complexity of industrial wind turbine installations have increased significantly over the last decades. The main focus of current studies of Wind Turbines (WTs) is to reduce the cost of energy in order to ensure that the wind generated electricity is competitive with the other generation sources. Operational & Maintenance (O&M) costs constitute a significant share of the annual cost of WTs downtime. Analyses of WT farm maintenance costs show that up to 40% of these costs is related to unexpected component failures which lead to costly unscheduled repairs [28]. Several studies [105] reveal that the effective association between cost of energy and O&M costs leads to a profitable operation of wind turbine. Therefore, increasing the availability and optimizing the maintenance process are crucial tasks from an industrial perspective in order to obtain a significant reduction of revenue losses.

The wind turbine system is composed of several subsystems as the pitch system, the drive train, the generator and the power converter. Faults occurring in some of these components impact significantly the availability of WTs to produce electricity and increase the maintenance costs. This is due to their high failure rate and/or their downtime. Therefore, early fault diagnosis of these critical components can enhance significantly the WT availability and reduce their maintenance costs. The pitch system and power converter are examples of these WT critical components. They are used to optimize the energy production and to keep it constant at its optimal value. Moreover, the WT must be shut down when the wind speed is too high in order to ensure WT safety. This task is accomplished by the controller based on the use of the pitch system. Therefore, faults in the pitch system and the power converter result in costly turbine down-time and contribute significantly to WT vulnerability. In addition, pitch system and power converter faults produce a large amount of alarms in the control center. This increases the mental task of human operators of supervision by analyzing a huge number of alarms. Hence, being able to correctly diagnose these components faults at early stage can increase wind turbine availability and reliability and reduce its maintenance costs to a great extent.

Faults impacting a component can be either discrete impacting the configuration or the discrete mode of the component or parametric affecting its continuous dynamics. Sensors or actuators stuck-on or stuck-off are examples of discrete faults. Abnormal deviation in the nominal values of resistors or capacitors is an example of parametric faults. This abnormal deviation from the nominal value decreases the ability (performance) of the component (e.g., resistor, capacitor) to accomplish its task. Parametric faults occur often in progressive manner. Their value starts to deviate from its nominal value over time leading to decrease progressively the performance until arriving to unacceptable predefined level or value leading to activate an alarm. These faults are generally named as drift-like faults.They entail an evolution in the WT normal operating conditions to a failure through degraded operating conditions. They are intrinsic changes in the property of the system, which make it evolve and change its dynamics. Therefore, Detecting the drift (degradation) in early stage helps to reduce the maintenance costs and to increase the WT availability.

Consequently, on-line early drift-like fault diagnosis of critical WTs components is crucial in order to ensure optimal and safe operation in spite of faults impacting WTs performance. However, this is a challenging task because [79], 1) the measurements of wind turbines are not enough reliable due to the high uncertainty of wind speed and to the turbulence around the rotor plane, 2) the non-linearity of the wind turbine dynamics, 3) the occurrence of certain faults (e.g., blade pitch motor faults) in operation conditions (power optimization region) in which fault consequences are hidden, 3) the actions of the control feedback which compensate the fault effects and 4) the low volume of data (imbalance data) describing the faults according to the data coming from normal operation conditions which makes the fault prediction task difficult.

1.2 Contributions

In the literature,there are several methods [30],[55],[57],[66],[96],[41],[74] that are used to achieve fault diagnosis in WTs. They achieve the fault diagnosis by reasoning over differences between desired or expected behavior, defined by a model, and observed behavior provided by sensors. They can be classified into two main categories of methods: internal and external methods [79]. The internal methods [95] use a mathematical or structural model to represent the relationships between measurable variables by exploiting the physical knowledge or/and experimental data about the system dynamics. However, they suffer from the necessity to depth information about system behavior and failures which is hard to obtain for complex and strong non-stationary systems as wind turbines. An alternative to overcome this problem is the external methods [55]. They consider the system as a black box and use exclusively a set of measurements or/and heuristic knowledge about system dynamics to build a mapping from the measurement space into a decision space.

Therefore, the contributions of this thesis focus on the use of external methods in particular machine learning and data mining approaches.

Although machine learning and data mining approaches have been applied successfully to the fault diagnosis of WTs, they suffer from some major drawbacks. Firstly, they require a priori enough and representative knowledge (data) about all faulty behaviors, 2) they require a discriminant representation or feature space sensitive to WT normal operation conditions and each of the faulty behaviors and 3) they do not integrate a mechanism to detect incipient (drift-like) faults in their early stage. Consequently, this thesis dissertation proposes an on-line adaptive machine learning and data mining scheme in order to achieve the drift-like fault diagnosis in WTs, in particular pitch system and power converter. This scheme is composed of five main steps: processing and data analysis, classifier design, drift monitoring and updating and interpretation steps.

The proposed scheme is based on the decomposition of the wind turbine into several components. Then, a classifier is designed and used to achieve the diagnosis of faults impacting each component. The goal of this decomposition into components is twofold: 1) to facilitate the isolation of faults and 2) to increase the robustness of the scheme in the sense that when the classifier related to one component is failed, the classifiers for the other components continue to achieve the diagnosis for faults in their corresponding components. This scheme has also the advantage to take into account the WT hybrid dynamics. Indeed, some WT components (as pitch system and power converter) manifest both discrete and continuous dynamic behaviors. In each discrete mode, or a configuration, different continuous dynamics are defined. Defining a feature space in each of these discrete modes may allow to increase the discrimination power (sensitivity) of the corresponding features to the components normal and/or failure operation conditions. Finally, this scheme can consider only data samples about normal operation conditions. Any drift from the characteristics representing these normal operation conditions is considered as an evolution towards a failure. When a failure is confirmed, the data samples representing this failure are used to update the classifier structure by integrating a new class to its data base. This helps to overcome the problem of imbalanced data or the absence of data about some faults in a WT component.

The specific contributions of this dissertation are as follows (see Figure 1.1):

- *A generic on-line and adaptive machine learning and data mining scheme* in the sense that any machine learning (supervised and unsupervised learning) and data mining (feature selection and extraction, etc.) can be used. A mechanism based on the use of a set of drift indicators is used in order to detect a drift and to confirm it. These indicators observe a serious change in the characteristics of the data samples representing the WT components normal operation conditions. Finally, an expert will be asked to provide an interpretation to the detected changes or drift. This interpretation is then used as a short-term prediction about the tendency of the future development of the current situation. This prediction may be useful to formulate a control action to modify the dynamics of the WT in order to accommodate the fault consequences.

- *A hybrid dynamic classifier* that able to change its decision function as well as its feature space according to the system internal state (discrete mode). This allows to keep the useful patterns representative of the drift and therefore to detect it in its early stage. Indeed, when a drift starts to occur in one discrete mode, its consequences may manifest within this discrete mode. However when the WT component changes its discrete mode, the drift consequences may not be visible and therefore the data samples within that discrete mode are not useful to detect the drift. Moreover, these data samples may delay the drift detection time. This is because they impact adversely the representativeness or usefulness of the data samples gathered during the discrete modes where the drift consequences are visible.

1.3 Organization

The structure of the thesis manuscript is as follows:

- **Chapter 2: Wind turbine fault diagnosis.** In this chapter, the WT description and the interests, motivations and challenges of achieving the diagnosis of faults impacting its performance are presented. Then, the different methods of the literature used to achieve the fault diagnosis of WTs are studied and compared. The goal is to focus the research in this manuscript on the category of methods allowing to answer the challenges of WTs fault diagnosis and to reach the goals related to their operational and maintenance costs as well as their availability and safety. This alternative is based on the use of on-line and adaptive learning scheme allowing achieving an early fault diagnosis for critical wind turbine components. Therefore, a review of on-line and adaptive machine learning methods is presented in order to define the framework and the structure of the scheme to be used to achieve an on-line and early diagnosis of faults impacting the performance of WT critical components.

- **Chapter 3: Hybrid dynamic classifier for simple and multiple drift-like faults diagnosis in pitch system.** This chapter presents the first contribution of this thesis which is an approach to achieve the drift like fault diagnosis of pitch system. The latter comprises two redundant sensors and one actuator for each of the three vertical blades of the wind turbine. The pitch system controller controls the angle of attack of the blades to the wind in order to extract a maximum of kinetic energy and to avoid rotor over-speed at high winds speed. Therefore, the pitch system has two different control modes according to the wind speed. In the first control mode, the normal and failure operation conditions cannot be discriminated because of the small pitch angles and the high variability of wind speed. Likewise, the normal and failure behaviors of pitch actuators cannot be separated because the actuators are not active (powered on) since the pitch angle is maintained at 0 degree. While, in the second control mode, the normal and failure operation conditions are separated. The developed approach in this chapter diagnoses the faults impacting the normal behavior of pitch system sensors and actuators. To achieve that, two feature spaces are used: the first feature space is sensitive to the normal operating conditions of the pitch system sensors; while the second feature space is sensitive to the pitch system actuator normal behavior. Two drift indicators are used in order to detect the evolution (degradations) of the normal operation conditions of pitch system (sensors, actuators). Only the patterns gathered when the pitch system is in control mode 2 are used since in the latter the normal and failure behaviors can be separated. These patterns represent the potential evolving class. The first drift indicator is based on the use of Euclidean distance between the gravity centers of the normal and evolving classes; while the other drift indicator is based on the use of Mahalanobis distance between the normal class patterns and the gravity center of the evolving class. The drift-like fault in pitch system impacts all the features of the feature space. This justifies the use of the distance between the normal and evolving classes according to all the features. The interest of these two indicators is that the Mahalanobis indicator is used to detect a drift and the Euclidean indicator to confirm it. Indeed Mahalanobis distance is more sensitive to low speed drifts since it takes into account all the patterns of the normal class.

- **Chapter 4: Hybrid dynamic classifier for single and multiple drift-like faults diagnosis in power converters.** The approach developed in this chapter presents the second contribution of this thesis which aims at achieving a drift like fault diagnosis of WT electronic power converter. The latter controls the flow of current (electrical energy) from the generator by adjusting its frequency. The power converter has several different discrete modes. The parameters describing the continuous dynamics in each mode depends on the discrete mode in which the power converter is. Therefore, the parameters sensitive to a certain parametric fault depend on the power converter discrete mode. Thus, the features of the feature space sensitive to a certain parametric fault depend on the power converter discrete mode. Consequently, the developed approach in this chapter defines a feature space in response to the power converter discrete mode. The drift (degradation) indicator is defined for each sensitive feature based on the use of the Euclidean distance between the gravity centers of normal and evolving classes. When a drift is detected by one indicator (according to one sensitive feature in a discrete mode), this drift can be then confirmed and its source (the degraded capacitor) isolated by another drift indicator (according to another sensitive feature). The proposed approach in this chapter is also used to achieve the multiple drift like faults detection and isolation since the multiple faults involve a drift according to several features, each is sensitive to one element (one power converter capacitor). This is because each feature is sensitive to a drift generated by one element (e.g., one capacitor in the power converter).

The benchmark developed by [73] is used to generate the fault scenarios in pitch system. This benchmark simulates a realistic generic three blade horizontal variable speed wind turbine with a full power converter coupling. However, this benchmark is developed in this thesis in order to generate drift-like faults scenarios in the pitch system sensors and actuators. Moreover, the power converter used in this benchmark is modeled by a first order transfer function.In order to generate the drift-like faults impacting the intrinsic parameters (nominal values of capacitors) of the power converter, a benchmark of three cell converter adapted to WT is developed in this thesis using Matlab-Simulink environment and Stateflow toolbox.

- **Chapter 5: Conclusion and future work.** This chapter summarizes the contributions of this dissertation, discuss their limitations and presents the future directions in order to improve the proposed approaches.

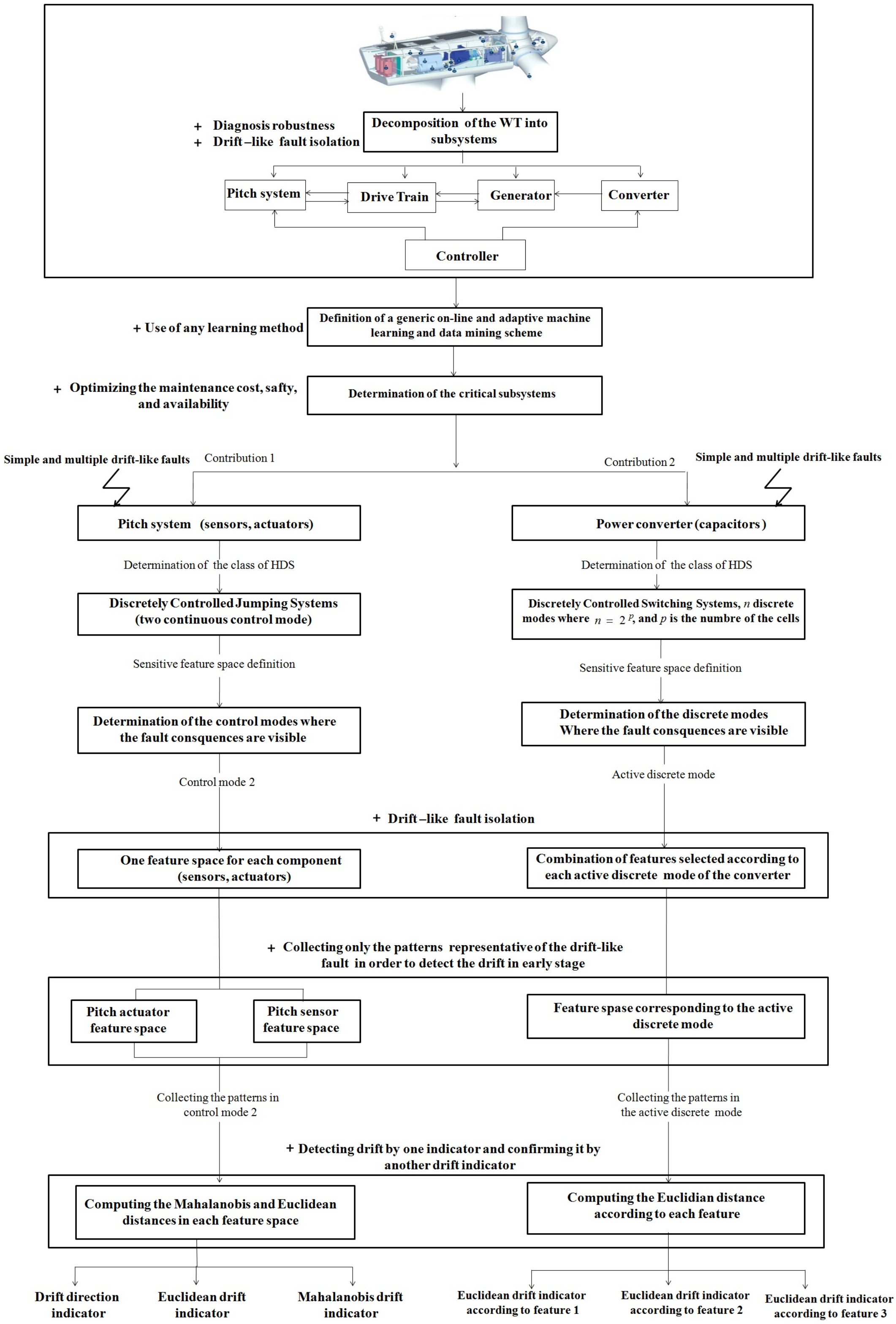

Figure 1.1: Global scheme for the thesis's contributions.

1.4 List of publications

Journal papers

- H. Toubakh, M. Sayed-Mouchaweh. Hybrid Dynamic Classifier for Drift-like Fault Diagnosis in a Class of Hybrid Dynamic Systems: Application to Wind Turbine Converters. Neurocomputing, Elsevier, 171:1496-1516, 2015.

- H. Toubakh, M. Sayed-Mouchaweh. Hybrid dynamic data-driven approach for drift-like fault detection in wind turbines. Evolving Systems, Springer, 6:115-129, 2014.

Conference papers

- H.Toubakh, M.Sayed-Mouchaweh, A.Fleury and J.Boonaert. Hybrid dynamic data mining scheme for drift-like fault diagnosis in multicellular converters. In Third International Conference on Technological Advances in Electrical, Electronics and Computer Engineering (TAEECE), Beyrouth, Lebanon, IEEE, pp. 56-61, 2015.

- H. Toubakh, M.Sayed-Mouchaweh. Advanced data mining approach for wind turbines fault prediction. In Proceedings of second European conference of the prognostics and health management society, Nantes, France, Vol. 5: pp. 288-296, 2014.

- B. Abichou, D. Flôrez, M. Sayed-Mouchaweh, H. Toubakh, B. Franois, N. Girard. Fault Diagnosis Methods for Wind Turbines Health Monitoring: a Review. In Proceedings of second European conference of the prognostics and health management society, Nantes, France, Vol.5: pp. 297-304, 2014.

- H. Toubakh, M. Sayed-Mouchaweh and E. Duviella. Advanced pattern recognition approach for fault diagnosis of wind turbines. In Machine Learning and Applications (ICMLA), 2013 12th International Conference, Miami, USA, IEEE, pp. 368-373, 2013.

Research seminars

- H. Toubakh, M. Sayed-Mouchaweh. Hybrid Dynamic Classifier for Single and Multiple Drift-like Fault Diagnosis in a Class of Hybrid Dynamic Systems: Application to Wind Turbine Converters. Réunion commune groupe de travail S3 et H2M, Grenoble, France, 2015.

- H. Toubakh, M. Sayed-Mouchaweh. Hybrid Classifier for Drift-like Fault Diagnosis in Wind Turbine Converters. In 11^{th} European Academy of Wind Energy (EAWE) PhD Seminar, Stuttgart, Germany, 2015.

- H.Toubakh, M.Sayed-Mouchaweh, E.Duviella. Diagnostic adaptatif et prédictif pour la maintenance prévisionnelle des système dynamiques complexes: application à un système éolien, 2 éme Journée Régionales des Doctorants en Automatique (JRDA), Valencienne, France, 2013.

Chapter 2

Wind turbine fault diagnosis

Contents

2.1 Background and definitions

Wind turbines (WTs) are mechanical devices that convert the kinetic energy of wind into the electrical energy through a rotating generator. In general, WTs can be classified into vertical-axis and horizontal-axis ones according to the position of WT rotor,see Figure 2.1. This manuscript focuses on horizontal-axis WTs since they are the most used ones. They use a three-bladed rotor design with an active yaw system keeping the rotor oriented upwind [28].

Wind turbines operate, generally, in severe and remote environments and require frequent schedule maintenance. In addition, the tower height and the rotor size become larger to capture more energy. This makes the inspection and maintenance task more difficult and the turbine more sensitive and vulnerable to wind speed. Therefore, it is essential to reduce the costs related to the WT operations and maintenance (O&M) in order to increase the competitiveness of this clean energy source according to the traditional ones. Indeed, O&M costs may reach 25% to 30% of the energy generation cost [58]. One of the main sources for the O&M costs

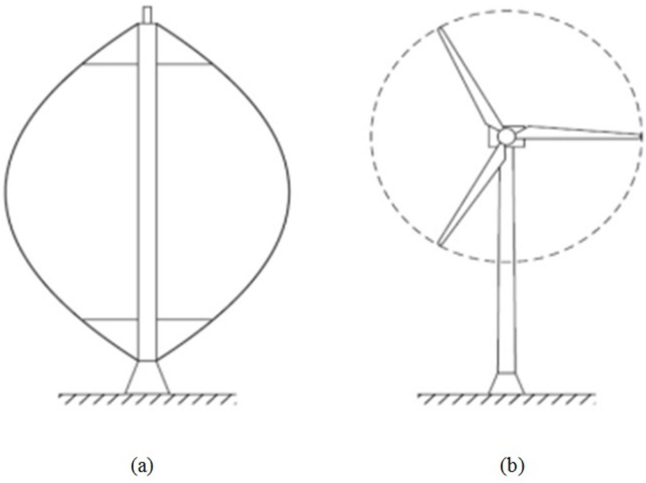

Figure 2.1: (a) Vertical-axis WTs (b) Horizon-axis WTs.

is the unscheduled maintenance due to unexpected failures. This can be costly not only for the maintenance support but also for the produced energy. Moreover, the accidents, in particular the fatal ones, of WTs increases year over year [83]. Therefore, an automated health monitoring system can reduce the O&M costs as well as the lost production time and ensure the WTs security and safety by detecting and isolating faults before becoming expensive, critical or catastrophic.

Operational state of a WT varies from fully operational to malfunction and shutdown. Their monitoring can be achieved by human operators of supervision using SCADA (Supervisory Control and Data Acquisition) system. The latter is a standard installation on large WTs where its data is collected continuously from the sensors fitted to the different WTs as well as their subassemblies (components). SCADA system [54] records the values of multiple operational and environmental parameters as well as systems potential or emerging faults. The SCADA data coming from the individual WTs in a wind farm is transmitted to a central point in order to allow human operators of supervision to monitor the health status and performance of these WTs. Although SCADA data is a rich resource about the health state of WTs, the human operators of supervision need to analyze a huge amount of data, historical alarms and detailed fault logs in order to schedule efficient and optimal maintenance actions. Moreover, SCADA system does not allow to achieve a precise localization of affected components. Therefore, several components of WT are suspected and additional time is required to isolate the component responsible of the occurrence of this fault. This will lead to increase the time of WT unavailability as well as its cost maintenance. To overcome this problem, Condition Monitoring System (CMS) is used. The latter allows to record data at much higher data rates. However, its cost is much higher than SCADA system due to the higher sampling rate and installation as well as to the additional processing, storing and analyzing costs. Moreover, the analyzing and monitoring tasks of human operators of supervision will be much more difficult due to the avalanche of data coming from SCADA and CMS systems. Consequently, the design of an on-line fault diagnosis system allowing to analyze automatically the huge amount of data (SCADA and CMS) and to detect the occurrence of a fault in early stage and to isolate its source (component) is essential to reduce significantly the operational and maintenance costs and to increase the availability and safety of WTs.

In this chapter, the different methods of the literature used to achieve the fault diagnosis of WTs are studied and compared. The goal is to focus the research in this manuscript on the category of methods allowing to answer the challenges of WTs fault diagnosis and to reach the goals related to their operational and maintenance costs as well as their availability and safety.

2.1.1 Wind turbine description

As we have seen in the introduction, the most recent and used WTs are horizontal-axis based with three blades. Having the rotor positioned on the top of the tower creates a more efficient system as more wind energy is produced. These turbines also have a nacelle, which is held up by the tower and contains the gearbox and the generator. A yaw system, which is turning the nacelle and rotor to face the wind, enables the turbine to capture the highest amount of energy. Figure 2.2 shows the components involved in a three bladed horizontal-axis WT.

A brief description of WT components is given below see Figure 2.2:

- **Blades:** capture the wind energy.
- **Pitch system:** composed of one actuator and two redundant pitch angle sensors : controls the direction of the blades to face the wind.
- **Low-speed shaft:**is the axe rotated by the rotor.
- **High-speed shaft:**drives the generator.
- **Drive train:** increases the speed of the low-speed shaft to a suitable value that is required by the electricity generator.
- **Generator:** is a device that converts the mechanical energy to electrical energy.
- **Converter:** controls the speed of the generator by adjusting the electrical power frequency in order to optimize the energy production.
- **Controller:** controls the pitch angle of the blades as well as well as the angular speed of the generator in response to the current wind speed in order to optimize the energy production and to ensure safety.
- **Transformer:** is used for the grid integration of wind power.

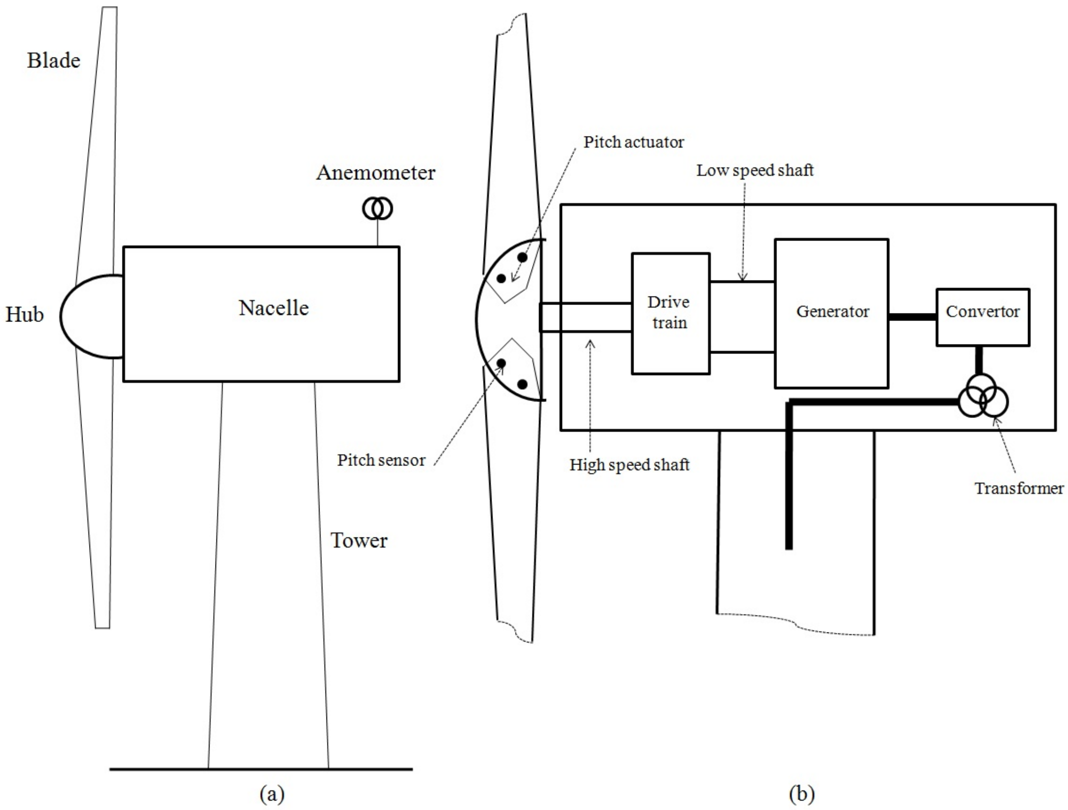

Figure 2.2: (a) General outline of the WT seen from the outside. (b) Major parts of the WT seen from the inside.

- **Nacelle:** is a large cover on the top of the tower used to protect the mechanical transmission system.
- **Anemometer:** measures the wind speed and conveys it to the controller.
- **Tower:** made from steel lattice or tubular steel. As the wind speed increases with height, taller towers capture more energy and generate more electricity.

For the exploitation of the energy supplied by the wind, several designs of horizontal axis WTs can be used. We can classify them into two categories:

2.1.1.1 Fixed speed wind turbine generator

Usually equipped with a squirrel cage induction generator SCIG [27], whose speed variations are limited as it is shown in Figure 2.3. Power can only be controlled through pitch angle variations [27].

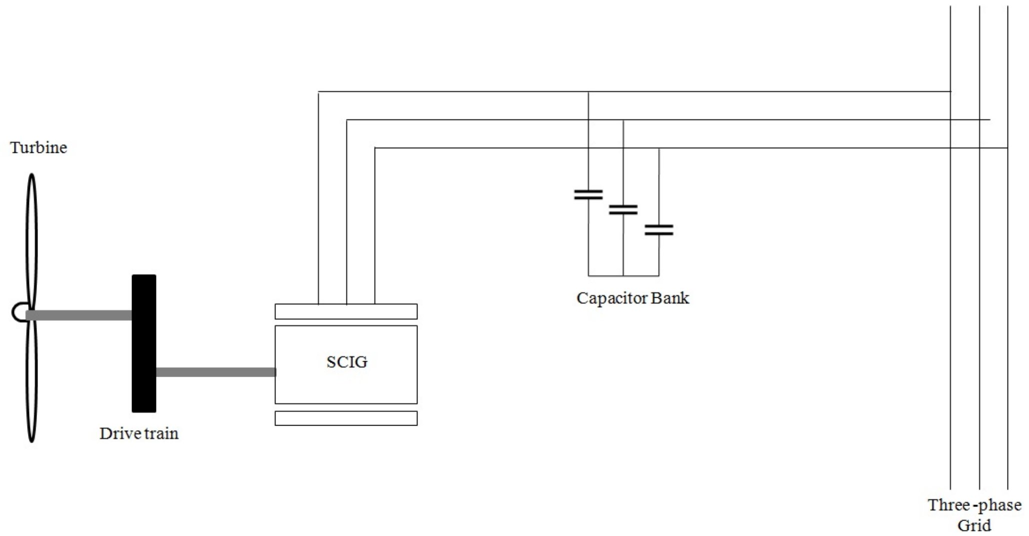

Figure 2.3: Fixed speed induction generator.

2.1.1.2 Variable speed wind turbine generator

Allowing the WT to operate at the optimum tip-speed ratio and hence at the optimum power coefficient for a wide wind speed range. The two most variable speed wind generators widely used are the Doubly Fed Induction Generator (DFIG) and the converter driven synchronous generator [27]:

- DFIG is basically a standard, wound rotor induction generator with a voltage source converter connected to the slip-rings of the rotor. The stator winding is coupled directly to the grid and the rotor winding is connected to the power converter as shown in Figure 2.4. The converter system enables two transfer ways of power. The grid side converter (converter 2 in Figure 2.4) provides a DC supply to the rotor side converter (converter 1 in Figure 2.4)that produces a three phases variable frequency supply to the generator rotor via slip rings. The variable voltage into the rotor at slip frequency enables variable speed operation. Manipulation of the rotor voltage permits the control of the generator operating conditions. In case of low wind speeds, the drop in rotor speed may lead the generator into a sub synchronous operating mode. During this mode, DFIG rotor absorbs power from the grid .

- Converter driven synchronous generator uses a synchronous generator that can either be an electrically excited synchronous generator (EESG) or a permanent magnet synchronous generator (PMSG). To enable variable-speed operation, the synchronous generator is connected to the network through a variable frequency converter, which completely decouples the generator from

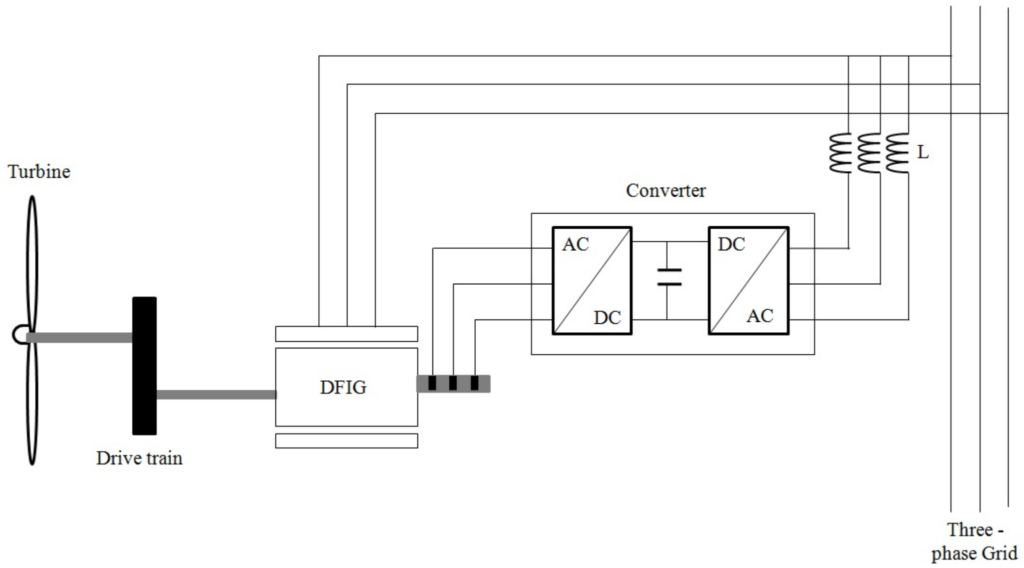

Figure 2.4: Scheme of the the Doubly Fed Induction Generator (DFIG).

the network. The electrical frequency of the generator may vary as the wind speed changes; while the network frequency remains unchanged. The rating of the power converter in this WT corresponds to the rated power of the generator plus losses. The schematic diagram of the converter driven synchronous generator is shown in Figure 2.5.

The comparison between the fixed speed and variable speed WTs shows that variable speed operation of WTs presents certain advantages over constant speed operation. Variable speed WTs allow obtaining higher energy yields and lower power fluctuations than fixed speed WTs. Moreover, variable speed WTs produce more reduced loads in the mechanical parts than fixed speed WTs. When comparing torque mode control and speed mode control strategies, literature review shows that speed mode control strategy follows wind speed, in order to achieve maximum power coefficient, more accurately, and the higher the speed control loop bandwidth is, the better the tracking is. Nevertheless, as a consequence, it produces more power fluctuations, since speed is rigidly imposed to the turbine. So, from power quality point of view, torque mode control strategy presents better behavior because speed is not directly imposed to the turbine and this control strategy lets the WT to freely change rotational speed during the transient.Therefore,in this dissertation, variable speed WTs based on doubly fed induction generator (DFIG) will be considered.

2.1.2 Wind turbine as a hybrid dynamic system

Many physical systems are Hybrid Dynamic Systems (HDS)[23]. Generally speaking, HDS are mixture of continuous dynamics and discrete events. These continuous

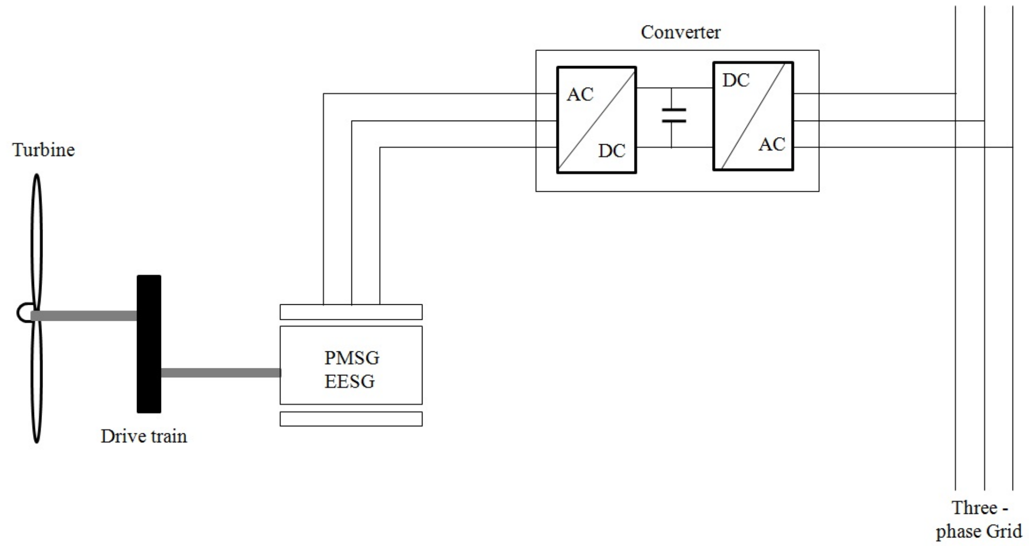

Figure 2.5: Scheme of the converter driven synchronous generator.

and discrete dynamics not only coexist, but interact and changes occur both in response to discrete instantaneous events and to the continuous dynamics described by differential or difference equations. Several WT components in WT, as pitch system and the converter, can be described as HDS.

The one tank water level control system represents a simple example of the HDS (see Figure 2.6). This example exhibits the continuous dynamics represented by the level of the tank and the discrete dynamics represented by the discrete modes of the pump (pump on (P_{on}), pump off (P_{off})) and the valve (valve opened (VO), valve closed (VC)). The discrete mode of the pump or the valve is changed in response to a discrete control command event sent by the discrete controller. As an example, if the initial discrete mode of the pump, respectively the valve, is 'pump off', respectively 'valve closed', then the control command event 'start pump', respectively 'Open valve', will change the pump discrete mode to 'pump on', respectively 'valve open'. The continuous dynamic evolution of the tank level x depends on the discrete modes of the pump and the valve. The tank filling is assured by flow rate O_p when the pump is on. The tank emptying is assured by flow rate O_V when the valve is opened. Therefore, the one tank water level control system is a HDS.

The hybrid dynamics of the one tank system example are modeled by a hybrid automaton. For the simplicity, we consider that the pump is always powered on. In this case, this hybrid automaton is depicted in Figure 2.7 and is defined by the tuple :

$$G = (Q, \Sigma, flux, Init, \delta) \tag{2.1}$$

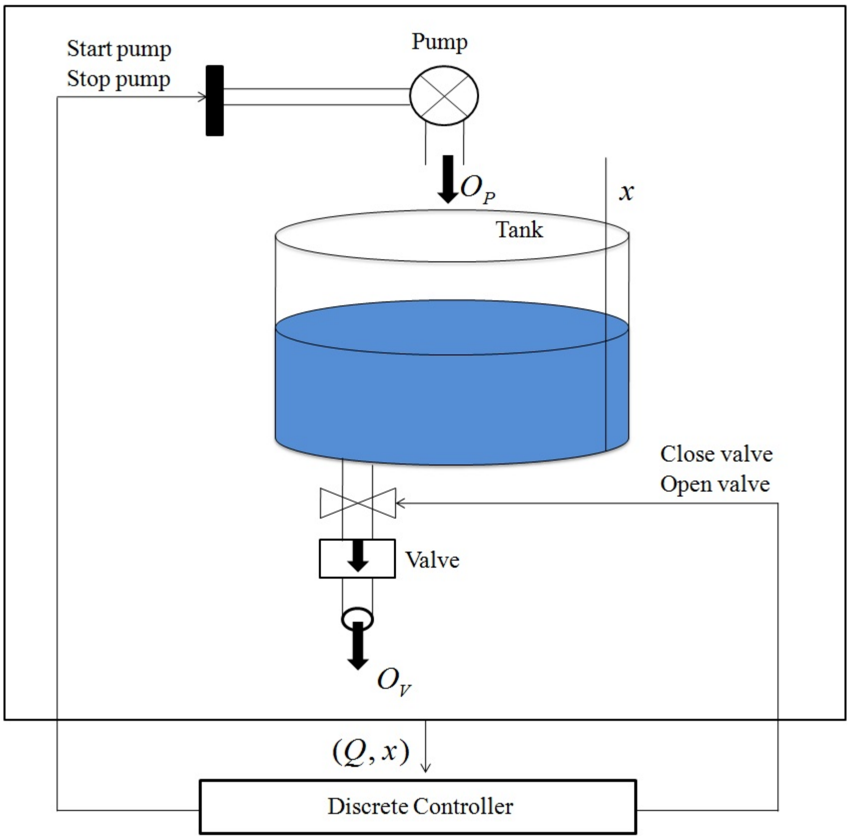

Figure 2.6: One tank water level control system.

where:

- $Q = \{(q_1 = P_{on} - VC), (q_2 = P_{on} - VO)\}$ is the set of discrete states including, pump on and valve closed (q_1) and pump on and valve opened (q_2);

- $\Sigma = \{\text{Open valve}, \text{Close valve}\}$

- $flux$: is the dynamic evolution $\dot{x}$ of the tank level x in each discrete state $q \in Q$;

- δ is the state transition function. As an example $\delta(P_{on} - VC, \text{Open valve}) = P_{on} - VO$;

- $Init : (P_{on} - VC, \dot{x} = O_p)$ is the initial conditions of the HDS (tank example)

Four particular classes of HDS can be distinguished according to the influence of the continuous dynamics on the evolution of the discrete events and conversely [10] [23]:

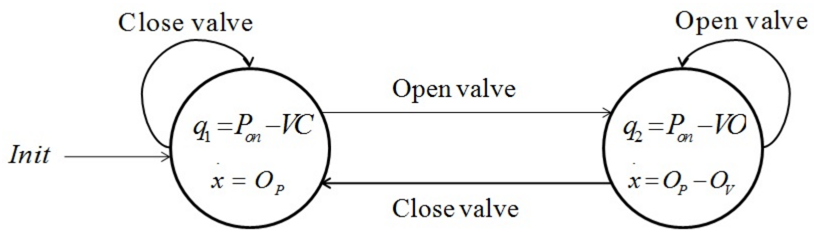

Figure 2.7: Hybrid automaton modeling the hybrid dynamics of the one tank system example of Figure 2.6.

2.1.2.1 Autonomous Switching Systems (ASS)

In this class of HDS, the continuous dynamics ($\dot{X}$) change when the continuous state (X) reaches some areas in the continuous state space. These systems are inherently hybrid, including discrete and continuous elements. One example is an electric circuit constituted by continuous elements (resistance and inductance) and the discrete elements (a switch and a diode) [68].

2.1.2.2 Discretely Controlled Switching Systems (DCSS)

In this class of hybrid dynamic systems, the continuous dynamics ($\dot{X}$) change instantly in response to a control signal(external input). Continuous systems supervised by a discrete controller are an example of discretely controlled switching based systems, Example of this class of HDS is one tank water level control system [63] (see Figure 2.6 and Figure 2.7).

2.1.2.3 Autonomous Jumping Systems (AJS)

In this class, the continuous state variables (X) change discontinuously when they reach a certain region in the continuous space states. Example of this class of HDS the ball bouncing from a massive wall [23].

2.1.2.4 Discretely Controlled Jumping Systems (DCJS)

In these systems, the continuous state variables (X) change discontinuously under the influence of an external action (e.g., a command) as the case for electromagnetic systems with pulse inputs [10].

2.1.3 Faults in wind turbine

Like every other complex system, WTs are prone to faults that can affect their performance and increase the production and exploitation costs. The faults are abnormalities that affect one or more properties of the system, which can lead to a failure or to a breakdown (shut down) of the system. They can occur in different parts or components of the WT. The objective of diagnosis is to detect the occurrence of a fault and to establish which possible faults or combinations of faults match the observed system behavior. In the literature, faults are classified according to their location, their time evolution or their nature.

2.1.3.1 Classification of faults according to their location,

As shown in Figure 2.8, faults may manifest in different parts of the system, namely, the actuators, the system, the sensors and the controller.

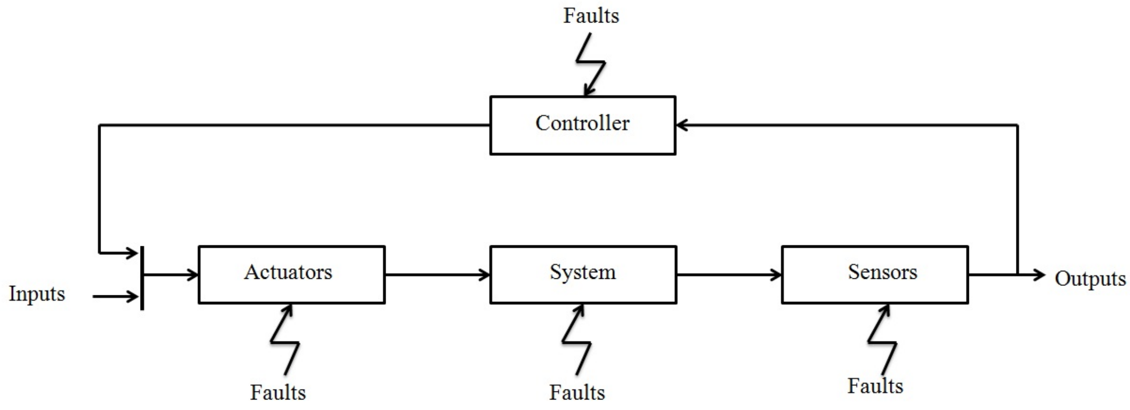

Figure 2.8: Location of faults in WT.

- **Actuator faults** [9]: They act at the operational part of the WT and deteriorate the signal input of the system. They result in total or partial failure of an actuator acting on the system. An example of a total failure of one actuator is an actuator which remains 'stuck' at a position resulting in the inability to control the system through the actuator. Partial failure actuators are actuators reacting similarly to the rated speed but only partly, that is with some degradation in their action on the system. In WT system, several actuator faults are possible to appear; these faults are either electrical, mechanical, hydraulic or pneumatic. The actuator fault in WT can occur in the pitch system , in generator or in converter. The occurrence of these faults will change the system performance like offset or change the dynamics of the actuator.

- **Sensor faults** [13]: A partial failure sensor produces a signal with varying degrees of consistency with the true value of the variable to be measured. This can result in a reduction of the displayed value relative to the true value, or the presence of a skew or increased noise preventing proper reading. A total sensor failure produces a value that is not related to the measured variable. In WT system, a number of possible sensor faults may occur. These faults are either electrical or mechanical faults in the position sensors, and can result in either a fixed value or a gain factor on the measurements.

- **System faults** [13]: These are faults resulting in breakage or deterioration of a system component reducing its capacity to perform a task. For instance a WT system the system fault may occur in the drive train where the friction changes with time. The occurrence of this fault changes the system parameters.

- **Controller faults** [63]: They impact the controller outputs. Indeed in this case, the controller does not respond properly to its inputs sensor reading. Controller faults are very dangerous in the case of WTs because they impact directly the WT safety and energy production.

2.1.3.2 Classification of faults according to their force of occurrence and time evolution

The operating conditions of WTs or one of its components change from normal to faulty either abruptly or gradually. According to the force of occurrence of the faults and to their time evolution, faults can be abrupt, intermittent or gradual.

- **Abrupt faults** [63],[92]: Manifest at full magnitude immediately and they are defined as a malfunction of a component that must be replaced or repaired. This type of faults is characterized by an abrupt evolution of the variables value of the corresponding element (see Figure 2.9). Several abrupt failures may occur in WTs, it may be a sensor, actuator or system fault. WTs are prone to either sensor, actuator or system abrupt faults. In this dissertation abrupt faults are not considered.

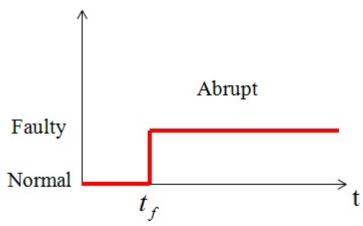

Figure 2.9: Abrupt faults in WTs.

- **Intermittent faults** [111]: These are a special case of abrupt faults with the property that the signal returns randomly to its normal value (see Figure 2.10). In this dissertation intermittent faults are not considered.

- **Gradual faults (Drift-like faults)** [14]: They entail a progressive evolution (degradation) of the operating conditions of the system from normal to a failure (see Figure 2.11). Consequently, the system begins to malfunction (degraded behavior) until the failure takes over completely. The diagnosis of

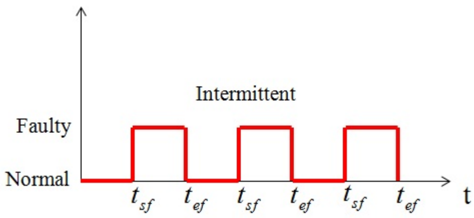

Figure 2.10: Intermittent faults evolution.

gradual faults is a challenging task due to the difficulty to distinguish between normal fluctuations of the system and abnormal drift in its operating conditions. In this dissertation only gradual faults (drift-like faults) are considered. The goal is to detect a drift from normal to faulty operating conditions in its early stage in order to provide enough time to human operators to achieve appropriate corrective actions to decrease the maintenance costs and to increase the availability of WTs.

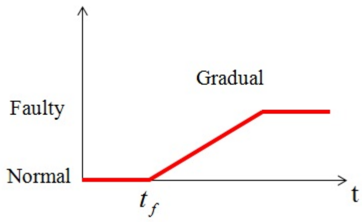

Figure 2.11: Gradual faults evolution.

2.1.3.3 Classification of faults according to their nature

In HDS, there are two types of faults which can adversely impact their continuous and discrete behaviors:

- **Parametric faults** [44] are associated with changes in parameter values, and are useful for modeling degradation in the system's components. Parametric faults are considered to be abnormal deviations of parameter values in continuous modes of operation. This dissertation focuses on the diagnosis of this type of faults since they entail a drift or degradation in the WT performance.

- **Discrete faults** [20] affect the system discrete dynamics and are considered either as the occurrence of unobservable events and/or reaching discrete fault modes. In this dissertation discrete faults are not considered.

2.2 Fault diagnosis in wind turbines

The complicated design of WTs makes very difficult and even dangerous to access the turbines. Thus, it is crucial to design an automated diagnostics system in order to achieve the fault detection and isolation. In order to evaluate the performance of the designed diagnosis method and to compare its performance to other methods of the literature, a benchmark representing the different components of a variable speed WT is used. This benchmark allows to generate several scenarios of normal and failure operating conditions for different WT components. This benchmark was proposed by the KK-electronic [73] to international competition to find the best diagnosis approach according to predefined evaluation criteria. The WT modeled by this benchmark is a three blade horizontal axis variable speed turbine with a full converter. The conversion from wind energy to mechanical energy in terms of a rotating shaft can be controlled by changing the aerodynamics of the turbine by pitching the blades or by controlling the rotational speed of the turbine relative to the wind speed. The mechanical energy is converted to electrical energy by a generator fully coupled to a converter. Between the rotor and the generator, a drive train is used to increase the rotational speed from the rotor to the generator. The converter can be used to set the generator torque, which consequently can be used to control the rotational speed of the generator as well as the rotor.

A system overview can be seen in Figure 2.12. This figure shows the relationships between: Blade & Pitch System, Drive Train, Generator & Converter, and Controller. Since it is a three blade turbine, each blade pitch angle is measured by two redundant sensors for each blade in order to ensure physical redundancy. The reference pitch angle provided by the controller in response to the current wind speed is applied to each blade based on an independent pitch actuator. The generator and rotor speeds are also measured by two duplicated sensors. The instrumentation of the WT benchmark model is resumed in Table 2.1.

Variable	Number of sensors	Notation
Generator speed	2	$\omega_{g,m_1}, \omega_{g,m_2}$
Rotor speed	2	$\omega_{r,m_1}, \omega_{r,m_2}$
Pitch position measurements	2 sensors/blade	$\beta_{1,m_1}, \beta_{1,m_2}, \beta_{2,m_1}, \beta_{2,m_2}, \beta_{3,m_1}, \beta_{3,m_2}$
The electrical power generated by Generator	1	$P_{g,m}$
Generator torque	1	$\tau_{g,m}$
Wind speed	1	v_w

Table 2.1: Variables and their corresponding sensors for the WT benchmark.

Figure 2.12 shows the overall WT model structure where v_w denotes the wind speed, $\tau_r$the rotor torque, ω_r the rotor speed, τ_g the generator torque, the converter torque, ω_r the generator speed, β_r the pitch angle control reference, β_m the measured pitch angles, $\omega_{r,m}$ the measured rotor speed, $\tau_{g,m}$ the measured generator torque, $\omega_{g,m}$ the measured generator speed, P_g the measured generated electrical power, $\tau_{g,r}$ the generator torque reference, and P_r the power reference.

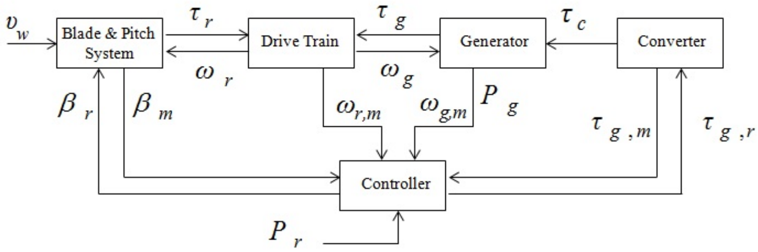

Figure 2.12: Overview of the WT system, source of the benchmark data.

In this benchmark model, a number of faults are considered. They are covering different kinds of possible faults in the WT. In Table 2.2, these different kinds of faults are listed. These faults have different degrees of severity and drift speeds (fast, medium, slow) as we can see in Table 2.3. Some are very serious and should result in a fast safe shut down of the WT and others are less severe in the way that the controller can be designed in order to accommodate these faults.

N.	Fault	Role	Notation	Type
1	Pitch angle sensor faults	Pitch position measurements	$\Delta\beta_{k,m_i}$ $k=1,2,3, i=1,2$	Fixed Value
2	Pitch angle sensor faults	Pitch position measurements	$\Delta\beta_{k,m_i}$, $k=1,2,3, i=1,2$	Gain Factor Value
3	Pitch angle sensor faults	Pitch position measurements	$\Delta\beta_{k,m_i}$, $k=1,2,3, i=1,2$	Fixed Value
4	Rotor speed Sensor faults	Rotor speed measurements	$\Delta\omega_{r,m_i}, i=1,2$	Gain Factor
5	Sensor faults	Rotor speed measurements	$\Delta\omega_{r,m_i}, i=1,2$	Fixed Value
6	Sensor faults	Generator speed measurements	$\Delta\omega_{g,m_i}, i=1,2$	Gain Factor
7	Actuator faults	Converter torque measurements	$\Delta\tau_g$	Offset
8	Pitch actuator faults	Changing the pitch blade position	$\Delta\beta_k, k=1,2,3$	Changed Dynamics
9	Pitch actuator faults	Changing the pitch blade position	$\Delta\beta_k, k=1,2,3$	Changed Dynamics
10	Drive train fault	Changing the speed ratio between turbine and generator	$\Delta\omega_r$,$\Delta\omega_g$	Changed Dynamics

Table 2.2: Faults considered in the WT benchmark model.

In this thesis, the sensors and actuator faults are considered and more specifically the drift-like faults that can affect the normal operating conditions of the pitch

N.	Consequence	Severity	Drift speed
1	False measurement,reconfigure system	Low	Medium
2	False measurement,reconfigure system	Low	Medium
3	False measurement,reconfigure system	Low	Medium
4	False measurement,reconfigure system	Low	Medium
5	False measurement,reconfigure system	Low	Medium
6	False measurement,reconfigure system	Low	Medium
7	Slow torque control,indicates serious problems	High	Fast
8	Leakage, slow control	High	Medium
9	Air in oil, slow control	Medium	Slow
10	Increased level of drive train vibrations	Medium	Very slow

Table 2.3: Severity and time of development of the considered faults.

system and the converter in the WT. The drift-like faults manifest as gradual change in WT operating conditions. Wind turbine begins to malfunction until the failure takes over completely. The detection of this drift from normal to faulty operating conditions in its early stage can help, as mentioned before, provide time to take appropriate corrective actions in order to decrease the maintenance costs and to increase the availability and the production.

2.3 Review of wind turbine fault diagnosis methods

There are several methods in the literature that are used to perform fault diagnosis in WTs. Based on the difference between the desired or expected behavior of the model and the observed behavior provided by sensors. Methods diagnosis generally can be devided into two categories: general purpose methods and component based methods [79]. In the case of general purpose methods, the model represents the specific behavior of the WT based on its parameters, as wind speed, generated electrical power, air temperature etc. Thresholds are used to define as alarm levels that indicate significant changes in the turbine behavior. Exceeding these alarm levels, due to the occurrence of a fault, leads to a drop in performance or to completely shut down the turbine and to wait for a remote restart or repair. Therefore, trend analysis of some representative signals using signal processing and data mining techniques can help to detect the fault occurrence in early stages. These methods have the advantage to be cost-effective since no need for a prior knowledge about the relationships between WT components. However, they do not provide a specific or precise diagnosis about the faulty components.

Component-based methods [30] are used to detect faults of one specific component of WT. The failure of this component, e.g., gearbox, blade pitch system, is normally critical according to its maintenance costs or/and its frequency occurrence. These methods provide reliable and precise diagnosis. However, a depth analysis is required to determine the highly sensitive parameters and features to normal and faulty behaviors of the monitored component [79] [5]. This dissertation focuses on components based approaches.

The fault diagnosis methods in WTs can also be divided into internal and external methods (see Figure 2.13). These methods are presented in the following subsections.

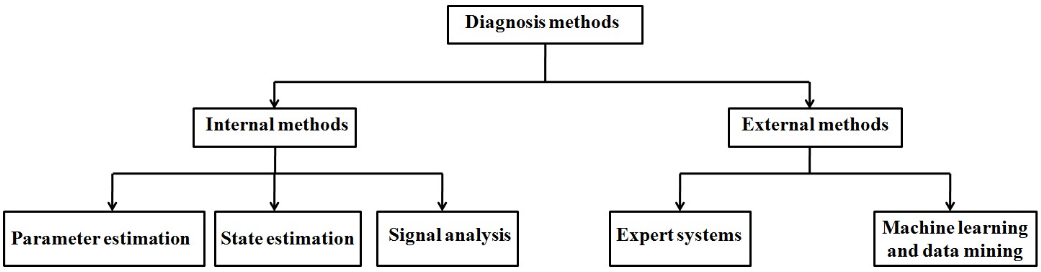

Figure 2.13: Fault diagnosis methods

2.3.1 Internal methods

The internal methods [15] [72] [91] use a mathematical or/and structural model to represent the relationships between measurable variables by exploiting the physical knowledge or/and experimental data about the system dynamics. These variables represent the internal parts of the WT. The response of the mathematical model is compared to the observed values of variables in order to generate indicators used as a basis for the fault diagnosis. Generally, the model is used to estimate the system state, its output or its parameters. The difference between the system and the model responses is monitored. Then, the trend analysis of this difference can be used to detect changing characteristics of the system resulting from a fault occurrence. The internal methods used to achieve the fault diagnosis of WTs are divided into three main categories:

1. Parameter estimation based approaches,
2. Observer based approaches,
3. Signal or feature based approaches.

2.3.1.1 Parameter estimation based approaches

Parameter estimation based methods rely on the fact that faults in systems are often reflected by variation of physical parameters such as, mass, damping, stiffness, etc. Faults can therefore be diagnosed by directly estimating the relevant parameters. If the estimated parameter value deviates from the nominal parameter value, then a fault has occurred [37],(see Figure 2.14).

For better understand of the principle and the application of parameter estimation based approaches in WTs, we take the example of the pitch system. In the benchmark model, a hydraulic pitch system is used. The state representation of the nominal pitch system dynamics is defined as follows [74]:

$$\begin{aligned} \dot{x_p} &= A_p x_p + B_p\left(\beta_r + \beta_f\right) \\ y_p &= C_p x_p \\ A_p &= \begin{bmatrix} 0 & 1 \\ -\omega_n^2 & -2\zeta\omega_n \end{bmatrix} \\ B_p &= \begin{bmatrix} 0 \\ \omega_n^2 \end{bmatrix} \\ C_p &= \begin{bmatrix} 0 & 1 \end{bmatrix} \end{aligned} \tag{2.2}$$

The state vector $x_p = \begin{bmatrix} \dot{\beta_i} & \beta_i \end{bmatrix}^T$ is composed of pitch angular speed $\dot{\beta_i}$, and angular position β_i for each blade $i : (i = 1, 2, 3)$. y_p is the measured pitch position, β_r is the pitch angle position reference provided by the controller, and β_f is the feedback pitch angle . ω_n, ζ are the parameters of the pitch system where ω_n represent the natural frequencies and ζ is the damping ratio.

A general procedure for fault diagnosis using parameter estimation consists of the following 5 steps [43],[32]:

1. Determination of the relationships between the model parameter vector θ and the physical parameter vector ζ:

$$\theta = f\left(\zeta\right) \tag{2.3}$$

2. Estimation of the model parameter vector θ using the inputs and outputs of the system, resulting in the estimate $\overset{\wedge}{\theta}$.

3. Construction of the physical parameter vector from the estimated parameter vector $\overset{\wedge}{\theta}$.

$$\overset{\wedge}{\zeta} = f^{-1}\left(\overset{\wedge}{\theta}\right) \tag{2.4}$$

   and computation of the deviation with respect to the nominal value, i.e. $\Delta\zeta = \zeta - \overset{\wedge}{\zeta}$. The deviation $\Delta\zeta$ takes the role of the residual.

4. Faults can be diagnosed by using $\Delta\zeta$ and the known relations between the faults and the parameters.

Figure 2.14, Equation 2.3 and Equation 2.4, show how the pitch system can be diagnosed by parameter estimation approaches. An estimator will be run in parallel to pitch system to estimate the internal parameter ζ of the pitch system . Then the estimated parameter ζ_e will be compared with its corresponding nominal value

ζ. The difference between the real ζ parameter value and estimated ζ_e parameter value, can be used as information about the status (normal/faulty) of the pitch system operating conditions.

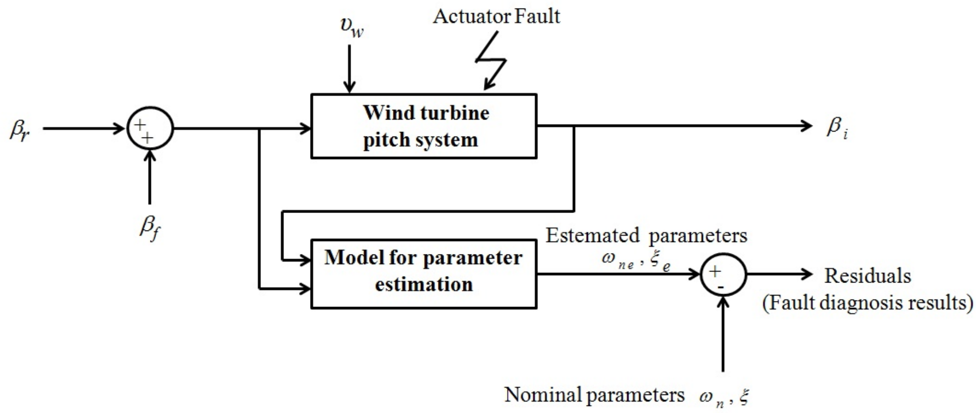

Figure 2.14: Parameter estimation approach.

There are several approaches based on the parameters estimation, that were developed for fault diagnosis of WTs. Among these approaches we can cite [91]. In this work, the authors proposed a procedure for the fault detection and isolation of a WTs using fuzzy models identified from uncertain input-output measurements. The considered faults in this work are defined in Table 2.2. In the same optical, in [57] a Kalman-like observer is used to estimate the parameters of the pitch actuator to detect and isolate the faults that may affect it.

The major advantages of parameter estimation based approaches are:

- Determination of the size and time-variant behavior of a fault,
- This approach is very interesting in the case of fault tolerant control,
- Do not need an additional hardware components to be implemented,

On another hand the major drawbacks of these approaches are:

- Require perfect physical knowledge about the system dynamics which is hard to obtain for complex systems,
- Powerless tool in handling non-linear dynamic systems,
- Conditions of estimation remain very restrictive and return to the physical parameters of the system is not always possible.
- Do not scale well to large scale systems with huge number of discrete modes.

2.3.1.2 State estimation based approaches

The main idea behind observer-based methods is to estimate the outputs of the system from the measurements or subsets of the measurements through use of observers. Subsequently, the estimation residual, can be computed as the difference between the estimated output and the measured output [37],(see Figure 2.15). This residual can be used for the purpose of fault diagnosis. In the nominal case the model used by the observer and the real system should correspond well, which would lead to a zero residual. In case of a fault, the residual would be nonzero. To see how such a residual is constructed, consider the state-space of pitch system.

$$\begin{aligned} \dot{x_p} &= A_p x_p + B_p (\beta_r + \beta_f) \\ y_p &= C_p x_p \end{aligned} \tag{2.5}$$

The state of the pitch system can be estimated with an observer as follows :

$$\begin{aligned} \dot{\hat{x}}_p &= A_p \hat{x}_p + B_p (\beta_r + \beta_f) + L \left(y_p - \hat{y}_p \right) \\ \hat{y}_p &= C_p \hat{x}_p \end{aligned} \tag{2.6}$$

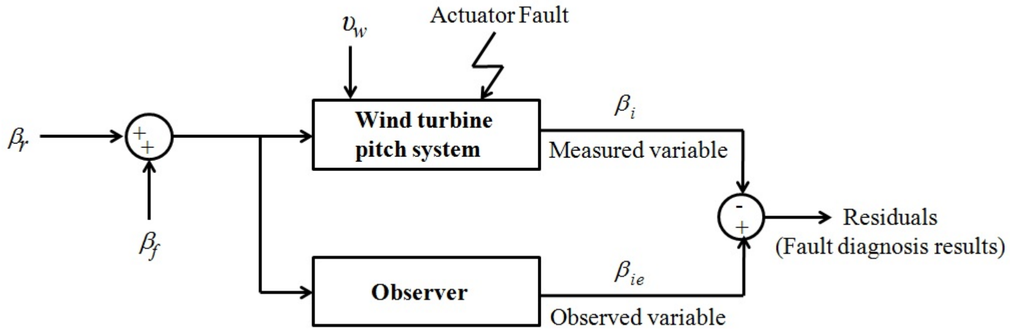

Figure 2.15: Observer-based approach.

where $\hat{x}_p$ and $\hat{y}_p$ denote the estimates of x_p and y_p, respectively. The matrix L is the observer gain, which determines the behavior of the observer. The residual that is of interest for fault diagnosis is $y_p - \hat{y}_p$. Using a single observer is not sufficient for fault isolation. For this purpose, several observer schemes can be used [32]. The Unknown input observers are useful when the wind speed is an important input of the model. Example of these approaches applied to the diagnosis in the WT we can cite [112],[15]. In both work, observer-based fault detection and isolation schemes are proposed for the WT faults defined in Table 2.2. In the same intention in [72], an unknown input observer is used for sensor faults detection in a WT defined in Table 2.2.

The major advantages of state estimation based approaches are:

- In many cases the measurements necessary to control the process are sufficient for the fault diagnosis algorithm so that no additional sensors have to be installed,
- Only the information about normal operating conditions are taken into account,
- Do not need an additional hardware components to be implemented.

On another hand the major drawbacks of this approach are:

- A need of an accurate analytical model of the real system,
- Powerless tool in handling of non-linear dynamic systems,
- Adaptability is not large because any change to the system necessarily requires the modification of the model or its structure,
- Several conditions must be verified before being applied,
- Do not scale well to large scale systems with huge number of discrete modes.

2.3.1.3 Signal analysis based approaches

Signal analysis approaches are based on time and frequency domains analysis without any explicit mathematical model. Only knowledge about suitable fault features is required. Fault features can be derived from raw signals (vibration, acoustic emission, electrical signatures) in order to evaluate the system operating state (see Figure 2.16). We take the same example of pitch system to explain the application of signal based approaches for fault diagnosis of WT. The measured output of the pitch system (the measured pitch position) will be processed, and frequency transformation is applied on this signal to extract informative features; these features will be compared with some suitable predefined thresholds. The latter define the normal (acceptable) variation of these features. When, the value of one or more of these features is greater than the corresponding threshold, a fault is detected. The figure 2.16 shows the different steps to diagnose the faults in the pitch system based on the signal analysis.

A Review of recent advanced approaches using this category of techniques applied to the fault diagnosis in WTs was given in [64]. Example of these approaches applied on the WT we can cite the work in [102]. In this paper, a continuous wavelet transform-based approach is applied to enhance the damage-detection capability of WT blades. The authors in [31] use SCADA and CMS signals for failure detection and diagnosis of the WT gearbox. More details about these techniques are given in [47].

The major advantages of signal analysis or feature based approaches are:

- Easier to be implemented if a sophisticated data acquisition systems and sensors exists,

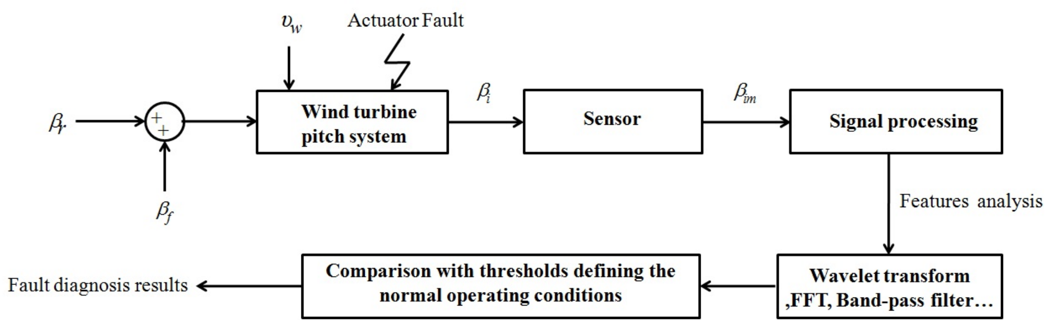

Figure 2.16: Scheme of signal analysis or feature based approach.

- Useful for analyzing signals that show oscillations with long periods (electric current, pressure, temperature ...)

On another hand the major drawbacks of these approaches are:

- Sensitivity to measurement noise when those which coincide with the frequency area of interest,
- Measurement signals are non-stationary and even more complex in a WT,
- A need of a sampling frequency for the reconstruction of the signal while minimizing frequency losses.

The application of internal approaches for fault diagnosis of WTs is a difficult task due to the WT complexity and to the strong non-stationary character of its environment.

2.3.2 External methods

The external methods consider the system as a black box, in other words, they do not need any mathematical model to describe the system dynamical behaviors. They use exclusively a set of measurements or/and heuristic knowledge about system dynamics to build a mapping from the measurement space into a decision space. They include expert systems machine learning and data mining techniques.

2.3.2.1 Expert systems

Traditional expert systems for fault diagnosis are rule-based systems [84] ,[108],[87], in which the heuristic knowledge of experts is captured in the form of empirical associations which relate symptoms to the faults that produce them. In a rule-based expert system, much of the knowledge is represented as conditional sentences (IF-THEN rule-based) relating symptoms with one another to a certain failure. In a real WT gearbox fault case, rules can be defined as [15]:
Rule

IF
The cooler oil temperature is NORMAL **AND**
Gearbox main bearing temperature is HIGH
THEN
Failure in the gearbox main bearing is CERTAIN

NORMAL and HIGH can be defined using fuzzy logic.

The major advantages of expert systems are:

- It is very efficiency for systems that are difficult to model,
- Very simple to be implemented,
- A high level results very easy to be understood by human operators,
- Short processing time thank for rules which describe directly the human reasoning.

On another hand the major drawbacks of these approaches are:

- The knowledge acquisition is hard to obtain,
- It is very domain dependent, lack of generality for other applications,
- Rules are not adapted for system evolutions,
- The integration or/and excluding of new rule may entail a serious problem of consistency in the inference engine,
- Very maladapted for the explication of a result in the case of failures propagation.

The most widely used scheme for alarm analysis, especially in the process control industry, is based on fault trees [117]. Fault trees provide a graphical representation of cause-effect relationships of faults in a system. Starting from a goal violation, or a system failure event that is indicated by an alarm condition, a fault tree is built by reasoning backwards from the system failure to basic or primal failures that represent the root cause of the failure. The primary drawbacks of this approach are: (i) fault trees require a great deal of effort in their construction and (ii) they are difficult for handling feedback systems. Example of these methods applied to fault diagnosis of WTs, we can cite the methods proposed on [117] and [36]. In [117], an expert system based on fault tree analysis was developed in order to make timely and accurate diagnosis for gearbox. The authors in [36] presented a robust, accurate expert system for the classification and detection of WT pitch faults.

2.3.2.2 Machine learning and data mining based approaches

When a process is too complex or poorly known to be monitored through analytical models and if signal analysis techniques do not allow an unambiguous diagnosis, machine learning (ML) and data mining (DM) approaches can be used. ML&DM approaches serve to learn the complex model exclusively from available historical data [103]. The model about the system behavior is built by learning from data in order to link the input or observation space to the output or decision space. These approaches are used when the knowledge about the system behavior is incomplete, and thus insufficient to construct an accurate model. ML&DM approaches consist of the following steps (see Figure 2.17) [79]: data preparation, data preprocessing and labeling, data analysis, model learning and model validation. Figure 2.17 shows the general scheme used by machine learning approaches to achieve fault diagnosis of pitch system. The measured outputs of the pitch system undergo different processing steps. Features will be extracted from measured variables in order to take decision about operating conditions of the pitch system.

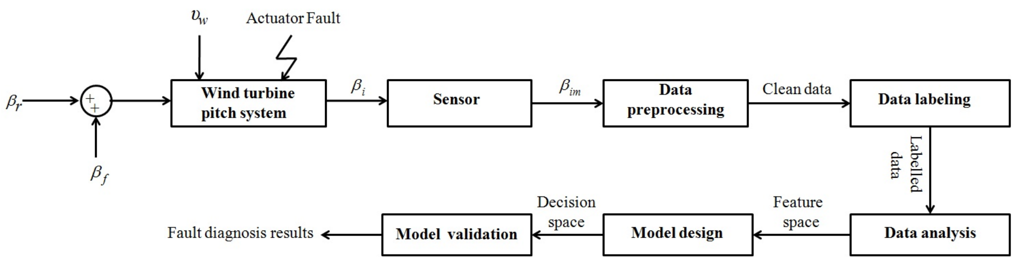

Figure 2.17: General scheme used by ML&DM techniques for fault diagnosis.

In general, there is no perfect mathematical solution for engineering problems. For this reason, machine learning techniques can provide a way of overcoming this issue [80]. Scientists in different fields attempt to employ historical data to develop algorithms that can learn the behavior of systems [106][94]. ML techniques provide the ability to learn without being explicitly programmed for systems [109]. This technique develops algorithms that are able to find different patterns in data and adjust program actions according to the training dataset.

The major advantages of ML DM approaches are:

- Ability to learn without to a priori physical knowledge of the system,
- Very simple to be implemented,
- Powerful tool of handling non-linear and multi variable problems,
- Ability to learn on-line.

On another hand their major drawbacks are:

- Require a priori knowledge (data) about all faulty behaviors,
- It is not able to determine the size of a fault,
- There is no general rule for choosing feature space.

There are several ML&DM methods used to achieve the fault diagnosis of WTs, as support vector machines (SVM) [71], [55], Neural Networks (NN) [88], [81], Auto-adaptive Dynamical Clustering (AUDyC) [60], Self-Feature Organization Map (SOFM) [50], K Nearest Neighbors (KNN) [97], Genetic Algorithm (GA) [104]. They can be classified into different categories of algorithms as follow:

- **Supervised learning methods,** that requires the training data to be fully labelled. For example in fault diagnosis application each data instance is assigned with either a normal or abnormal class. Any unseen data is compared against the trained model to determine to which class it belongs. Examples of supervised learning algorithms are Nave Bayes (NB) [8], KNN [97], and SVM [71] [55]. To explain how this category of approaches can be applied to make the fault diagnosis of WT, we take the example of KNN approach and we apply it to achieve the fault diagnosis of pitch system actuator. The features sensitive to the pitch actuator normal and failure operation conditions are extracted in order to define the feature space. In the latter, two classes, represented as restricted areas in the feature space, are defined: the pitch actuator normal and failure classes. Then, a classifier based on the use of KNN is used to assign a new pattern, representing the pitch actuator current operating conditions, to the pitch actuator normal or failure classes.

- **Unsupervised learning methods,** that build classifers using patterns without any class label. The unsupervised learning algorithm itself needs to determine what those classes are and how to separate them. The most well-known unsupervised learning algorithms are k-Means Clustering [55], Fuzzy c-means (FCM) [66], SOFM [50] and AUDyC [60]. To explain how this category of approaches can be applied to make the fault diagnosis of WT, we take the example of K-Means Clustering approach and how it can be applied to the fault diagnosis of pitch actuator system. The features sensitive to the normal and pitch actuator failure operation conditions are extracted in order to define the feature space. K-Means Clustering approaches will be applied to detect the number of classes or clusters in data base. The obtained clusters will be validated using some meaningful criteria [7]. Then an expert will indicate which class or cluster is represents the normal or failure operation conditions.

2.3.3 Comparison and discussion

Based on this study, major advantages and drawbacks of each category of WTs fault diagnosis approaches are listed here after:

- The internal methods, in particular parameter estimation based ones, have the advantage of identifying the abnormal physical parameters rather than faulty signal signatures that are more dependent on the load condition [65].The major advantages of these methods are the ability to 1) detect both the abrupt and progressive failures via trend analysis and 2) give a precise decision or isolation of a failure. However,they require a sufficiently accurate a priori knowledge to construct a mathematical or analytic model for the monitored system. This is hard to achieve in the case of complex non-linear systems as WTs. Signal analysis based approaches are easier to implement if a sophisticated data acquisition systems and sensors exists. However, successful implementation of such approaches is dependent on the construction of suitable fault-related features and reliable thresholds since subjective and unproven ones may result in wrong alerts [113]. Moreover, they suffer from 1) the necessity to depth information about system behavior and failures which is hard to obtain for complex and strong non-stationary systems as WTs and 2) the sensitivity of the fault detection to model design errors and measurements noises.

- The external methods consider the system as a black box, in other words, they do not need any mathematical model to describe the system dynamical behaviors. They use exclusively a set of measurements or/and heuristic knowledge about system dynamics to build a mapping from the measurement space into a decision space. They include expert systems and data mining techniques. These methods are suitable for systems that are difficult to model. They are simple to implement and require short processing time. However, since the obtained models are not transparent, the obtained results are hard to be interpreted and demonstrated. Machine learning and data mining methods achieve multi-dimensional analysis based on the combination of several sensors that monitor the same component. Moreover, their performance is highly dependent on the selection of training data set which must represent all operating modes (normal and failure) for the WT.

As a synthesis of this state of the art, some criteria are proposed to compare these two categories of diagnosis methods (see Table 2.4). Such comparison could support the choice of the suitable fault diagnosis approach with respect to the initial needs. Chosen criteria for this comparison are the following:

1. Systems non-stationary nature: ability to separate the actual degradation and environmental or load effects.
2. Needed knowledge: ability to construct model without need to a priori knowledge.
3. System complexity: ability to deal with system hierarchical levels (local component or global system point of view).
4. Adaptability: Ability to handle the system evolutions.

Table 2.4 shows the rank of each category of methods regarding each of the criteria. A category is accorded the first rank (+++) when it satisfies the best the previous corresponding criteria.

Type	Methods	non-stationarity	knowledge	complexity	Adaptability
Internal methods	Parameter estimation	++	+	+	+
	State estimation	+	+	+	+
	Signal analysis	+	++	++	+
External methods	Expert systems	+	+++	++	+
	Machine learning and data mining	+++	+++	+++	+++

Table 2.4: Comparison of fault diagnosis methods.

Based on Table 2.4 machines learning and data mining methods represent the best solution to achieve the fault diagnosis of WTs. However, they suffer from several drawbacks -) they require a sufficient number of patterns according to each fault behavior in order to obtain an efficient diagnosis model, -) they are usually insensitive to the occurrence of undefined or unpredicted fault, -) since the obtained models are not transparent, the obtained results are hard to be interpreted and demonstrated and -) they are not adapted to detect drift-like faults representing the component degradation. Therefore, it is interesting to develop an advanced systematic methodology and architecture of fault diagnostics in WTs able to: 1) separate any abnormal change caused by components degradation from normal change due to environmental (e.g. weather conditions) or load (e.g. electricity network status) effects and 2) describe WT dynamical behaviors (normal/degraded/faulty) without the need to depth a priori knowledge. One solution to achieve these tasks is the use of on-line and adaptive machine learning and data mining approaches which will be detailed in the next subsections.

2.4 Review of on-line and adaptive machine learning methods

2.4.1 Concept drift definition

In general a system is subject to drift when an incipient fault causes intrinsic changes in its parameters. This results in change in the properties of the data that are generated by this system. Thus, the environment is non-stationary. In the context of early diagnosis, what is needed is an algorithm that can model data in non-stationary environments, in the aim of extracting indicators for health assessment and diagnosis.

Conventional modeling algorithms proceed in an off-line manner. A model is constructed from the historical data and then applied on-line on the incoming data. In on-line manner, this model is used to fulfill its task, which could be prediction, decision making, etc. However, these conventional methods fail when the environment generating the data is subject to change. These changing environments or so called non-stationnary environments will induce incorrect outputs from the model. The changing in the environment is known as concept drift. It refers to a slowly changing environment. For abrupt changes, the term concept shift, is used. For both, 'concept drift' and 'concept shift' adaptation of the model is required. This gave rise to evolving modeling techniques that are designed to cope with changing environments. Their aim is to continuously give an authentic representation of the environment. Thus, their structure as well as their parameters could persistently be subject to changes. Evolving models are also referred to adaptive models.

During the classification task, a learning model L attempts to predict the class label $y_i\,(i = 1, \ldots\ldots, c)$ of the incoming instance x. This prediction is based on estimating the distribution D which represents the joint probability $P\,(x, y_i)$. Hence, when referring to a particular distribution D_t at time t (i.e, a particular joint probability $P\,(x, y_i)$ at time t) we define it as concept:

$$D_t = \{P_t\,(x, y_1)\,, P_t\,(x, y_2)\,, \ldots\ldots, P_t\,(x, y_c)\} \tag{2.7}$$

Thus a concept drift occurs when there is a change in the joint probability between two time points t_0 and t_1 :

$$P_{t_0}\,(x, y_i) \neq P_{t_1}\,(x, y_i) \tag{2.8}$$

There are two essential types of concept drift:

- Real concept drift,
- Virtual concept drift.

2.4.1.1 Real Concept Drift

Refers to changes in the posterior probability $P\,(y_i\,|x\,)$ which means that the target concept of the same values of attributes changes. This kind of drift directly affects the decision boundaries, which in turn decreases the learner performance [49] [2] [45]. Generally, for handling real concept drift, many techniques rely on the prediction feedbacks or the performance indicators of the learner.

Let us take the example of Figure 2.18.a showing two classes in two dimensional feature space. The class with the label N represents the normal operation conditions of a machine while the class with the label $F1$ indicates a failure operation conditions. Let us suppose that an abrupt (shift) drift has occurred indicating the occurrence of new failure mode $F2$ (class with label $F2$ in Figure 2.18.b). The classifier will misclassify these patterns by considering them as belonging to $F1$ while they represent a new failure operation. The decision boundary of the classifier must be update in order to take into account the occurrence of this abrupt drift as it is depicted in Figure 2.19.

2.4.1.2 Virtual Concept Drift

Refers to changes in the class conditional probability $P\,(x\,|y_i\,)$ without affecting the posterior probability $P\,(y_i\,|x\,)$ in the sense that, the data distribution within

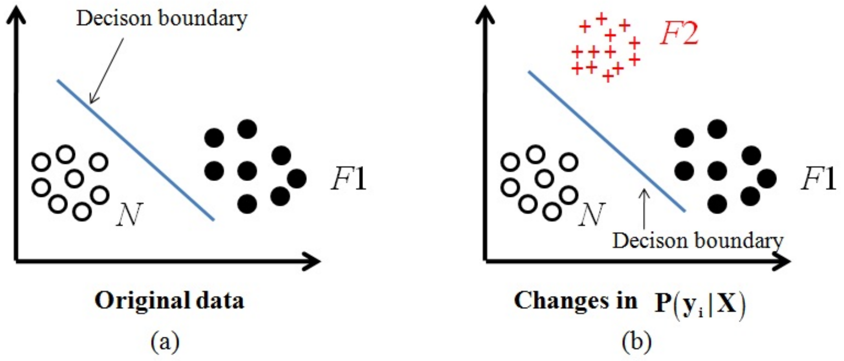

Figure 2.18: Tow classes in 2 dimensional feature space: (a) before the real drift (b) after the real drift

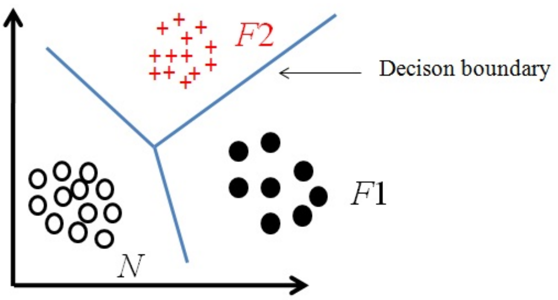

Figure 2.19: Updating the boundary decision in response to the occurrence of a real abrupt drift for the example of Figure 2.18.(a)

the same class changes without affecting the decision boundaries [49], [2], [45]. Generally, for handling the virtual concept drift, many techniques focus on the input data distribution and track changing in the class conditional.

Let us take the example of Figure 2.18.a but let us suppose that a degradation has occurred in the normal operation conditions of the machine. As long as this degradation is greater than the threshold defining the failure, the machine is considered to be in normal functioning. The drift resulted from the degradation will move the class N but without impacting the decision boundary. Indeed the new location of the class N after the drift remains in the Normal region and there is no misclassified patterns yet. Therefore, this drift is virtual since it does not impact the decision boundary or the performance of the classifier (see Figure 2.20).

2.4.2 Concept drift characteristics

The changes (Concept drift) are characterized by their speed, their severity, their dynamic, their frequency of occurrence and detectability [59],[67]. These characteristics are necessary to determine the most efficient indicators for detecting these

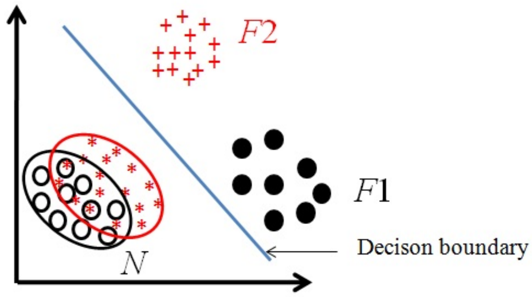

Figure 2.20: Virtual drift representing a degradation of the normal class N of the example of Figure 2.18.(b).

changes and the most appropriate way to update the classifier following the occurrence of these changes.

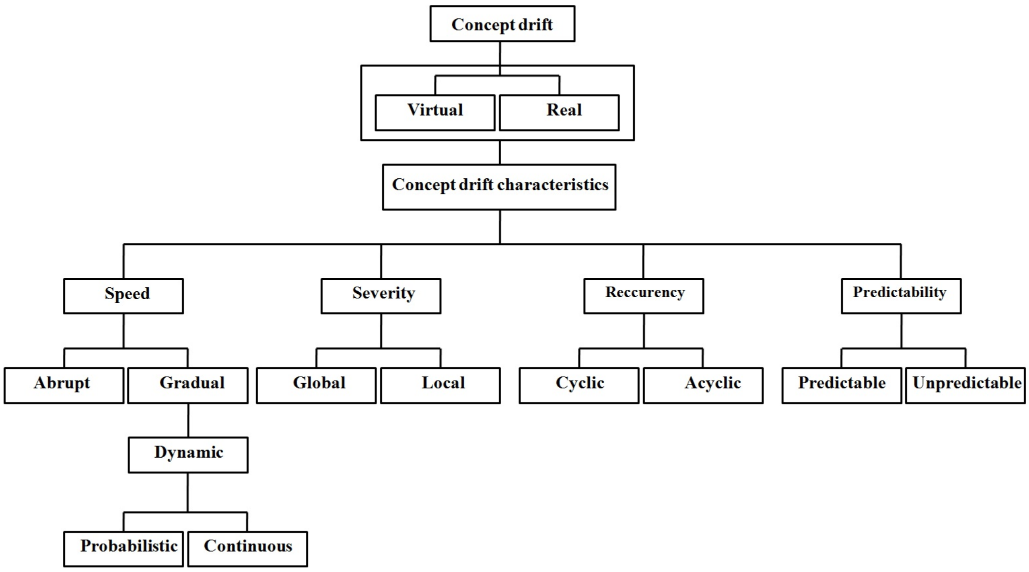

Figure 2.21: Concept drift characteristics

2.4.2.1 Speed of drift

The duration of drift, also called drifting time or drift width, is the number of times steps for a new concept to replace the old one. According to [67] speed is the inverse of the drifting time. In the sense that a higher speed is related to a lower number of time steps and a lower speed is related to higher number of time steps. According to speed, drifts can be categorized as either abrupt or gradual.

- **Abrupt drift** occurs when the new concept suddenly replaces the old one in short drifting time. This kind of drift immediately deteriorates the learner performance, as the new concept quickly substitutes the old one. This kind of drift have similar behavior as abrupt faults (see Figure 2.9).

- **Gradual drift** occurs when the drifting time is relatively large. This kind of drift is harder to detect since it creates a period of uncertainly between stable states. There are two types of gradual drift: probabilistic and continues.

  1. **The gradual probabilistic drift** refers to a period when both new and old concepts are active. In the sense that there is a weighted combination between data source S_1 sampled from the old concept and S_2 sampled from the new concept. As time passes, the probability of sampling from source S_1 decreases whereas the probability of sampling from source S_2 increases until the new concept totally replaces the old one. This kind of drift have similar behavior as intermittent faults (see Figure 2.10).
  2. **The gradual continuous drift** is when the concept itself continuously changes from the old to the new concept, by suffering small modifications at every time step. Notice that these changes are so small that they are only noticed during a long time period. This kind of drift have similar behavior as gradual faults (see Figure 2.11).

2.4.2.2 Severity of drift

The criterion severity has been used in [49],[77],[14]. It refers to the amount of change caused by the drift. According to severity criterion, there are local and global drifts.

- **Local concept drift** [86],[45],[49], we may define it as changes that occur in some regions of the instance space. Hence, when looking at the overall instance space, we notice that only some subsets are affected by the drift. The time until local concept drift is detected can be arbitrarily long. This is due to the rarity of data samples representing the drift, which in turn makes it difficult to confirm the presence of drift. Moreover, in some cases, local drift can be considered as noises by confusion, which makes the model unstable. Hence, to overcome the instability, the model has to 1) effectively differentiate between local changes and noises, and 2) deal with the scarcity of instances that represent the drift in order to effectively update the learner.

- **Global concept drift** [86],[45],[49],is easier to detect since it affects the overall instance space. in such case, the difference between the old and the new concept is more noticeable and the drift can be earlier detected.

The studied faults in this thesis manifest both local and global drift behaviors. Indeed, when the consequences of a fault are visible only in certain discrete modes, then the regions of the feature space representing the dynamic behavior of the WTs will be impacted by the fault. While the other regions describing the normal dynamic behavior of the system in the other discrete modes will not be impacted by the fault (see Figure 2.22). When the fault consequences are visible in all discrete modes (see Figure 2.23), then all the feature space describing the normal dynamic behavior of the system (WTs) will be impacted.

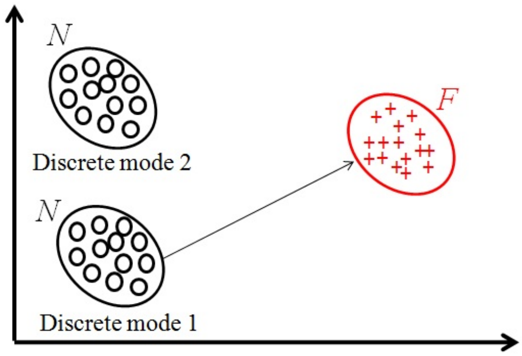

Figure 2.22: Fault entailing a local drift in the feature space for an example of a hybrid dynamic system with two discrete modes. Only the zone of the feature space occupied by the patterns of discrete mode 1 is impacted by the drift-like fault

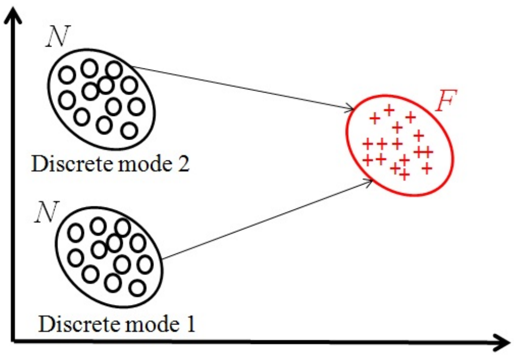

Figure 2.23: Fault entailing a global drift in the feature space for an example of a hybrid dynamic system with two discrete modes. All the zones of the feature space are impacted by the drift-like fault since the latter is active in all the system's discrete modes

2.4.2.3 Recurrency

Yet another characteristic of drift concerns recurrent concepts, i.e., previously active concepts that may reappear after some time. As stated in [67] recurrent drifts can have cyclic or unordered behavior.

- **The cyclic recurrent drift** may occur according to a certain periodicity or due to a seasonable trend. For instance, in the WT power converter, when fault impact a capacitor of multicellular converter, the output voltage of multicellular converter increase when the faulty capacitor is solicited, then in the others discrete mode where the faulty capacitor is not solicited the output voltage return to previous level. So the drift reappear according to the sequence of discrete modes decided by the controller.

- **The acyclic recurrent drift** may not be certainly periodic, i.e., it is not clear when the concept may reappear. For instance, in the WT pitch system, the pitch angle may increase due to the increase of wind speed, then return to previous level in the others wind speed.

It is worth underlining that the emergence of recurrent concepts can be abrupt or gradual; moreover they can locally or globally affect the instance space. Hence in reality often mixtures of many types of drifts can be observed during the transition phases.

2.4.2.4 Predictability

A change (concept drift) can be predictable or random. a concept drift is predictable when its occurrence follows a dynamic or a special mechanism. A random change is a change that cannot be predictable because its occurrence does not follow any rule nor mechanism.

2.4.3 Handling concept drift

Obviously, the drifting methods are expected to deal with the instability of the learner when drifts occur, in the sense that they have to effectively manage evolving data, otherwise their accuracy will degrade. There are two main categories of drift handling methods: blind and informed methods.

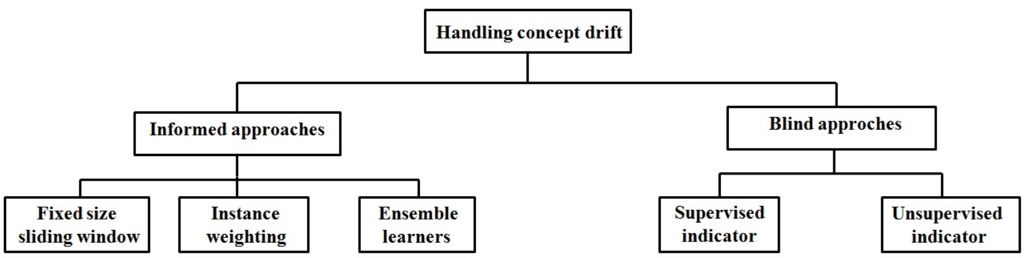

Figure 2.24: Handling concept drift methods

2.4.3.1 Informed methods

Generally, the choice of the method is related to the intention behind handling concept drift. For instance, in monitoring and control applications, it is primordial to detect anomalous activities and out-of-control behaviors. Such situation is often formulated as a detection task where the drift needs to be signaled. For this purpose, the Informed methods are the most appropriate as they explicitly detect drifts using triggering mechanisms. The triggering mechanisms, also known by drift detection mechanisms, are useful when we expect to provide description about the occurrence and the time detection of the drifts. From machine learning perspective, these mechanisms may monitor the performance indicators of a learner [45][6] [48], the estimators of data distributions[17] [61][40] [25] or the learners structure and parameters [35] in order to detect a drift.

They are reactive, because when a drift is detected, they can either relearn the model from scratch or update it using a recent selection of data (data window). In summary, the Informed methods process by:

Recent studies [6],[34],[82], 39 have opted to windows with dynamic size which are adaptively adjusted when a drift is detected thanks to triggering mechanisms. These triggering mechanisms, also known by drift detection mechanisms, are useful when we expect to provide description about the occurrence and the detection time of drift.

1. Detecting the drift.

2. Deciding which data to keep and which ones to forget.

3. Revising the current learner when significant change has been detected.

2.4.3.2 Blind methods

The Blind methods implicitly adapt the learner to the current concept at regular intervals without any drift detection. They discard old concepts at a constant speed independently of whether changes are happening or not. These approaches can be of good interest for handling gradual continuous drifts where the dissimilarity between consecutive data sources is not quite relevant to trigger a change.

Hereafter, some strategies used by Blind methods in order to handle drift:

- **Fixed size sliding window** [70], where the learner is periodically updated according to a fixed number of instances stored in the first-in-first-out (FIFO) data structure. In the sense that, whenever a new instance arrives, it is saved to memory and the oldest one is discarded.

- **Instance weighting,** where the learner is periodically updated according to the weighted instances from the training set. The instances can be weighted according their age, i.e., the most recent data should have the highest weights [21][76]; or according to their representativity to the current concept using entropy measure[107]

- **Ensemble learners** where the ensemble is continuously adapting its structure to represent the current concept. A possible strategy is replace the loser: where the individual learners are re-evaluated and the worst one is replaced by a new one trained on recent data. [53].

2.4.3.3 Informad Vs Blind methods discussion

The main limitation of Blind methods is slow reaction to drifts, because the updating process is the same whatever the drift is abrupt or gradual. Moreover, the regular updates can be too costly as the amount of arriving data may be overwhelming. These approaches can work well only if the speed and severity of the change are known before or if we have rigorous instructions provided by an expert about the nature of the drift, but this is rarely the case.

In the other side; the Informed methods are more reactive as they update the learner only if a drift is alarmed, which in turn may save time and resources. Moreover, they are able to provide useful description about the drift like: speed, severity, occurrence and the time detection. It is worth to underline that when using the Informed methods, we are not only concerned by preserving the learner performance, but also controlling the false alarms and the missed detections. Image, in monitoring and control applications, that at each false alarm, the system signals that there are some sensors which do not work anymore and may be replaced. In such case, an unnecessary maintenance is made and may cause time and resources waste.

2.4.4 Drift indicators

Generally the drift indicators can be divided into two big groups based on the monitoring measure to be used supervised indicators based approach and unsupervised indicators based approach. This measure is related to the drift type and the availability of prediction feedbacks.

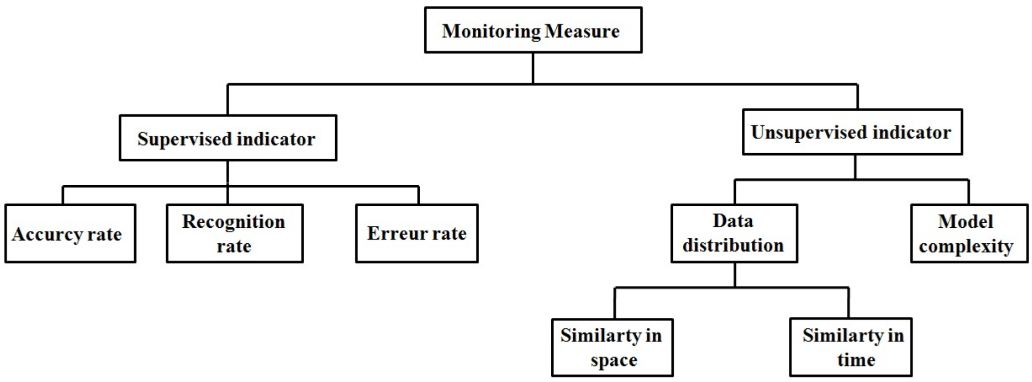

Figure 2.25: Type of drift monitoring indicators.

2.4.4.1 Supervised indicators based approach

Initially, the handling drift methods were focused on preserving the learner performance; thus they were interested by handling Real Concept Drifts. The main key for handling this type of drift relies on monitoring the learner feedbacks indicator like:

- **Error rate** is calculated as the sum of the number of misclassified points divided by the total number of classified Points. Generally for handling drifts, some approaches monitor the frequency of classification errors or the distance between two error of classification.

- **Recognition rate** is calculated for each class as number of correctly classified points in a class on the total number of points that should be in this class),

- **Accuracy rate** is calculated for each class as the number of points correctly classified in a class on the total number of points classified in this class).

		True class	
		Class1 Faulty operating condition	Class2 Normal operating condition
Predicted class	Class1 Faulty operating condition	a	b
	Class 2 Normal operating condition	c	d

Table 2.5: Classification matrix to extract supervised indicators

$$\text{Classification error rate} = \frac{\mathrm{b}+\mathrm{c}}{\mathrm{a}+\mathrm{b}+\mathrm{c}+\mathrm{d}} \tag{2.9}$$

$$\text{Recognition rate of class 1} = \frac{\mathrm{a}}{\mathrm{a}+\mathrm{c}} \tag{2.10}$$

$$\text{Recognition rate of class 2} = \frac{\mathrm{d}}{\mathrm{b}+\mathrm{d}} \tag{2.11}$$

$$\text{Accuracy rate of class 1} = \frac{\mathrm{a}}{\mathrm{a}+\mathrm{b}} \tag{2.12}$$

$$\text{Accuracy rate of class 2} = \frac{\mathrm{d}}{\mathrm{c}+\mathrm{d}} \tag{2.13}$$

These indicators have the advantage of being reliable and independent of the learning methodology used to construct the classifier. However, they operate in supervised mode, it means, they require the availability of the true class label of point already classified by the classifier. So this could delay the detection of changes, if the true class of point classified is not available immediately as is the case in most real applications. These three measures can be used together to obtain a more reliable and sensitive indicator of change in particular in the case where a class is much smaller than the other.

2.4.4.2 Unsupervised drift indicators

The unsupervised indicators are used for handling drifts that do not affect the decision boundaries. i.e. Virtual Concept Drift. Moreover, they are useful for detecting change when the prediction feedback is delayed, which can be of good interest for many real world applications where data are partially labeled.

These measures can be based on monitoring:

- **Data distribution** Quantifying the change in data distribution can be processed according to two dimensions:

  1. **Similarity in time:** How the data distribution is evolving from a time stamp to another. A drift occurs when there is a significant change between two distributions D_0 and D_1. This change is quantified by some dissimilarity measures: $Diss_{t_0(D_0,D_1)}$ at time point t_0. For this purpose many measures can be used like: The Sequential Probability Ratio Test (SPRT), CUSUM test and PHT test.
  2. **Similarity in space:** How the data distribution is evolving according to the feature space.The drift in data distribution may affect the class memberships or the repartition of the instances in the feature space. This drift can be quantified by a dissimilarity measure: $Diss_\lambda\ (D_0, D_1)$. For this purpose many measures can be used like: Euclidean distance [99][96], Mahalanobis distance [96] and Kullback distance[100]. This category of indicators will be developed in this dissertation.

- **Model Complexity:** The model complexity measure is based on monitoring the structure and/or the parameters of the model. For instance, the explosion of the number of rules for rule-based classifiers or the number of support vectors for SVM method can inform about an unusual model behavior . These indicators can perfectly operate in unsupervised mode. However, they can only be applicable to some specific classifiers.

Unsupervised drift indicators are independent of the learning methodology used and operate in non-supervised mode (not need to know the true class label of each point after its classification by the classifier). In this thesis this type of indicator will be used because we considered that just a data representing the normal operating condition are available, the faulty data, representing the failure operating conditions of components or subsystems, are considered to be a priori unknown.

The monitoring measure represent an essential point in concept drift tracking. Hereafter some promising ideas and future trends related to these measures:

- Combining supervised and unsupervised indicators for monitoring concept drift can be a promising tendency. To the best of our knowledge, this idea is not yet developed in the literature (until writing this dissertation). Combining supervised and unsupervised indicators can be beneficial for two reasons:

1. Handling different types of drift in the same time, in the sense that supervised indicators are used for handling Real Drift whereas unsupervised indicators are used for handling Virtual Drift.
2. Early detection, because some kind of Virtual drift may evolve and become Real drift, hence by using unsupervised indicators, we can expect the change; then by using the supervised indicators we can confirm it.

- Very often, in real world applications, data is heterogeneous and often can be represented over set of categorical attributes as well as numerical ones. In the last decade, many approaches have focused on detecting changes in numerical data; whereas the problem of detecting drift in categorical time-evolving data have not been considered extensively so far and remains a challenging issue [39][29][19][24].

2.5 Focus of research

The work in this dissertation will focus on the development of a hybrid dynamic classifier able to achieve the early diagnosis of drift-like faults in WTs, in particular pitch system and power converters system considered as two different class of hybrid dynamic systems. Few approaches have been proposed to achieve early fault diagnosis of WTs. These methods require a priori knowledge (data) about all faulty behaviors and do not integrate a mechanism to detect a drift by analyzing the characteristics of incoming data in order to update the model parameters and structure in response to this drift. Moreover, they do not integrate the hybrid dynamic aspect of the WT subsystems as pitch system and power converters. Consequently, the diagnosis performance (diagnosis delay) is decreased significantly for faults occurring in WT critical subsystems as pitch system and converters. The aim of this thesis is to propose an approach to achieve an early diagnosis of drift-like faults. This approach comes as an answer to the challenges that were defined in this chapter. The developed approach in this thesis, does not require an exhaustive amount of historical data. In addition, it does not require any physical knowledge concerning the degradation mechanisms. In this approach, the model parameters and structure are updated continuously according to the novelties and changes in either its internal dynamical states or in its environment conditions. This update enables a continuous learning of the system behaviors leading to improve or at least to maintain its performance over time. The developed approach builds a hybrid classifier able to change its decision function as well as its feature space according to the system internal state (discrete mode) and to abnormal changes (e.g., faults) in its environments. This allows to keep the useful patterns representative of the drift and therefore to detect it in its early stage. Consequently, detecting and following this drift can help to predict the occurrence of failure. The proposed data-mining scheme is presented in see Figure 2.26.

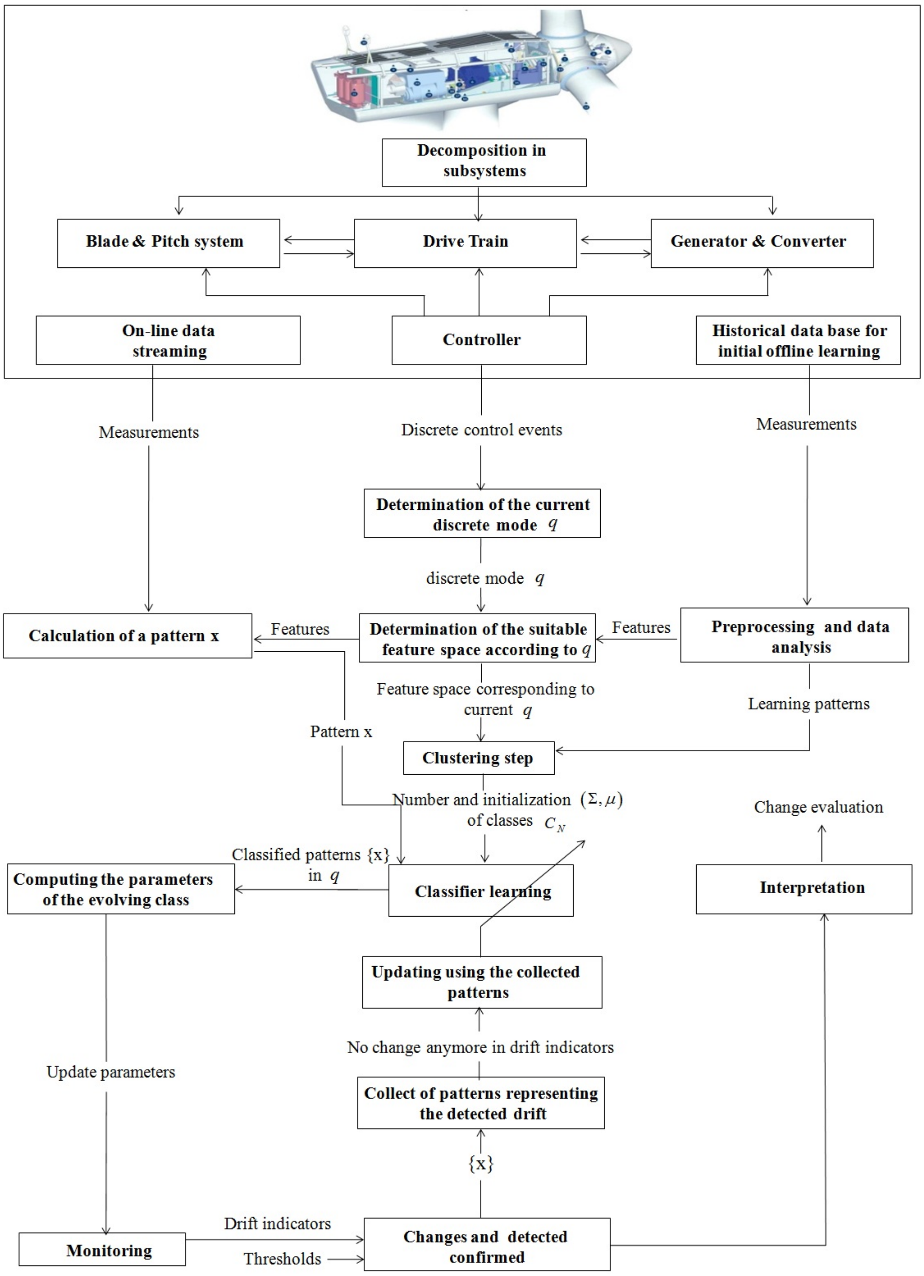

Figure 2.26: Proposed on-line adaptive scheme used in order to achieve drift-like fault diagnosis of WTs.

2.6 Summary

In this chapter, the interest, motivation and challenges of achieving an early fault diagnosis in WTs are discussed. The goal is to show the importance of an early fault diagnosis to increase the availability and safety of WTs and to reduce their maintenance costs. Then, the different methods of the literature used to achieve the fault diagnosis of WTs are presented and classified according to two main categories: internal and external methods. Due to the increasing complexity of WTs and to the non-linear evolution of their dynamics in non-stationary environments, it is not feasible to design a model analytically. Therefore, internal methods may not allow achieving a reliable and efficient fault diagnosis. The external methods may represent an alternative since they do not need any mathematical model to describe the system dynamical behaviors. Machine learning and data mining approaches are particularly interesting to achieve the fault diagnosis in WTs since they build a generic model regardless the application domain. They use exclusively a set of measurements or/and heuristic knowledge about the system operating states to build a mapping from the measurement space into a decision space.

Although machine learning and data mining approaches were applied successfully to the fault diagnosis of WTs, they suffer from several drawbacks. Firstly, a prior knowledge about all the faulty behaviors in WT is required. This is very hard to obtain. Secondly, these approaches do not include any mechanism to detect a drift of the operating conditions from normal to a failure. Thirdly, they do not consider the interactions between the continuous and discrete dynamics of certain WT components. Considering these interactions can help to increase the discrimination between the normal and operating conditions in the feature space. Therefore, they are not able to achieve an early fault diagnosis of drift-like faults. An alternative to overcome these short comes could be the use of on-line and self-adaptive machine learning and data mining scheme allowing to detect an abnormal drift in its early stage and to update the model parameters and structure in order to include the new information about the occurrence of this new failure. In Chapter 3, this on-line and self-adaptive machine learning and data mining scheme will be developed and applied to achieve the drift-like fault diagnosis of the pitch system.

Chapter 3

Hybrid dynamic classifier for simple and multiple drift-like faults diagnosis in wind turbine pitch system

Contents

3.1 Introduction

The search for alternative clean energy is undoubtedly becoming more and more important in modern societies. The growing interest in wind energy production has led to the design of sophisticated wind turbines (WTs). Like every other complex and heterogeneous system, WTs are faced to the occurrence of faults that can impact their performance as well as their security. Therefore, it is crucial to design a reliable automated diagnostic system in order to achieve fault detection and isolation in early stage.

This chapter presents a new data-driven based approach in order to achieve a reliable drift monitoring and diagnosis of simple and multiple drift-like fault that can affect wind turbine pitch system. This approach takes into account the different dynamical behaviors of WTs according to the wind speed. The goal is to detect a drift from normal operating conditions using only the recent and useful data. Initial off-line modeling allows constructing initial classes based on the historical data set. These classes characterize the operating conditions of the pitch system (normal/faulty) and are represented by restricted zones in the feature space. The latter is formed by sensitive features to pitch actuator and sensor operating conditions in order to distinguish any drift from normal to fault operating conditions. The modeling tool is an algorithm called AuDyC (Auto-Adaptive Dynamical Clustering) used to initialize the classes that will be dynamically updated.

In this work, two two-dimensional feature spaces are constructed, one for the sensor faults and one for the actuator faults. The faulty classes, representing the failure operating conditions of pitch actuator and sensor, are considered to be a priori unknown. There are two known classes in advance. The first class represents the pitch actuator normal operating conditions and the second class represents the pitch sensor normal operating conditions. It considers gradual degradations in pitch actuator or sensor operating condition as a drift in the characteristics of normal class over time. Detecting and following this drift can help to predict the occurrence of pitch actuator or sensor failure.

The drift-like fault is monitored using two drift indicators: one to detect a drift and the second one to confirm it. When the drift is detected by the first indicator, a warning is emitted to human operators. Then, the second drift indicator confirms this drift in order to inform human operators of the necessity to react by taking the adequate correction actions.

The proposed data-driven approach is composed of five main steps: processing and data analysis, clustering and classification, drift monitoring, updating and interpretation steps.

Chapter 3 is organized as follows. Firstly, the WT benchmark and the generated fault scenarios are described. Then, the proposed approach to achieve drift-like fault detection of pitch actuators and sensors is detailed. Finally, the obtained results based on the use of the WT benchmark are presented.

3.2 Challenges and motivations of fault diagnosis in wind turbine pitch system

Fault diagnosis of WTs is a challenging task because of the high variability of the wind speed and the confusion between faults and noises as well as outliers. However, the fault diagnosis of pitch system is particularly a challenging task because of (i) the occurrence of pitch system faults in power optimization zone in which the fault consequences are hidden and (ii) the actions of the control feedback which compensate the fault effects. The role of the pitch system is to adjust the pitch of a blade by rotating it depending on the pitch angle position reference provided by the controller. The latter decides the pitch angle position reference according to the wind speed in order to allow an optimum energy production.

The operating conditions of WTs, or one of its components, change from normal to faulty either abruptly or gradually. In the case of gradual change, WT begins to malfunction (degraded behavior) until the failure takes over completely. The detection of this drift from normal to faulty operating conditions in its early stage can help providing a time to achieve appropriate corrective actions leading to decrease the maintenance costs and to increase the availability time. Therefore, developing a drift monitoring and diagnosis module for pitch system is of particular interest for WTs industry due to their operational & maintenance costs as well as their essential role in optimizing the energy production.

Few approaches have been proposed to achieve early fault diagnosis of WTs, in particular pitch actuators and sensors. This is due to the fact that modeling component degradation in strong non-linear and complex non-stationary environments is very hard task. Examples of these methods, we can cite genetic algorithm [56], neural network, the boosting tree algorithm, and support vector machine [55]. These methods do not integrate a mechanism to detect a drift by analyzing the characteristics of incoming data and to update the model parameters and structure in response to this drift. Therefore, they do not achieve a reliable early diagnosis. Consequently, the diagnosis performance (diagnosis delay) is decreased significantly for faults occurring in WT critical subsystems as pitch systems ones.

3.3 Pitch system within wind turbines

The wind turbine model under study is composed of five principal parts: the blades, the drive train, the generator with the converter, and the controller (see Figure 3.1). It can be seen that the blades are fixed to the main axis, which in turn is connected to the generator through the drive train. The generator is electrically connected to the converter, which in turn is connected to a transformer. The blades are pitched by the pitch actuators.

The controller operates in four zones (see Figure 3.2). Zone 1 is the start-up of the turbines, zone 2 is power optimization, zone 3 is constant power production and zone 4 is no power production due to a too high wind speed.

In order to handle transitions between the control modes, the controller checks the operating zone in which the WT is by observing the wind speed. The transitions between the control modes change the dynamics of the pitch system. Each control

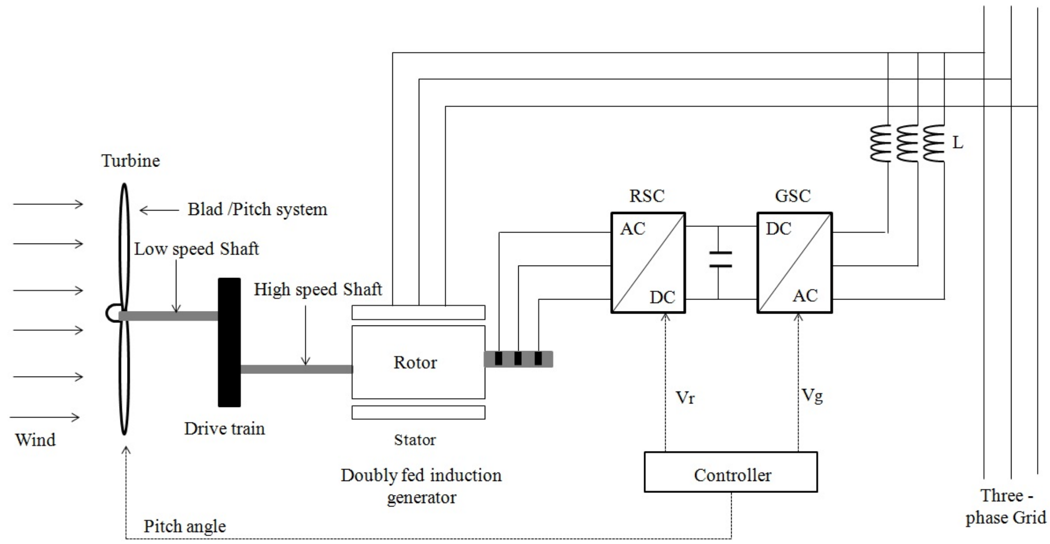

Figure 3.1: Wind turbine components.

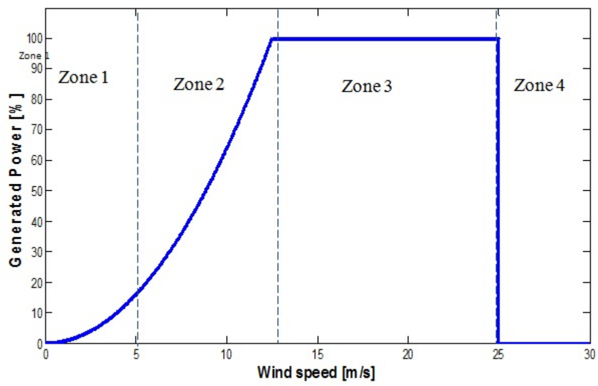

Figure 3.2: Reference power curve for the WT depending on the wind speed.

mode is active in one zone thus it is modeled by a finite state automaton. Each zone is represented by a state in which a specific control mode or strategy is defined. According to the wind speed, the control mode changes by switching from one mode or state to another mode or state. This switching between control modes is achieved by discrete events. As an example, if the WT was initially in control mode related to the zone 1, as long as the wind speed is less than a predefined threshold (5 m/s in Figure 3.2) E_{11} will be generated. E_{11} keeps the WT in control mode 1. If the wind speed is greater than the predefined threshold for zone 1 (5 m/s in Figure 3.2), The event E_{12} is generated leading to switch the WT from the control mode related to zone 1 to the control mode related to zone 2 (see Figure 3.3). Same reasoning can be applied for the other events.

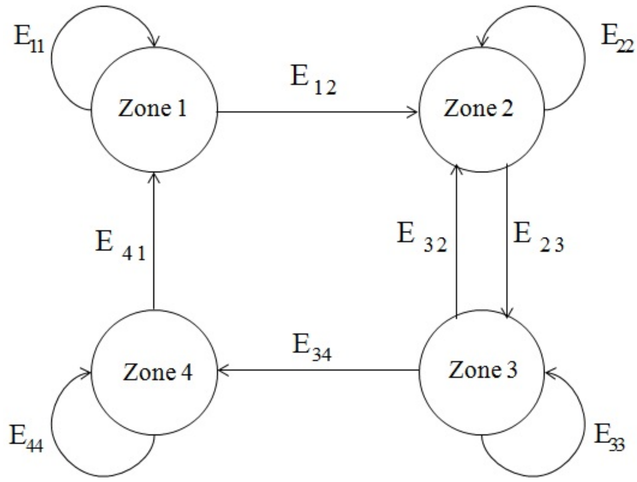

Figure 3.3: Controller operating zones modeled by a finite state automaton.

The focus of this benchmark model is on the operation of WT in zones 2 and 3. Two control strategies are applied to optimize the energy production and keep it constant at its optimal value: the converter torque control in zone 2 and the blades angle control in zone 3 (see Figure 3.4). In zone 2, the WT is controlled so that it produces as much energy as possible. To do so, the blades angle is maintained equal to 0°and the tip speed ratio is kept constant at its optimal value. The latter is regulated by the rotating speed control by tuning the converter torque. Once the optimal power production is achieved, the blades angle control maintains the converter torque constant and adjusts the rotating speed by controlling the blades angle. The latter modifies the transfer of the aerodynamic power of the wind on the blades. In this work, the controller modes are modeled by a finite state automaton containing two states (see Figure 3.4). In the following, zones 2 and 3, respectively, correspond to control modes 1 and 2:

Control Mode 1: In this control mode, the power optimum value is achieved by setting the pitch reference to zero $\beta[t] = 0$ and the reference torque to the converter $\tau_{g,r}$ as follows:

$$\tau_{g,r} = K_{opt} \times \left(\frac{\omega_g\,[t]}{N_g}\right)^2 \tag{3.1}$$

N_g is the gear ratio and n is the sampling time.
Where

$$K_{opt} = \frac{1}{2}\rho A R^3 \frac{C_{P_{\max}}}{\lambda^3_{opt}} \tag{3.2}$$

with ρ the air density, A the area swept by the turbine blades, $C_{P_{max}}$ the maximum value of power coefficient, and λ_{opt} the optimal value of λ is found as the optimum point in the power coefficient C_P mapping of the WT. The power coefficient mapping characterizes the efficiency of energy and it depend on λ and β.

Control Mode 2: In this mode, the major control actions are handled by the pitch system using a Proportional Integral (PI) controller trying to keep $\omega_g[t]$ at ω_g.

$$\beta_r(t) = \beta_r(t-1) + k_p.e(t) + (k_i.T_s.k_p).e(t-1) \tag{3.3}$$

When $e(t) = \omega_r(t) - \omega_{nom}$. In this case the converter reference is used to suppress fast disturbances:

$$\tau_{g,r}(t) = \frac{P_r(t)}{\omega_t(t)} \tag{3.4}$$

The control mode should switch from mode 1 to mode 2 if the following condition is satisfied:

$$E_{23} : \omega_g(t) \geq \omega_{nom} \tag{3.5}$$

The satisfaction of this condition generates a discrete event, E_{23}, allowing the switching from control mode 1 to control mode 2. The goal to obtain P_g equal to P_r. This condition is satisfied when the wind speed is greater than predefined threshold for zone 2 (12.5 m/s in Figure 3.2). Likewise, the control mode should switch from control mode 2 to control mode 1 if the following condition is satisfied:

$$E_{32} : \omega_g(t) < \omega_{nom} - \omega_\Delta \tag{3.6}$$

Where ω_{nom} is the nominal generator speed and ω_Δ is a small offset subtracted from the nominal generator speed to introduce some hysteresis in the switching scheme, thereby avoiding that the control modes are switching all the time [74]. The satisfaction of this condition generates a discrete event, E_{32}, allowing the switching from control mode 2 to control mode 1. This condition is satisfied when the wind speed is less than the wind speed threshold defined for zone 3 (12.5 m/s in Figure 3.2).

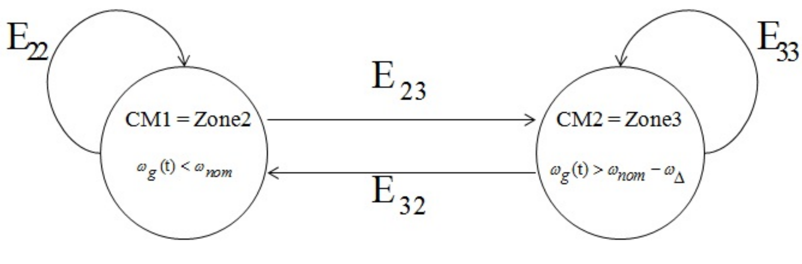

Figure 3.4: Controller modes modeled by a finite state automaton

As we said before, the benchmark model allows simulating the WT behavior in two power zones: 1) zone 2 (power optimization) where τ_g is controlled and β_r is equal to zero and; 2) zone 3 (optimal energy production) where τ_g is kept β_r

constant and is controlled. In this chapter, we focus on pitch actuator and sensor faults as it is discussed in subsection 2.

3.4 Pitch system description

The considered WT is horizontal-axis based with three blades. Each blade is equipped with an actuator. The role of the pitch actuator is to adjust the pitch of a blade by rotating it; Each actuator is provided by the same pitch angle reference β_r. The pitch angle of a blade is measured on the cylinder of the pitch actuator, each pitch position (angle) β_{m_i} where $i \in \{1, 2, 3\}$ is measured with two sensors where index m_i represents the i^{th} sensor of the corresponding variable (see Figure 3.5). The pitch system feedback β_f is an internal variable used to model the pitch position error caused by sensor faults:

$$\beta_f = \beta_r - \frac{1}{2}(\beta_{k,m1} + \beta_{k,m2}) \tag{3.7}$$

The controller is fed by the mean value of the readings of the two sensors. Hence, this sensor fault is modeled as a change in the pitch references, meaning that a sensor fault resulting in changed mean value should also change the pitch reference accordingly [74].

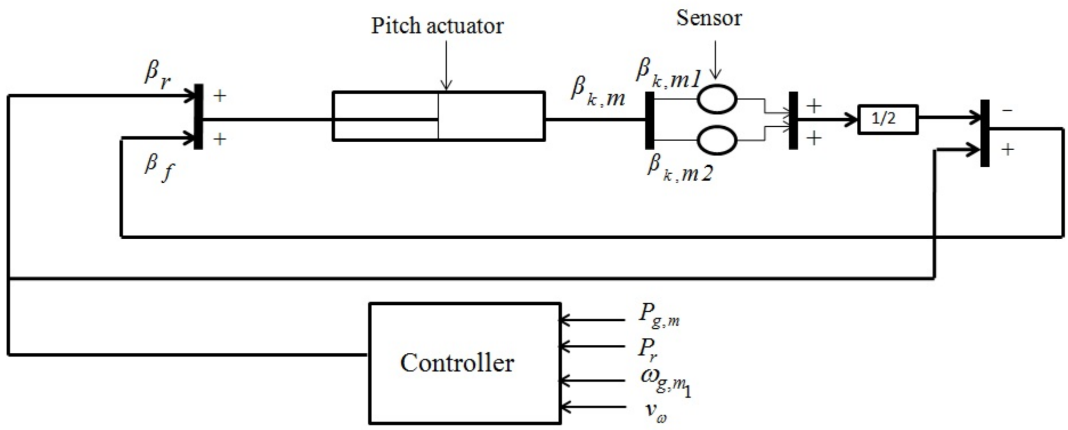

Figure 3.5: Block diagram of pitch system for the blade k, $(k = 1, 2, 3)$

3.5 Pitch system modeling

The hydraulic pitch system is modeled in the benchmark as a closed loop of dynamic system. The state representation of the nominal pitch system dynamics is defined as follows [74]:

$$\begin{aligned}
&\dot{x}_p = A_p x_p + B_p\left(\beta_r + \beta_f\right) \\
&y_p = C_p x_p \\
&A_p = \begin{bmatrix} 0 & 1 \\ -\omega_n^2 & -2\zeta\omega_n \end{bmatrix} \\
&B_p = \begin{bmatrix} 0 \\ \omega_n^2 \end{bmatrix} \\
&C_p = \begin{bmatrix} 0 & 1 \end{bmatrix}
\end{aligned} \tag{3.8}$$

The state vector $x_p = \begin{bmatrix} \dot{\beta}_k & \beta_k \end{bmatrix}^T$ is composed of pitch angular speed $\dot{\beta}_k$, and position β_i for each blade $k : (k = 1, 2, 3)$. y_p is the measured pitch position, β_r is the pitch angle position reference provided by the controller, and β_r is the feedback pitch system (see Figure 3.5). ω_n, ζ are the parameters of the pitch system where ω_n represent the natural frequencies and ζ is the damping ratio.

The pitch system represent a hybrid dynamic system and especially it belongs to the class of Discretely Controlled Jumping Systems (DCJS). The pitch system state variable $x_p = \begin{bmatrix} \dot{\beta}_k & \beta_k \end{bmatrix}^T$ changes discontinuously under the influence of an external action defined by Equation 3.5and 3.6.

3.6 Pitch system drift-like fault scenarios generation

In this chapter the types of fault which are considered in this work are pitch actuator and pitch sensor faults. The following subsections detail the generation of several scenarios representing drift-like faults with three different speeds in pitch actuator, in pitch sensor β_{m1} and pitch sensor β_{m2}, and in both pitch sensors β_{m1} and β_{m2}.

3.6.1 Actuator drift-like fault

The pitch actuator fault considered in this chapter is caused by air content increase in the actuators oil. This fault is modeled as a gradual change in the parameters ω_n, ζ of pitch actuator n°3 [74]. Nine scenarios for this fault are generated in order to simulate slow, moderate and high degradation speeds represented by slow, moderate and high drift speeds. Each drift speed scenario is generated at three different time instances. Thus, parameters ω_n, ζ are changed linearly from ω_{n1}, ζ_1 to ω_{n2}, ζ_2 in a period of 30s, 60s and 90s, corresponding respectively to fast, moderate and slow drift speeds. Then, the fault remains active for 200s. Finally the parameters decrease again to return to their initial values (see Figure 3.6). The pitch actuator faults scenarios are summarized in Table3.1.

Fault N	Drift speed	Pitch Actuator drift-like fault	Period
F_{1h}	30s (High)	$\omega_{n1}, \zeta_1 \rightarrow \omega_{n2}, \zeta_2$	3400s-3630s
F_{1m}	60s (Medium)	$\omega_{n1}, \zeta_1 \rightarrow \omega_{n2}, \zeta_2$	3400s-3660s
F_{1s}	90s (Slow)	$\omega_{n1}, \zeta_1 \rightarrow \omega_{n2}, \zeta_2$	3400s-3690s
F_{2h}	30s	$\omega_{n1}, \zeta_1 \rightarrow \omega_{n2}, \zeta_2$	3500s-3730s
F_{2m}	60s	$\omega_{n1}, \zeta_1 \rightarrow \omega_{n2}, \zeta_2$	3500s-3760s
F_{2s}	90s	$\omega_{n1}, \zeta_1 \rightarrow \omega_{n2}, \zeta_2$	3500s-3790s
F_{3h}	30s	$\omega_{n1}, \zeta_1 \rightarrow \omega_{n2}, \zeta_2$	3600s-3830s
F_{3m}	60s	$\omega_{n1}, \zeta_1 \rightarrow \omega_{n2}, \zeta_2$	3600s-3860s
F_{3s}	90s	$\omega_{n1}, \zeta_1 \rightarrow \omega_{n2}, \zeta_2$	3600s-3890s

Table 3.1: Pitch actuator drift-like fault scenarios.

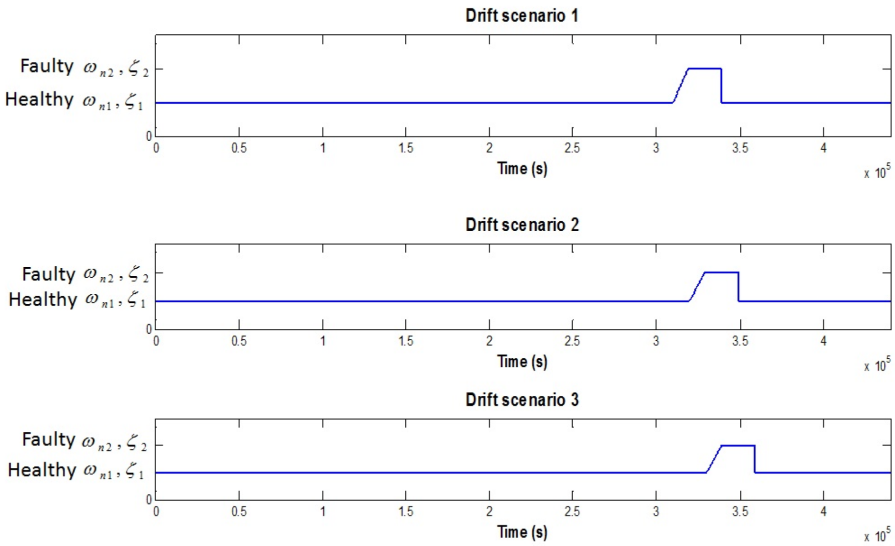

Figure 3.6: Actuator drift-like fault scenarios corresponding to high drift speed in 3 different time instances.

3.6.2 Sensor drift-like fault

Each blade is equipped with an actuator. Each actuator is provided by the same pitch angle reference β_r. In addition, each pitch position, (angle) β_{mi} is measured with two sensors where index i represents the i^{th} sensor of the corresponding variable. The fault scenarios related to simple drift-like fault in pitch sensor n°1 and sensor n°2 and multiple drift-like fault in both pitch position sensor n°1 and sensor n°2 in blade n°3 are summarized respectively in Table 3.2, Table 3.3 and Table 3.4. The state representation of the pitch system after the integration of a fault in sensor β_{mi}, $i \in \{1, 2\}$ is defined as follow:

$$\begin{aligned} \dot{x_p} &= Ax_p + Bu \\ y_p &= Cx_p + f(t) \\ f(t) &= \lambda_i.(t_b - t_e) \end{aligned} \tag{3.9}$$

Therefore the parameter λ_i, $i \in \{1, 2\}$ is used in the simulation to generate a fault in sensor β_{mi} during the time period $(t_b - t_e)$ where t_b is the start time and t_e is the end time of sensor drift-like fault.

3.6.2.1 Simple drift-like fault in sensor β_{m1}

In this chapter the simple drift-like fault scenarios in pitch sensor 1 (β_{m1}) scenarios are modeled as a gradual change in the coefficient λ_1 of pitch sensor n°1 in blade n°3 where t_b is the beginning of the drift and t_e is the end of the drift. Nine scenarios for simple sensor drift-like fault are generated in order to simulate slow, moderate and high degradation speeds represented by slow, moderate and high drift speeds (see Figure 3.2). Each drift speed scenario is generated at three different time instances. Thus, parameter λ_1 is changed linearly from λ_{1N} to λ_{1F} in a period of 30s, 60s and 90s, corresponding respectively to high, moderate and slow drift speeds. Then, the fault remains active for 200s. Finally the parameter λ_1 decreases again to return to its initial value λ_{1N} (see Figure 3.7 for the case of high drift speed in sensor 1 (β_{m1})).

Fault N°	Drift speed	Simple drift-like fault in pitch sensor β_{m1}	Period
F_{4h}	30s (High)	$\lambda_{1N} \rightarrow \lambda_{1F}$	2500s-2730s
F_{4m}	60s (Medium)	$\lambda_{1N} \rightarrow \lambda_{1F}$	2500s-2760s
F_{4s}	90s (Slow)	$\lambda_{1N} \rightarrow \lambda_{1F}$	2500s-2790s
F_{5h}	30s	$\lambda_{1N} \rightarrow \lambda_{1F}$	2600s-2830s
F_{5m}	60s	$\lambda_{1N} \rightarrow \lambda_{1F}$	2600s-2830s
F_{5s}	90s	$\lambda_{1N} \rightarrow \lambda_{1F}$	2600s-2890s
F_{6h}	30s	$\lambda_{1N} \rightarrow \lambda_{1F}$	2700s-2930s
F_{6m}	60s	$\lambda_{1N} \rightarrow \lambda_{1F}$	2700s-2960s
F_{6s}	90s	$\lambda_{1N} \rightarrow \lambda_{1F}$	2700s-2990s

Table 3.2: Simple drift-like fault scenarios in pitch sensor 1 (β_{m1}).

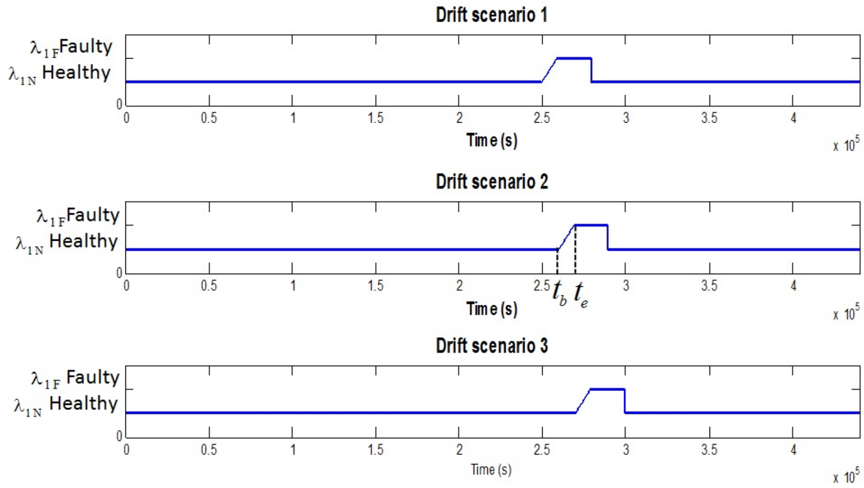

Figure 3.7: Simple drift-like fault scenarios in pitch sensor 1 (β_{m1}), corresponding to high drift speed in 3 different time instances t_b is the beginning time of the drift and t_e is the end of the drift.

3.6.2.2 Simple drift-like fault in sensor β_{m2}

The simple drift-like fault scenarios in pitch sensor 2 (β_{m2}) scenarios are modeled as a gradual change in the coefficient λ_2 of pitch sensor n°2 in blade n°3 where t_b is the beginning of the drift and t_e is the end of the drift. As for the case of simple drift-like fault in pitch sensor β_{m1} scenarios, nine scenarios for simple sensor drift-like fault are generated in order to simulate slow, moderate and high degradation speeds represented by slow, moderate and high drift speeds (see Figure 3.3). Each drift speed scenario is generated at three different time instances. Thus, parameter λ_2 is changed linearly from λ_{2N} to λ_{2F} in a period of 30s, 60s and 90s, corresponding respectively to high, moderate and slow drift speeds. Then, the fault remains active for 200s. Finally the parameter λ_2 decreases again to return to its initial value λ_{2N} (see Figure 3.7 for the case of high drift speed in sensor 2, (β_{m2})).

3.6.2.3 Multiple sensor drift-like fault

In this chapter the generated scenarios of the multiple drift-like fault in pitch sensor 1 (β_{m1}) and sensor 2 (β_{m2}) are modeled as a gradual change at the same time in the drift coefficient (λ_1 and λ_2) of both pitch sensors n°1 and pitch sensors n°2 in blade n°3. As for the case of simple drift-like fault in pitch sensor scenarios, nine scenarios for multiple sensor drift-like fault are generated in order to simulate slow, moderate and high degradation speeds representing by slow, moderate and high drift speeds (see Table 3.4). Each drift speed scenario is generated at three different time instances. Thus, parameters λ_1 and λ_2 are changed linearly from λ_{1N} and λ_{2N} to λ_{1F} and λ_{2F} in a period of 30s, 60s and 90s, corresponding respectively to high, moderate and slow drift speeds. Then, the fault remains active for 200s.

Fault N°	Drift speed	Simple drift-like fault in pitch sensor β_{m2}	Period
F_{7h}	30s (High)	$\lambda_{2N} \to \lambda_{2F}$	2800s-3030s
F_{7m}	60s (Medium)	$\lambda_{2N} \to \lambda_{2F}$	2800s-3060s
F_{7s}	90s (Slow)	$\lambda_{2N} \to \lambda_{2F}$	2800s-3090s
F_{8h}	30s	$\lambda_{2N} \to \lambda_{2F}$	2900s-3130s
F_{8m}	60s	$\lambda_{2N} \to \lambda_{2F}$	2900s-3130s
F_{8s}	90s	$\lambda_{2N} \to \lambda_{2F}$	2900s-3190s
F_{9h}	30s	$\lambda_{2N} \to \lambda_{2F}$	3000s-3230s
F_{9m}	60s	$\lambda_{2N} \to \lambda_{2F}$	3000s-3260s
F_{9s}	90s	$\lambda_{2N} \to \lambda_{2F}$	3000s-3290s

Table 3.3: Simple drift-like fault scenarios in pitch sensor 2 (β_{m2}).

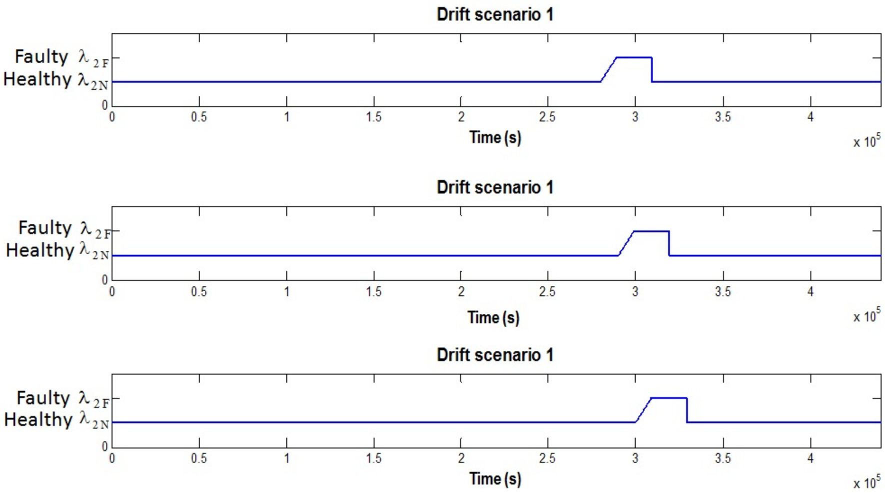

Figure 3.8: Simple drift-like fault scenarios in pitch sensor 2 (β_{m2}), corresponding to high drift speed in 3 different time instances.

Finally the parameter decreases again to return to their initial values (see Figure 3.9 for the case of high drift (degradation) speed in both sensor 1 (β_{m1}) and sensor 2 (β_{m2})).

Fault N°	Drift speed	Multiple drift-like fault in in pitch sensors β_{m2} and β_{m2}	Period
F_{10h}	30s (High)	$\lambda_1 N \to \lambda_{1F}$ and $\lambda_{2N} \to \lambda_{2F}$	3100s-3330s
F_{10m}	60s (Medium)	$\lambda_1 N \to \lambda_{1F}$ and $\lambda_{2N} \to \lambda_{2F}$	3100s-3360s
F_{10s}	90s (Slow)	$\lambda_1 N \to \lambda_{1F}$ and $\lambda_{2N} \to \lambda_{2F}$	3100s-3390s
F_{11h}	30s	$\lambda_{1N} \to \lambda_{1F}$ and $\lambda_{2N} \to \lambda_{2F}$	3200s-3430s
F_{11m}	60s	$\lambda_{1N} \to \lambda_{1F}$ and $\lambda_{2N} \to \lambda_{2F}$	3200s-3460s
F_{11s}	90s	$\lambda_{1N} \to \lambda_{1F}$ and $\lambda_{2N} \to \lambda_{2F}$	3200s-3490s
F_{12h}	30s	$\lambda_{1N} \to \lambda_{1F}$ and $\lambda_{2N} \to \lambda_{2F}$	3300s-3530s
F_{12m}	60s	$\lambda_{1N} \to \lambda_{1F}$ and $\lambda_{2N} \to \lambda_{2F}$	3300s-3560s
F_{12s}	90s	$\lambda_{1N} \to \lambda_{1F}$ and $\lambda_{2N} \to \lambda_{2F}$	3300s-3590s

Table 3.4: Multiple drift-like fault scenarios in pitch sensors (β_{m1}) and (β_{m2}).

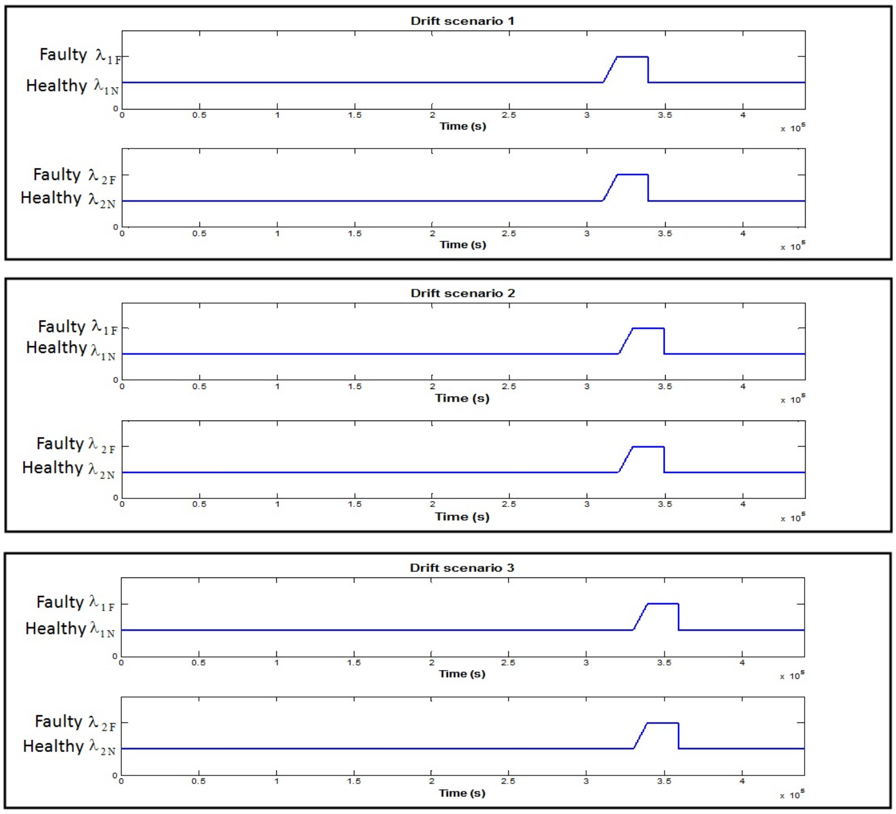

Figure 3.9: Multiple sensor drift-like fault scenarios in sensors (β_{m1}) and (β_{m2}) corresponding to high drift speed in 3 different time instances.

3.7 Proposed approach

In this section, hybrid dynamic data-driven approach is developed in order to achieve condition monitoring and drift like fault detection of pitch actuator and sensor. It performs predictive diagnosis by detecting a drift of the system operating conditions from normal to faulty modes. The proposed approach is based on 5 steps developed in the following subsections (see Figure 3.10).

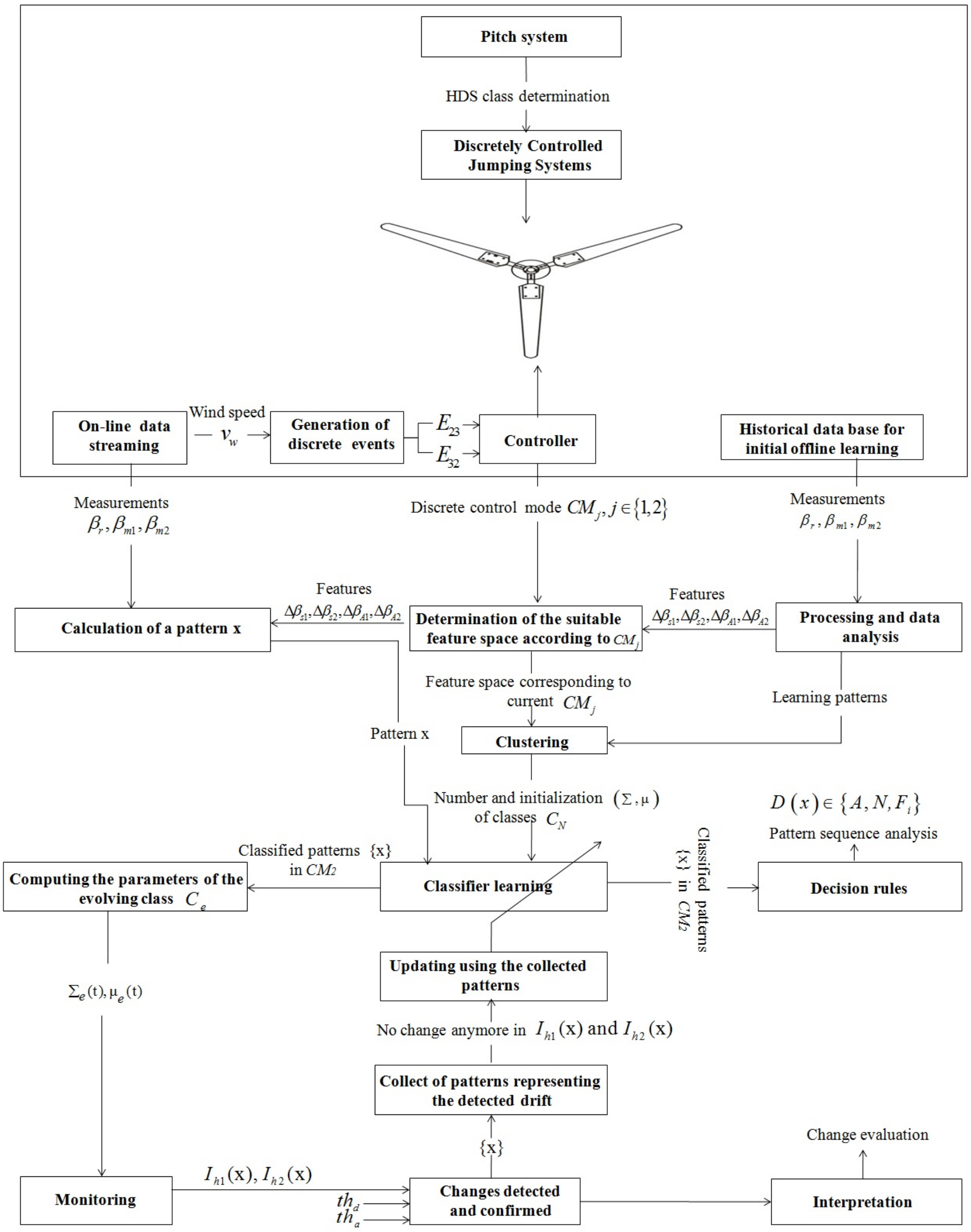

Figure 3.10: Proposed on-line adaptive scheme steps.

3.7.1 Processing and data analysis

This step aims at finding the features that are sensitive to the system operating conditions in order to construct the feature space. A feature space representing the operating conditions of each assembly of WT is defined, this feature space will be responsible of the detection and isolation of faults impacting this components. The research of sensitive features is based on the signals provided by the pitch sensors as well as the prior knowledge about the system dynamics. These features are chosen in order to maximize the discrimination between operating conditions in the feature space. In this chapter, two of two-dimension feature spaces are constructed; one for the sensor fault and one for the actuator fault [96]. The goal of the feature space use, at the level of component, is to facilitate the drift-like fault isolation and to enhance the diagnosis robustness.

The position of the pitch actuators is measured by two redundant sensors for each of the three pitch positions $\beta_{k,mi}$, $k = 1, 2, 3$, $i = 1, 2$, with the same reference angle β_r provided to each of them. In order to enhance the robustness against noise, the measurements are filtered by a first order filter using time constant $\tau = 0.06$.

Actuator feature space: For the drift like fault detection and isolation of the actuator fault, the residual $\Delta\beta_{Am}$, $m = 1, 2$ is based on the physical redundancy along with the physical features of the actuator and of the variability of the control pitch command $V(\beta_r)$.

Both features are residuals $\Delta\beta_{Am}$, $A = 1, 2$ computed by Equation 3.10 and Equation 3.11. Residuals $\Delta\beta_{Am}$, $A = 1, 2$, are generated by the comparison between the pitch angle measurement $\beta_{k,mi}$, $k = 1, 2, 3$, $i = 1, 2$ and the reference value of the pitch angle β_r (see Figure 3.5). The strong variability of the wind speed leads to a strong variability of the control pitch command which can increase the residuals in the normal functioning mode. To overcome this problem which can cause false alarms, the residuals are computed within a time window in order to take into account the control variability $V(\beta_r)$. The size of this time window is determined experimentally to achieve a tradeoff between the delay of drift detection and false drift detection.

$$\Delta\beta_{A1} = \frac{|\beta_r - \beta_{k,m1}|^2}{V(\beta_r)} \tag{3.10}$$

$$\Delta\beta_{A2} = \frac{|\beta_r - \beta_{k,m2}|^2}{V(\beta_r)} \tag{3.11}$$

$$V(\beta_r) = variance(\beta_r) \tag{3.12}$$

Sensor feature space: For the drift like fault detection and isolation of the sensor faults, we propose to explore the physical redundancy in order to generate residuals as follows:

$$\Delta\beta_{s1} = |\beta_r + \beta_f - \beta_{m1}| \tag{3.13}$$

$$\Delta\beta_{s2} = |\beta_r + \beta_f - \beta_{m2}| \tag{3.14}$$

To do so, the residual $\Delta\beta_{sn}$, $n = 1, 2$, is generated by the comparison between the pitch angle measurement β_{mi}, $i = 1, 2$, $m = 1, 2, 3$ and the command computed by the sum of the desired value of the pitch angle β_r and the feedback pitch system β_f (see Figure 3.5). The residual is computed within a time window which is tuned to be several times the actuator time response.

The evolution of these residuals with respect to each of the two sensors is considered as meaningful features. Indeed, the residual $\Delta\beta_{s1}$ respectively $\Delta\beta_{s2}$, is equal to zero when the corresponding sensor β_{m1} respectively β_{m2}, is in normal operating conditions. When, the sensor $\beta_m 1$ respectively $\beta_m 2$, is in faulty operating conditions, the residual $\Delta\beta_{s1}$, $\Delta\beta_{s2}$ will be different of zero because this sensor will not measure the new value of command $(\beta_r + \beta_f)$ (see Figure 3.5). Indeed, the command $(\beta_r + \beta_f)$ will change in order to compensate the difference between the two sensors due to the fault of sensor β_{m1} respectively β_{m2}.

3.7.2 Classifier learning and updating

The clustering looks to determine the number of classes contained in the learning set and to initialize their parameters. The classification aims at designing a classifier able to assign a new pattern to one of the learnt classes in the feature space. A new pattern characterizes the actual operating conditions (normal or faulty in response to the occurrence of a certain fault) of the system. Examples of these approaches, we can cite [26] and the references therein.

Auto-adaptive Dynamical Clustering Algorithm (AuDyC) [69] is selected in this thesis in order to achieve both clustering and classification. AuDyC computes the parameters of initial classes based on the statistical properties of data which are the mean and the variance-covariance matrix. These classes characterize the normal operating conditions of pitch actuators and sensors. AuDyC was chosen because it is unsupervised classification method and is able to model streams of patterns since it always reflects the final distribution of patterns in the features space. It uses a technique that is inspired from the Gaussian mixture model [69]. Let E^d be a d-dimensional feature space. Each feature vector $x \in E^d$ is called a pattern. The patterns are used to model Gaussian prototypes P^j characterized by a center $\mu_{Pj} \in R^{d\times 1}$ and a covariance matrix $\sum_{Pj} \in R^{d\times d}$. Each Gaussian prototype characterizes a class. A minimum number of N_{win} patterns are necessary to define one prototype, where N_{win} is a user-defined threshold. A class models operating conditions and gathers patterns that are similar one to each other. The similarity criterion that is used is the Gaussian membership degree. Faults will affect directly this distribution and this will be seen through the continuously updated parameters. More details about AuDyC related to merging classes, splitting classes, rules of recursive adaptation, similarity criteria, etc., can be found in [69].

3.7.2.1 Actuator operating conditions classifier

Figure 3.11 shows the classes representing normal and failure operating conditions of pitch actuator in the feature space constituted by the two residuals defined by Equation 3.10 and Equation 3.11. Due to the WT non-stationary environments, an overlapping region is created between the normal and failure classes (see Figure3.11). In this region, the consequences of the fault are hidden because the actuators are not solicited or are solicited for small angles. In both cases, normal and failure classes overlap because of pitch sensors noises and low wind speed (see Figure 3.11 and Figure 3.12).

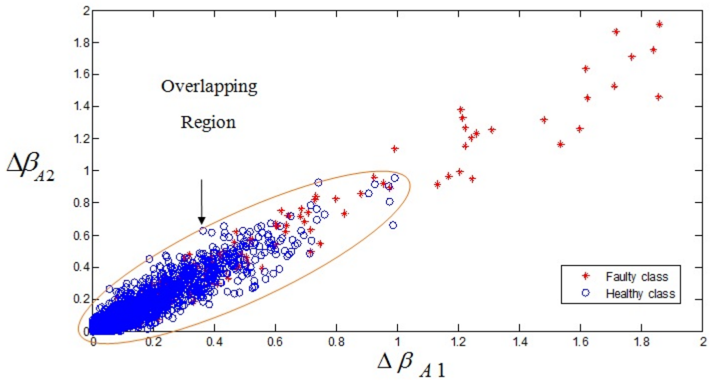

Figure 3.11: Large view of overlapping region for the third pitch actuator normal and failure operating conditions.

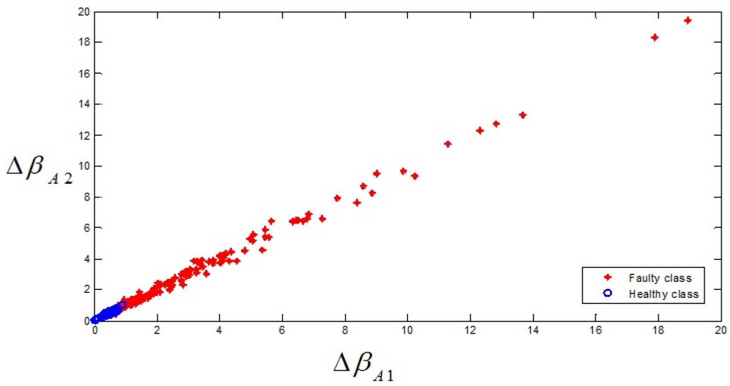

Figure 3.12: Feature space of the third pitch actuator normal and failure operating conditions.

In order to distinguish as much as possible the operating conditions (normal/faulty) and to improve the misclassification rate of the classifier, the normal and failure classes are split into three classes 1, 2 and 3 and the pitch actuator dynamics are represented by two different control modes. The first one corresponds to zone 2 representing the case of low wind speed; while the second control mode represents the case of zone 3 corresponding to high wind speed; (see Figure 3.13). Class 1 is the ambiguity class. It gathers the patterns representing pitch actuator normal or faulty operating conditions. This class represents the control mode 1. Class 2 represents the normal operating conditions in control mode 2. Class 3 represents pitch actuator failure class in control mode 2.

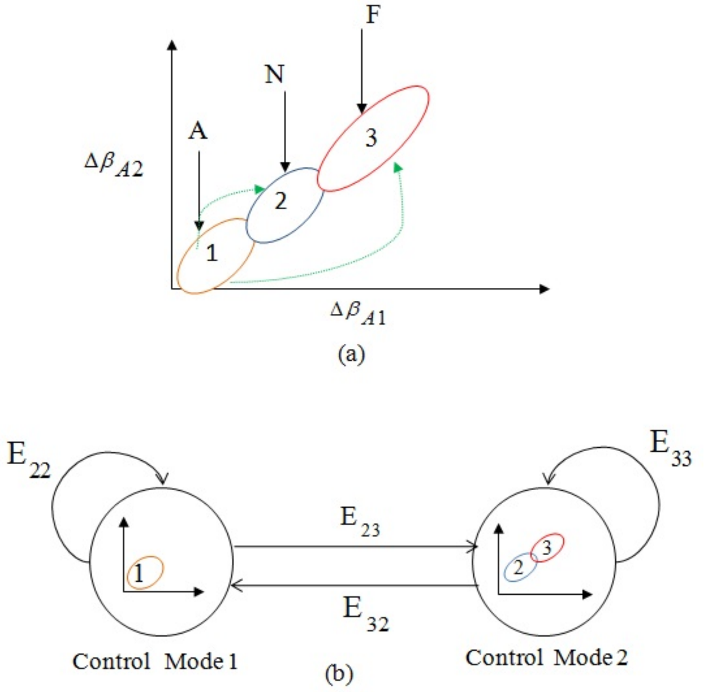

Figure 3.13: (a) Actuator decision space. (b) Control modes 1 and 2 modeled by a finite state automaton.

3.7.2.2 Sensor operating conditions classifier

In the sensor feature space, four classes are considered: the fault of sensor 1, β_{m1} , the fault of sensor 2, β_{m2}, the fault of both sensor 1,β_{m1} and sensor 2 β_{m2}, and the normal functioning. Figure 3.14 shows the classes representing normal and failure operating conditions of pitch sensor in the feature space constituted by the two residuals defined by Equation 3.13 and 3.14. In zone 2, the effects of this fault are hidden because the actuators are not operated. Moreover, it is strongly difficult to distinguish the fault occurrence to the noise in the case of small angles. Therefore an overlapping region is created between the normal and failure classes (see Figure 3.14 and Figure 3.19).

In order to answer the challenges inherent to the system operation, the normal and failure classes are split into five classes and the pitch actuator dynamics are represented by two different control modes in the same way as for actuator fault. The first one corresponds to the case of zone 2 low wind speed; while the second control mode represents the case of zone 3 high wind speed (see Figure 3.20). Class 1 is the ambiguity class. It gathers the patterns representing pitch sensor normal or faulty operating conditions. This class represents the control mode 1. Class 2 represents the normal operating conditions class in control mode 2. Class 3 represents failure class caused by simple drift-like fault in pitch sensor 1, β_{m1} in

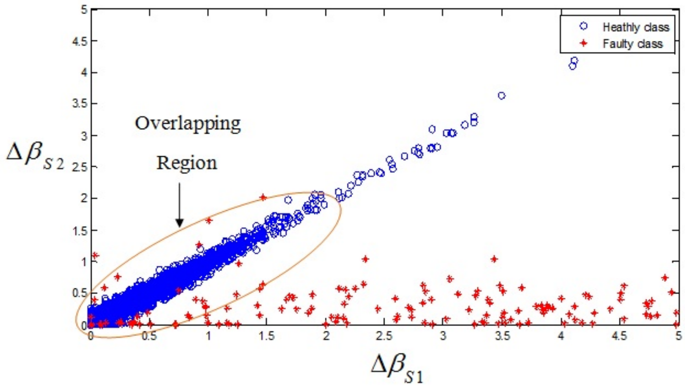

Figure 3.14: Large view of overlapping region for the pitch sensor normal and failure operating conditions in case of simple fault in pitch sensor 1, (β_{m1}).

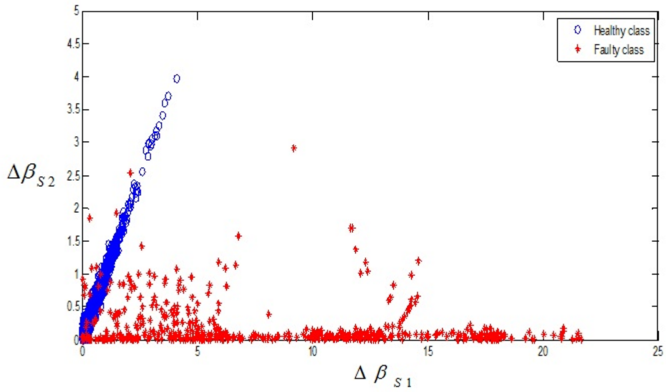

Figure 3.15: Feature space of the pitch sensor normal and failure operating conditions in case of simple fault in pitch sensor 1, (β_{m1}).

control mode 2, class 4 represents failure class caused by simple drift-like fault in pitch sensor 2, β_{m2} in control mode 2 and class 5 represents failure class caused by multiple drift-like fault in pitch sensor 1, β_{m1} and sensor 2, β_{m2} in control mode 2.

The updating step aims at reacting to the changes in classes characteristics in the feature space. AuDyC continuously updates the classes parameters by using the recursive adaptation Rules 3.15 and 3.16. In such a way, its validity and performance over time is preserved.

$$\mu_e(t) = \mu_e(t-1) + f(\mu_e(t-1), x^{new}, x^{old}, N_{win}) \tag{3.15}$$

$$\sum_e(t) = \sum_e(t-1) + g(\sum_e(t-1), \mu_e(t-1), x^{new}, x^{old}, N_{win}) \tag{3.16}$$

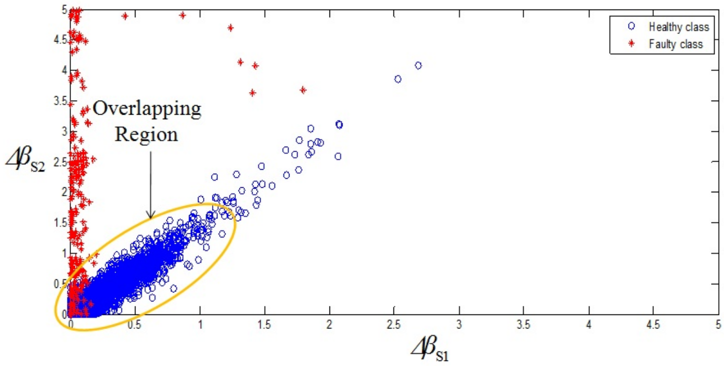

Figure 3.16: Large view of overlapping region for the pitch sensor normal and failure operating conditions in case of simple fault in pitch sensor 2, (β_{m2}).

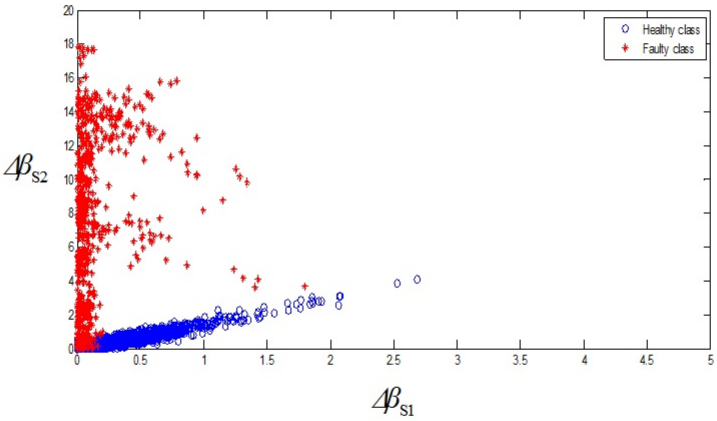

Figure 3.17: Feature space of the pitch sensor normal and failure operating conditions in case of simple fault in pitch sensor 2, (β_{m2}).

where x^{new} and x^{old} are respectively, the newest and the oldest arrived pattern in the time window N_{win} .

Initial off-line modeling allows the construction of initial classes that characterize knowledge from historical data. The historical data are usually sensor data that are saved. AuDyC is used to initialize the parameters of classes that will be dynamically updated. Knowledge of failure modes given from (labeled) historical data can help building a classification scheme for fault diagnosis. However, in reality, these data are hard to obtain.

In this work, we suppose that only data corresponding to normal operating conditions (normal classes) are known in advance. The training of the process by applying AuDyC is made based on features that are extracted from historical sensor data once finished; the class corresponding to normal operating conditions

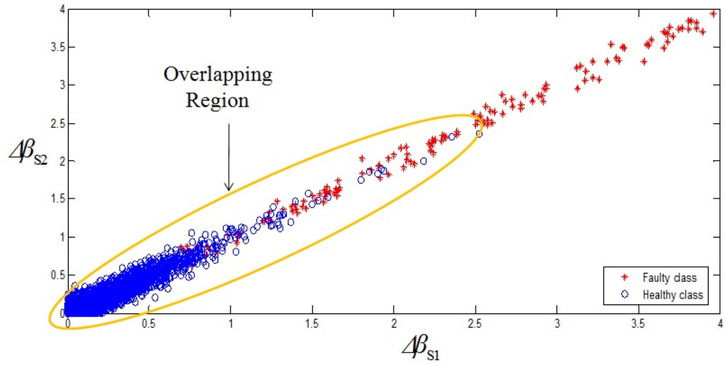

Figure 3.18: Large view of overlapping region for the pitch sensor normal and failure operating conditions n case of multiple fault in pitch sensor 1, (β_{m1}) and pitch sensor 2, β_{m2}.

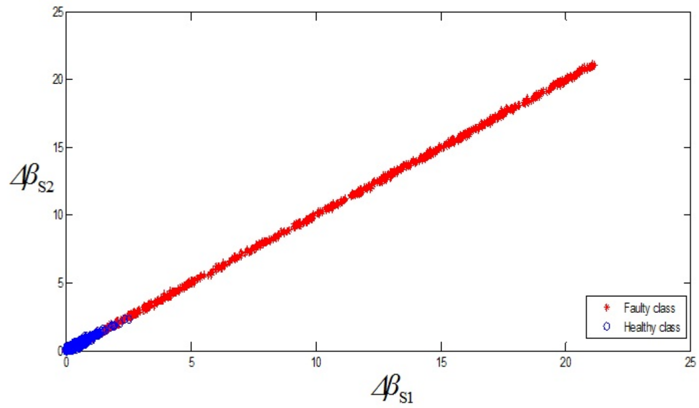

Figure 3.19: Feature space of the pitch sensor normal and failure operating conditions in case of multiple fault in sensor β_{m1} and β_{m2}.

is retained. We denote this class by $C_N = (\mu_N, \Sigma_N)$.

In on-line functioning, the parameters of C_N are dynamically updated by AuDyC for each new pattern arrived in control mode 2. This yields changes in the class parameters which continuously reflect the distribution of the newest arriving patterns. We denote by $C_e = (\mu_e, \Sigma_e)$ the evolving classes in feature space. We have $C_e(t = 0) = (\mu_e, \Sigma_e) = C_N$.

In control mode 1 of pitch actuator or sensor, normal and faulty behaviors cannot be distinguished. Thus, in the proposed approach, the decisions about the status (normal/faulty) of patterns located in this region are delayed. Therefore in this case, the classifier will not be updated in order to avoid integrating in the drift time window useless patterns. In order to detect the drift as soon as possible, AuDyC updates the classes parameters by using a window that contains only the

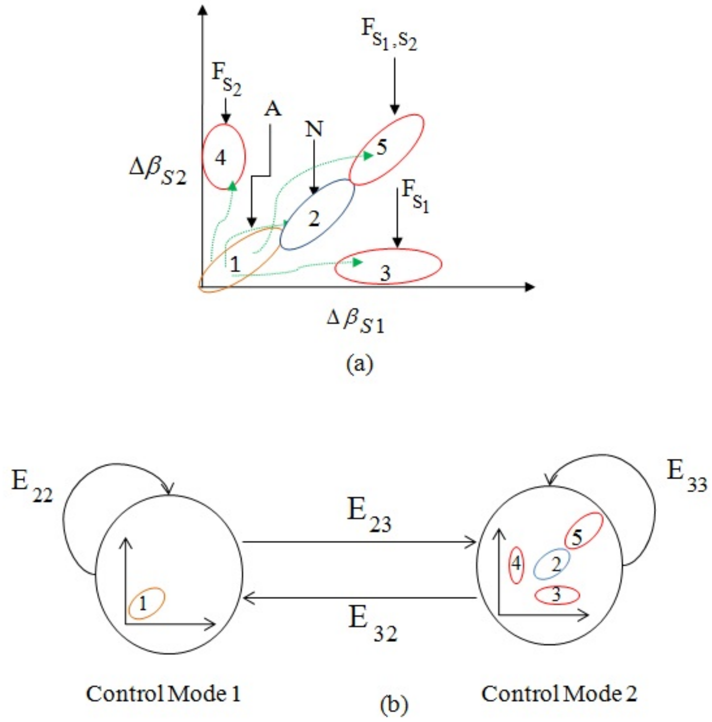

Figure 3.20: (a) Sensor decision space. (b) Control modes 1 and 2 modeled by a finite state automaton.

patterns belonging to control mode 2. AuDyC is dynamic by nature in the sense that it continuously updates the parameters of the classes as new patterns arrive.

3.7.3 Pattern decision analysis

When a new pattern is classified in the ambiguity class (A), in actuator or sensor feature space, assigning it to normal or failure operating conditions is a risky decision since normal and failure classes are overlapped in this region of the feature space. In order to reduce this risk, the decision about the status (normal or faulty) of any pattern classified in this region is delayed by assigning the label A (ambiguity decision). Then, this ambiguity can be removed by analyzing the past and future decisions of this pattern. The analysis of the pattern decision sequence is achieved by using a set of decision rules allowing assigning to ambiguity patterns label N or label F (normal or faulty) as follows. Let us suppose that $X_A = \{x_t, x_{t+1}, \ldots, x_{t+n}\}$ is a set of patterns associated with decision A. Let x_{t-1} be the previous pattern arrived just before x_t . Let $D(x_{t-1}) \in \{A, N, F_i\}$ be the decision of this pattern. Let x_{t+n+1} the pattern arrived just after x_{t+n}. Let $D(x_{t+n+1}) \in \{A, N, F_i\}$ be the decision for this pattern. Then, the decision can be updated as follows:

$$D(x_{t-1}) = N \wedge D(x_{t+n+1}) = N \Rightarrow D(x) = N, \forall x \in X_A \tag{3.17}$$

$$D(x_{t-1}) = F \wedge D(x_{t+n+1}) = F \Rightarrow D(x) = F, \forall x \in X_A \tag{3.18}$$

$$D(x_{t-1}) = N \wedge D(x_{t+n+1}) = F \Rightarrow D(x) = A, \forall x \in X_A \tag{3.19}$$

$$D(x_{t-1}) = F \wedge D(x_{t+n+1}) = N \Rightarrow D(x) = A, \forall x \in X_A \tag{3.20}$$

Where $\wedge$ refers to And logical operation.

Rule 3.19 signifies that the fault has occurred somewhere in control mode 1 where its consequences on the pitch system dynamical behavior can be observed. Rule 3.20 indicates that the failure has disappeared in the control mode 1 either because of maintenance actions or because the fault is intermittent.

3.7.4 Drift monitoring and interpretation

The key problem of drift monitoring is to distinguish between variations due to stochastic perturbations and variations caused by unexpected changes in a systems state. If the sequence of observations is noisy, it may contain some inconsistent observations or measurements errors (outliers) that are random and may never appear again. Therefore, it is reasonable to monitor a system and to process observations within time windows in order to average and reduce the noise influence. Moreover, the information about possible structural changes within time windows can be interpreted and processed more easily. As a result, a more reliable classifier update can be achieved by monitoring within time windows. The latter must include enough of patterns representing the drift.

To distinguish the useful patterns, the pitch actuator and sensor dynamics are represented by two different control modes. In the control mode 2, the degradation consequences of pitch actuator or sensor can be observed. Therefore, all patterns in this mode are useful to be analyzed and to be included in the drift time window. In the control mode 1, the degradation consequences are masked. Patterns representing normal operating conditions cannot be distinguished from patterns representing pitch actuator or sensor degradations. Therefore in this case, no decision (normal/drift) will be taken in order to avoid integrating in the drift time window useless patterns.

The proposed scheme makes use of classes parameters (Mean, Variance-covariance matrix) which are dynamically updated at each time but only with the patterns belonging to control mode 2. Drift indicators are defined based on these parameters and the detection of faults inception will be made based on their values. We define two drift indicators $I_{h1}(x), I_{h2}(x)$ as follows:

$$I_{h_1}(x) = d_{Mah}(C_N, \mu_e) \tag{3.21}$$

$$I_{h_2}(x) = d_E(\mu_N, \mu_e) \tag{3.22}$$

Where d_{Mah} and d_E are, respectively, the Mahalanobis and Euclidean metrics.

Euclidean metric computes the distance between the center μ_n of the normal class C_N and the center μ_e of evolving class C_e; on the other side Mahalanobis metric computes the distance between the normal class C_N and the evolving class center μ_e. Therefore, these two distances are calculated as follows:

$$d_{Mah}\left(C_N, \mu_e\right) = \sqrt{\left(\mu_N - \mu_e\right) \quad \Sigma_N^{-1} \quad \left(\mu_N - \mu_e\right)^T} \tag{3.23}$$

$$d_E\left(\mu_N, \mu_e\right) = \sqrt{\left(\mu_N - \mu_e\right) \times \left(\mu_N - \mu_e\right)^T} \tag{3.24}$$

The drift is detected when the Mahalanobis indicator $I_{h_1}(x)$, defined by Equation 3.21, exceeds a certain threshold th_d:

$$I_{\mathrm{h1}}(x) > th_d \Rightarrow \text{drift is detected} \tag{3.25}$$

After the drift detection, the drift is confirmed when Euclidean indicator $I_{h2}(x)$ defined by Equation 3.22, exceeds th_d as follows:

$$I_{\mathrm{h2}}(x) > th_d \Rightarrow \text{drift is confirmed} \tag{3.26}$$

The selection of th_d is motivated statically by taking three σ (standard deviations) of the data in the normal operating conditions.

In the case of pitch sensor faults, three scenarios may appear in the sensor feature space: fault impacting sensor 1 (β_{m1}), fault impacting sensor 2 (β_{m2}) or fault impacting both sensors (β_{m1} and β_{m2}) at the same time. The direction of the evolving class in the sensor feature space depends on which of these scenarios happened. Therefore, for sensor fault isolation, we use a drift direction indicator in order to monitor the direction of the evolving class. This will allow to determine which of these three scenarios happened and hence to isolate the abnormal drift source. When drift occurs, the evolving class will migrate from normal operating condition to failure. The direction indicator Dr and direction isolation DI are used to isolate the sensor which caused the drift-like fault. The idea is to consider the angle θ_1 respectively θ_2, between the vector μ_e relating the center of the evolving class and the origin of the feature space, and the vector μ_{e1} respectively μ_{e2} relating the origin with the projection of the center of the evolving class according to feature 1 respectively feature 2, of the feature space. These angles define the movement direction of the evolving class.

In order to calculate θ_1 and θ_2, the scalar products between $\overrightarrow{\mu_{e1}}$ and $\overrightarrow{\mu_e}$ and between $\overrightarrow{\mu_{e2}}$ and $\overrightarrow{\mu_e}$ are calculated as follows:

$$\overrightarrow{\mu_e}(x) \cdot \overrightarrow{\mu_{e1}}(x) = \|\mu_e(x)\| \cdot \|\mu_{e1}(x)\| \cdot \cos\theta_1 \tag{3.27}$$

$$\overrightarrow{\mu_e}(x) \cdot \overrightarrow{\mu_{e2}}(x) = \|\mu_e(x)\| \cdot \|\mu_{e2}(x)\| \cdot \cos\theta_2 \tag{3.28}$$

If the drift is detected and confirmed by the two drift indicators $I_{h1}(x)$ and $I_{h2}(x)$, then the drift isolation (to determine if sensor 1 or sensor 2 or both is the source of this drift) is achieved as follows:

$$\text{If } Dr = \theta_1 - \theta_2 > th_a \text{ and } \theta_1 > \theta_2 \Rightarrow DI = 1: \ \text{fault in sensor } 1(\beta_{m1}) \tag{3.29}$$

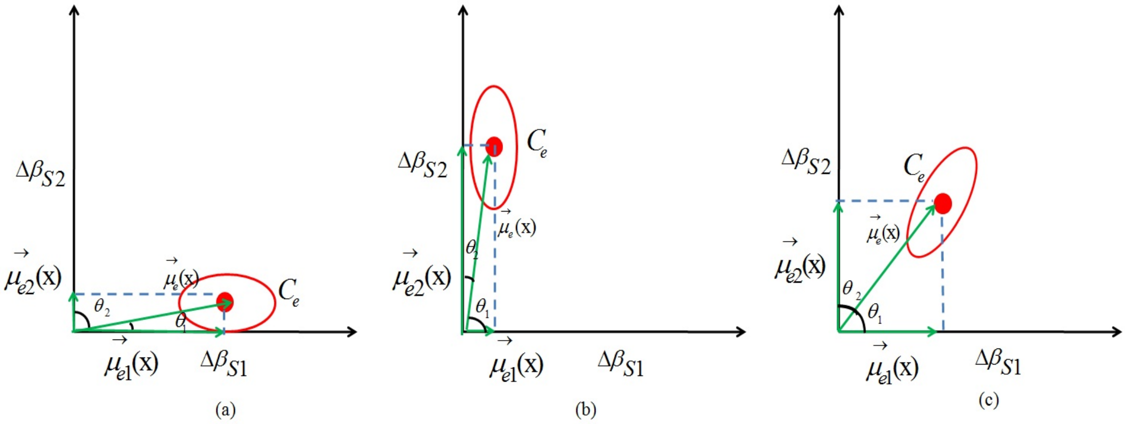

Figure 3.21: Drift direction angles in the pitch sensor feature space in the case of (a) simple drift-like fault in pitch sensor 1 (β_{m1}), (b) simple drift-like fault in pitch sensor 2 (β_{m2}), (c) multiple drift-like fault in both pitch sensors (β_{m1}) and (β_{m2}).

$$\text{If } Dr = \theta_1 - \theta_2 > th_a \text{ and } \theta_2 < \theta_1 \Rightarrow DI = 2: \text{ fault in sensor } 2\,(\beta_{m2}) \tag{3.30}$$

$$\text{If } Dr = \theta_1 - \theta_2 < th_a \Rightarrow DI = 3: \text{ fault in both sensors}\,(\beta_{m1}\text{and}\beta_{m2}) \tag{3.31}$$

where th_a is the angle threshold. th_a is defined according to the variation of patterns within the normal class C_N. Therefore, th_a is determined experimentally using the patterns belonging to C_N.

The interpretation step aims at interpreting the detected changes within the classifier parameters and structure. This interpretation is then used as a prediction about the tendency of the future development of the WT current situation. This prediction is useful to formulate a control or maintenance action.

3.7.5 Discussion on the choice of drift-like fault indicators for pitch system

The choice of a method to handle drifting data can be influenced by the availability of prediction feedback. If the true labels are immediately, or shortly, available after the prediction, then methods based on supervised drift indicators may be used. However, if data are partially labeled and the prediction feedback is delayed then, methods based on unsupervised indicators are the most appropriate. In the context of drift-like fault detection, the expert feedback is not available and the occurrence of a fault must be inferred in order to alarm human operators of supervision. The fault alarm allows them to take the suitable maintenance actions. Moreover in this thesis, we suppose that only data corresponding to normal operating conditions (normal class) are available in advance. Therefore, the drift-like fault impacting the characteristics of a normal class does not affect the decision boundaries.

Consequently for the drift-like fault diagnosis, only unsupervised drift detection indicators can be used. The unsupervised drift indicators are mainly classified into time similarity based and space similarity based indicators. The time similarity based indicators monitor the similarity over time of the data distribution [62]. Generally in the literature, the similarity in time can be quantified using hypothesis tests as Sequential Probability Ratio Test (SPRT) [52], CUSUM test [12], Page-Hinkley test [89] etc. The space-similarity based indicators monitor the change in the dispersion of data in the feature space (e.g., class centroid moving in the feature space). Hence, the localization of data within a class changes (evolves) in the feature space although their distribution remains the same.

Generally in the literature, the similarity in space can be quantified using distance measures as Euclidean distance [99], Heterogeneous Euclidean-Overlap distance [4], [93], Mahalanobis distance [78],[98], Hellinger distance [16], [61],[40], [25], Entropy measure [107], Kullback distance [100] etc. The drift like faults entail a change in the space (location) occupied by the class representing the normal operation conditions. This space change is characterized by a movement of the normal class to occupy another zone in the feature space. Therefore, drift indicators based on the distance similarity between data in the feature space are considered in this thesis to achieve the drift like fault detection in pitch system.

Therefore in this chapter, a drift space-similarity indicator based on two drift indicators; one based on the use of Euclidean distance for the drift detection and the other indicator based on Mahalanobis distance for the drift confirmation. The reason behind the use of two distance metrics (Euclidean and Mahalanobis ones) in the same time is to exploit the complementarity between them. Indeed, the Mahalanobis metric calculates the distance between the gravity center of the evolving class and all the patterns of the initial (normal) class. This will give more reactivity in case of change; while the Euclidean metric confirms this change by calculating the distance between the gravity center of the initial (normal) class and the gravity center of evolving class.

3.8 Experimentation and obtained results

The failures of pitch actuators and sensors are caused by a continuous degradation of its performance over time. This degradation can be seen as a continuous drift of the normal operating conditions characteristics (normal class) of the pitch actuator and sensor. Detecting and following this drift can help to predict the occurrence of the pitch actuator and sensor failures. The two monitoring indicators defined by Equation 3.21 and Equation 3.22 are used to detect and to confirm this drift for the nine scenarios of the pitch actuator fault and the twenty-seven scenarios of simple and multiple drift-like fault in pitch sensors are defined in section 2.

3.8.1 Actuator drift-like fault

Figure 3.22 and Figure 3.23, represent, respectively, first and second residual used in the pitch actuator feature space in presence of an abnormal drift in pitch actuator performance. We can see that both residual $\Delta\beta_{A1}$ and $\Delta\beta_{A2}$ are impacted by the occurrence of the abnormal drift in pitch actuator.

Table 3.5 show the values of the drift indicators $I_{h_1}(x)$ and $I_{h_2}(x)$ for the nine defined drift-like fault scenarios. These values represent the required time (starting from the drift beginning) to detect and confirm the drift occurrence. Thus, they can be used as an evaluation criterion to measure the time delay to detect a drift before its end.

Fault N	Drift speed	I_{h1}	I_{h2}	Period
F_{1h}	30s(High)	7s	11.10s	3400s-3630s
F_{1m}	60s(Medium)	14.40s	28.70s	3400s-3660s
F_{1s}	90s (Slow)	28.70s	31.40s	3400s-3690s
F_{2h}	30s	10.70s	11.50s	3500s-3730s
F_{2m}	60s	18.50s	21.40s	3500s-3760s
F_{2s}	90s	21.30s	31.60s	3500s-3790s
F_{3h}	30s	9.90s	10.70s	3600s-3830s
F_{3m}	60s	13.00s	20.30s	3600s-3860s
F_{3s}	90s	22.70s	29.30s	3600s-3890s

Table 3.5: Results of drift-like fault detection and confirmation in pitch actuator for the nine drift scenarios.

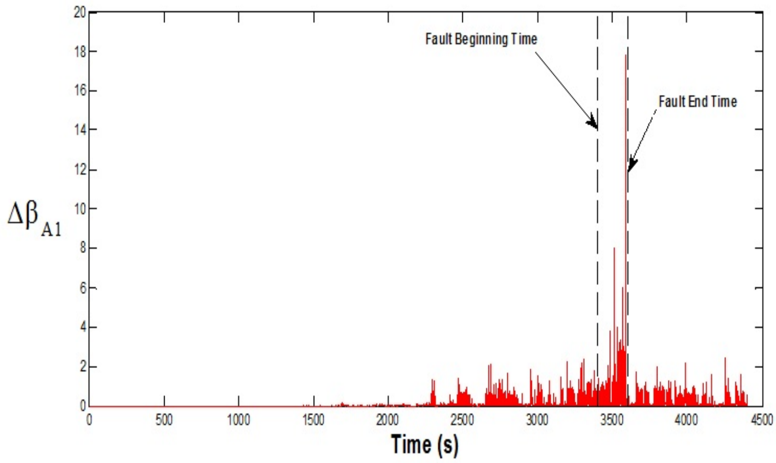

Figure 3.22: First residual used in the pitch actuator feature space.

Figures 3.24 and 3.25 show the obtained results using the two drift detection indicators $I_{h_1}(x)$ and $I_{h_2}(x)$, for pitch actuator. The degradation is observed when the pitch actuator operate in control mode 2. The drift like fault in pitch actuator

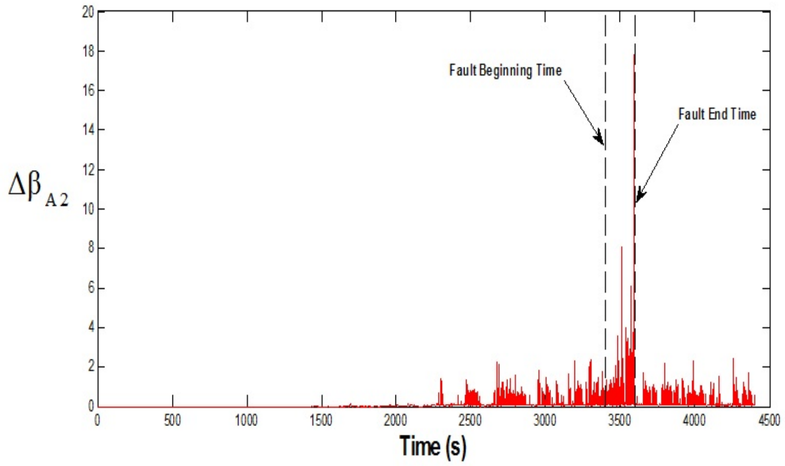

Figure 3.23: Second residual used in the pitch actuator feature space.

is successfully detected by both indicator $I_{h_1}(x), I_{h_2}(x)$ for all drift speeds (see Figure 3.24 and Figure 3.25).

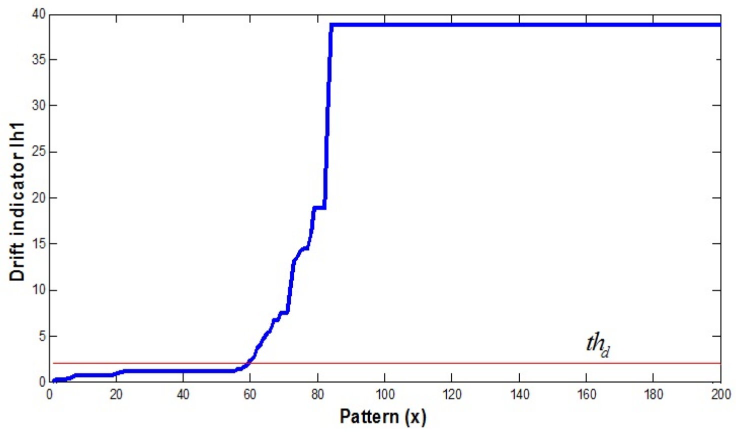

Figure 3.24: Drift indicator $I_{h_1}(x)$ based on Mahalanobis distance of the third pitch actuator.

The drift-like fault in pitch actuator is detected in early stage before the end of this drift (arriving to the failure mode due to drift fault in pitch actuator). As an example, in the case of a drift of slow speed (F3s) (see Table 3.5), the pitch actuator reaches the failure mode resulting from a drift-like fault in ω_n, ζ after 90 seconds of the beginning of the drift (degradation in ω_n, ζ). In the proposed approach, this drift is detected 21.30 seconds and confirmed 31.60 seconds after its beginning. Therefore, the drift like fault in pitch actuator is confirmed approximately 60 seconds before its end. This enables to achieve an early fault diagnosis and therefore

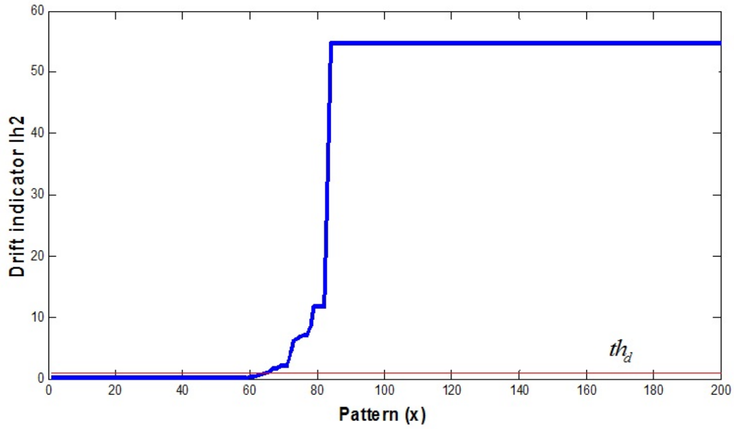

Figure 3.25: Drift indicator $I_{h_2}(x)$ based on Euclidean distance of the third pitch actuator.

helps the human operators of supervision to take efficiently the right actions.

3.8.2 Simple drift-like fault in sensor β_{m1}

Figure 3.26 and Figure 3.27 represent, respectively, first and second residuals used in the pitch sensor feature space in presence of an abnormal drift in pitch sensor 1, β_{m1}. We can see in the case of an abnormal drift in pitch sensor 1, β_{m1}, that only residual $\Delta\beta_{s1}$ is impacted, while residual $\Delta\beta_{s2}$ has similar behavior as the one without abnormal drift in β_{m1}.

Table 3.6 show the values of the drift indicators $I_{h_1}(x)$ and $I_{h_2}(x)$ for the nine defined drift-like fault scenarios. These values represent the required time (starting from the drift beginning) to detect and confirm the drift occurrence. Thus, they can be used as an evaluation criterion to measure the time delay to detect a drift before its end.

Fault N	Drift speed	I_{h1}	I_{h2}	Period
F_{4h}	30s(High)	5.25s	11.00s	2500s-2730s
F_{4m}	60s(Medium)	8.60s	18.70s	2500s-2760s
F_{4s}	90s (Slow)	14s	26.30s	2500s-2790s
F_{5h}	30s	6.90s	13.30s	2600s-2830s
F_{5m}	60s	11.50s	20.20s	2600s-2860s
F_{5s}	90s	14.25s	27.10s	2600s-2890s
F_{6h}	30s	6.05s	11.90s	2700s-2930s
F_{6m}	60s	12.60s	23.50s	2700s-2960s
F_{6s}	90s	15.10s	29.40s	2700s-2990s

Table 3.6: Results of simple drift-like fault detection and confirmation in pitch sensor 1 (β_{m1}), for the nine drift scenarios.

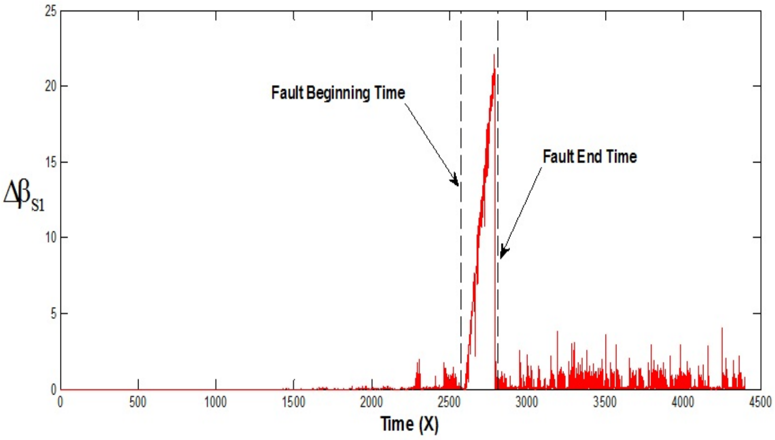

Figure 3.26: First residual used in the pitch sensor feature space in the case of the simple drift-like fault in pitch sensor 1 (β_{m1}).

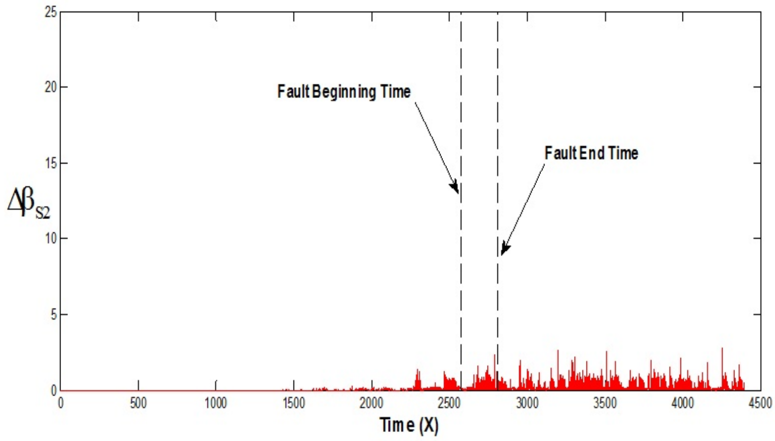

Figure 3.27: Second residual used in the pitch sensor feature space in the case of the simple drift-like fault in pitch sensor 1 (β_{m1}).

Figures 3.28 and 3.29 show the obtained results using the two drift detection indicators $I_{h_1}(x)$ and $I_{h_2}(x)$, for simple drift-like fault in pitch sensor β_{m1}. The degradation is observed when the pitch actuator operate in control mode 2, the drift like fault in pitch sensor is successfully detected by both indicator $I_{h_1}(x)$ and $I_{h_2}(x)$, for all drift speeds (see Figure 3.28 and Figure 3.29).

The drift-like fault in pitch sensor 1 (β_{m1}), is detected in early stage before the end of this drift (arriving to the failure mode due to drift fault in pitch sensor). As an example, in the case of a drift of slow speed (F6s) (see Table 3.6), the pitch

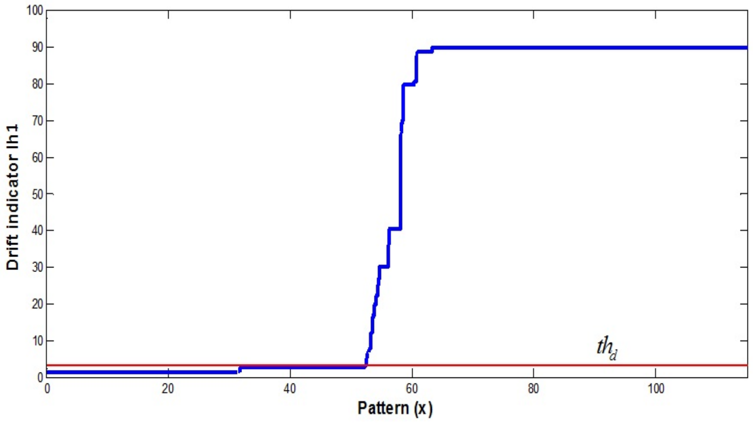

Figure 3.28: Drift indicator $I_{h_1}(x)$ based on Mahalanobis distance of the simple drift-like fault in pitch sensor 1 (β_{m1}).

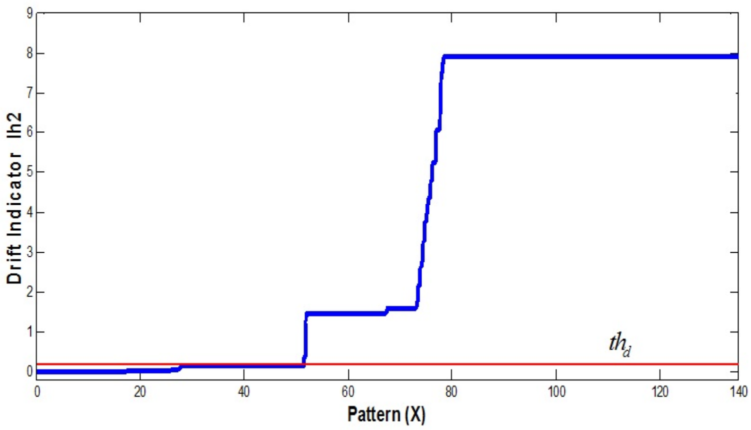

Figure 3.29: Drift indicator $I_{h_2}(x)$ based on Euclidean distance of the simple drift-like fault in pitch sensor 1 (β_{m1}).

sensor reaches the failure mode resulting from a drift-like fault in λ_1 (degradation in λ_1) after 90 seconds of the beginning of the drift. In the proposed approach, this drift is detected 15.10 seconds and confirmed 29.40 seconds after its beginning. Therefore, the drift like fault in pitch sensor is confirmed 60 seconds before its end. This enables to achieve an early fault diagnosis and therefore helps the human operators of supervision to take efficiently the right actions.

Figure 3.30 and Figure 3.31 represent, respectively, evolving class angle and the direction indicator of the pitch sensor fault. These figures show the obtained results in presence of simple drift-like fault in pitch sensor 1, based on Figure 3.30 and Figure 3.31 the sensor 1 (β_{m1}), fault is successfully isolated by the direction indicator. Indeed, the direction angle shows that the evolving class exceeds the angle threshold (see Figure 3.21.a). Based on Equation 3.29, the drift-like fault in sensor 1 (β_{m1}), is isolated (see Figure 3.37).

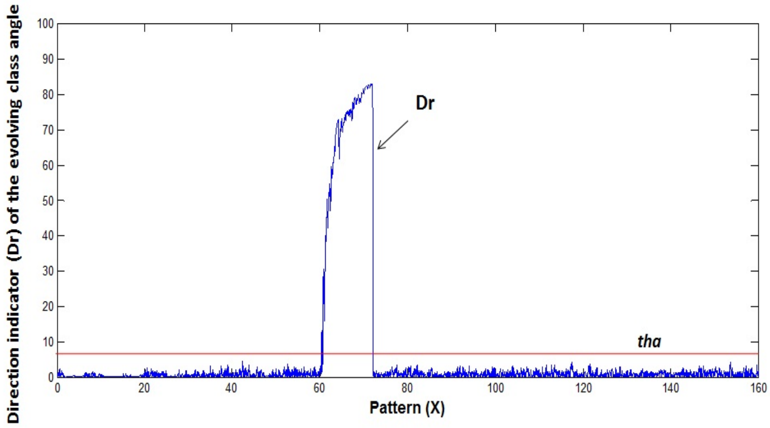

Figure 3.30: Direction indicator Dr of the evolving class angle of the simple drift-like fault in pitch sensor 1 (β_{m1}).

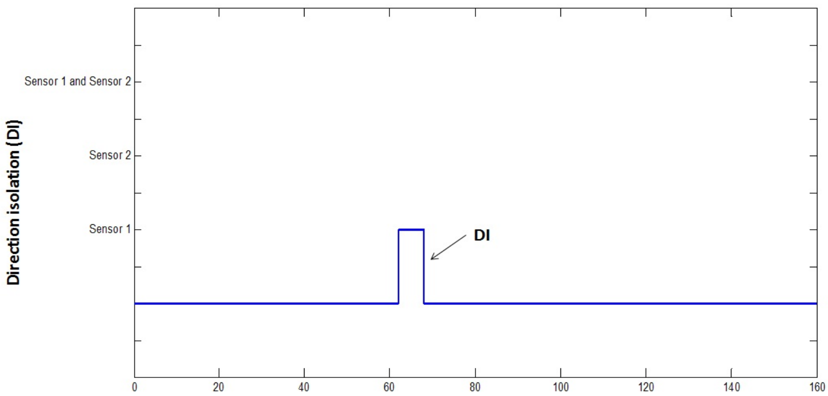

Figure 3.31: Direction isolation DI of the simple drift-like fault in pitch sensor 1 (β_{m1}).

3.8.3 Simple drift-like fault in sensor β_{m2}

Figure 3.32 and Figure 3.33 represent, respectively, first and second residuals used in the pitch sensor feature space in presence of an abnormal drift in pitch sensor sensor 2, β_{m2}. We can see in the case of an abnormal drift in pitch sensor 2, β_{m2}, that only residual $\Delta\beta_{s2}$ is impacted, while residual $\Delta\beta_{s1}$ has similar behavior as the one without abnormal drift in β_{m2}.

Table 3.7 show the values of the drift indicators $I_{h_1}(x)$ and $I_{h_2}(x)$ for the nine defined drift-like fault scenarios. These values represent the required time (starting from the drift beginning) to detect and confirm the drift occurrence. Thus, they can be used as an evaluation criterion to measure the time delay to detect a drift before its end.

Fault N	Drift speed	I_{h1}	I_{h2}	Period
F_{7h}	30s(High)	6.07s	12.15s	2800s-3030s
F_{7m}	60s(Medium)	8.90s	19.05s	2800s-3060s
F_{7s}	90s (Slow)	14.20s	27s	2800s-3090s
F_{8h}	30s	5.70s	11.80s	2900s-3130s
F_{8m}	60s	8.25s	18.40s	2900s-3160s
F_{8s}	90s	13.70s	26.18s	2900s-3190s
F_{9h}	30s	6.90s	12.70s	3000s-3230s
F_{9m}	60s	9s	20.30s	3000s-3260s
F_{9s}	90s	14.90s	28.10s	3000s-3290s

Table 3.7: Results of simple drift-like fault detection and confirmation in pitch sensor 2(β_{m2}), for the nine drift scenarios.

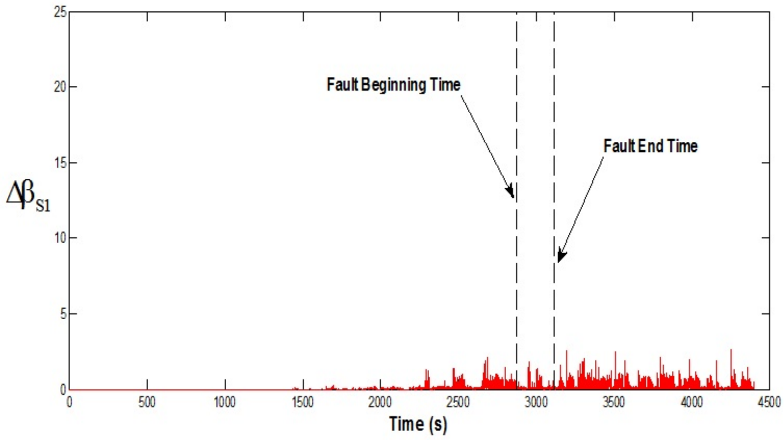

Figure 3.32: First residual used in the pitch sensor feature space in the case of the simple drift-like fault in pitch sensor 2 (β_{m2}).

Figures 3.34 and 3.35 show the obtained results using the two drift detection indicators $I_{h_1}(x)$ and $I_{h_2}(x)$, for simple drift-like fault in pitch sensor 2 (β_{m2}). The degradation is observed when the pitch actuator operate in control mode 2, the drift-like fault in pitch sensor 2 is successfully detected by both indicators $I_{h_1}(x)$ and $I_{h_2}(x)$ for all drift speeds (see Figure 3.34 and Figure 3.35).

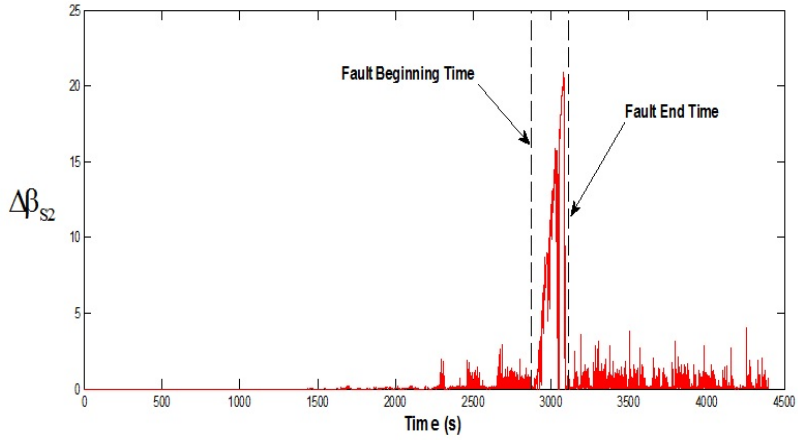

Figure 3.33: Second residual used in the pitch sensor feature space in the case of the simple drift-like fault in pitch sensor 2 (β_{m2}).

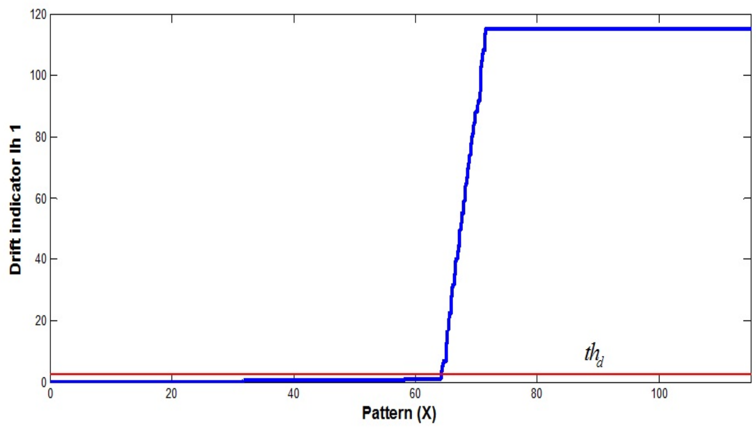

Figure 3.34: Drift indicator $I_{h_1}(x)$ based on Mahalanobis distance of the simple drift-like fault in pitch sensor 2 (β_{m2}).

The drift-like fault in pitch sensor 2 (β_{m2}), is detected in early stage before the end of this drift (arriving to the failure mode due to drift fault in pitch sensor). As an example, in the case of a drift of slow speed (F9s) (see Table 3.7), the pitch sensor reaches the failure mode resulting from a drift-like fault in λ_2 (degradation in λ_2) after 90 seconds of the beginning of the drift. In the proposed approach, this drift is detected 14.90 seconds and confirmed 28.10 seconds after its beginning. Therefore, the drift like fault in pitch sensor is confirmed 60 seconds before its end. This enables to achieve an early fault diagnosis and therefore helps the human operators of supervision to take efficiently the right actions.

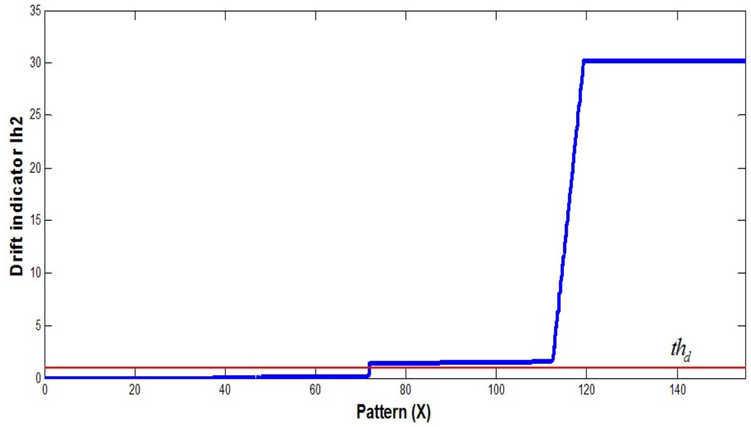

Figure 3.35: Drift indicator $I_{h_2}(x)$ based on Euclidean distance of the simple drift-like fault in pitch sensor 2 (β_{m2}).

For the drift isolation, Figure 3.36 and Figure 3.37 are used. They represent, respectively, evolving class angle and the direction indicator of the pitch sensor fault. These figures show the obtained results in presence of simple drift-like fault in pitch sensor 2, based on Figure 3.36 and Figure 3.37 the sensor 2 (β_{m2}), fault is successfully isolated by the direction indicator. Indeed, the direction angle shows that the evolving class exceeds the angle threshold (see Figure 3.21.b). Based on Equation 3.30, the drift-like fault in sensor 2 (β_{m2}), is isolated (see Figure 3.37).

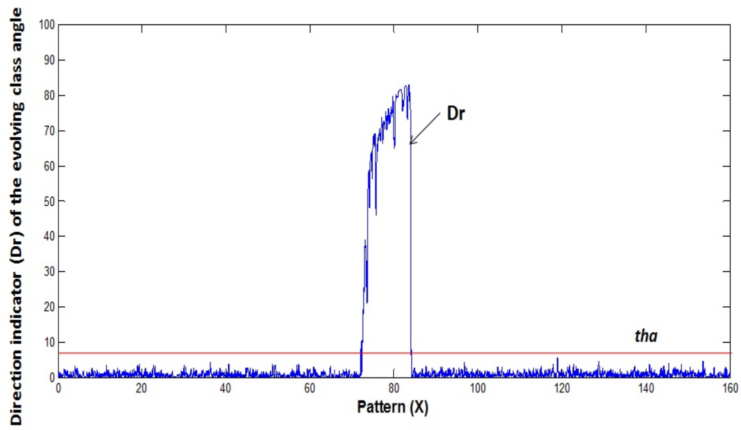

Figure 3.36: Direction indicator Dr of the evolving class angle of the simple drift-like fault in pitch sensor 2 (β_{m2}).

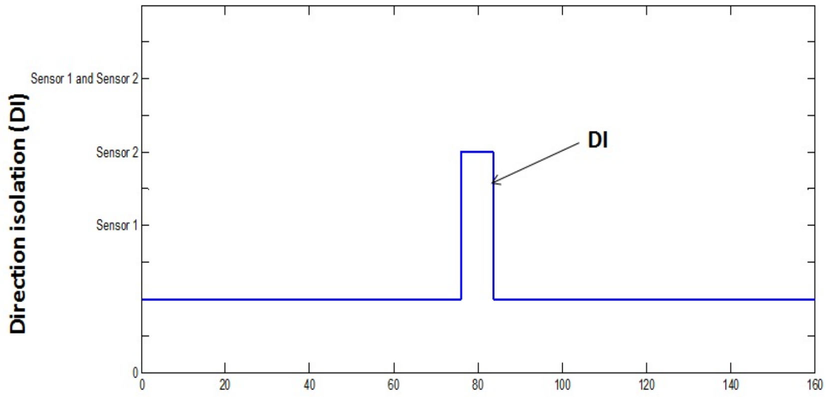

Figure 3.37: Direction isolation DI of the simple drift-like fault in pitch sensor 2 (β_{m2}).

3.8.4 Multiple drift-like fault in sensors β_{m1} and β_{m2}

Figure 3.38 and Figure 3.39 represent, respectively, first and second residuals used in the pitch sensor feature space in presence of an abnormal drift in both pitch sensor β_{m1} and β_{m2} at the same time. We can see that both residual $\Delta\beta_{s1}$ and $\Delta\beta_{s2}$ are impacted by the occurrence of the abnormal drift in β_{m1} and β_{m2}.

Table 3.8 show the values of the drift indicators $I_{h_1}(x)$ and $I_{h_2}(x)$ for the nine defined drift-like fault scenarios. These values represent the required time (starting from the drift beginning) to detect and confirm the drift occurrence. Thus, they can be used as an evaluation criterion to measure the time delay to detect a drift before its end.

Fault N	Drift speed	I_{h1}	I_{h2}	Period
F_{10h}	30s(High)	5.04s	10.9s	3100s-3330s
F_{10m}	60s(Medium)	9s	19.04s	3100s-3360s
F_{10s}	90s(Slow)	13.68s	26.23s	3100s-3390s
F_{11h}	30s	6.55s	15.50s	3200s-3430s
F_{11m}	60s	10.05s	19.30s	3200s-3460s
F_{11s}	90s	13.80s	27.50s	3200s-3490s
F_{12h}	30s	7.10s	16.10s	3300s-3530s
F_{12m}	60s	9.55s	22.80s	3300s-3560s
F_{12s}	90s	14.70s	28.25s	3300s-3590s

Table 3.8: Results of multiple drift-like fault detection and confirmation in pitch sensor 1 (β_{m1}), and pitch sensor 2 (β_{m2}), for the nine drift scenarios.

Figures 3.40 and 3.41 show the obtained results using the two drift detection indicators $I_{h_1}(x)$ and $I_{h_2}(x)$, for multiple pitch sensor fault. The degradation is observed when the pitch actuator operate in control mode 2. The drift like fault

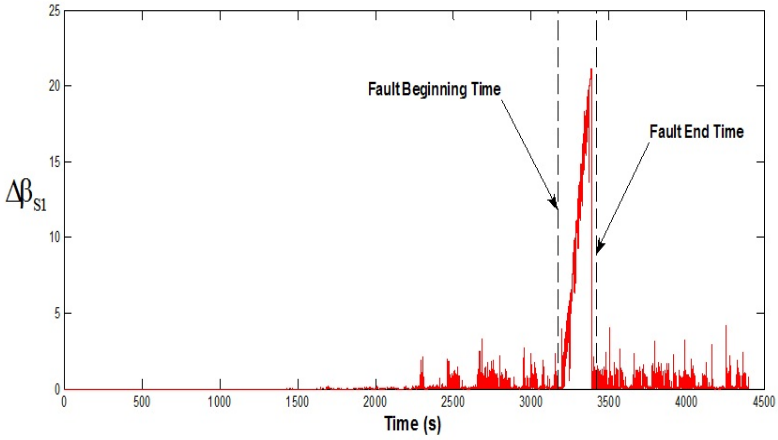

Figure 3.38: First residual used in the pitch sensor feature space in the case of the multiple drift-like fault in pitch sensor 1 (β_{m1}), and sensor 2 (β_{m2}).

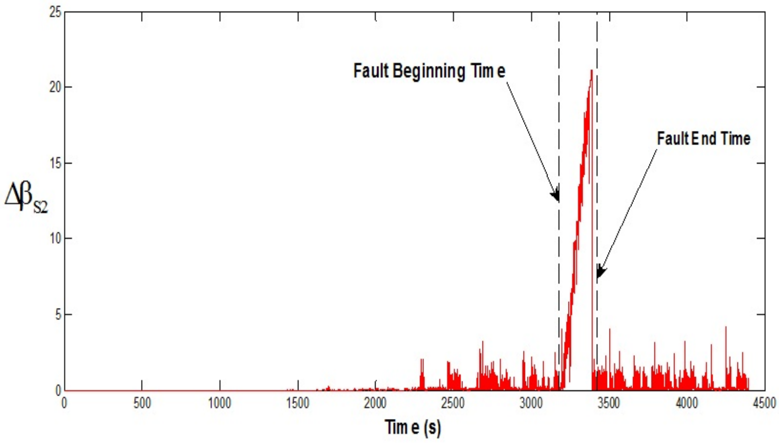

Figure 3.39: Second residual used in the pitch sensor feature space in the case of the multiple drift-like fault in pitch sensor 1 (β_{m1}), and sensor 2 (β_{m2}).

in pitch sensor is successfully detected by both indicator $I_{h_1}(x)$ and $I_{h_2}(x)$ for all drift speeds in both sensors (see Figure 3.40 and Figure 3.41).

The multiple drift-like faults in pitch sensors are detected in early stage before the end of these drifts (arriving to the failure mode due to drift fault in both pitch sensors). As an example, in the case of a drift of slow speed (F12s) (see Table 3.8), the pitch sensors reache the failure mode resulting from a drift-like fault in λ_1 and λ_2 (degradation in λ_1 and λ_2) after 90 seconds of the beginning of the drift. In the proposed approach, this drift is detected 14.70 seconds and confirmed 28.25

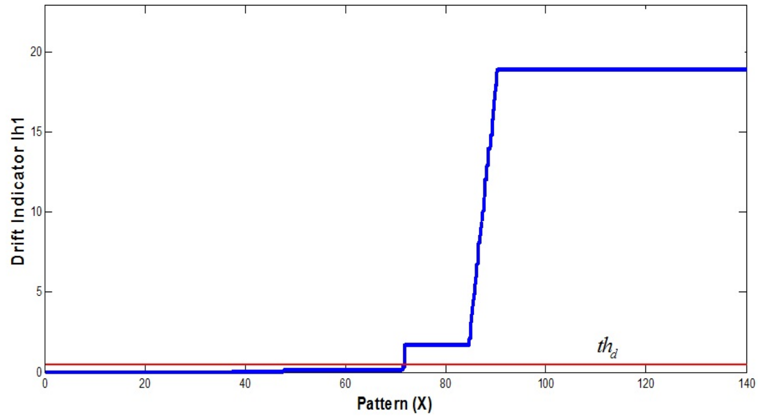

Figure 3.40: Drift indicator $I_{h_1}(x)$ based on Mahalanobis distance of the multiple drift-like fault in both pitch sensor 1 (β_{m1}), and sensor 2 (β_{m2}).

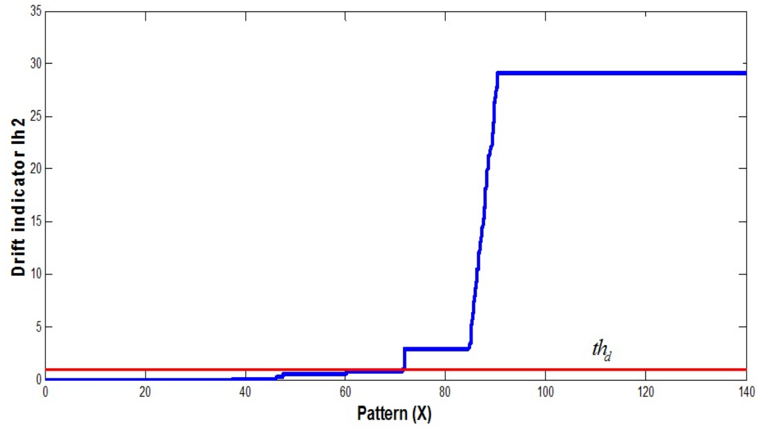

Figure 3.41: Drift indicator $I_{h_2}(x)$ based on Euclidean distance of the multiple drift-like fault in both pitch sensor 1 (β_{m1}), and sensor 2 (β_{m2}).

seconds after its beginning. Therefore, the multiple drift-like fault in pitch actuator is confirmed 60 seconds before its end. This enables to achieve an early fault diagnosis and therefore helps the human operators of supervision to take efficiently the right actions.

For the drift isolation, Figure 3.42 and Figure 3.43 are used. They represent, respectively, evolving class angle and the direction indicator of the pitch sensor fault. These figures show the obtained results in presence of a multiple drift-like fault in both pitch sensors β_{m1} and β_{m2}, as we can see in Figure 3.42 and Figure 3.43 the fault is successfully isolated by the direction indicator. Indeed, the direction angle shows that the evolving class evolve within the axe of the normal class (see

Figure 3.21.c). Based on Equation 3.30, the multiple drift-like isolation in both pitch sensors is isolated (see Figure 3.43).

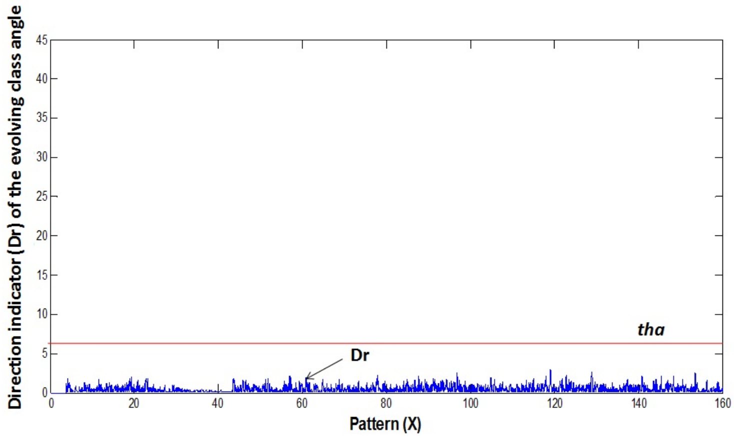

Figure 3.42: Direction indicator Dr of the evolving class angle of the multiple drift-like fault in both pitch sensor 1 (β_{m1}), and pitch sensor 2 (β_{m2}).

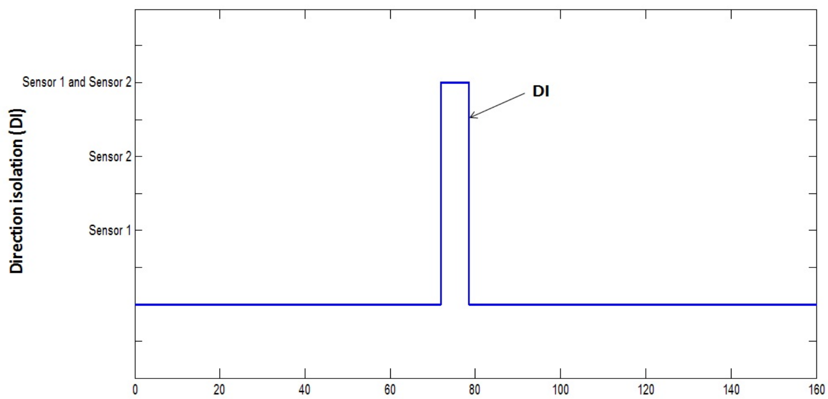

Figure 3.43: Direction isolation DI of the multiple drift-like fault in both pitch sensor 1 (β_{m1}), and sensor 2 (β_{m2}).

3.9 Summary

In this chapter, an approach of condition monitoring and drift like fault detection was developed. It is based on the use of a classifier able to achieve a reliable drift monitoring and early diagnosis of actuator and sensor parametric faults . This approach considers the system switching between several control modes. This approach based on the monitoring of the drift of the characteristics of classes representing the normal operating conditions of pitch system in each control mode. These characteristics are described by the mean and variance covariance matrix of these classes. They are monitored using two indicators in order to monitor and follow the drift. Both are defined based on the computation of the distance between the class representing normal operating conditions and the evolving class. The first indicator is based on the Mahalanobis distance and is used to detect the drift; while the second indicator is based on Euclidean distance and is used to confirm the drift. The drift indicators have detected successfully all drift scenarios of three speeds in early stage before the end of this drift for the case of simple and multiple drift-like faults in pitch system.

In chapter 4, the proposed hybrid dynamic classifier in Chapter 3 will be developed in order to achieve the drift like fault diagnosis of the power converter which is another critical WT component. The converter belongs to another class of hybrid dynamic system which is discretely controlled continuous; while the pitch system is represented by a discretely externally triggered jumping system. Therefore, the feature space and the drift indicators will be defined and used differently for the converter than the ones in the pitch system.

Chapter 4

Hybrid dynamic classifier for simple and multiple drift-like faults diagnosis in wind turbine power converter

Contents

4.1 Introduction

The behaviors of Hybrid dynamic systems (HDS) are described as a combination of continuous and discrete dynamics. There are several classes of HDS [23]. Discretely Controlled Continuous Systems (DCCS) is an important class of HDS in which the system switches between several discrete modes in response to discrete control events issued by a discrete controller. Fault diagnosis of these systems requires taking into account both their discrete and continuous dynamics. The faults can occur as abnormal change in the values of parameters describing the system dynamics in a discrete mode and are termed as parametric faults. When this change in parameters values is gradual, the parametric faults are called incipient. In this case, they entail a drift in the system operating conditions until the failure takes over completely. Detecting this drift in early stage allows reducing the power production losses as well as the WT unavailability and maintenance costs. However, this drift is observed when the system is in the discrete modes where the dynamics (operating conditions) described by the affected parameters are active. Therefore, the fault diagnosis system must take into account the discrete modes in which the system is.

This chapter proposes a data-mining based approach in order to build a classifier able to achieve a reliable drift monitoring and early diagnosis of faults that can affect WT converters. This approach considers the converter as a DCCS. Therefore, it takes into account the converter continuous dynamics in each discrete mode. The continuous dynamics are described in a feature space sensitive to normal operating conditions in the corresponding discrete mode. Therefore, the feature space is dynamic in the sense that the classifier selects the discriminant features according to each discrete operating mode.

The normal operating conditions of the converter are represented by a set of restricted zones in the feature space, called classes. The faulty classes, representing the failure operating conditions of converter system, are considered to be a priori unknown. Converter degradation is considered as a continuous drift in the characteristics of the normal classes over time. This drift is characterized by a change in data characteristics in the normal classes. The proposed approach monitors this change by using a drift indicator for each attribute of the feature space, in order to detect a drift and isolate its origin as soon as possible. When the drift is detected by one indicator, a warning is emitted in order to inform human operators of the necessity to react by taking the adequate correction actions. To achieve that, the proposed approach builds a hybrid classifier able to change its decision function as well as its feature space according to the system internal state (discrete mode) and to abnormal changes (e.g., faults) in its environments. This allows to keep the useful patterns representative of the drift and therefore to detect it in its early stage. Consequently, detecting and following this drift can help to predict the occurrence of converter failure.

The main goal of our approach is to take advantage of the hybrid dynamic aspect of the WT components in order to improve the diagnosis performance (decreasing the fault diagnosis delay time). The hybrid dynamic aspect can exist in different ways in the system. For this reason, the approach must be developed and adapted for each wind turbine WT component according to the HDS class to which it belongs. The work realized in chapter 3 presents an approach to achieve the drift like fault diagnosis of pitch system (composed of two redundant sensors and an actuator for each of the three vertical blades). The major differences between the approach proposed in chapter 3 and the one proposed in this chapter can be summarized as follows:

- The way how the WT component changes its continuous dynamic: each approach is applied to a different class of hybrid dynamic systems. In chapter 3, the pitch system is represented by a discretely externally triggered jumping system; while in this chapter, the converter is represented by a discretely controlled continuous system.

- The feature space definition: in chapter 3, one feature space with the same features is used for all the different discrete modes; while in this chapter, the feature space is dynamic in the sense that the classifier selects the discriminant features according to each discrete operating mode.

- The drift indicator definition and using: in chapter 3, two drift detection indicators: one based on the use of Euclidean distance for the drift detection and the other based on Mahalanobis distance for the drift confirmation. In addition, a third indicator based on the determination of the evolution direction of the drift in the feature space is used to isolate the element (pitch sensor) generating this drift. While in this chapter,one drift indicator based on the Euclidean distance for each feature in a certain discrete mode is used for drift detection and isolation. Each feature is sensitive to a drift generated by one element (e.g., one capacitor in the converter).

Chapter 4 is organized as follows. Firstly, the WT system, in particular the converter, and the generated fault scenarios are described. Then, the proposed approach to achieve drift like fault detection of converters is detailed. Then, the results based on the use of a simulator of a WT converter are presented. Finally, the conclusion and the future work are discussed.

4.2 Challenges and motivations of fault diagnosis in wind turbine converters

Fault diagnosis of WTs is a challenging task because of the high variability of the wind speed and the confusion between faults and noises as well as outliers. The role of the converter is to adjust the generator torque depending on the reference torque provided by the controller. The latter decides the reference torque according to the wind speed in order to allow an optimum energy production. Therefore faults of the converter result in costly turbine down-time.However, the fault diagnosis of converter system is particularly a challenging task because of i) the unknown aerodynamic torque related to the wind speed ii) no sensor redundancy, as for some other WTs components, since one sensor is used to measure the converter output (torque) and iii) the switching between several discrete modes which can hide the converter faults consequences.

The operating conditions of WTs, or one of its components, change from normal to faulty either abruptly or gradually. In the case of gradual change, WT begins to malfunction (degraded behavior) until the failure takes over completely. The detection of this drift from normal to faulty operating conditions in its early stage can help providing a time to achieve appropriate corrective actions leading to decrease the maintenance costs and to increase the availability time. Therefore, developing a drift monitoring and diagnosis module for converter system is of particular interest for WTs industry due to their operational & maintenance costs as well as their essential role in optimizing the energy production.

The growing interest in wind energy production has led to the design of sophisticated WTs. Like every other complex and heterogeneous system, WTs are faced to the occurrence of faults that can impact their performance as well as their security. Therefore, it is crucial to design a reliable automated diagnostic system in order to achieve fault detection and isolation in early stage. Some works in literature on fault diagnosis of WTs presented some statistics about most costly subsystems to repair and their failures frequency [50],[3]. In [50], the indicators used to determine the ranks of the subsystems are failure rate and downtime per turbine per year. Figure 4.1 is a graphical representation of the average failure rate and down time.

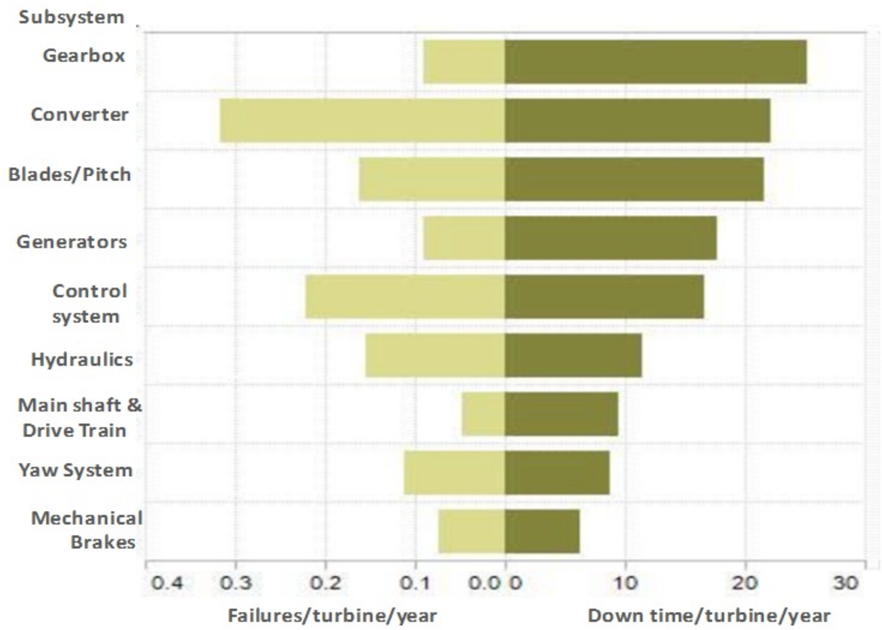

Figure 4.1: Literature review summary of failure rate and downtime per turbine per year

Based on these statistics, we can find that converters fail most frequently. These failures result in the second rank in the downtime per turbine, per year after gearbox. Therefore, diagnosing faults in converters is essential in order to improve the WTs availability and to reduce their maintenance costs.

Few approaches have been proposed to achieve early fault diagnosis of WT converters. This is due to the fact that modeling converter degradation in strong nonlinear and complex non-stationary environments is very hard task. Examples of these methods, we can cite self-organizing feature map neural network [114], fuzzy logic[51], Fuzzy system consisting of a set of piecewise affine TakagiSugeno models [90]. These methods require a priori knowledge (data) about all faulty behaviors and do not integrate a mechanism to detect a drift by analyzing the characteristics of incoming data in order to update the model parameters and structure in response to this drift. Therefore, they do not achieve a reliable and early diagnosis. Consequently, the diagnosis performance (diagnosis delay) is decreased significantly for faults occurring in WT critical subsystems as converters.

4.3 Converters within wind turbines

The wind turbine model used in this chapter is the same used in chapter 3, composed of five parts: the blades, the drive train, the generator, the converter, and the controller (see Figure 2.12 and Figure 4.2).

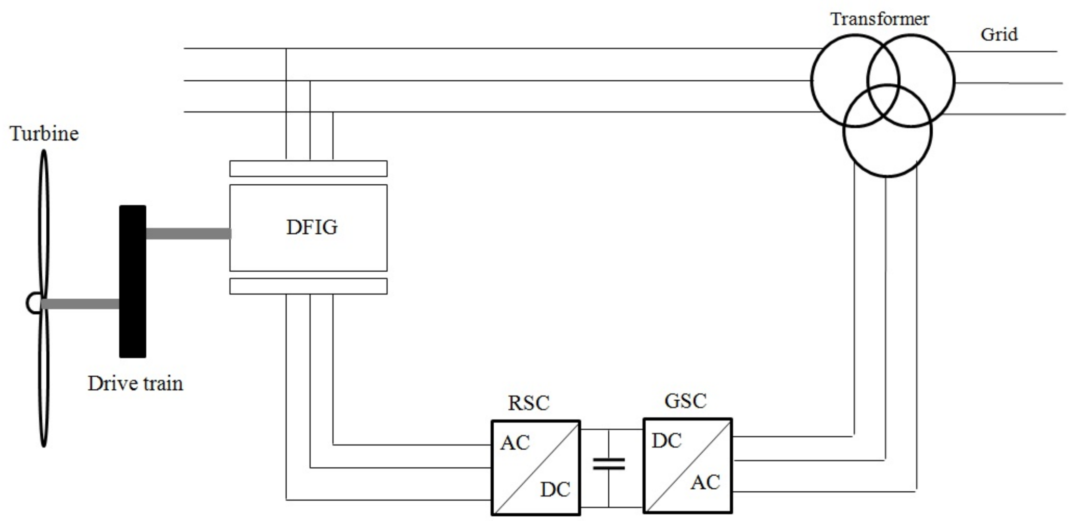

Figure 4.2: Converter architecture in the wind turbine energy system

The controller operates in four zones (see Figure 3.2). Zone 1 is the start-up of the turbine, zone 2 is power optimization, zone 3 is constant power production and zone 4 is no power production due to a too high wind speed.The focus of converter control of WT is in zones 2 and 3. Two control strategies are applied to optimize the energy production and keep it constant at its optimal value: the converter torque control τ_c in zone 2 and the blades angle control in zone 3. In zone 2, the WT is controlled so that it produces as much energy as possible. To do so, the blades angle is maintained equal to 0 and the tip speed ratio is kept constant at its optimal value. The latter is regulated by the rotating speed control by tuning the converter torque. Once the optimal power production is achieved, the blades angle control maintains the converter torque constant and adjusts the rotating speed by controlling the blades angle. The latter modifies the transfer of the aerodynamic power of the wind on the blades.

There are several control strategies [27], [1] used in the literature in order to achieve a maximum of power production and to ensure the WT safety. Their design depends on the WT structure, type and size. In this chapter, variable speed WTs based on doubly fed induction generator (DFIG) are considered [101]. This chapter focuses only on the early fault diagnosis of WTs based on DFIG.

The doubly fed induction generator (DFIG) structure consists of two converters, one is in the grid side and called the grid side converter (GSC) and the other one is in the DFIG side and called the rotor side converter (RSC). The DFIG is implemented and supplied by the grid power through the stator while the rotor is connected to the grid through two converters forming a double conversion: alternative current (AC) to direct current (DC) in the grid side (GSC) and DC to AC in the rotor side (RSC) (see Figure 4.2).

4.4 Multicellular converter description

In this chapter, we study the fault diagnosis of multicellular converter system(MCCS) implemented in RSC. It is used in DC-AC conversion to control the currents of the three phases of DFIG with maximum power point tracking (MPPT). The multicellular converters consist of serial cells (see Figure 4.3). Each cell contains tow switches with complementary values. If one is closed the other is open and vice versa. These switches are controllable by control signal S_j (see Figure 4.3). S_j is equal to 1 when the upper switch of the cell is conducting and 0 when the lower complementary switch of the cell is conducting [22].

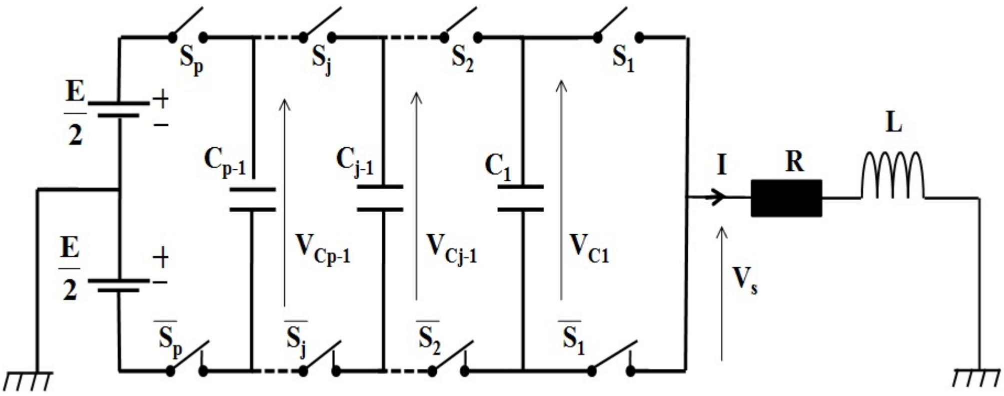

Figure 4.3: Multicellular converter system

Let p be the number of discrete cells (switches) in the MCCS, then $n = 2^p$ is the number of its normal discrete states (modes). The output voltage can take $p + 1$ levels and $p - 1$ reference voltage of the floating capacitors as follows:

$$V_{C_j,ref} = j\frac{E}{p}, j = 1, ..., p - 1 \tag{4.1}$$

The MCCS dynamics evolution is written as follows [22]:

$$\begin{cases} \dot{I} = -\frac{R}{L}I + \frac{E}{L}S_p - \sum\limits_{j}^{p-1} \frac{V_{C_j}}{L}\left(S_{j+1} - S_j\right) - \frac{E}{2L} \\ \dot{V_{C_j}} = \frac{I}{C_j}\left(S_{j+1} - S_j\right), j = 1, ..., p - 1 \end{cases} \tag{4.2}$$

where I represents the current flowing from source E towards load and V_{C_j} is the reference voltage of the floating capacitors.

In this application, the multicellular converters are used to control the currents of the DFIG rotor. The RSC consists of three identical three cell converters implemented in parallel and power supplied by the DC-link voltage E which is controlled by the converter in GSC (see Figure 4.4).

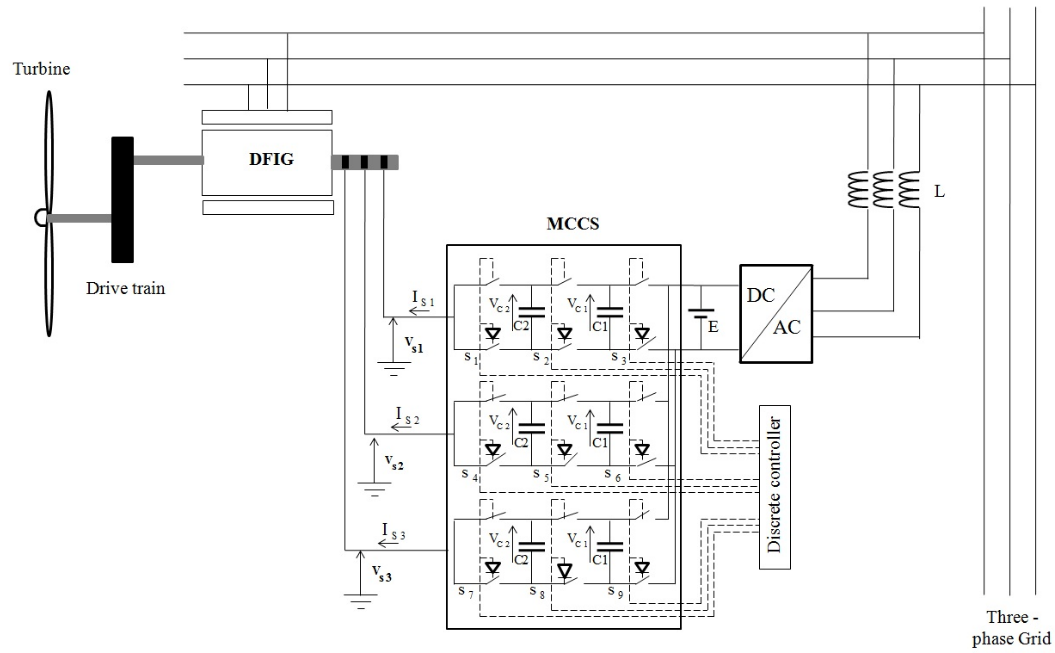

Figure 4.4: Architecture of the block DFIG-MCCS

4.5 Multicellular converter modeling

In this chapter, a three cell converter is used in DC-AC configuration. Therefore, $2^3 = 8$ discrete modes, $(3 + 1)$ levels of reference output voltage $V_{S,ref}$ (see Figure 4.5) , and $(3\ \ 1 = 2)$ reference voltages of the floating capacitors $V_{C_1,ref} = \dfrac{E}{3}$ and

Discrete Mode q_n	h_{q1}	h_{q2}	h_{q3}	h_{q4}	h_{q5}	h_{q6}	h_{q7}	h_{q8}
$h^1_{q^1_i}$	0	1	0	1	0	1	0	1
$h^2_{q^2_i}$	0	0	1	1	0	0	1	1
$h^3_{q^3_i}$	0	0	0	0	1	1	1	1
$V_{S,ref}$	$\frac{-E}{2}$	$\frac{-E}{6}$	$\frac{-E}{6}$	$\frac{E}{6}$	$\frac{-E}{6}$	$\frac{E}{6}$	$\frac{E}{6}$	$\frac{E}{2}$

Table 4.1: Different discrete modes associated with the discrete states of the cells and the reference output voltage $V_{S,ref}$ for the three cell converter.

$V_{C_2,ref} = \dfrac{2E}{3}$ are defined see Equation 4.1. Each discrete state or mode $q_i(i = 1, ..., 8)$ is a combination of the discrete mode of each discrete cell (switch). As an example, the discrete mode $q = (S_1off, S_2off, S_3off)$ is a combination of the discrete modes S_joff,$(j = 1, 2, 3)$ of the discrete switches $S_j, (j = 1, 2, 3)$. Let $h_{q_i} = (h^1_{q^1_i} h^2_{q^2_i} h^3_{q^3_i})$ be the discrete output of the discrete mode $q_i = (q^1_i q^2_i q^3_i)$. Table 4.1 presents the reference output voltage $V_{S,ref}$ at each discrete mode q. Figure 4.5 shows the change of $V_{S,ref}$ according to the change in the discrete mode of the MCCS.

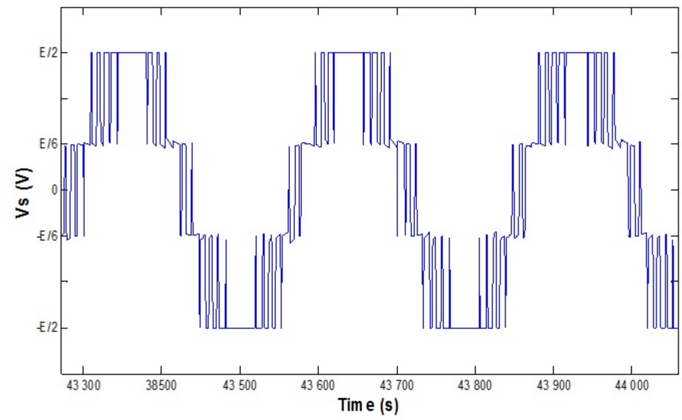

Figure 4.5: Output voltage of three cell converter

The continuous dynamics of the converter are described by the state variables vector $X = \begin{bmatrix} V_{C_1} & V_{C_2} & I \end{bmatrix}^T$, where V_{C_1} and V_{C_2} represent, respectively, the floating voltage of capacitors C_1 and C_2. I represents the current flowing from source E towards DFIG rotor through three elementary switching, S_j, $j \in \{1, 2, 3\}$, cells. The latter represent the system discrete dynamics. Each discrete switch S_j, has two discrete modes: S_j opened $\left(h^j_q = 0\right)$ or closed $\left(h^j_q = 1\right)$, where h^j_q is the state discrete output of S_j. The control of this system has two main tasks: -) balancing the voltages between the switches and -) regulating the load current to a desired value. To accomplish that, the controller changes the switches discrete states from opened to closed or from closed to opened by applying discrete commands close CS_j or open OS_j to each discrete switch j (see Figure 4.3). Thus, the considered example is a DCSS since it has different continuous dynamics (related to capacitors) in each discrete mode and the switching between these discrete modes is controlled by the discrete control events. The three cell converter dynamics evolution is written as follows [22]:

$$\begin{cases} \dot{V}_{C_1} = -h_q^1 \frac{1}{C_1} I + h_q^2 \frac{1}{C_1} I \\ \dot{V}_{C_2} = -h_q^2 \frac{1}{C_2} I + h_q^3 \frac{1}{C_2} I \\ \dot{I} = -\frac{R}{L} I + h_q^1 \frac{1}{L} V_{C1} + h_q^2 \frac{1}{L} (V_{C_2} - V_{C_1}) + h_q^3 \frac{1}{L} (E - V_{C_2}) - \frac{E}{2L} \end{cases} \tag{4.3}$$

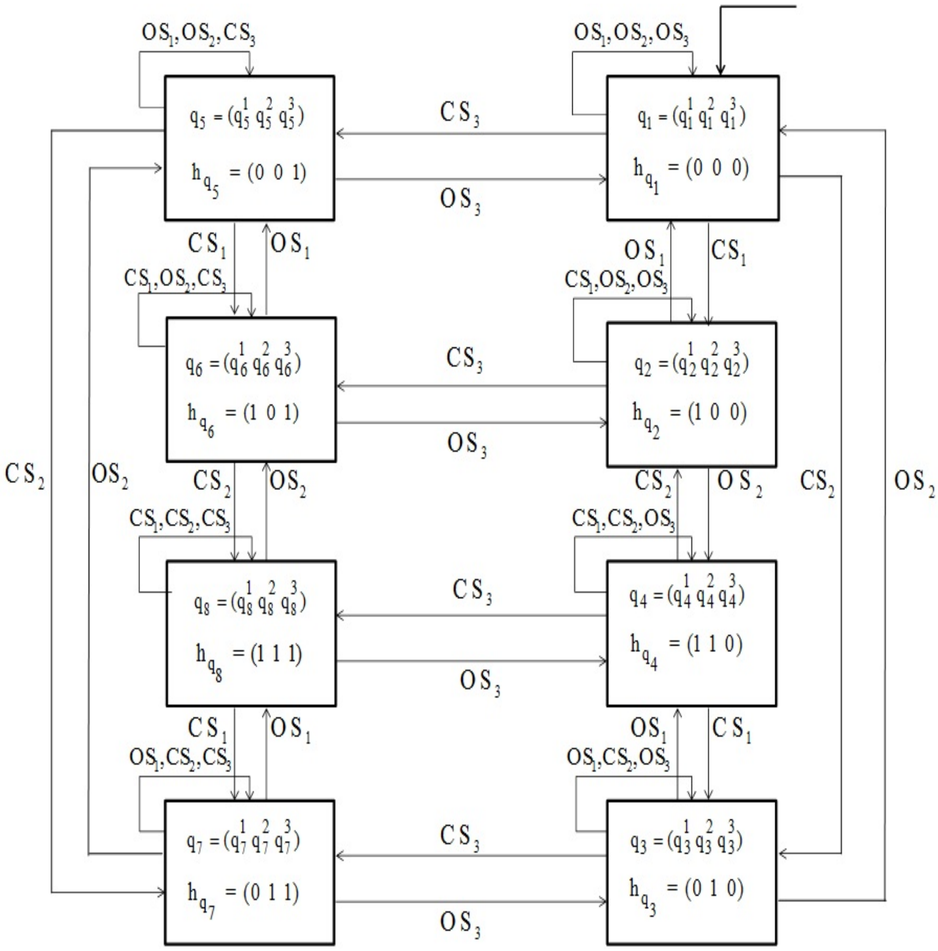

Figure 4.6: Discrete event model represented by a finite state automaton to describe the discrete modes of the three cells converter.

4.6 Multicellular converter drift-like fault scenarios generation

In this chapter, the parametric faults impacting adversely the continuous dynamics of the capacitors charge and discharge in the rotor side converter (RSC) are considered. The Multi Cellular Converter System (MCCS) aims at ensuring a reference output voltage (V_S). Then V_S with the corresponding reference current I allow to provide an adequate torque used by the generator to produce an optimal amount of energy in response to the current wind speed. Due to degradation phenomena, this ability drops over time. This drop will impact adversely the generator performance and consequently the produced energy. When this drop arrives to a certain level, defined as unacceptable, the converter is considered working in fault operating conditions.

The degradation in converter performance is considered to be related to the chemical aging of its capacitors. As the electrolyte capacitor degenerates, the equivalent serial resistance (ESR) rises, which causes the output voltage of the converter to drop (see Figure 4.8). The lower voltage is fed into the inverter (generator) and consequently it contributes to a reduced turbine power output. In addition, the controller will try to compensate this drop in the output voltage by changing the state (opened, closed) of the switches (transistors of type IGBT (Insulated Gate Bipolar Transistors). Therefore, these switches will become more active leading to rise the power cabinet temperature due to more switching which in its turn reduces the power output of the WT.

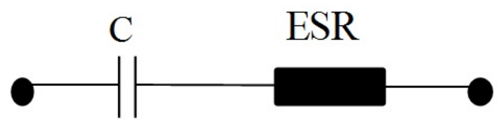

Figure 4.7: Simplified diagram of the equivalent serial resistance (ESR) of a real capacitor.

In order to simulate these parametric faults in converter capacitors, a gradual increase in the nominal value of ESR is generated. This gradual increase in ESR will directly impact the voltage of the floating capacitors (see Figure 4.8):

$$V_{C_j} = \frac{1}{C_j} \int I dt + ESR_j.I \tag{4.4}$$

The following subsections detail the generation of several scenarios representing drift-like faults with three different speeds in C_1, in C_2 and in both C_1 and C_2.

4.6.1 Simple parametric drift-like fault in capacitor C_1

In this chapter the simple parametric fault in C_1 is modeled as a gradual increase in equivalent serial resistance (ESR) of capacitor C_1 (see Figure 4.10). Thus, the

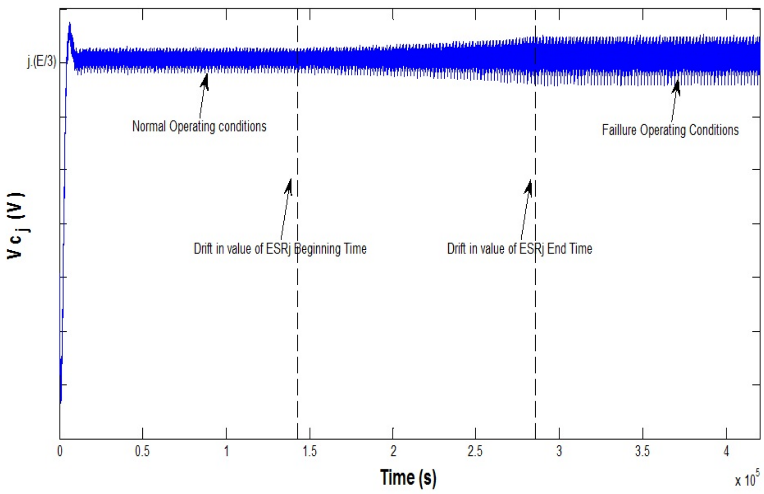

Figure 4.8: Voltage of the floating capacitors in the case of gradual increase in the nominal value of ESR_j

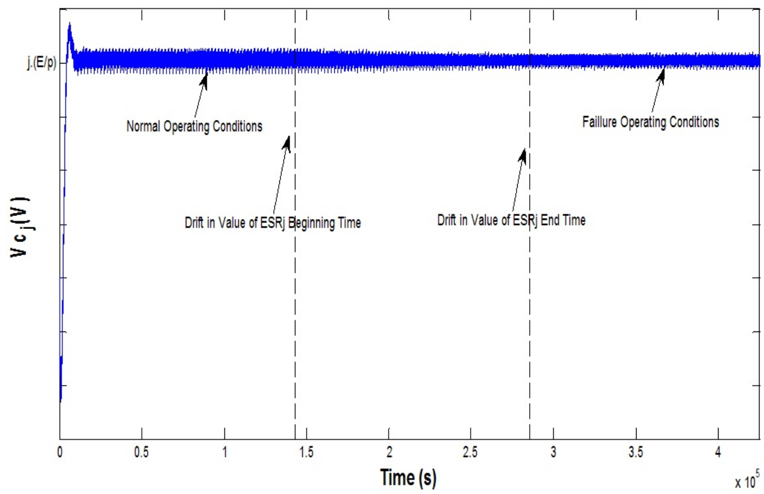

Figure 4.9: Voltage of the floating capacitors in the case of gradual decrease in the nominal value of ESR_j

nominal value of ESR_1 related to C_1 is increased linearly from ESR_{1N} to ESR_{1F} in a period of 5s, 10s and 15s corresponding, respectively, to high, moderate and slow drift speeds.

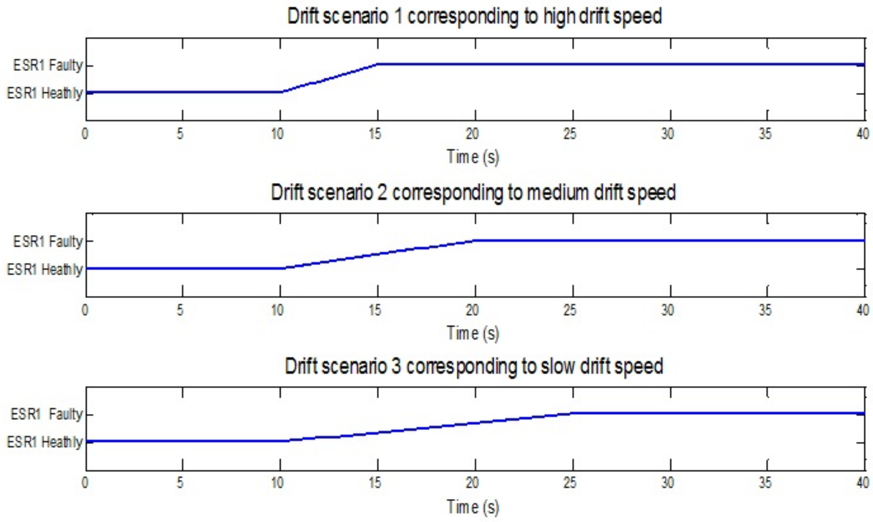

Figure 4.10: Converter drift-like fault scenarios related to capacitor C_1.

4.6.2 Simple parametric drift-like fault in capacitor C_2

The nominal value of the equivalent serial resistance ESR_2 of capacitor C_2 is increased linearly from ESR_{2N} to ESR_{2F} in a period of 5s, 10s and 15s corresponding, respectively, to high, moderate and slow drift speeds (see Figure 4.11).

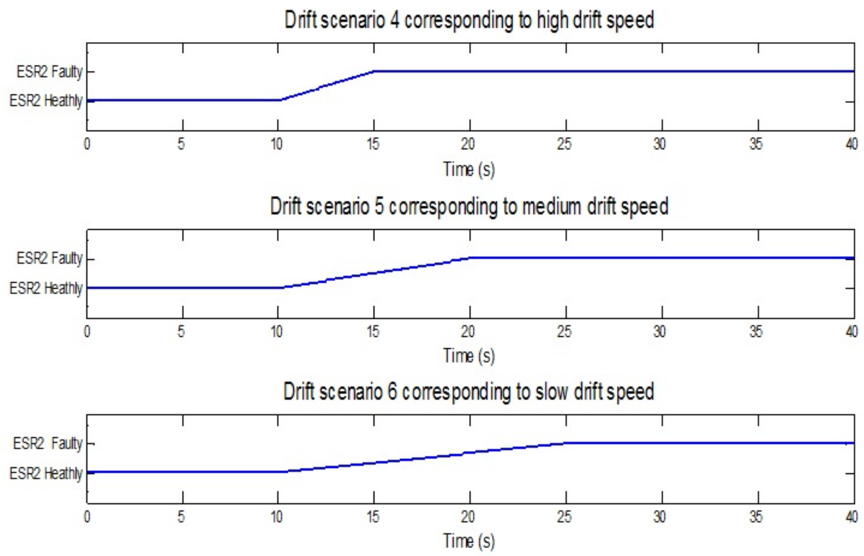

Figure 4.11: Converter drift-like fault scenarios related to capacitor C_2.

4.6.3 Multiple parametric drift-like fault in C_1 and C_2

The multiple parametric fault are modeled as a gradual increase at the same time in the equivalent serial resistances (ESR_1 and ESR_2) of both capacitors C_1 and C_2 . The multiple fault scenarios for this fault are generated in order to simulate degradations represented by a drift in the nominal value of ESR_1 and ESR_2 . Thus, ESR_1 and ESR_2 are increased linearly from ESR_{1N} and ESR_{2N} to ESR_{1F} and ESR_{2F} in a period of 5s, 10s and 15s corresponding respectively to high, moderate and slow drift speeds (see Figure 4.12).

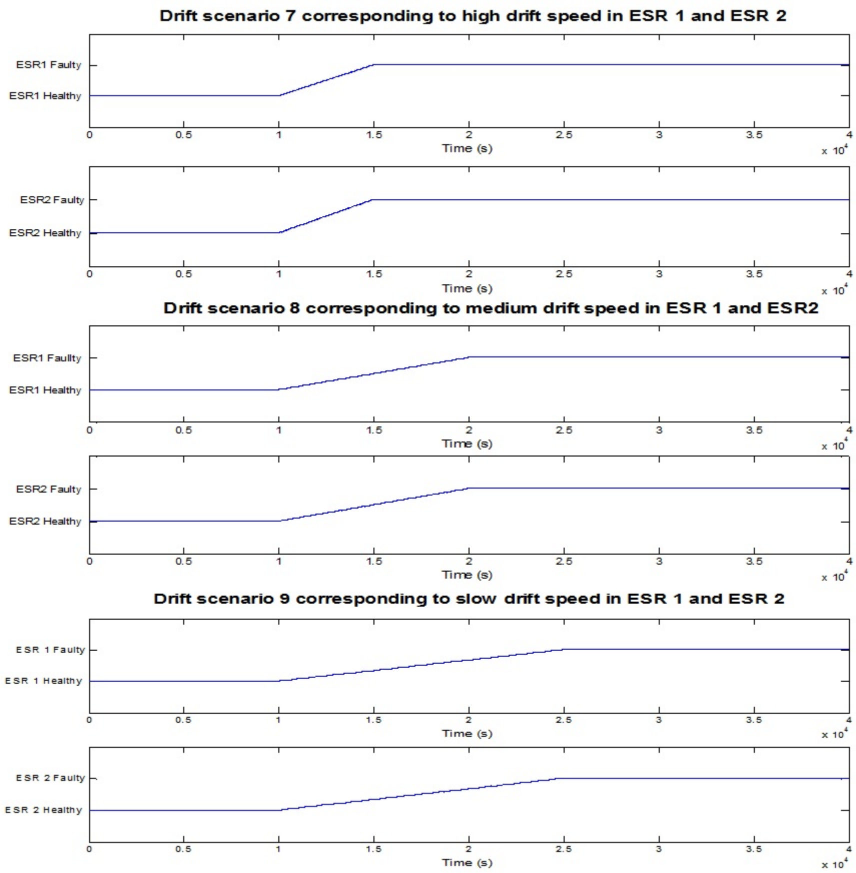

Figure 4.12: Converter multiple drift-like fault scenarios related to capacitor C_1 and C_2 .

The objective of simulating different degradation (drift) speeds is to test the robustness of the proposed approach in detecting drifts of different dynamics (speeds). The converter faults scenarios are summarized in Table 4.2.

Fault N	Drift speed	Converter Fault	Type
F1h	5s(Fast)	$ESR_{1N} \rightarrow ESR_{1F}$	Simple fault in C_1
F2m	10s(Medium)	$ESR_{1N} \rightarrow ESR_{1F}$	Simple fault in C_1
F3s	15s(Slow)	$ESR_{1N} \rightarrow ESR_{1F}$	Simple fault in C_1
F4h	5s	$ESR_{2N} \rightarrow ESR_{2F}$	Simple fault in C_2
F5m	10s	$ESR_{2N} \rightarrow ESR_{2F}$	Simple fault in C_2
F6s	15s	$ESR_{2N} \rightarrow ESR_{2F}$	Simple fault in C_2
F7h	5s	$ESR_{1N} \rightarrow ESR_{1F}$ and $ESR_{2N} \rightarrow ESR_{2F}$	Multiple fault in C_1andC_2
F8m	10s	$ESR_{1N} \rightarrow ESR_{1F}$ and $ESR_{2N} \rightarrow ESR_{2F}$	Multiple fault in C_1 and C_2
F9s	15s	$ESR_{1N} \rightarrow ESR_{1F}$ and $ESR_{2N} \rightarrow ESR_{2F}$	Multiple fault in C_1andC_2

Table 4.2: Generated converter drift-like fault scenarios.

4.7 Proposed approach

In this section, hybrid dynamic data-mining scheme is proposed in order to achieve condition monitoring and drift-like fault diagnosis for the three cell converter. This scheme performs early diagnosis by detecting a drift of the system operating conditions from normal to faulty modes.

The proposed data-mining scheme is composed of the following main steps: processing and data analysis, classification, drift monitoring and updating and interpretation steps (see Figure 4.13).

4.7.1 Processing and data analysis

This step aims at finding the features sensitive to the system operating conditions in each discrete mode in order to construct the feature space. The research of sensitive features is based on the signals provided by the converter sensors as well as the prior knowledge about the system physics and dynamics. These features are chosen in order to maximize the discrimination power between the different operating conditions modes in the feature space. In this work, a dynamical feature space sensitive to normal operating conditions in each discrete mode is defined. The feature space is dynamic in the sense that the features are selected according to the current discrete mode. This allows to choose the useful patterns representative of the drift and therefore to detect it in its early stage [41].

4.7.1.1 Dynamical feature space construction

For the drift like fault diagnosis of the faults related to capacitors, we propose to explore the physical knowledge in order to construct the feature space. In the latter, the features are represented by residuals R_{r,q_n} $r : (r = 1, 2, 3)$ where r is the number of features in the feature space. The residual is generated by the comparison between the voltage measurement and its reference value, see Equation 4.5,4.6 and

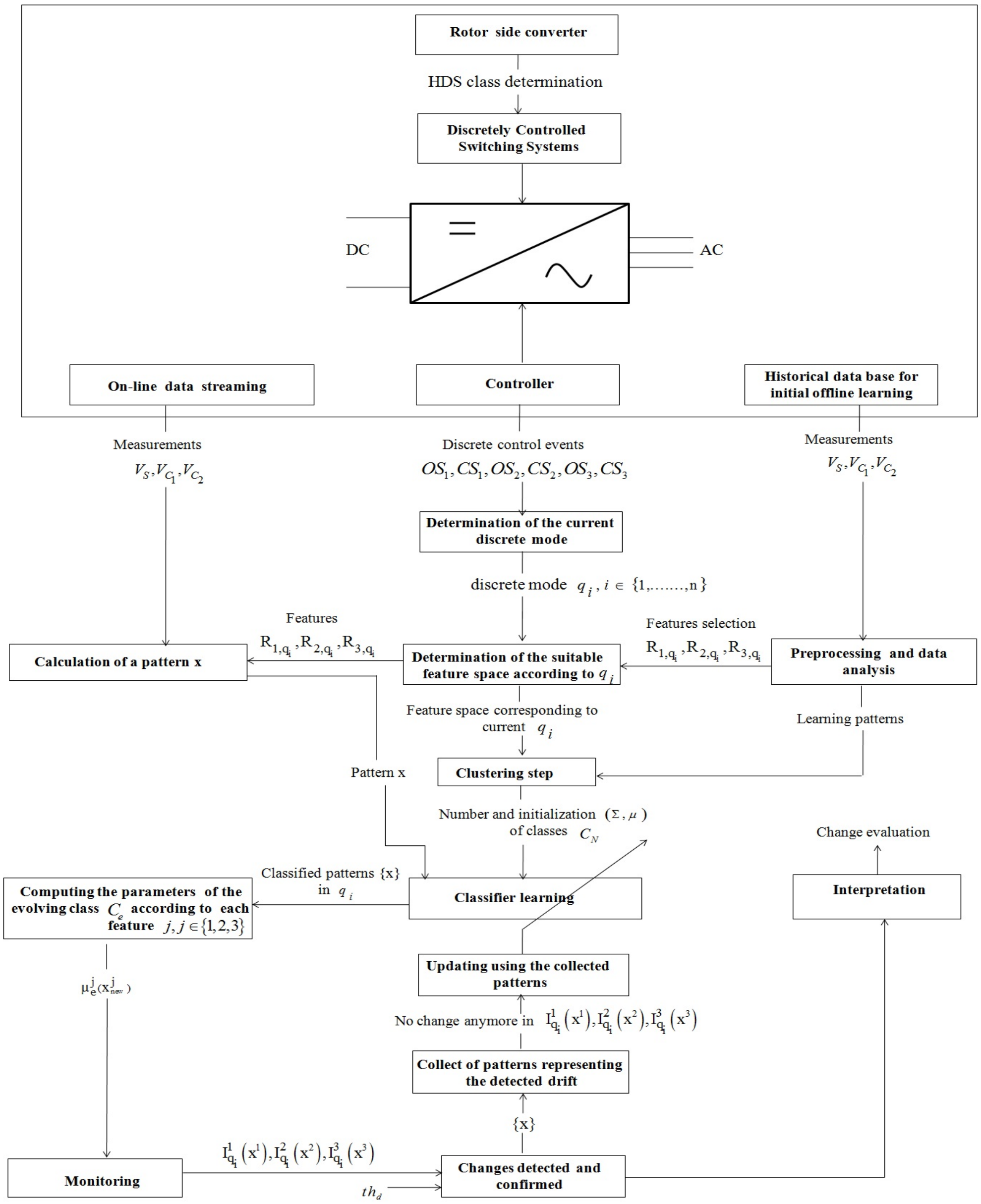

Figure 4.13: Proposed on-line adaptive scheme steps.

Equation 4.7. In this work, three features are defined and the classifier choses the discriminant combination of the features according to each discrete mode as follows:

Feature 1: This feature is related to capacitor C_1 . It is generated by the residual R_{1,q_i} as follows:

$$R_{1,q_i} = V_{C_1,m} - \left(V_{C_1,ref} = \frac{E}{3}\right) \tag{4.5}$$

R_{1,q_i} is computed by the comparison between the real voltage measurement $V_{C_1,m}$ of C_1, and its voltage reference value $V_{C_1,ref} = \dfrac{E}{3}$. R_{1,q_i} is less than a threshold, th, when C_1 is working in active normal operating conditions. R_{1,q_i} is greater than a threshold, th, when C_1 starts to deviate from its normal (nominal) value due to chemical aging effects as example.

Feature 2: This feature is related to capacitor C_2. It is generated by the residual R_{2,q_i} as follows:

$$R_{2,q_i} = V_{C_2,m} - \left(V_{C_2,ref} = \frac{2E}{3}\right) \tag{4.6}$$

R_{2,q_i} is computed by the comparison between the real voltage measurement $V_{C_2,m}$ of C_1, and its voltage reference value $V_{C_1,ref} = \dfrac{2E}{3}$. R_{2,q_i} is less than a threshold, th, when C_2 is working in active normal operating conditions. R_{2,q_i} is greater than a threshold, th, when C_2 starts to deviate from its normal (nominal) value due to chemical aging effects as example.

Feature 3: This feature is related to the converter output and is generated by the residual R_{3,q_i} as follows:

$$R_{3,q_i} = V_{S,m} - V_{S,ref} \tag{4.7}$$

R_{3,q_i} is computed by the comparison between the real output voltage measurement $V_{S,m}$, and its output voltage reference value $V_{S,ref}$. The latter has different values according to the converter discrete mode q_i (see Table 4.1). Therefore, R_{3,q_i} updates its V_{S,ref,q_i} depending on the current converter discrete mode q_i. R_{3,q_i} is equal to zero when the converter continuous dynamics (described by the nominal capacitors values) are in normal operating conditions (no parametric faults related to capacitors).

The number of feature spaces is equal to the number of discrete modes of the multicellular converter. In each discrete mode q_i, the current I has different paths to circulate through the switches and capacitors (see Figure 4.14). This information allows us to remove or add a residual in the feature space because in the case where I cannot circulate through a capacitor, the fault impact of this capacitor cannot be observed (see Figure 4.10). For this reason, the classifier selects the discriminant features according to the discrete mode in which these features are sensitive to normal operating conditions. In this chapter, q_1 and q_8 are not considered because in these two discrete modes there is no current floating through the capacitors (i.e., the parametric faults consequences cannot be observed). For the other $\{q_2\ , q_3, q_4, q_5, q_6, q_7\}$, the classifier selects its feature space in response to the current discrete mode q_i as follows:

Feature space in q_i \ Feature R_n	R_1	R_2	R_3
Feature space in q_2	+	−	+
Feature space in q_3	+	+	+
Feature space in q_4	−	+	+
Feature space in q_5	−	+	+
Feature space in q_6	+	+	+
Feature space in q_7	+	−	+

Table 4.3: Feature space matrix where + and - indicate, respectively, the add and the delete of the corresponding feature in the feature space.

In discrete mode 2 (q_2), R_{1,q_2},R_{3,q_2} represent the attributes of the feature space as follows:

$$q_2 \begin{cases} R_{1,q_2} = V_{C_1,m} - \left(V_{C_1,ref} = \frac{E}{3}\right) \\ R_{3,q_2} = V_{S,m} - \left(V_{S,ref,q_2} = -\frac{E}{6}\right) \end{cases} \tag{4.8}$$

As an example, for the feature space defined in mode 2 (q_2), these features are selected because R_{1,q_2} is impacted in q_2 by a parametric fault in capacitor C_1 since the latter is solicited (I circulates through C_1) and R_{3,q_2} is sensitive in q_2 to a parametric fault in C_1 or C_2. R_{3,q_2} is adapted according to the voltage reference value in q_2. R_{2,q_2} is not selected because C_2 is not solicited in q_2 (i.e., the current does not float through C_2 when the converter is in q_2) (see Figure 4.14). The features of the feature space defined in mode 3 (q_3), are selected because in q_3, C_1 and C_2 related to $R_{1,q_3}, 2, q_3$ are solicited. R_{3,q_3} is selected and adapted according to the voltage reference value in since it is sensitive to a parametric fault in C_1 or C_2 (see Figure 4.14). While, the features of the feature space defined in mode 4 (q_4), are selected because in q_4, C_2 is solicited and R_{3,q_4} is adapted according to the voltage reference value in q_4. R_{1,q_4} is not selected because C_1 is not solicited in q_4 (i.e., the current does not float through C_1) (see Figure 4.14). Likewise, the feature space for each of the remaining discrete modes can be defined.

Feature \ q_n	q_1	q_2	q_3	q_4	q_5	q_6	q_7	q_8
R_1	-	F_{C_1}	F_{C_1}	-	-	F_{C_1}	F_{C_1}	-
R_2	-	-	F_{C_2}	F_{C_2}	F_{C_2}	F_{C_2}	-	-
R_3	-	F_{C_1}	F_{C_1},F_{C_2},$F_{C_1C_2}$	F_{C_2}	F_{C_2}	F_{C_1},F_{C_2},$F_{C_1C_2}$	F_{C_1}	-

Table 4.4: Sensitivity of residuals R_1 ,R_2 and R_3 to the parametric faults in C_1 (indicated by the fault label F_{C_1}) and in C_2 (indicated by the fault label F_{C_1}) in each discrete mode q_i of the multicellular converters.

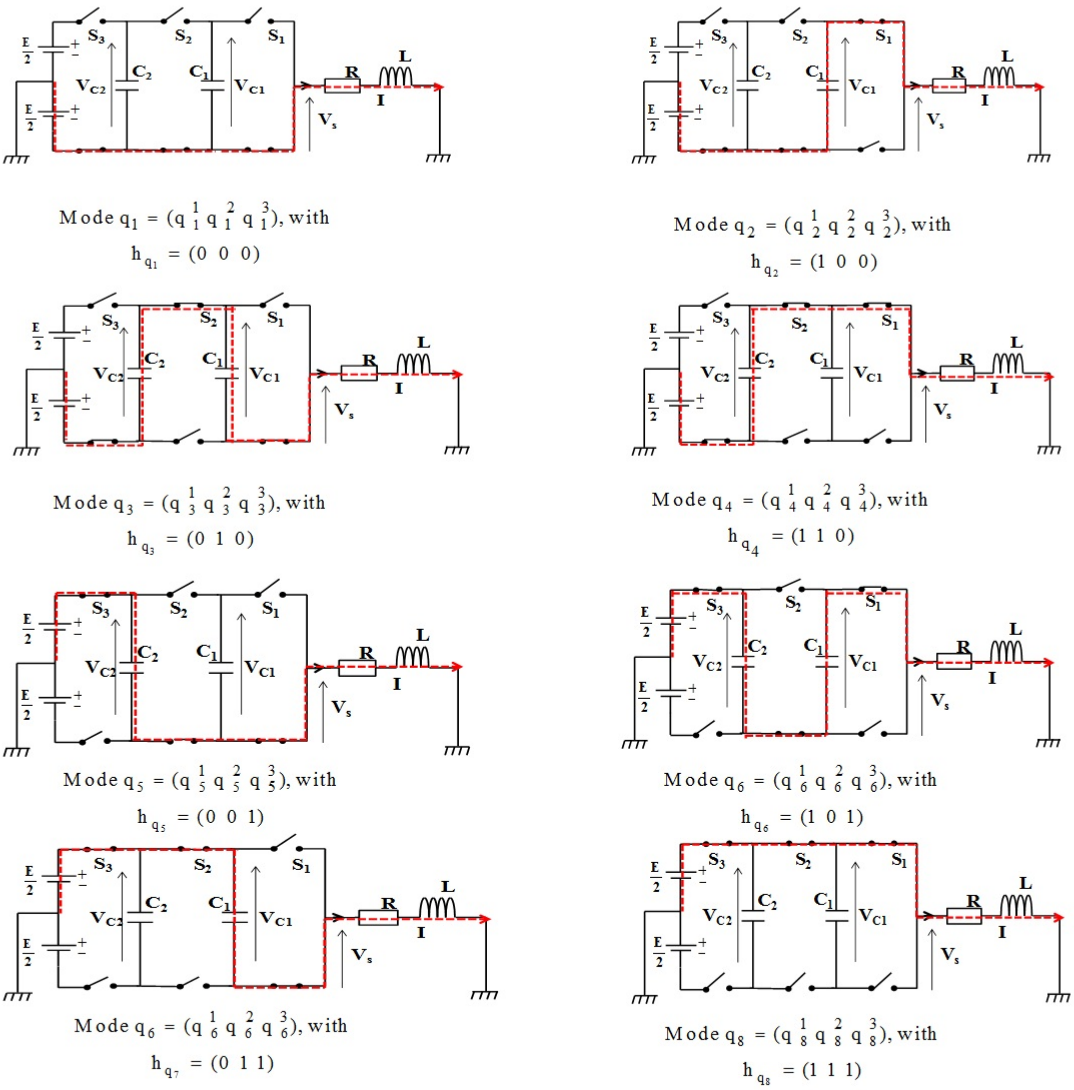

Figure 4.14: Different discrete modes of a three-cell converter

4.7.2 Classifier learning and updating

The classifier learning aims at designing a classifier able to assign a new pattern to one of the learnt classes in the feature space. A new pattern characterizes the current operating conditions (normal or faulty in response to the occurrence of a parametric fault) of the system. The updating aims at reacting to the changes in the classifier environment by updating its parameters and structure. The goal is to preserve the classifier validity and performance over time. Examples of these approaches, can be found in [26], [18],[86] and the references therein. Without loss of generality, we use the Auto-adaptive Dynamical Clustering Algorithm (AuDyC) [69] to achieve the classification and updating tasks. AuDyC computes the parameters of initial classes based on the data statistical properties which are the mean and the variance-covariance matrix. AuDyC was chosen because it is unsupervised classification method and it is able to model streams of patterns since it always reflects the final distribution of patterns in the features space. It uses a technique that is inspired from the Gaussian mixture model [69] Let E^d be a d-dimensional feature space. Each feature vector $x \in E^d$ is called a pattern. The patterns are used to model Gaussian prototypes P^j characterized by a center $\mu_{Pj} \in R^d$ and a variance-covariance matrix $\Sigma_{Pj} \in R^{d \times d}$. Each Gaussian prototype characterizes a class. A minimum number of N_{win} patterns are necessary to define one prototype, where N_{win} is a user-defined threshold.

A class models operating conditions and gathers patterns that are similar one to each other. The similarity criterion that is used is the Gaussian membership degree. Faults will affect directly this distribution and this will be seen through the continuously updated parameters. More details about AuDyC can be found in [69] and the references therein. AuDyC continuously updates the classes parameters (mean covariance matrix) by integrating the newest arrived pattern X_{new} and by removing the oldest pattern X_{old} in the time window W_t containing N_{win}. It achieves this update using recursive adaptation rules [69] in order to preserve the classifier validity and performance over time. AuDyC is used firstly off-line in order to construct the classifier and to characterize the parameters of the initial classes (representing normal operating conditions of MCCS) based on the use of the historical data set. The latter is based on the collection of sensor data during a certain time of system operation.

In this work, we suppose that only data corresponding to normal operating conditions (normal class) are available in advance. The class corresponding to normal operating conditions is denoted by $C_N = (\mu_N, \Sigma_N)$. In on-line functioning, the parameters (μ_N, Σ_N) of C_N for each feature space are dynamically updated by AuDyC for each new pattern. We adopt the assumption that in the normal behavior, the system is in an invariant or stable state or regime (i.e., not oscillatory). The system in this stable state may vary within the operating regime boundaries defined by the three standard deviations in term of data density.

The data is collected on-line continuously during the system run. Some of the new patterns reinforce and confirm the information (parameters) contained in the previous data. In this case, (μ_N, Σ_N) of C_N will be updated by AuDyC. However, other patterns can indicate a change in the information (normal class parameters) contained in the previous data. This change is considered to be the result of a fault development. Therefore, any pattern X_{new} entailing a change in the parameters of the system normal class (converter) greater than the three standard deviations in term of data density will be considered as a pattern of a new class. The latter is called the evolving class and denoted by C_e.

AuDyC is dynamic by nature in the sense that it continuously updates the parameters of the classes as new patterns arrive without taking into account the current discrete mode. This creates two problems. Firstly, a change in the system characteristics can be related to a change in the normal operating conditions (system discrete mode or its discrete dynamics) and not due to a fault. Secondly, in the case of the occurrence of a parametric fault, only patterns characterizing the behavior of the system in the discrete modes where this parametric fault is active are representative of the drift. All the other patterns are useless and will delay the drift detection and confirmation. Therefore, using AuDyC in the proposed scheme improves its performance by converting the designed classifier by AuDyC to a hybrid dynamical classifier. The latter detects and confirms a drift in its early stage thanks to the use of only representative patterns of a drift resulting from a parametric fault.

4.7.3 Drift monitoring and interpretation

The key problem of drift monitoring is to distinguish between variations due to stochastic perturbations and variations caused by unexpected changes in a systems state. If the sequence of observations is noisy, it may contain some inconsistent observations or measurements errors (outliers) that are random and may never appear again. Therefore, it is reasonable to monitor a system and to process observations within time windows in order to average and reduce the noise influence. Moreover, the information about possible structural changes within time windows can be interpreted and processed more easily. As a result, a more reliable classifier update can be achieved by monitoring within time windows. The latter must include enough of patterns representing the drift.

The proposed scheme is based on the use of a dynamical evolving time window . The size of the latter is defined by the number of patterns representing the current drift. This size depends on the drift speed. If the drift speed is high, then its size will be small; while when the drift is slow the window size will be high in order to include sufficient of patterns representing the drift (degradation dynamics). starts when the drift is detected and ends when one drift indicator at least remains stable. When another drift indicator remains stable, this will confirm the end of the drift. Therefore, the size of the evolving time window is determined and confirmed dynamically according to the drift speed. In this work these indicators are based on the Euclidean distance according to each attribute between the class C_N representing normal operating conditions and the evolving class C_E in the feature space Figure 4.15. To achieve that, let $C_N = \left(\mu_N^1, \mu_N^2, \mu_N^3, ..., \mu_N^d\right)$ and $C_e = \left(\mu_e^1, \mu_e^2, \mu_e^3, ..., \mu_e^d\right)$ be represented by the gravity center (the mean value) of its probability density according to each attribute j of the feature space. When a new pattern $x_{new} = (x_{new}^1, x_{new}^2, ..., x_{new}^j, ...x_{new}^d)$ is classified in the evolving class C_e , the gravity center $\left(\mu_e^1(x^1), \mu_e^2(x^2), \mu_e^3(x^3), ..., \mu_e^d(x^d)\right)$ of C_e will be updated recursively in order to take into account the information carried by X_{new}.

Let $I_{q_i}^j\left(x_{new}\right)$ be the drift indicator measuring the Euclidean distance between the gravity centers of C_N and C_e according to each attribute j, $j = 1, \ldots, d$,when the system is in the discrete operating mode q_i . $I_{q_i}^j\left(x_{new}\right)$ is calculated as follows:

$$I_{q_i}^j\left(x_{new}^j\right) = d_E\left(\mu_N^j, \mu_e^j\right), j = 1, \ldots, d; i = 1, \ldots, n \tag{4.9}$$

where d_E is the Euclidean metric calculated as follows:

$$d_E\left(\mu_N^j, \mu_e^j\right) = \left|\mu_N^j - \mu_e^j(x_{new}^j)\right| \tag{4.10}$$

Since there are d = 3 features (attributes) for the MCCS with n = 8 discrete modes, then there are three drift indicators $I_{q_i}^1\left(x^1\right), I_{q_i}^2\left(x^2\right), I_{q_i}^3\left(x^3\right)$ used to measure the Euclidean distance between the gravity centers of C_N and C_e according to each feature and in each discrete mode q_i :

$$I_{q_i}^1\left(x_{new}^1\right) = d_E\left(\mu_N^1, \mu_e^1\right) \tag{4.11}$$

$$I_{q_i}^2\left(x_{new}^2\right) = d_E\left(\mu_N^2, \mu_e^2\right) \tag{4.12}$$

$$I_{q_i}^3\left(x_{new}^3\right) = d_E\left(\mu_N^3, \mu_e^3\right) \tag{4.13}$$

The indicators $I_{q_i}^1\left(x^1\right), I_{q_i}^2\left(x^2\right), I_{q_i}^3\left(x^3\right)$ keep always the greatest distance over time. Therefore, $I_{q_i}^1\left(x^1\right), I_{q_i}^2\left(x^2\right), I_{q_i}^3\left(x^3\right)$ will be calculated as follows:

$$I_{q_i}^j\left(x_{new}^j\right) = \begin{cases} d_E\left(x_{new}^j\right) if d_E\left(x_{new}^j\right) > d_E\left(x_{t-1}\right) \\ d_E\left(x_{t-1}\right) otherwise \end{cases} \tag{4.14}$$

The greatest distance is choose over time in order to use the patterns representing the best the drift in normal operating conditions. This will help to better represent the drift dynamics (speed) over time.

A drift is detected when one at least of these drift indicators exceeds a threshold th . In this case, an alarm is activated to warn human operators of supervision about a potential drift. The other indicators sensitive to this drift will confirm this alarm (drift). The selection of th is motivated statically by taking three σ(standard deviations) of the data in the normal operating conditions. This value represents a good trade-off between false detection and missed detection of drift when the severity of the drift is low.

Euclidean distance is used as metric in order to compute the distance between two centers (mean values) μ_N^j and μ_e^j of the probably densities of the normal and evolving classes according to each feature. The reason behind the use of a drift indicator for each feature is to detect and isolate a drift as soon as possible. Indeed, each feature (residual for the converter) is sensitive for a drift (degradation) resulting from a parametric fault in a continuous component (a capacitor in the converter). When a parametric fault impacts a continuous component, the gravity center of the normal class C_N according to the sensitive feature will be changed. Therefore, following the change in the data density characteristics of the normal class according to the sensitive features can help to isolate the origin (affected continuous component) of this drift. Also, the indicators impacted by this drift can be used to confirm the occurrence and the origin of this drift (degradation of a continuous component as a capacitor).

The interpretation step aims at interpreting the detected changes in the classifier parameters and structure. This interpretation may then be used as a short-term prognosis about the tendency of the future development of the current situation. This prognosis is useful to formulate a control action to modify the dynamics of a

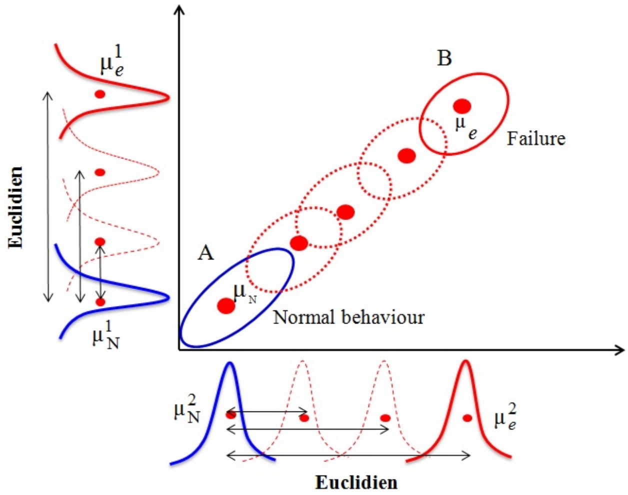

Figure 4.15: Drift indicators according to each attribute of the feature space measuring the Euclidean distance between the gravity centers of normal and evolving classes.

system. For instance, let suppose that we have two classes A and B. Let suppose that class A represents the normal operating conditions (e.g., capacitor is working normally) while class B is a fault state (capacitor value is outside of its nominal interval). When the interpretation step provides the result The systems state has been moved away from class A and is approaching class B, this means that the system needs to be repaired, adjusted or reconfigured. The goal is to inverse its tendency, to move towards a fault state, by forcing it to return to the normal operating conditions. In addition, this step may provide the Remaining Useful Life (RUL) of a system before the failure. RUL is used in Condition-based Maintenance (CBM) to schedule required repair and maintenance actions prior to breakdown (failure state) [33].

4.7.4 Discussion on the choice of drift-like fault indicators for power converter

The discussion on the choice of indicators for drift-like fault diagnosis of pitch system in chapter 3, the criteria to choose the drift indicators were highlighted (see subsection 3.7.5). The drift indicators based on the use of distance measures may cope with high dimensional data, especially where there are many irrelevant attributes. Therefore in this chapter, a drift space-similarity indicator based on the use of the Euclidean distance according to each feature is used to monitor a drift in the spatial characteristics of the distribution of data samples in the normal class according to this feature. This allows handling drifts, where the relevance of features during classification changes over time due to the occurrence of a drift-like fault. In addition, the use of a drift indicator according to each attribute allows not only the drift detection but also the isolation of the source generating this drift (e.g., the capacitor deviating from its nominal value in the converter). Finally, Euclidean distance measure for each feature is useful for the context of multiple drift like faults detection and isolation since the multiple faults involve a drift according to several features. Each feature may be sensitive to a drift generated by one element (e.g. one capacitor in the converter).

Therefore in this chapter, the drift indicators are based on one dimensional Euclidean distance [42]. The other distance measures cannot be compared with the one used in this chapter since these measures, for instance Mahalanobis and Kullback distances, need at least two dimensions in order to calculate the variance covariance matrix.

4.8 Experimentation and obtained results

The used converter is characterized by $E = 600V$, $C_1 = 40muF$and $C_2 = 40muF$. The failures of converter are caused by a continuous degradation of its performance over time. This degradation can be seen as a continuous drift of characteristics of the normal operating conditions (normal class) due to a simple parametric faults in capacitor C_1 , in capacitor C_2 and multiple parametric faults in both capacitors C_1 and C_2 (see Table 4.2). Detecting and following this drift can help the prediction of the occurrence of the converter failures.

Drift indicators $I^1_{q_i}\left(x^1\right), I^2_{q_i}\left(x^2\right)$ and $I^3_{q_i}\left(x^3\right)$, for a pattern x according to each of the three defined features at a discrete mode q_i, are used to detect and to confirm this drift for the nine drift scenarios of C_1 defined in Table 4.2. Table 4.5 shows the values of the drift monitoring indicators $I^1_{q_i}\left(x^1\right), I^2_{q_i}\left(x^2\right)$ and $I^3_{q_i}\left(x^3\right)$ for the drift scenarios of simple and multiple parametric drift-like fault of capacitors defined in Table 4.2. These values represent the required time (starting from the drift beginning) to detect the drift occurrence. Thus, they can be used as an evaluation criterion to measure the time delay to detect a drift before its end. The results are shown by discrete mode q_i to better highlight the contribution of this scheme.

Fault N	Type	Drift speed	I^1	I^2	I^3
F1h	Simple fault in C_1	5s	0.74s	No detection	1.65s
F2m	Simple fault in C_1	10s	1.78s	No detection	3.34s
F3s	Simple fault in C_1	15s	2.71s	No detection	5.09s
F4h	Simple fault in C_2	5s	No detection	0.79s	1.71s
F5m	Simple fault in C_2	10s	No detection	1.81s	3.42s
F6s	Simple fault in C_2	15s	No detection	2.77s	5.17s
F7h	Multiple fault in C_1 and C_2	5s	0.75s	0.78s	1.09s
F8m	Multiple fault in C_1 and C_2	10s	1.79s	1.77s	2.87s
F9s	Multiple fault in C_1 and C_2	15s	2.69s	2.65s	4.10s

Table 4.5: Results of capacitors C_1 and C_2 drift detection and confirmation.

4.8.1 Simple parametric drift-like fault in C_1

Figure 4.16 and Figure 4.17 represent, respectively, $V_{C_1,m}$ and $V_{C_2,m}$ of both C_1 and C_2 in presence of an abnormal drift in C_1. We can see that, $V_{C_1,m}$ is impacted by the occurrence of abnormal drift (degradation) in the nominal value of C_1; while $V_{C_2,m}$ is not sensitive (remains unchanged) to this drift in C_1. This can be justified as follows. The dynamics of charge and discharge ($\dot{V_{C_1}}$ and $\dot{V_{C_2}}$) of C_1 and C_2 are defined by Equation 4.2. It is clear that an abnormal drift (change) in the nominal value of C_1 will impact only $\dot{V_{C_1}}$; while $\dot{V_{C_2}}$ does not depend on the value of C_1.

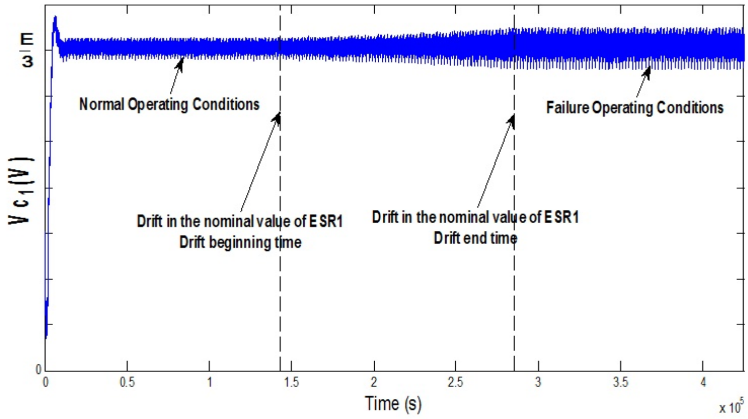

Figure 4.16: Voltage measurement $V_{C_1,m}$ in three cell converter.

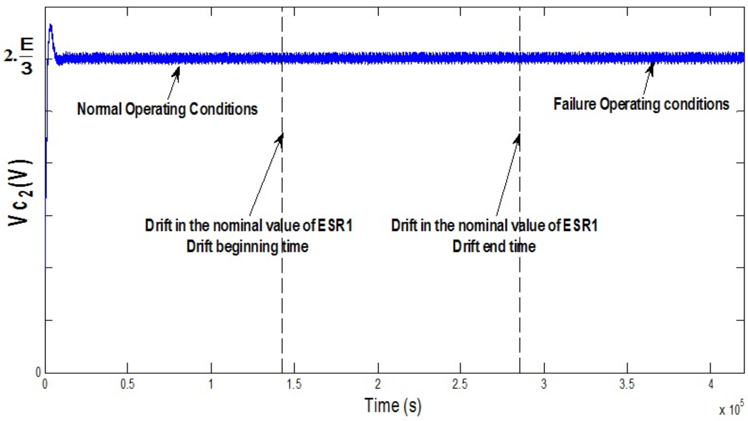

Figure 4.17: Voltage measurement $V_{C_2,m}$ in three cell converter.

Based on Table 4.4, the features R_1 and R_3 are sensitive to a parametric fault in C_1 when the converter is in one of the following discrete modes: q_2 , q_3 , q_6 and q_7; while R_2 is not sensitive to this drift in C_1. Therefore, the drift indicators, $I^1_{q_i}\left(x^1\right)$

and $I^3_{q_i}\left(x^3\right)$, based on, respectively,R_1 and R_3, are used to infer the occurrence of an abnormal drift in C_1 ; while $I^2_{q_i}\left(x^2\right)$ is not sensitive to this drift. Figure 4.18 and Figure 4.19 show these indicators in the discrete mode q_2. Figure 4.20, Figure 4.21 and Figure 4.22 show these indicators in the discrete mode q_3. Figure 4.23 and Figure 4.24 show these indicators in the discrete mode q_4. The figures showing $I^1_{q_i}\left(x^1\right)$, $I^2_{q_i}\left(x^2\right)$ and $I^3_{q_i}\left(x^3\right)$ in the discrete modes q_5,q_6 and q_7 are not integrated in the chapter because they have similar behavior as in q_2, q_3 and q_4.

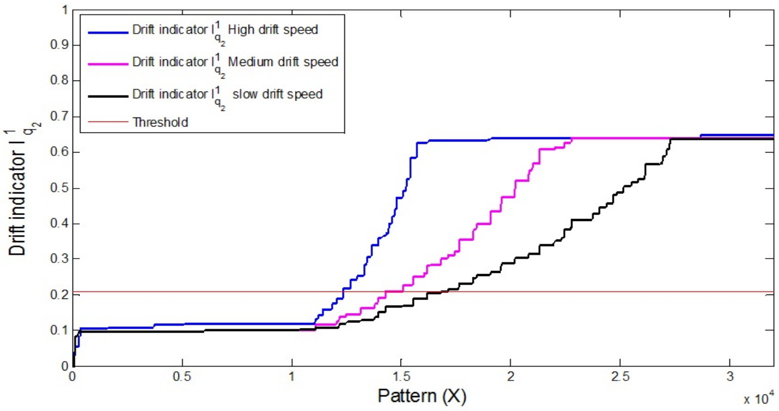

Figure 4.18: Drift indicator $I^1_{q_2}\left(x^2\right)$ for attribute 1 and according to each drift speed in the feature space of discrete mode q_2.

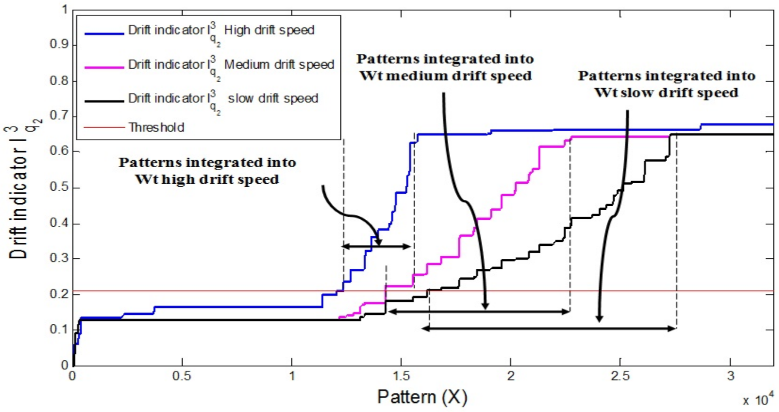

Figure 4.19: Drift indicator $I^3_{q_2}\left(x^3\right)$ for attribute 2 and according to each drift speed in the feature space of discrete mode q_2.

In the discrete mode q_3 and q_6 Where C_1 and C_2 are solicited, the drift-like fault in C_1 is successfully detected by $I^1_{{q_i}^1}\left(x^1\right)$ and $I^3_{{q_i}^3}\left(x^3\right)$ for all drift speeds (see Figure 4.18 and Figure 4.20). However, it cannot be detected by the indicator $I^2_{{q_i}^2}\left(x^2\right)$ (see Figure 4.19), because $I^2_{{q_i}^2}\left(x^2\right)$ is sensitive only to drift like fault in C_2

(see Table 4.4). Same results can be obtained for these indicators in discrete mode q_6.

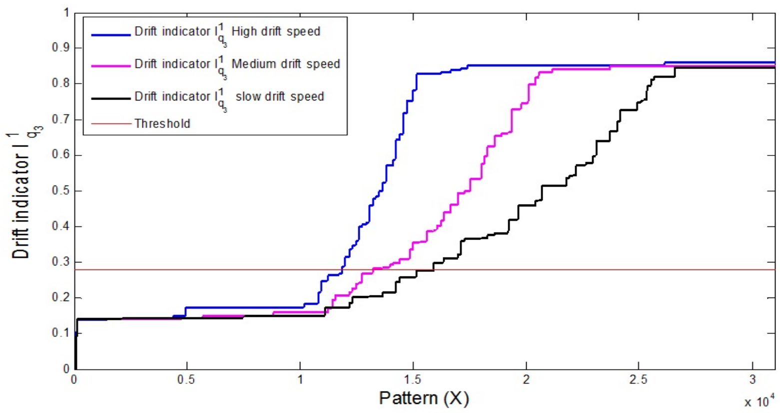

Figure 4.20: Drift indicator $I_{q3}^{1}\left(x^{1}\right)$ for attribute 1 and according to each drift speed in the feature space of discrete mode q_3.

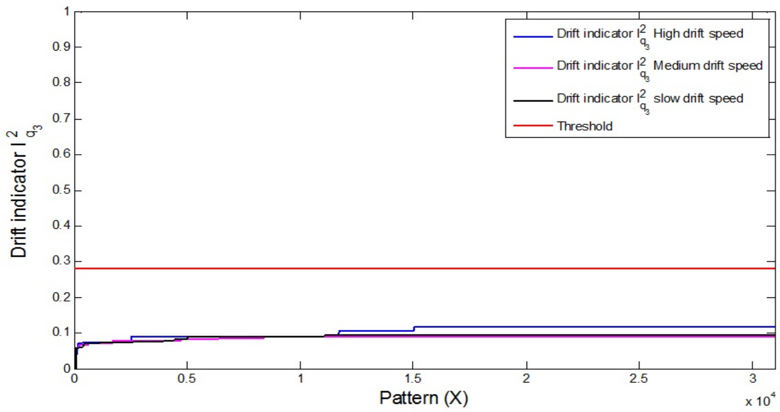

Figure 4.21: Drift indicator $I_{q3}^{2}\left(x^{2}\right)$ for attribute 2 and according to each drift speed in the feature space of discrete mode q_3.

In the discrete mode q_4 and q_5 Where only C_2 is solicited, the drift-like fault in C_1 can not be detected by all indicators $I_{q_i{}^1}^{1}\left(x^{1}\right)$,$I_{q_i{}^2}^{2}\left(x^{2}\right)$ and $I_{q_i{}^3}^{3}\left(x^{3}\right)$ for all drift speeds (see Figure 4.21 and Figure 4.22), because in the discrete mode q_4 and q_5 the drift indicators are sensitive only to drift like fault in C_2 (see Table 4.4). Same results can be obtained for these indicators in discrete mode q_5.

Based on these figures, we can see that the drift like fault in C_1 is successfully detected by both indicators $I_{q_i{}^1}^{1}\left(x^{1}\right)$ and $I_{q_i{}^3}^{3}\left(x^{3}\right)$ for all drift speeds (see Figure 4.20 and Figure 4.22). However, it cannot be detected by the indicator $I_{q_i{}^2}^{2}\left(x^{2}\right)$ (see

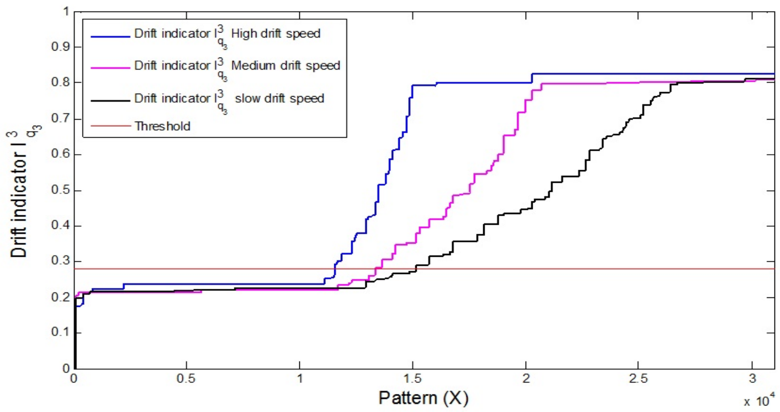

Figure 4.22: Drift indicator $I^3_{q3}\left(x^3\right)$ for attribute 3 and according to each drift speed in the feature space of discrete mode q_3.

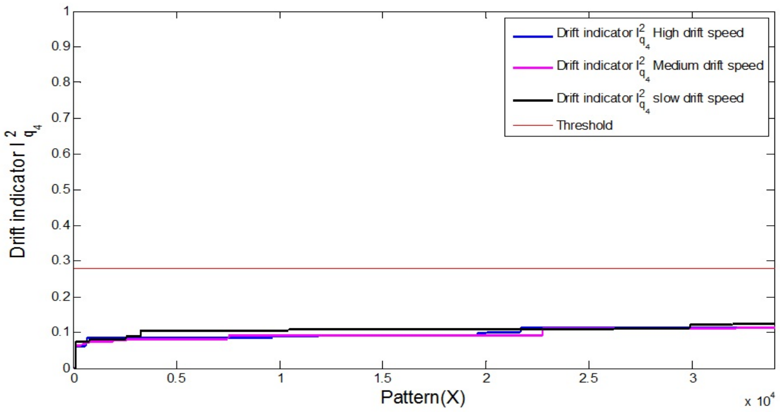

Figure 4.23: Drift indicator $I^2_{q4}\left(x^2\right)$ for attribute 1 and according to each drift speed in the feature space of discrete mode q_4.

Figure 4.22) because $I^2_{q_i{}^2}\left(x^2\right)$ is sensitive only to drift like fault in C_2 (see Table 4.4).

As it can be seen in Table 4.5, $I^3_{q_i{}^3}\left(x^3\right)$, related to the converter output V_{C_S}, detects always the drift after its detection by $I^1_{q_i{}^1}\left(x^1\right)$. For this reason in this work, $I^3_{q_i{}^3}\left(x^3\right)$ is used to confirm the occurrence of a drift-like fault in C_1.

The drift like fault in C_1 is detected in early stage before the end of this drift (arriving to the failure mode due to a parametric fault in C_1). As an example, in the case of a drift of slow speed (F3s) (see Table 4.5), the converter reaches the failure mode resulting from a parametric fault in C_1 after 15 seconds of the beginning of the drift (degradation in C_1). In the proposed approach, this drift is detected 2.7 seconds and confirmed 5.09 seconds after its beginning. Therefore, the

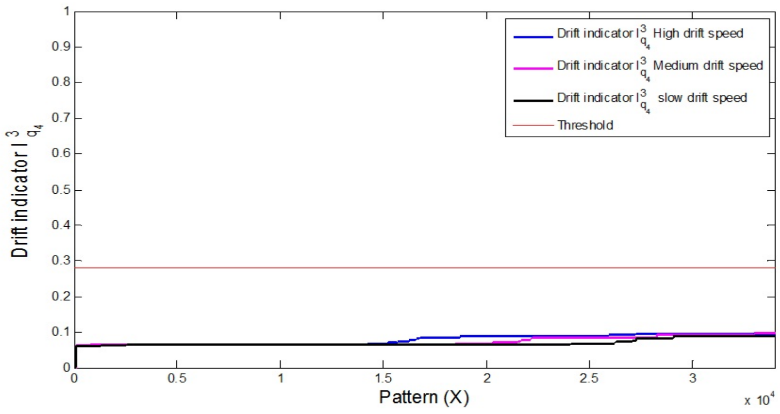

Figure 4.24: Drift indicator $I^3_{q_i^3}\left(x^3\right)$ for attribute 3 and according to each drift speed in the feature space of discrete modeq_4.

drift like fault in C_1 is confirmed 10 seconds before its end. This allows achieving an early fault diagnosis and therefore helps the human operators of supervision to take efficiently the right actions.

4.8.2 Simple parametric drift-like fault in C_2

Figure 4.25 and Figure 4.26 represent, respectively $V_{C_1,m}$ and $V_{C_2,m}$ of both C_1 and in presence of an abnormal drift in C_2. As we have seen in the case of an abnormal drift in C_1 (Subsection 4.1), an abnormal drift in C_2 impacts only $V_{C_2,m}$; while $V_{C_1,m}$ has the same behavior as the one without a drift in C_2.

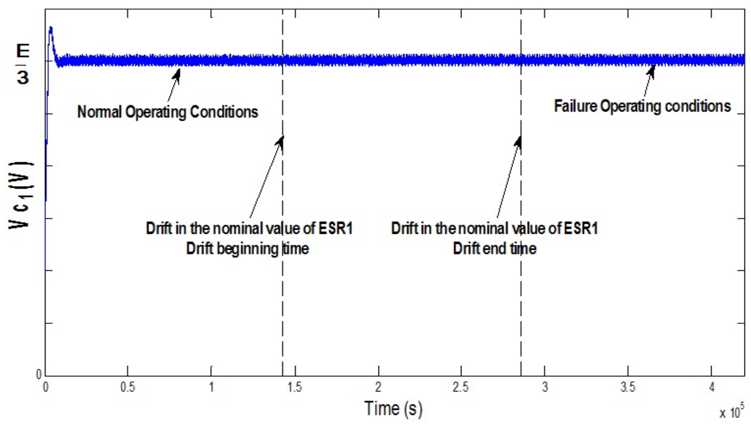

Figure 4.25: Voltage measurement $V_{C_1,m}$ in three cell converter.

The degradation is observed when the system is in the discrete modes where the dynamics (operating conditions) described by the affected parameters are active.

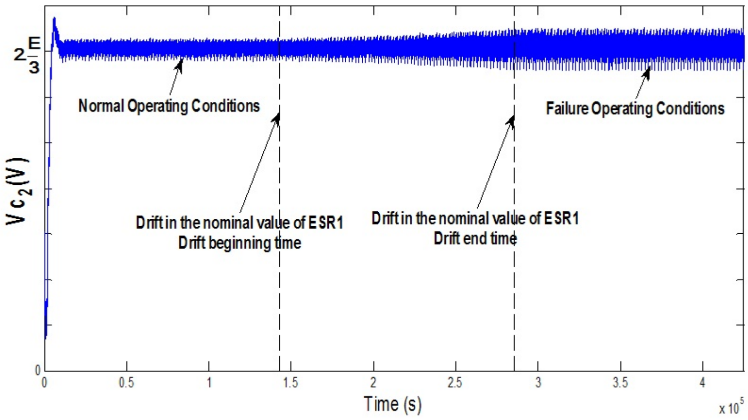

Figure 4.26: Voltage measurement $V_{C_2,m}$ in three cell converter.

In the discrete modes q_3,q_4,q_5 and q_6 where C_2 is solicited, the drift-like fault in C_2 is successfully detected by both indicator $I^2_{q_i{}^2}\left(x^2\right)$ and $I^3_{q_i{}^3}\left(x^3\right)$ for all drift speeds (see Figure 4.30 and Figure 4.31 for the discrete mode q_3). However, it cannot be detected by the indicator $I^1_{q_i{}^1}\left(x^1\right)$ (see Figure 4.29 for the discrete mode q_3), because $I^1_{q_i{}^1}\left(x^1\right)$ is sensitive only to drift like fault in C_1 (see Table 4.4).

Figure 4.27 and Figure 4.28 show these indicators in the discrete mode q_2. Figure 4.29, Figure4.30 and Figure 4.31 show these indicators in the discrete mode q_3. Figure 4.32 and Figure 4.33 show these indicators in the discrete mode q_4. The figures showing $I^1_{q_i}\left(x^1\right)$, $I^2_{q_i}\left(x^2\right)$ and $I^3_{q_i}\left(x^3\right)$ in the discrete modes q_5,q_6 and q_7 are not integrated in the chapter because they have similar behavior as in q_2, q_3 and q_4.

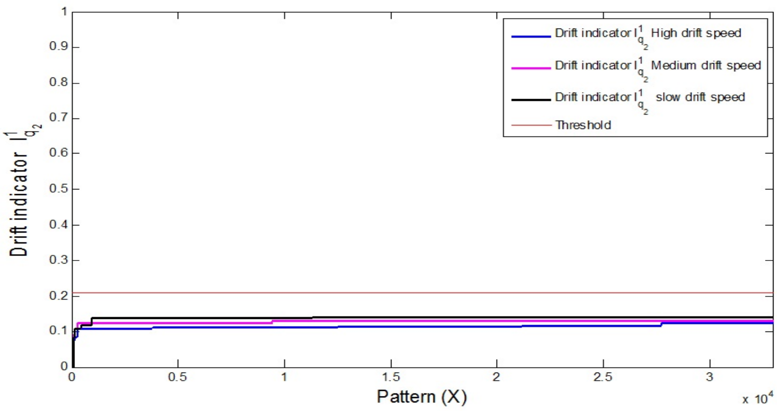

Figure 4.27: Drift indicator $I^1_{q_2}\left(x^2\right)$ for attribute 1 and according to each drift speed in the feature space of discrete mode q_2.

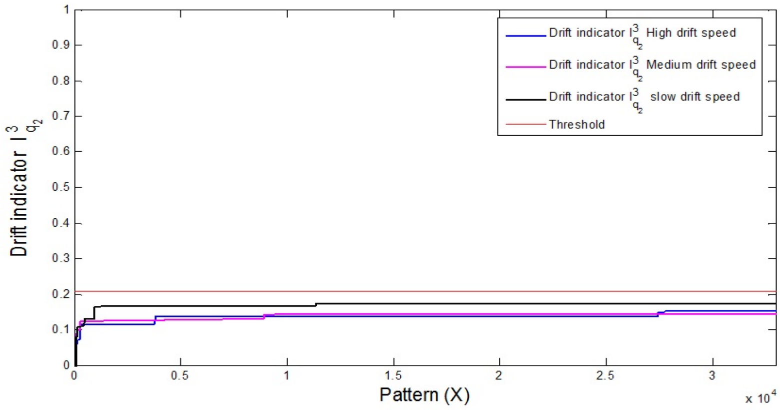

Figure 4.28: Drift indicator $I^3_{q_2}\left(x^3\right)$ for attribute 2 and according to each drift speed in the feature space of discrete mode q_2.

In the discrete mode q_3 and q_6 Where C_1 and C_2 are solicited, the drift like fault in C_2 is successfully detected by $I^2_{q_i 2}\left(x^2\right)$ and $I^3_{q_i 3}\left(x^3\right)$ for all drift speeds (see Figure 4.27 and Figure 4.28). However, it cannot be detected by the indicator $I^2_{q_i 2}\left(x^2\right)$ (see Figure 4.28), because $I^1_{q_i 1}\left(x^1\right)$ is sensitive only to drift like fault in C_1 (see Table 4.4). Same results can be obtained for these indicators in discrete mode q_6.

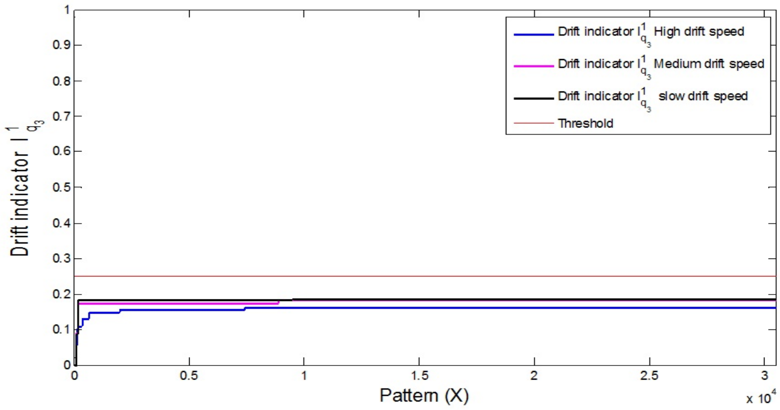

Figure 4.29: Drift indicator $I^1_{q_3}\left(x^1\right)$ for attribute 1 and according to each drift speed in the feature space of discrete mode q_3.

In the discrete mode q_4 and q_5 Where only C_2 is solicited, the drift-like fault in C_2 is successfully detected by $I^2_{q_i 2}\left(x^2\right)$ and $I^3_{q_i 3}\left(x^3\right)$ for all drift speeds (see Figure 4.32 and Figure 4.33), because in the discrete mode q_4 and q_5 the drift indicators are sensitive only to drift like fault in C_2 (see Table 4.4). Same results can be obtained for these indicators in discrete mode q_5.

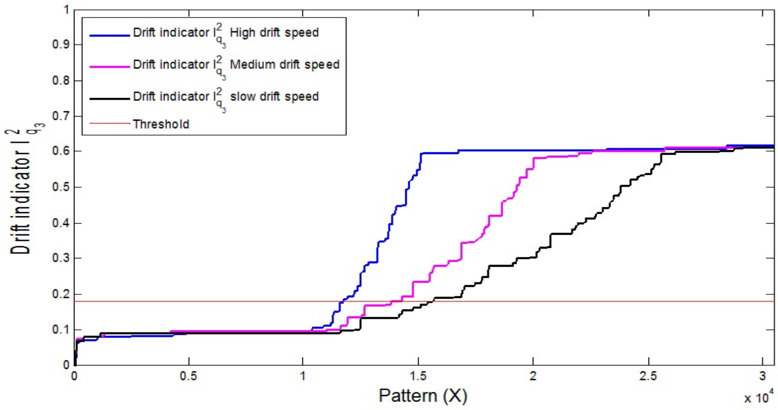

Figure 4.30: Drift indicator $I^2_{q3}\left(x^2\right)$ for attribute 2 and according to each drift speed in the feature space of discrete mode q_3.

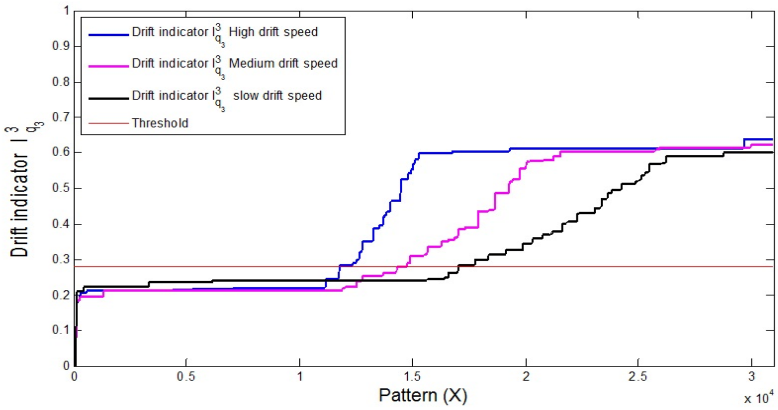

Figure 4.31: Drift indicator $I^3_{q3}\left(x^3\right)$ for attribute 3 and according to each drift speed in the feature space of discrete mode q_3.

Based on these figures, we can see that the drift-like fault in C_2 is successfully detected by both indicators $I^1_{q_i}\left(x^1\right)$ and $I^3_{q_i}\left(x^3\right)$ for all drift speeds (see Figure 4.29 and Figure 4.31). However, it cannot be detected by the indicator $I^2_{q_i}\left(x^2\right)$ (see Figure 4.30) because $I^2_{q_i}\left(x^2\right)$ is sensitive only to drift like fault in C_1 (see Table 4.4).

As it can be seen in Table 4.5, $I^3_{q_i}\left(x^3\right)$, related to the converter output V_S ,detects always the drift after its detection by $I^1_{q_i}\left(x^1\right)$. For this reason in this work, $I^3_{q_i}\left(x^3\right)$ is used to confirm the occurrence of a drift like fault in C_1 .

The drift-like fault in C_2 is detected in early stage before the end of this drift (arriving to the failure mode due to a parametric fault in C_1). As an example, in the case of a drift of slow speed (F3s) (see Table 4.5), the converter reaches

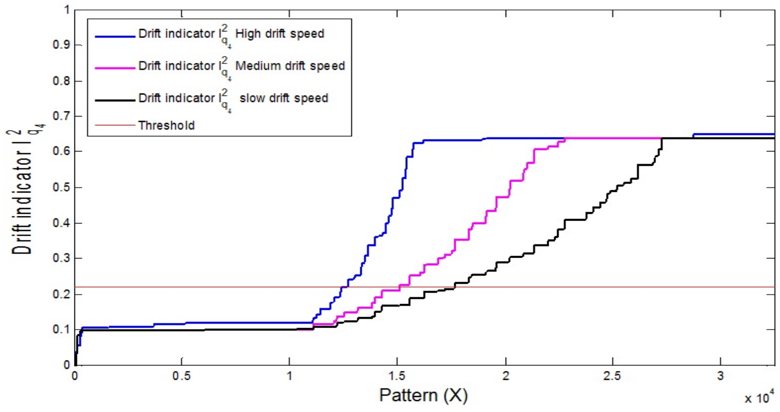

Figure 4.32: Drift indicator $I_{q4}^{2}\left(x^{2}\right)$ for attribute 1 and according to each drift speed in the feature space of discrete mode q_4.

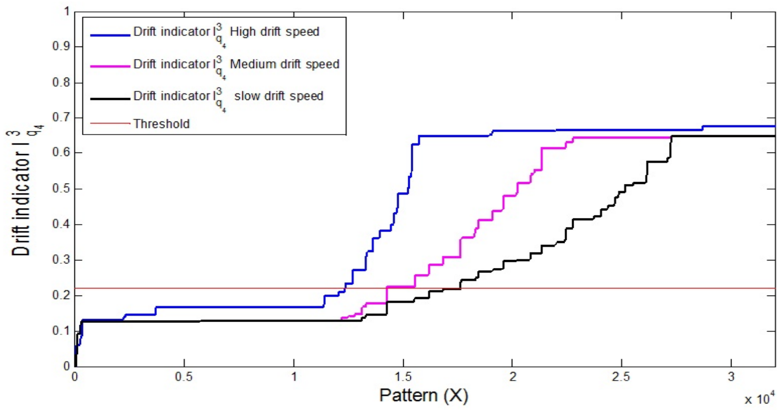

Figure 4.33: Drift indicator $I_{q_4{}^3}^{3}\left(x^{3}\right)$ for attribute 3 and according to each drift speed in the feature space of discrete mode q_4.

the failure mode resulting from a parametric fault in C_1 after 15 seconds of the beginning of the drift (degradation in C_2). In the proposed approach, this drift is detected 2.7 seconds and confirmed 2.77 seconds after its beginning. Therefore, the drift-like fault in C_2 is confirmed 10 seconds before its end. This allows achieving an early fault diagnosis and therefore helps the human operators of supervision to take efficiently the right actions.

4.8.3 Multiple parametric drift-like fault in C_1 and C_2

Figure 4.34 and Figure 4.35 represent, respectively, $V_{C_1,m}$ and $V_{C_2,m}$ of the three cell converter in presence of an abnormal drift in both C_1 and C_2 at the same time. We can see that both $V_{C_1,m}$ and $V_{C_2,m}$ are impacted by the occurrence of the abnormal drift in C_1 and C_2.

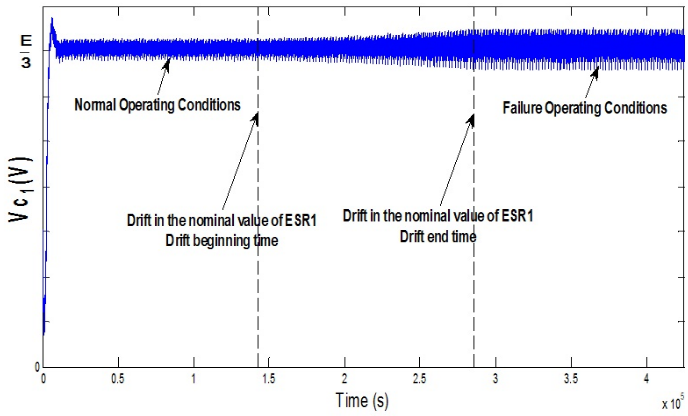

Figure 4.34: Voltage measurement $V_{C_1,m}$ in three cell converter.

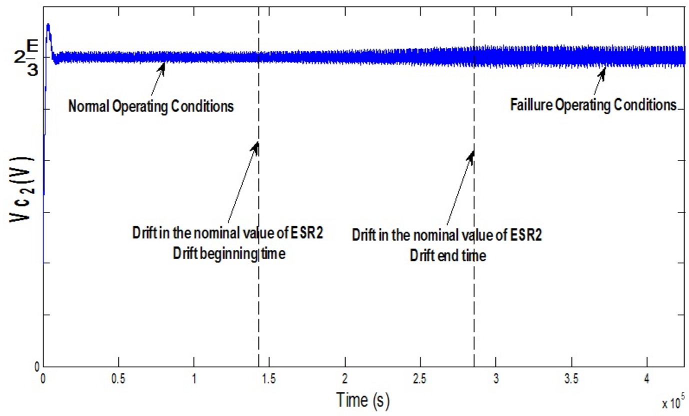

Figure 4.35: Voltage measurement $V_{C_2,m}$ in three cell converter.

The degradations are observed when the converter is in the discrete modes where the dynamics (operating conditions) described by the affected parameters are active. In the discrete modes q_2 and q_7 where only C_1 is solicited (see Table 4.4), the drift-like fault in C_1 is successfully detected by both indicator $I_{q_i}^1\left(x^1\right)$ and $I_{q_i}^3\left(x^3\right)$ for all drift speeds (see Figure 4.36 and Figure 4.37 for the case of discrete mode q_2).

In the discrete modes q_4 and q_5 where only C_2 is solicited, the drift-like fault in C_2 is successfully detected by both indicator $I_{q_i}^2\left(x^2\right)$ and $I_{q_i}^3\left(x^3\right)$ for all drift speeds (see Figure 4.38 and Figure 4.39 for the case of the discrete mode q_4).

In the discrete modes q_3 and q_6 where C_1 and C_2 are solicited, the drift like fault in C_1 and C_2 is successfully detected by all indicators $I_{q_i}^1\left(x^1\right)$, $I_{q_i}^2\left(x^2\right)$ and

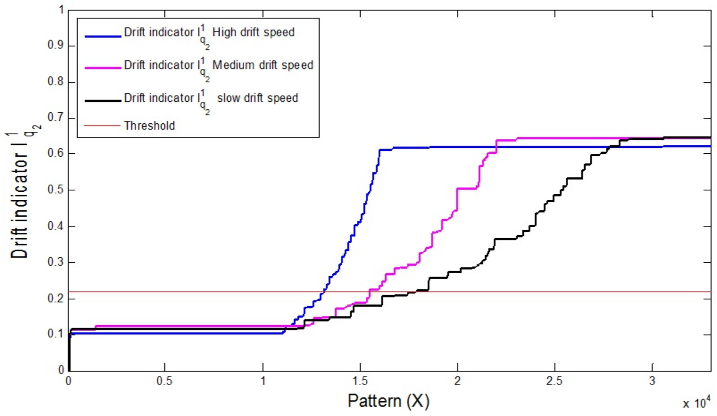

Figure 4.36: Drift indicator $I^1_{q2}\left(x^2\right)$ for attribute 1 and according to each drift speed in the feature space of discrete mode q_2.

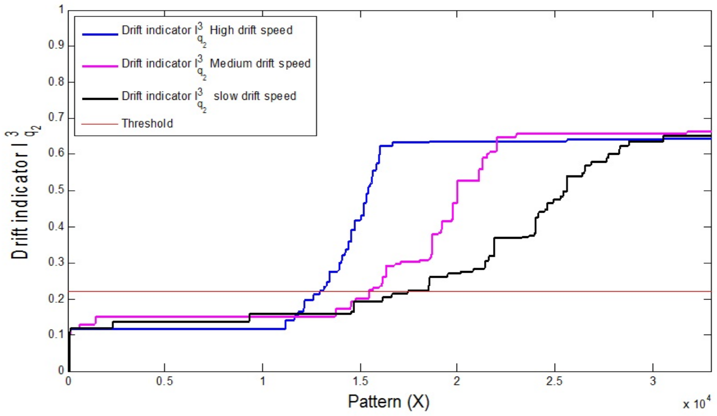

Figure 4.37: Drift indicator $I^3_{q2}\left(x^3\right)$ for attribute 2 and according to each drift speed in the feature space of discrete mode q_2.

$I^3_{qi}\left(x^3\right)$ for all drift speeds (see Figure 4.40,Figure 4.41 and Figure 4.42 for the case of the discrete mode q_3).

Therefore, according to the sequence of discrete modes decided by the controller of the three cell converter in response to the variation of the load conditions, the multiple abnormal drifts in both C_1 and C_2 are successfully detected by both indicator $I^1_{qi}\left(x^1\right)$ and $I^2_{qi}\left(x^2\right)$ confirmed by $I^3_{qi}\left(x^3\right)$ for all drift speeds.

The drift like faults in both C_1 and C_2 are detected in early stage before the end of each of these drifts (arriving to the failure mode due to a multiple parametric faults in C_1 and C_2). As an example, in the case of a drift of slow speed (F9s) (see Table 4.5), the converter reaches the failure mode resulting from a parametric fault inC_1 and C_2 after 15 seconds of the beginning of the drift (degradation in

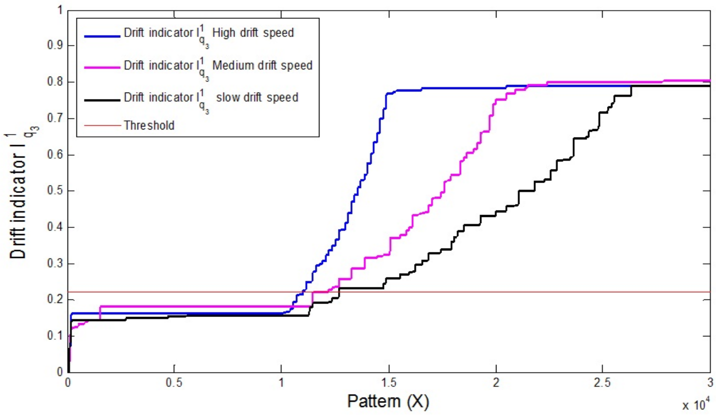

Figure 4.38: Drift indicator $I^1_{q4}\left(x^1\right)$ for attribute 1 and according to each drift speed in the feature space of discrete mode q_4.

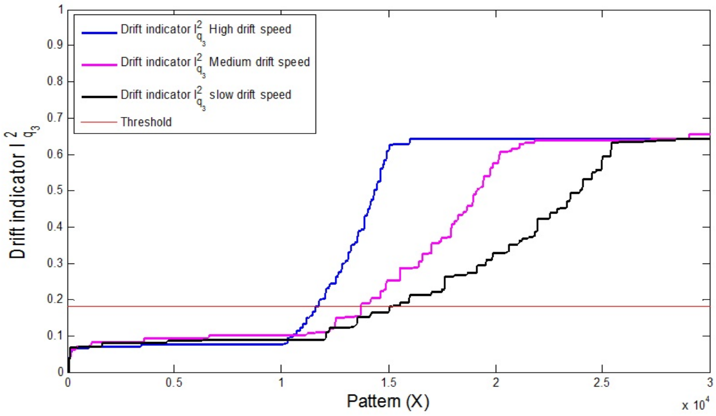

Figure 4.39: Drift indicator $I^2_{q4}\left(x^2\right)$ for attribute 2 and according to each drift speed in the feature space of discrete mode q_4.

C_1 and C_2). In the proposed approach, this drift is detected after 2.33 seconds of the occurrence of both drifts and confirmed at 4.10 seconds after its beginning. Therefore, the drift like faults in C_1 and C_2 are confirmed 11 seconds before its end. This allows to achieve an early fault diagnosis and therefore it helps the human operators of supervision to take efficiently the right actions.

The occurrence of multiple faults was confirmed by $I^3_{qi}\left(x^3\right)$. The confirmation time of multiple faults is less than the one of simple faults, because in the case of multiple faults is V_S impacted by both faults in C_1 and C_2 (see Equation4.3). Therefore, the deviation in V_S from the nominal operation conditions will be increased leading $I^3_{qi}\left(x^3\right)$ to be greater than the confirmation threshold earlier than the case of a simple drift like fault scenario.

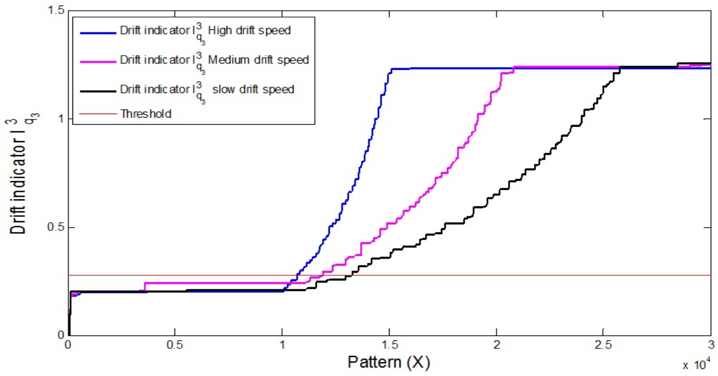

Figure 4.40: Drift indicator $I^1_{q3}\left(x^1\right)$ for attribute 3 and according to each drift speed in the feature space of discrete mode q_3.

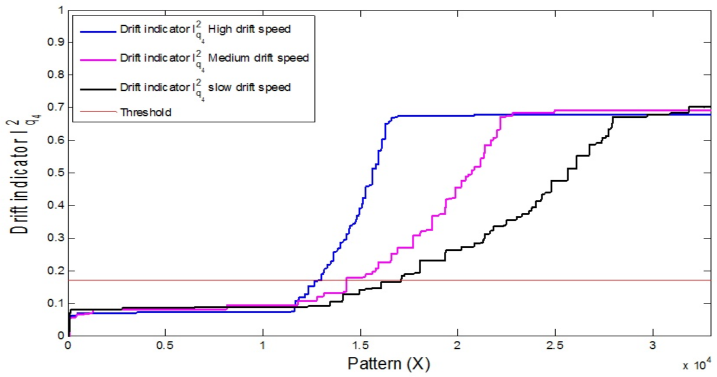

Figure 4.41: Drift indicator $I^2_{q3^2}\left(x^2\right)$ for attribute 1 and according to each drift speed in the feature space of discrete mode q_3.

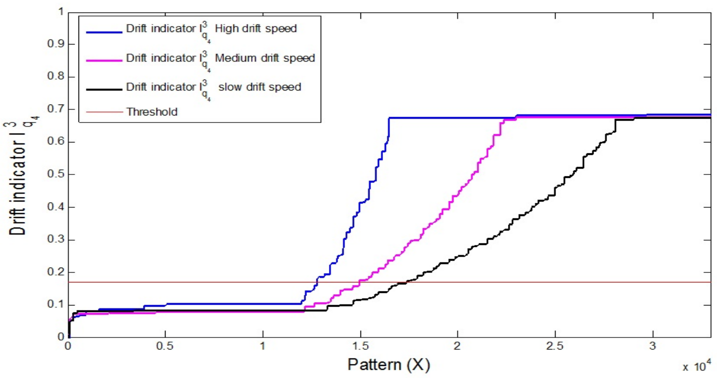

Figure 4.42: Drift indicator $I^3_{q3^3}\left(x^3\right)$ for attribute 3 and according to each drift speed in the feature space of discrete mode q_3.

4.9 Summary

In this chapter a data-mining based approach is proposed in order to build a classifier able to achieve a reliable drift monitoring and early diagnosis of simple and multiple parametric faults that can affect WT converters. This approach considers the converter as a special class of hybrid dynamic systems called Discretely Control Continuous System (DCCS). Therefore, it takes into account the converter continuous dynamics in each discrete mode.

The continuous dynamics are described in a feature space sensitive to normal operating conditions in the corresponding discrete mode. Therefore, the feature space is dynamic in the sense that the classifier selects the discriminant features according to each discrete operating mode. Converter degradation is considered as a continuous drift in the characteristics (gravity center) of the normal classes over time. The proposed approach monitors this change by using a drift indicator for each attribute of the feature space, in order to detect a drift and isolate its origin as soon as possible.

The proposed approach was applied to achieve the drift-like fault diagnosis in the nominal value of the capacitors C_1 and C_2 of a three cell converter using three different drift (decrease) speeds (high, moderate and low). This drift in the nominal value of capacitors C_1 and C_2 simulates degradation due to an abnormal deviation in the nominal value of one capacitor (simple fault scenario) and in both capacitors (multiple faults scenario). The drift indicators have detected successfully the drift for these three speeds in early stage before the end of this drift.

Future work related to the validation of the proposed approach will focus on the drift like fault of other WT critical components as the generator and drive train as well as the use of other indicators to detect drifts of other types or natures. The future work and directions related to the extension of the proposed scheme will be discussed in chapter 5.

Chapter 5

General conclusion and future work

Contents

5.1 Summary of contributions and discussion

Faults occurring in pitch system and power converter of wind turbines (WTs) impact significantly WTs availability to produce electricity and increases their maintenance costs. This is due to the high failure rate and/or the downtime of these two critical WTs components. Therefore, this thesis proposed an advanced automated on-line fault diagnosis approach for pitch system and power converter of WTs.

The diagnosis of faults in pitch system and power converters is a challenging task. This is because:

- Their dynamics are hybrid in the sense that they have several discrete modes (configurations) and in each of the latter, they have a different continuous dynamics. Therefore, modeling efficiently their behavior requires taking into account the discrete and continuous dynamics as well as the interactions between them,

- The consequences of faults impacting the continuous dynamics depend on the discrete mode in which the pitch system or the power converter is. In some discrete modes, the fault consequences may be hiden, compensated or invisible,

- The faults impacting the continuous dynamics of pitch system and power converters are drift-like faults. In the latter, the operation conditions changes progressively (as a drift) from normal to a failure. Achieving an early diagnosis of these faults requires efficient and reliable drift detection tools able to distinguish early between the normal variations due to changes in load or weather conditions and the abnormal drift due to the occurrence of a fault.

To answer these challenges, the work of this thesis is organized as follows. In Chapter 2, the different methods of the literature used to achieve the faults diagnosis of WTs are studied and compared. The goal is to justify the use of machine learning and data mining approaches as an alternative to overcome the complexity and non-linearity of WT dynamics and their non-stationary environments. In these approaches, no need to a mathematical knowledge about the WT dynamics. The model is built by learning using a set of historical data samples representing the WT dynamical behaviors.

However, machine learning and data mining approaches used to achieve the fault diagnosis of WTs suffer from two main drawbacks. Firstly, they require prior data samples about each fault behavior which is very hard to obtain. Secondly, they do not include any mechanism to detect a drift in order to achieve a diagnosis in early stage of the fault development.

Consequently, two contributions are developed in this thesis in order to overcome these two drawbacks. The first contribution is used to achieve the simple and multiple drift-like faults in the pitch system sensors and actuators and is detailed in Chapter 3. It is based on the use of a machine learning and data mining scheme that integrates a mechanism to monitor abnormal drifts in the normal operation conditions. Only data samples representing the latter are considered to be available in advance. They form a restricted area in the feature space called the normal class. When an abnormal drift occurs, the incoming data samples considered to represent the evolving class. Two drift indicators are used in order to detect the evolution (degradations) of the normal operation conditions of pitch system (sensors, actuators).The first drift indicator is based on the use of Euclidean distance between the gravity centers of the normal and evolving classes; while the other drift indicator is based on the use of Mahalanobis distance between the normal class patterns and the gravity center of the evolving class. The interest of these two indicators is that the Mahalanobis indicator is used to detect a drift and the Euclidean indicator to confirm it.

In order to detect and confirm the drift as early as possible, only representative data samples about the abnormal drift are gathered and used to compute the drift indicators. These representative data samples are available when the pitch system is in control mode 2 where the normal and failure behaviors can be separated. In the first control mode, the normal and failure operation conditions cannot be discriminated because of the small pitch angles (wich can not be separated from noises) and the high variability of wind speed or because the actuators are not active (powered on) since the pitch angle is maintained at 0 degree. The proposed approach has been applied to three different speeds of degradation (drift) in pitch system sensors and actuators in the case of simple and multiple drift-like faults. The obtained results showed that the abnormal drift in the sensors and actuators is detected and confirmed 60% before its end.

In Chapter 4, the machine learning and data mining scheme is developed to be able to achieve the simple and multiple drift-like fault diagnosis of power converter.

These faults affect the nominal values of the converter capacitors and lead to reduce its ability to control the flow of electrical energy from the generator. The proposed approach defines a feature space in response to the converter discrete mode. The drift (degradation) indicator is defined for each sensitive feature based on the use of the distance between the gravity centers of normal and evolving classes. When a drift is detected by one indicator (according to one sensitive feature in a discrete mode), this drift can be then confirmed by another drift indicator (according to another sensitive feature). The proposed approach has been applied to the simple and multiple drift like faults in power converter. The obtained results showed that these abnormal drifts were detected and confirmed in early stage of the fault development.

The major differences between the approach developed in Chapter 3 and the one proposed in Chapter 4 can be summarized as follows:

- The way how the WT component changes its continuous dynamic: each approach is applied to a different class of hybrid dynamic systems. In Chapter 3, the pitch system is represented by a discretely externally triggered jumping system; while in Chapter 4, the converter is represented by a discretely controlled continuous system;

- The feature space definition: in Chapter 3, one feature space with the same features is used for the different discrete modes; while in Chapter 4, the feature space is dynamic in the sense that the classifier selects the discriminant features according to each discrete operating mode;

- The drift indicator definition and using: in Chapter 3, two drift detection indicators are used based on Euclidean and Mahalanobis distances and one drift isolation indicator; while one drift indicator based on the Euclidean distance for each feature in a certain discrete mode is used for drift detection and isolation in Chapter 4.

5.2 Future directions

Throughout the development of this work, several extensions are possible in order to enrich the proposed scheme in order to achieve the supervision of WTs.These future directions are summarized as follows:

5.2.1 Fault prognosis and its interaction with the drift-like fault diagnosis

Early drift-like fault diagnosis is necessary to determine as fast as possible the components that must be replaced or repaired. The more the diagnosis is early, the more the maintenance actions are efficient. For instance, let suppose that we have two classes A and B. Let suppose that class A represents the normal operating conditions (e.g., capacitor is working normally) while class B is a failure state (capacitor value is outside of its nominal interval). When the drift occurs, the system's state moves away from class A and approaches class B. This means that the system needs to be repaired, adjusted or reconfigured. Let us take the example of Figure 5.1 showing a drift-like fault evolving case. If one catches the fault at 5 percent severity, one needs to replace only the component. If the fault is not caught until 10 percent severity, the subsystem must be replaced, and at failure, the entire system must be replaced [33].The prognosis model must give an estimate of the Remaining Useful Life (RUL). The latter indicates the remaining time for the component before being unable to accomplish its mission.This information is important in order to schedule the maintenance actions that optimize the availability and maintenance costs.

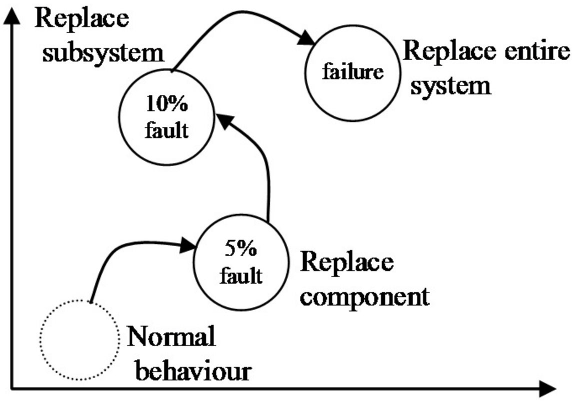

Figure 5.1: Evolving of a fault and its required maintenance actions.

However, the drift-like fault diagnosis does not provide any information about the time of the RUL. The estimating of the RUL can be achieved using fault prognosis techniques. Fault prognosis has a sense in the case of drift-like faults where the component performance starts to decrease over time until reaching an unacceptable level entailing declaring a failure.

A big challenge concerning prognosis modeling is the need for a reactive model that always takes into consideration current operating conditions which is the case in the proposed approach in this thesis. This means that the prognosis model that needs to be defined is an on-line prognosis model. The RUL estimation should be based on the actual dynamics of the drift. Thus, the knowledge of these dynamics should be updated in an on-line manner.The proposed scheme in this thesis implemented all the steps necessary before being able to obtain a prognosis result, (i,e, RUL). In order to compute the RUL on a component level, it is necessary to answer these basic questions:

1. Fault detection: Is there any degradation?
2. Fault isolation: Which component is degrading?
3. Fault Identification: What is the failure mode behind this degradation?
4. What is the severity of the degradation?

5. How estimate the dynamics of degradation (evolving class)?

6. RUL prediction: Determining the time to failure?

The first three questions related to the diagnosis module are already accomplished in the proposed scheme in this thesis. Indeed when the drift is detected and confirmed, the proposed approach in this thesis provides the element (sensor, actuator, capacitor) generating this drift.The answer to the question 4 can be achieved by determining the end of the drift. This can be achieved by observing the stabilization of the characteristics of the incoming data samples. In this case, these new data samples represent the failure mode. In other words, the evolving class stops to evolve and remains in its current region in the feature space. The answer for the two last questions can be done by storing all the data samples between the detection and the end of drift (patterns between class A and class B). These data samples represent the degradation dynamics and can be used to estimate the health or degradation indicator as well as the RUL by applying the regression techniques. For instance we can cite regression techniques inspired from the time series analysis domain, where a degradation indicator is considered as Auto- Regressive (AR) models [38], Moving Average (MA) models [11], a combination of these two models (ARMA models) [46],[75],Auto Regressive Integrated Moving Average (ARIMA) [75],[46] etc.

After the end of drift, the data samples representing the new failure mode (class B in Figure 5.1) will be integrated to the knowledge base of the proposed scheme by updating the classifier structure. Therefore, a new class will be learnt and the decision boundaries of the classifier will be update to include this new class in the feature space. This shows clearly the evolving and adaptive property of the proposed scheme as well as the strong relationship between the diagnosis task in this scheme and the prognosis task.

5.2.2 Fault tolerant control and its interaction with the drift-like fault diagnosis

The second future work is the enrichment of the proposed scheme by the integration of the Fault-Tolerant Control (FTC) module. Drift-like faults in WTs typically result in a slow abnormal decrease (degradation) in the generated power. The goal of FTC is to inverse the tendency of degradation to move towards a failure state. FTC accommodates its control actions or strategy in order to reduce the fault consequences on the system performance (e.g., WT energy production). The new control actions aim at modifying the dynamics of a WT and forcing it or its affected components to return to the normal operating conditions. This allows a reduction in the cost of losses in energy due to the fault occurrence. The most early the fault is detected the most efficient the fault accommodation is.

FTC systems are divided into two distinct classes [115],passive and active. Passive FTC systems [116], [85], introduce fault tolerance into a controllers. The latter are designed to be robust against a set of predefined faults [110]. Therefore there is no need for fault diagnosis.These robust controllersare designed off-line and do not adapt to the anticipated faults on-line. However since in this thesisthe faults are supposed to be unknown in advance, the FTC techniques must be adaptive allowing to enrich on-line the required control strategy to accommodate the new detected faults. Active FTC systems,in contrary to passive FTC systems, can adapt on-line to the occurrence of new faults. This on-line adaptation allows active FTC systems to deal with more faults and generally achieve better performance than passive FTC systems. An active FTC algorithm that has the ability to adapt to unanticipated fault conditions is therefore very desirable. Active FTC systems react to faults actively by reconfiguring control actions and by doing so the system (WT) can maintain an acceptable performance over time. To achieve that, the proposed scheme in this thesis should link the fault diagnosis to the active FTC accommodation technique.

Active FTC methods, require availability as soon as possible of detailed fault information (fault localization, its amplitude, its time occurrence, etc.) to accommodate faults. This information is then used by the adaptive controller to accommodate the faults that have occurred. In order to integrate a strategy of Active FTC in the proposed scheme, the interpretation task will be enriched in order to achieve the fault estimation (determine the amplitude of the diagnosed fault, its time occurrence, etc.) and importance evaluation (severity, priority, impact on the WTs availability etc.). Then, this information is provided to the active FTC module in order to define the most suitable control actions or strategy to be used to optimize the availability and cost maintenance in presence of this fault development (degradation).

5.2.3 Maintenance module and its interaction with the drift-like fault diagnosis

The costs of operation and maintenance of WTs are a significant part of the overall cost of WTs. The challenge for human operators of supervision is to achieve an efficient maintenance operation. Corrective or scheduled maintenance are widely implemented in the industry but it may not be optimized. More the fault is detected and isolated early, more the maintenance operation can be optimized. For this reason, the proposed scheme in this thesis achieved an early drift-like fault detection and isolation in order to diagnose the fault in its early stage.

During the maintenance process, WTs are required to be shutdown. This affects negatively the production. However, the longer the maintenance operations are, the more costly they are. Therefore, the maintenance process duration should be reduced as much as possible. Indeed, fault diagnosis is needed to determine precisely the component that must be reconditioned. The more the diagnosis is precise, the more the maintenance actions are effective. Furthermore, the maintenance can be expensive in emergency situations when equipment is suddenly damaged and the WT can no longer perform its function. In this case, maintenance actions should be done rapidly to get the system working. These actions are more costly because they were not expected. Thus, to avoid the occurrence of this kind of situations, preventive maintenance can be used by anticipating and correcting the failure of equipment before occurrence of excessive damage.

One of the most relevant preventive maintenance strategies is the Condition-Based Maintenance (CBM). The major advantage of CBM among the other approaches is the ability to incorporate a prognosis module [14]. Indeed, a prognosis module can provide a worthy solution to minimize unexpected situations when it is correctly connected to a diagnosis module. The prognosis model improves the planning for maintenance actions by estimating the Remaining Useful Life (RUL) of a physical asset. Therefore, it is interesting to integrate a CBM module into the developed adaptive learning scheme in this thesis. The maintenance actions (plan) can be defined using the drift indicators and the prognosis results (RUL) as inputs.

By integrating the prognosis, the active FTC and the CBM modules into the developed adaptive learning scheme in this thesis, the availability and safety of WTs will be maximized and their maintenance and exploitation costs will be minimized. Figure 5.2 shows the global adaptive learning scheme for the supervision of WTs.

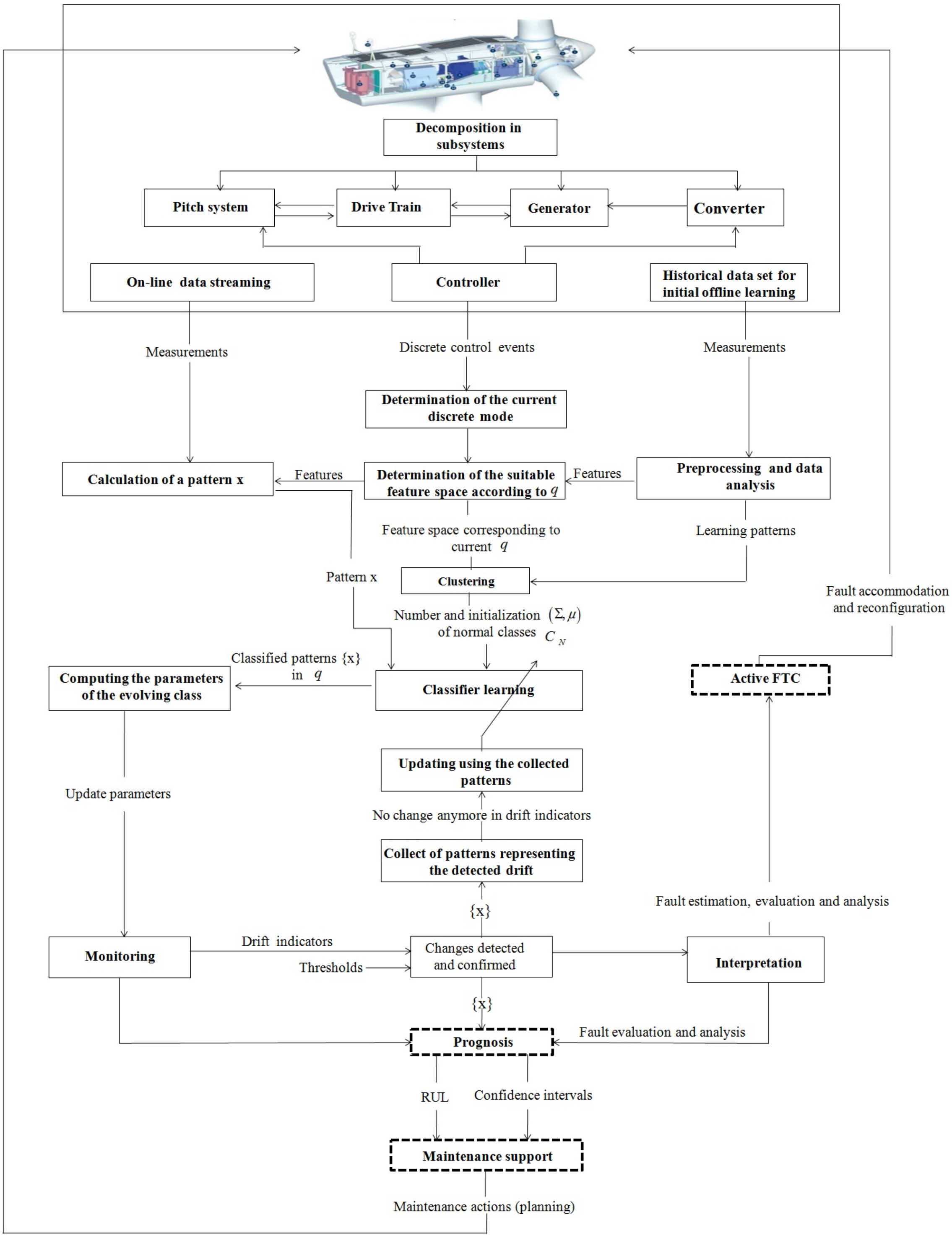

Figure 5.2: Global scheme for wind turbine supervision.

Bibliography

[1] N. Nait-Said A. Dahbi, M.Hachemi and M.S. Nait-Said. Realization and control of a wind turbine connected to the grid by using pmsg. *Energy Conversion and Management*, 84:346–353, 2014. (Cited on page 108.)

[2] V. Akila and G.Zayaraz. A brief survey on concept drift. In *Intelligent Computing, Communication and Devices*, pages 293–302. Springer, 2015. (Cited on pages 49 and 50.)

[3] Y. Amirata, M.E.H. Benbouzid, E. Al-Ahmara, B. Bensakerb, and S. Turria. A brief status on condition monitoring and fault diagnosis in wind energy conversion systems. *Renewable and Sustainable Energy Reviews*, 13:26292636, 2009. (Cited on page 106.)

[4] A.Tsymbal and S. Puuronen. Bagging and boosting with dynamic integration of classifiers. *Principles of Data Mining and Knowledge Discovery*, pages 195–206, 2000. (Cited on page 88.)

[5] B.Abichou, D. Florez, M. Sayed-Mouchaweh, H.Toubakh, B. Francois, and N. Girard. Fault diagnosis methods for wind turbines health monitoring: a review. (Cited on page 37.)

[6] M. Baena-Garc, J. del Campo-Avila, R.Fidalgo, A. Bifet, R. Gavalda, and R. Morales-Bueno. Early drift detection method. 2006. (Cited on page 55.)

[7] J.C. Bezdek. *Pattern recognition with fuzzy objective function algorithms*. Springer Science & Business Media, 1981. (Cited on page 46.)

[8] C. Bindi. *Automated on-line fault prognosis for wind turbine monitoring using SCADA data*. Phd, Durham University, 2014. (Cited on page 46.)

[9] K. Bouibed, L.Seddiki, K. Guelton, and H. Akdag. Actuator and sensor fault detection and isolation of an actuated seat via nonlinear multi-observers. *Systems Science & Control Engineering: An Open Access Journal*, 2(1):150–160, 2014. (Cited on page 32.)

[10] M.S. Branicky, V.S. Borkar, and S.K. Mitter. A unified framework for hybrid control: model and optimal control theory. *Automatic Control, IEEE Transactions on*, 43(1):31–45, Jan 1998. (Cited on pages 30 and 31.)

[11] P.J. Brockwell and R.A.Davis. *Introduction to time series and forecasting*. Springer Science & Business Media, 2006. (Cited on page 145.)

[12] M. Roveri. C. Alippi, G. Boracchi. Change detection tests using the ici rule. In *Neural Networks (IJCNN), The 2010 International Joint Conference on*, pages 1–7. IEEE, 2010. (Cited on page 88.)

[13] Y.J. Chain and A.K. Agrawal. Robustness studies of sensor faults and noises for semi-active control strategies using large-scale magnetorheological dampers. *Journal of Vibration and Control*, page 1077546314535947, 2014. (Cited on pages 32 and 33.)

[14] A. Chammas. *Drift Detection and Characterization for Supervision, Diagnosis and Prognosis of Dynamical Systems*. Phd, Lille1 University, 2014. (Cited on pages 33, 52 and 147.)

[15] W. Chen, S. X. Ding, A. Haghani, A. Naik, A. Q. Khan, and S. Yin. Observer-based fdi schemes for wind turbine benchmark. pages 7073–7078, 2011. (Cited on pages 38, 41 and 43.)

[16] D. Cieslak and N. Chawla. A framework for monitoring classifiers performance: when and why failure occurs? *Knowledge and Information Systems*, 18(1):83–108, 2009. (Cited on page 88.)

[17] D.A. Cieslak and N.V. Chawla. A framework for monitoring classifiers performance: when and why failure occurs? *Knowledge and Information Systems*, 18(1):83–108, 2009. (Cited on page 55.)

[18] B. S. J. Costa, P. P. Angelov, and L. A. Guedes. Real-time fault detection using recursive density estimation. *Journal of Control, Automation and Electrical Systems*, 25:428–437, 2011. (Cited on page 120.)

[19] B. Pfahringer D. Ienco, A. Bifet and P. Poncelet. Change detection in categorical evolving data streams. In *Proceedings of the 29th Annual ACM Symposium on Applied Computing*, pages 792–797. ACM, 2014. (Cited on page 59.)

[20] M. J. Daigle. *A qualitative event-based approach to fault diagnosis of hybrid systems*. Phd, Vanderbilt University, 2008. (Cited on page 34.)

[21] M. Datar, A. Gionis, P. Indyk, and R. Motwani. Maintaining stream statistics over sliding windows. *SIAM Journal on Computing*, 31(6):1794–1813, 2002. (Cited on page 55.)

[22] M. Defoort, M. Djemai, T. Floquet, and W. Perruquettii. Robust finite time observer design for multicellular converters. *International Journal of Systems Science*, 42:18591868, 2011. (Cited on pages 108, 109 and 111.)

[23] A.J. Van der Schaft. *An Introduction to Hybrid Dynamical Systems*. Springer, address = Amsterdam, 1999. (Cited on pages 28, 30, 31 and 103.)

[24] G. Ruggero D.Ienco, R. Pensa and R. Meo. From context to distance: Learning dissimilarity for categorical data clustering. *ACM Transactions on Knowledge Discovery from Data (TKDD)*, 6(1):1, 2012. (Cited on page 59.)

[25] G. Ditzler and R. Polikar. Hellinger distance based drift detection for nonstationary environments. In *Computational Intelligence in Dynamic and Uncertain Environments (CIDUE), 2011 IEEE Symposium on*, pages 41–48. IEEE, 2011. (Cited on pages 55 and 88.)

[26] D.Kolev, P.P. Angelov, G. Markarian, M.Suvorov, and S.Lysanov. Arfa: Automated real-time flight data analysis using evolving clustering, classifiers and recursive density estimation. In *Evolving and Adaptive Intelligent Systems (EAIS), 2013 IEEE Conference on*, pages 91–97, April 2013. (Cited on pages 78 and 120.)

[27] K. Elkington. *The Dynamic Impact of Large Wind Farmson Power System Stability*. Phd, KTH School of Electrical Engineering, 2012. (Cited on pages 26, 27 and 108.)

[28] M. Entezami. *Novel operational condition monitoring techniques for wind turbine brake systems*. PhD thesis, University of Birmingham, Year = 2013, Type = PhD,. (Cited on pages 15 and 23.)

[29] L. Bai X. Zhao F. Cao, J. Liang and C. Dang. A framework for clustering categorical time-evolving data. *Fuzzy Systems, IEEE Transactions on*, 18(5):872–882, 2010. (Cited on page 59.)

[30] J. M. P. Prez M. Papaelias F. P. G. Mrquez, A. M. Tobias. Condition monitoring of wind turbines: Techniques and methods. *Renewable Energy*, 46:169–178, 2012. (Cited on pages 16 and 37.)

[31] Y. Feng, Y. Qiu, H. Long C. J. Crabtree, and P. J. Tavner. Use of scada and cms signals for failure detection and diagnosis of a wind turbine gearbox, 2009. (Cited on page 42.)

[32] P.M. Frank. Fault diagnosis in dynamic systems using analytical and knowledge-based redundancy: A survey and some new results. *Automatica*, 26(3):459–474, 1990. (Cited on pages 39 and 41.)

[33] M. Roemer A. Hess B. Wu G. Vachtsevanos, F. L. Lewis. *Intelligent Fault Diagnosis and Prognosis for Engineering Systems*. Wiley, united states, 2012. (Cited on pages 124 and 144.)

[34] J. Gama and G. Castillo. Learning with local drift detection. In *Advanced Data Mining and Applications*, pages 42–55. Springer, 2006. (Cited on page 55.)

[35] G. Gert and P. Tomaso. Incremental and decremental support vector machine learning, 2000. (Cited on page 55.)

[36] J. L. Godwin and P. Matthews. Classification and detection of wind turbine pitch faults through scada data analysis. *Systems Engineering Procedia*, 4:1–11, 2013. (Cited on page 44.)

[37] R. Hallouzi. *Multiple-model based diagnosis for adaptive fault-tolerant control.* TU Delft, Delft University of Technology, 2008. (Cited on pages 38 and 41.)

[38] J.D. Hamilton. *Time series analysis*, volume 2. Princeton university press Princeton, 1994. (Cited on page 145.)

[39] M.S. Chen H.L. Chen and S.C. Lin. Catching the trend: A framework for clustering concept-drifting categorical data. *Knowledge and Data Engineering, IEEE Transactions on*, 21(5):652–665, 2009. (Cited on page 59.)

[40] T. R. Hoens, N. V. Chawla , and R. Polikar. Heuristic updatable weighted random subspaces for non-stationary environments. In *Data Mining (ICDM), 2011 IEEE 11th International Conference on*, pages 241–250. IEEE, 2011. (Cited on pages 55 and 88.)

[41] H.Toubakh and M.Sayed-Mouchaweh. Hybrid dynamic classifier for drift-like fault diagnosis in a class of hybrid dynamic systems: Application to wind turbine converters. *Neurocomputing*, 2015. (Cited on pages 16 and 116.)

[42] H.Toubakh, M. Sayed-Mouchaweh, A. Fleury, and J. Boonaert. Hybrid dynamic data mining scheme for drift-like fault diagnosis in multicellular converters. In *Technological Advances in Electrical, Electronics and Computer Engineering (TAEECE), 2015 Third International Conference on*, pages 56–61. IEEE, 2015. (Cited on page 125.)

[43] R. Isermann. Process fault detection based on modeling and estimation methodsa survey. *Automatica*, 20(4):387–404, 1984. (Cited on page 39.)

[44] R. Isermann. *Fault-diagnosis systems*. Springer, address = Germany, 2006. (Cited on page 34.)

[45] A. Bifet M. Pechenizkiy J. a. Gama, I. Zliobaite and A. Bouchachia. A survey on concept drift adaptation. *ACM Computing Surveys (CSUR)*, 46(4):44, 2014. (Cited on pages 49, 50, 52 and 55.)

[46] M. Koc J. Yan and J. Lee. A prognostic algorithm for machine performance assessment and its application. *Production Planning & Control*, 15(8):796–801, 2004. (Cited on page 145.)

[47] K.S.A. Jardine and D. Banjevic D. Lin. A review on machinery diagnostics and prognostics implementing condition-based maintenance. *Mechanical Systems and Signal Processing 20*, 20:14831510, 2006. (Cited on page 42.)

[48] I. Khamassi and M. Sayed-Mouchaweh. Drift detection and monitoring in non-stationary environments. In *Evolving and Adaptive Intelligent Systems (EAIS), 2014 IEEE Conference on*, pages 1–6, June 2014. (Cited on page 55.)

[49] I. Khamassi, M. Sayed-Mouchaweh, M. Hammami, and K. Ghdira. Self-adaptive windowing approach for handling complex concept drift. *Cognitive Computation*, pages 1–19. (Cited on pages 49, 50 and 52.)

[50] K. Kim, G. Parthasarathy, O. Uluyol, and W. Foslien. Use of scada data for failure detection in wind turbines. page 19, 2011. (Cited on pages 46 and 106.)

[51] Y. Ko and K. Lee. Fault diagnosis of a voltage-fed pwm inverter for a three-parallel power conversion system in a wind turbine. *Journal of Power Electronics*, 10(6):686 693, 2010. (Cited on page 107.)

[52] L. I. Kuncheva. Using control charts for detecting concept change in streaming data. *Bangor University*, 2009. (Cited on page 88.)

[53] L.I. Kuncheva. Classifier ensembles for changing environments. In *Multiple classifier systems*, pages 1–15. Springer, 2004. (Cited on page 55.)

[54] A. Kusiak. Condition monitoring of wind turbines: Techniques and methods. *International Journal of Production Research*, 44,:4175–4191, 2006. (Cited on page 24.)

[55] A. Kusiak and W. Li. The prediction and diagnosis of wind turbine faults. *Renewable Energy*, 36:16–23, 2011. (Cited on pages 16, 46 and 65.)

[56] A. Kusiak and A. Verma. A data-mining approach to monitoring wind turbines. *Sustainable Energy, IEEE Transactions on*, 3(1):150–157, Jan 2012. (Cited on page 65.)

[57] N. Laouti, S. Othman, M. Alamir, and N.Sheibat-Othman. Combination of model-based observer and support vector machines for fault detection of wind turbines. *International Journal of Automation and Computing*, 11:274–287, 2015. (Cited on pages 16 and 40.)

[58] B. C. P. Lau, E. W. M. Ma, , and M. Pecht. Review of offshore wind turbine failures and fault prognostic methods. *Prognostics and System Health Management Conference*, pages 231–236, 2012. (Cited on page 23.)

[59] M. Lazarescu, S. Venkatesh, and H. Bui. Using multiple windows to track concept drift. *Intelligent data analysis*, 8(1):29–60, 2004. (Cited on page 50.)

[60] Stéphane Lecoeuche and Christophe Lurette. Auto-adaptive and dynamical clustering neural network. In *Artificial Neural Networks and Neural Information ProcessingICANN/ICONIP 2003*, pages 350–358. Springer, 2003. (Cited on page 46.)

[61] R. Lichtenwalter and N. V. Chawla. Adaptive methods for classification in arbitrarily imbalanced and drifting data streams. In *New Frontiers in Applied Data Mining*, pages 53–75. Springer, 2010. (Cited on pages 55 and 88.)

[62] I. liobait. Combining time and space similarity for small size learning under concept drift. In *Foundations of Intelligent Systems*, pages 412–421. Springer, 2009. (Cited on page 88.)

[63] H. Louajri. *Centralized and Decentralized Fault Diagnosis of a Class of Hybrid Dynamic Systems: Application to Three Cell Converter*. Phd, Lille1 University, 2015. (Cited on pages 31 and 33.)

[64] B. Lu, Y. Li, X. Wu, and Z. Yang. A review of recent advances in wind turbine condition monitoring and fault diagnosis. In *Power Electronics and Machines in Wind Applications, 2009. PEMWA 2009. IEEE*, pages 1–7, June 2009. (Cited on page 42.)

[65] B. Lu and S.K. Sharma. A literature review of igbt fault diagnostic and protection methods for power inverters. *Industry Applications, IEEE Transactions on*, 45(5):1770–1777, Sept 2009. (Cited on page 47.)

[66] X. Luo and X. Huang. Fault diagnosis of wind turbine based on elmd and fcm. *The Open Mechanical Engineering Journal*, 8:716–720, year= 2014,. (Cited on pages 16 and 46.)

[67] L.L. Minku, A.P.Whit, and Y.Xin. The impact of diversity on online ensemble learning in the presence of concept drift. *Knowledge and Data Engineering, IEEE Transactions on*, 22(5):730–742, May 2010. (Cited on pages 50, 51 and 53.)

[68] M. Sayed Mouchaweh. Diagnostic des systmes dynamiques hybrides. *Intelligent data analysis*, pages 1–15, 2015. (Cited on page 31.)

[69] M.Traore, E. Duviella, and S. Lecoeuche. Comparison of two prognosis methods based on neuro fuzzy inference system and clustering neural network. In *7th IFAC Symposium on Fault Detection, Supervision and Safety of Technical Processes*, pages 91–97, 2009. (Cited on pages 78, 120 and 121.)

[70] S. Muthukrishnan, V.D.B. Eric, and Y.Wu. Sequential change detection on data streams. In *Data Mining Workshops, 2007. ICDM Workshops 2007. Seventh IEEE International Conference on*, pages 551–550. IEEE, 2007. (Cited on page 55.)

[71] S. Othman N. Laouti, N. Sheibat-Othman. Support vector machines for fault detection in wind turbines. pages 7067–7072, 2011. (Cited on page 46.)

[72] P. F. Odgaard and J. Stoustrup. Unknown input observer based scheme for detecting faults in a wind turbine converter. pages 161–166, 2009. (Cited on pages 38 and 41.)

[73] P. F. Odgaard, J. Stoustrup, and M. Kinnaert. Fault tolerant control of wind turbines a benchmark model. *7th IFAC Symposium on Fault Detection, Supervision and Safety of Technical Processes*, pages 155–160, 2009. (Cited on pages 19 and 35.)

[74] P.F. Odgaard, J. Stoustrup, and M. Kinnaert. Fault-tolerant control of wind turbines: A benchmark model. *Control Systems Technology, IEEE Transactions on*, 21(4):1168–1182, July 2013. (Cited on pages 16, 39, 68, 69 and 70.)

[75] H.T. Pham and B.S. Yang. Estimation and forecasting of machine health condition using arma/garch model. *Mechanical Systems and Signal Processing*, 24(2):546–558, 2010. (Cited on page 145.)

[76] C. Pinto and J. Gama. Incremental discretization, application to data with concept drift. In *Proceedings of the 2007 ACM symposium on Applied computing*, pages 467–468. ACM, 2007. (Cited on page 55.)

[77] J. Gama P.Kosina and R.Sebastiao. Drift severity metric. In *ECAI*, pages 1119–1120, 2010. (Cited on page 52.)

[78] R.S. Barros D.C. Vieira P.M. Gonalves, S.G. de Carvalho. A comparative study on concept drift detectors. *Expert Systems with Applications*, 41(18):8144–8156, 2014. (Cited on page 88.)

[79] R. Precup, P. Angelov, B. S. Jales Costa, and M. Sayed-Mouchaweh. An overview on fault diagnosis and nature-inspired optimal control of industrial process applications. *Computers in Industry*, 2015. (Cited on pages 16, 37 and 45.)

[80] A. Purarjomandlangrudi. Application of machine learning technique in wind turbine fault diagnosis. (Cited on page 45.)

[81] A. Rasit. Artificial neural networks applications in wind energy systems: a review. *Renewable and Sustainable Energy Reviews*, 49:534562, 2015. (Cited on page 46.)

[82] I. Renz. Adaptive information filtering: Learning in the presence of concept drifts. (Cited on page 55.)

[83] C. M. E. Robinson, E. S. Taylor, A. J. T. Morrison, and E. D. Sanderson. *Study and development of a methodology for the estimation of the risk and harm to persons from wind turbines*. RR968 Research Report, London, 2013. (Cited on page 24.)

[84] S. Russell and P. Norvig. Artificial intelligence: a modern approach. 1995. (Cited on page 43.)

[85] M. Witczak S. de Oca, V. Puig and L. Dziekan. Fault-tolerant control strategy for actuator faults using lpv techniques: Application to a two degree of freedom helicopter. *International Journal of Applied Mathematics and Computer Science*, 22(1):161–171, 2012. (Cited on page 145.)

[86] M. Sayed-Mouchaweh and E. Lughofer. *Learning in Non-Stationary Environments*. Springer, New York, 2012. (Cited on pages 52 and 120.)

[87] W.T. Scherer and C.C. White III. A survey of expert systems for equipment maintenance and diagnostics. In *Knowledge-Based System Diagnosis, Supervision, and Control*, pages 285–300. Springer, 1989. (Cited on page 43.)

[88] 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*, 25:18491875, 2011. (Cited on page 46.)

[89] A. Shaker and E. Lughofer. Self-adaptive and local strategies for a smooth treatment of drifts in data streams. *Evolving Systems*, 5(4):239–257, 2014. (Cited on page 88.)

[90] S. Simani, P. Castaldi, and A. Tilli. Datadriven modelling of a wind turbine benchmark for fault diagnosis application. *Transaction on control and Mechanical Systems*, 1(7):278–289, 2012. (Cited on page 107.)

[91] S. Simani, S. Farsoni, and P. Castaldi. Residual generator fuzzy identification for wind turbine benchmark fault diagnosis. *machines*, 2:275–298, 2014. (Cited on pages 38 and 40.)

[92] S.J.Yoo. Fault detection and accommodation of a class of nonlinear systems with unknown multiple time-delayed faults. *Automatica*, 50(1):255–261, 2014. (Cited on page 33.)

[93] P. Sobhani and H. Beigy. *New drift detection method for data streams*. Springer, 2011. (Cited on page 88.)

[94] R. S. Sutton and A. G. Barto. *Introduction to Reinforcement Learning*. MIT Press, Cambridge, 1998. (Cited on page 45.)

[95] S. Tabatabaeipour, P.F. Odgaard, T. Bak, and J. Stoustrup. Fault detection of wind turbines with uncertain parameters: a set-membership approach. *Energies*, 5(7):2424–2448, 2012. (Cited on page 16.)

[96] H. Toubakh and M. Sayed-Mouchaweh. Hybrid dynamic data-driven approach for drift-like fault detection in wind turbines. *Evolving Systems*, 6:115129, 2014. (Cited on pages 16, 58 and 77.)

[97] H. Toubakh, M. Sayed-Mouchaweh, and E.Duviella. Advanced pattern recognition approach for fault diagnosis of wind turbines. In *Machine Learning and Applications (ICMLA), 2013 12th International Conference on*, volume 2, pages 368–373, Dec 2013. (Cited on page 46.)

[98] H. Toubakhi and M. Sayed-Mouchaweh. Advanced data mining approach for wind turbines fault prediction. In *Proceedings of second European conference of the prognostics and health management society*, volume 5, pages 288–296, 2014. (Cited on page 88.)

[99] D.H. Tran. Automated change detection and reactive clustering in multivariate streaming data. *arXiv preprint arXiv:1311.0505*, 2013. (Cited on pages 58 and 88.)

[100] M. Traore, A. Chammas, and E. Duviella. Supervision and prognosis architecture based on dynamical classification method for the predictive maintenance of dynamical evolving systems. *Reliability Engineering & System Safety*, 136:120–131, 2015. (Cited on pages 58 and 88.)

[101] C. Srinivas T.R.K. Mada and K.S. Reddy. Doubly-fed induction generator for variable speed wind energy conversion systems-modeling and simulation. 2012. (Cited on page 108.)

[102] C. Tsai, C. Hsieh, and S. Huang. Enhancement of damage-detection of wind turbine blades via cwt-based approaches. *Energy Conversion, IEEE Transactions on*, 21(3):776–781, Sept 2006. (Cited on page 42.)

[103] V. Venkatasubramanian, R. Rengaswamy, S. N. Kavuri, and K. Yin. A review of process fault detection and diagnosis part iii: Process history based methods. *Computers and Chemical Engineering*, 27:327–346, 2003. (Cited on page 45.)

[104] A. Verma and A. Kusiak. Fault monitoring of wind turbine generator brushes: A data-mining approach. *Journal of Solar Energy Engineering*, 134:1–9, 2012. (Cited on page 46.)

[105] A. P. Verma. *Performance monitoring of wind turbines: a data-mining approach*. Phd, University of Iowa, 2012. (Cited on page 15.)

[106] V.N.Vapnik. An overview of statistical learning theory. *Neural Networks, IEEE Transactions on*, 10(5):988–999, Sep 1999. (Cited on page 45.)

[107] P. Vorburger and A. Bernstein. Entropy-based detection of real and virtual concept shifts. Technical report, Working Paper-University of Zurich, Department of Informatics, 2006. (Cited on pages 55 and 88.)

[108] R.C. De Vries. An automated methodology for generating a fault tree. *Reliability, IEEE Transactions on*, 39(1):76–86, 1990. (Cited on page 43.)

[109] S. Wisdom. The research of data mining in hhm technology based on association rule. *Computers and Chemical Engineering*, 2:1–9, 2012. (Cited on page 45.)

[110] M. Witczak, J. Korbicz P. Witczak, and C. Aubrun. Robust and efficient predictive ftc: Application to wind turbines. In *Control and Fault-Tolerant Systems (SysTol), 2013 Conference on*, pages 371–376. IEEE, 2013. (Cited on page 145.)

[111] Q. Xu, H. Yan, B. Jiang, D. Zhou, and Y. Zhang. Fault tolerant formations control of uavs subject to permanent and intermittent faults. *Journal of Intelligent & Robotic Systems*, 73(1-4):589–602, 2014. (Cited on page 33.)

[112] S.Zhao R.M. Ferrari M.M. Polycarpou X.Zhang, Q. Zhang and T. Parisini. Fault detection and isolation of the wind turbine benchmark: An estimation-based approach. In *Proceedings of IFAC world congress*, volume 2, pages 8295–8300, 2011. (Cited on page 41.)

[113] W. Yang, R. Court, and J. Jiang. Wind turbine condition monitoring by the approach of scada data analysis. *Renewable Energy*, 53:365–376, 2013. (Cited on page 47.)

[114] X. Youa and W. Zhangb. Fault diagnosis of frequency converter in wind power system based on som neural network. *Procedia Engineering*, 29:31323136, 2012. (Cited on page 107.)

[115] YZhang and J.Jiang. Bibliographical review on reconfigurable fault-tolerant control systems. *Annual reviews in control*, 32(2):229–252, 2008. (Cited on page 145.)

[116] Y. Jin Z. Qu, C.M. Ihlefeld and A. Saengdeejing. Robust fault-tolerant self-recovering control of nonlinear uncertain systems. *Automatica*, 39(10):1763–1771, 2003. (Cited on page 145.)

[117] Y. Zhi-Ling, W. Bin, D. Xing-Hui, and L. Hao. Expert system of fault diagnosis for gear box in wind turbine. *Systems Engineering Procedia*, 4:189–195, 2012. (Cited on page 44.)

Abstract:

This thesis addresses the problem of automatic detection and isolation of drift-like faults in wind turbines (WTs). The main aim of this thesis is to develop a generic on-line adaptive machine learning and data mining scheme that integrates drift detection and isolation mechanism in order to achieve the simple and multiple drift-like fault diagnosis in WTs, in particular pitch system and power converter. The proposed scheme is based on the decomposition of the wind turbine into several components. Then, a classifier is designed and used to achieve the diagnosis of faults impacting each component. The goal of this decomposition into components is to facilitate the isolation of faults and to increase the robustness of the scheme in the sense that when the classifier related to one component is failed, the classifiers for the other components continue to achieve the diagnosis for faults in their corresponding components. This scheme has also the advantage to take into account the WT hybrid dynamics. Indeed, some WT components (as pitch system and power converter) manifest both discrete and continuous dynamic behaviors. In each discrete mode, or a configuration, different continuous dynamics are defined. Defining a feature space in each of these discrete modes may allow to increase the discrimination power (sensitivity) of the corresponding features to the components normal and/or failure operation conditions. Finally, this scheme can consider only data samples about normal operation conditions. Any drift from the characteristics representing these normal operation conditions is considered as an evolution towards a failure. When a failure is confirmed, the data samples representing this failure are used to update the classifier structure by integrating a new class to its data base. This helps to overcome the problem of imbalanced data or the absence of data about some faults in a WT component.

Keywords: Drift-like fault detection, Machine learning, Data mining, Hybrid dynamic system, Multicellular converters, Pitch system, Wind turbine.

Résumé:

L'objectif principal de cette thèse est de développer un schéma générique et adaptatif basée sur les approches d'apprentissage automatique, intégrant des mécanismes de détection et d'isolation des défauts avec une force d’apparition progressive. Le but de ce schéma est de réaliser le diagnostic en ligne des défauts simple et multiple de type dérive dans les systèmes éoliens, et plus particulièrement dans le système du calage des pales et le convertisseur de puissance. Le schéma proposé est basé sur la décomposition du système éolien en plusieurs composantes. Ensuite, un classifieur est conçu et utilisé pour réaliser le diagnostic de défauts dans chaque composant. Le but de cette décomposition en composants est de faciliter l'isolation des défauts et d'augmenter la robustesse du schéma globale de diagnostic dans le sens que lorsque le classifieur lié à un composant est défaillant, les classifieurs liées aux autres composants continuent à réaliser le diagnostic des défauts dans leurs composants. Ce schéma a aussi l'avantage de prendre en compte la dynamique hybride de l’éolienne. La définition d'un espace de représentation dans chacun de ces modes discrets peut permettre d'augmenter la puissance de discrimination (sensibilité) des caractéristiques correspondant aux composantes en fonctionnement normal ou défaillant. Enfin, ce schéma ne considère que les données représentant le fonctionnement normal. Toute dérive à partir des caractéristiques représentatives du fonctionnement normal est considérée comme une évolution vers un mode de fonctionnement défaillant. Lorsqu'un défaut est confirmé, les données représentant ce défaut sont utilisées pour mettre à jour la structure du classifieur par l'intégration d'une nouvelle classe dans sa base de données. Cela permet de surmonter le problème de déséquilibre de données ou l'absence des données représentatives de certains défauts de l'éolienne.

Mots-clefs: Détection des défauts type dérive, Apprentissage automatique, Système dynamique hybride, Convertisseur multicellulaire, Système du calage de pales, Eolienne.

