

# Combination of Model-based Observer and Support Vector Machines for Fault Detection of Wind Turbines

Nassim Laouti<sup>1</sup> Sami Othman<sup>1</sup> Mazen Alamir<sup>2</sup> Nida Sheibat-Othman<sup>1</sup>

<sup>1</sup>Université de Lyon, Université Lyon 1, CNRS, CPE Lyon, UMR 5007, LAGEP, F-69616 Villeurbanne, France

<sup>2</sup>Gipsa-lab/CNRS, University of Grenoble, Rue de la Houille Blanche, 38400 Saint Martin d'Hères, France

**Abstract:** Support vector machines and a Kalman-like observer are used for fault detection and isolation in a variable speed horizontal-axis wind turbine composed of three blades and a full converter. The support vector approach is data-based and is therefore robust to process knowledge. It is based on structural risk minimization which enhances generalization even with small training data set and it allows for process nonlinearity by using flexible kernels. In this work, a radial basis function is used as the kernel. Different parts of the process are investigated including actuators and sensors faults. With duplicated sensors, sensor faults in blade pitch positions, generator and rotor speeds can be detected. Faults of type stuck measurements can be detected in 2 sampling periods. The detection time of offset/scaled measurements depends on the severity of the fault and on the process dynamics when the fault occurs. The converter torque actuator fault can be detected within 2 sampling periods. Faults in the actuators of the pitch systems represents a higher difficulty for fault detection which is due to the fact that such faults only affect the transitory state (which is very fast) but not the final stationary state. Therefore, two methods are considered and compared for fault detection and isolation of this fault: support vector machines and a Kalman-like observer. Advantages and disadvantages of each method are discussed. On one hand, support vector machines training of transitory states would require a big amount of data in different situations, but the fault detection and isolation results are robust to variations in the input/operating point. On the other hand, the observer is model-based, and therefore does not require training, and it allows identification of the fault level, which is interesting for fault reconfiguration. But the observability of the system is ensured under specific conditions, related to the dynamics of the inputs and outputs. The whole fault detection and isolation scheme is evaluated using a wind turbine benchmark with a real sequence of wind speed.

**Keywords:** Fault detection and isolation, wind turbine, Kalman-like observer, support vector machines, data-based classification.

## 1 Introduction

With the widespread use of wind turbines (WTs) as renewable energy systems, it is now important to include control and supervision in the system design. Fault detection and isolation (FDI) of WTs allows reducing maintenance costs, which is particularly important for offshore WTs. Online supervision should suggest the best maintenance time as a function of fault occurrence and wind speed in order to reduce operation and maintenance costs. Early detection of faults allows also avoiding degradation of the material and other side effects. Furthermore, fault detection is essential for control reconfiguration in order to ensure optimal power in case of partial fault. Even though the wind turbine functionality might be similar to rotating machinery, it involves a number of difficulties ranging from a high variability in the wind speed, aggression by the environment, measurement difficulties due to noise and vibrations, besides the fact that wind turbines are supposed to run continuously for several years. For these reasons, the development of methods for FDI in WT is increasingly important. Similarly, a number of fault tolerant control (FTC) approaches are also being applied to WT, but this is out of the scope of this paper.

FDI approaches can in general be classified as model-based or data-based: On one hand, model-based methods require a comprehensive model of the system. On the other

hand, success of data-based approaches is conditioned by the significance (amount and quality) of historical data and the mathematical method used to detect the patterns in data. However, training data is usually limited to some specific conditions that are typically normal, non faulty data, with limited variations of operating conditions. Limitations of both model-based and data-based approaches can be overcome by combining them in order to ensure optimal supervision. This represents the main idea of the present paper.

Reviews of WT monitoring and fault diagnosis were proposed by [1 – 3]. Both data- and model-based approaches were reported. Among model-based approaches, observers were applied to monitoring several parts of wind turbines. Reference [4] proposes an unknown input observer to detect sensor faults around the WT drive train. More focus has been drawn on the electrical conversion system in the wind turbines. Reference [5] proposes an observer-based solution to current and voltage sensors fault detection. Reference [6] presents an FDI solution to faults in a doubly fed wind turbine converter.

A number of approaches were used for FDI in WT, such as neural networks (NN) as well as statistical-based approaches. Neural networks were used for estimation of the generator power by [7]. Reference [8] shows that neural networks had a higher confidence level than polynomial regression-based model for FDI in gearbox bearing damages and stator temperature anomalies in the WT. Reference [9]

studies faults related to the accumulation of coal in the coal mill using statistical and dynamic-based approaches. They showed the importance of data selection in the statistical approach for supervision. Reference [10] compares different data-mining algorithms to extract models for FDI of WT (without isolation, except for diverter fault): NN, NN ensemble, boosting tree algorithm, and support vector machine (SVM). In normal situations, SVMs are more accurate in prediction and isolation. But in faulty situations, better prediction is obtained by NN ensemble and better evaluation of its severity by the boosting tree algorithm. The use of frequency domain was also found interesting for FDI of some vibration components in WT. Reference [11] uses the frequency domain model for FDI of tooth crack in the planetary gear using spectral methods.

The WT considered in this work is a horizontal axis variable speed turbine composed of three blades for which a benchmark was proposed by the companies *kk-electronic* and *MathWorks* and the University Aalborg<sup>[12]</sup>. Different faults are likely to occur in this benchmark: sensor faults (pitch positions, generator and rotor speeds), actuator faults (pitch positions, convertor torque) and system faults (drive train). These faults could be type stuck, scaled measurements or subject to offset (e.g., calibration error, interruption in data transmission and degradation of some components). Based on this benchmark, different solutions for FDI of the WT were proposed at an International Federation of Automatic Control (IFAC) competition in 2011<sup>[13–18]</sup>. The proposed solutions were satisfactory only for part of the possible faults. No solution was found convenient for faults related to the actuator of the convertor torque and system faults. Reference [13] only considered sensor faults using SVM for supervision. This method showed the best results in terms of detection time and number of false alarms for faults of type stuck measurements. The second best solution was proposed by [14] which is based on an estimation approach. The third solution is based on up-down counter solution given by [15]. Other approaches were also used, such as the concepts of sensitivity matrix<sup>[16]</sup>, piecewise affine models for pitch sensors<sup>[17]</sup>, parity equations followed by  $H_\infty$  optimization<sup>[18]</sup>, a data-based method<sup>[19]</sup>, and a method based on Kalman filter<sup>[20]</sup>.

In this work, both observers (Kalman-like) and a data-based approach (SVM) are used for FDI in WT using the mentioned benchmark. The paper is organized as follows. In Section 2, basic hints about SVM classification and Kalman-like observer are given. In Section 3, the wind turbine is described and the locations and types of faults are defined. In Section 4, SVM and observer implementation are presented showing the different tuning levels. In Section 5, simulation scenarios are presented to evaluate the efficiency and limitations of the proposed methodology using a real wind speed sequence.

## 2 Theoretical background

The objective of this work is to combine data-based and model-based approaches for FDI in WT. Among data-based approaches for FDI appear artificial neural

networks<sup>[8]</sup>, and statistical methods such as principal component analysis<sup>[21]</sup>, partial least square (PLS) and more recently support vector machines (SVM). This last approach will be considered due to its robustness and fastness, which makes it valid for online applications.

In model-based approaches, the use of observers represents the first choice for FDI<sup>[22–24]</sup>. Various observer structures have been proposed in the literature for linear and nonlinear systems. Among the proposed structures, the unknown input observer is widely used<sup>[25]</sup>; also appears the eigenstructure assignment approach<sup>[26]</sup> or the sliding mode observer<sup>[27]</sup>. For FDI in nonlinear applications, the extended Kalman filter was firstly applied without theoretical validation<sup>[28]</sup>. The theoretical FDI problem for nonlinear systems was initially introduced by [29]. Since that time, many techniques have been proposed for nonlinear systems. Garcia and Frank<sup>[30]</sup> gave a survey on the principal observer-based approaches to fault diagnosis of nonlinear systems. The authors in [31–33] proposed solving the FDI problem by combining the geometric decoupling techniques with the nonlinear observer synthesis. The observer form proposed in the present paper is obtained by input-output injection linearization.

### 2.1 Support vector machines

SVMs are based on the structural risk minimization principle using the statistical learning theory introduced in 1964 by Vapnik and Chervonenkis<sup>[34]</sup>. Only recently, SVMs were introduced as machine learning algorithms for classifying data from two different classes<sup>[35,36]</sup>. Basically, a binary support vector classifier constructs a separating hyperplane. The hyperplane should have the maximum margin which is the width up to which the boundary can be extended on both sides before it hits any data point. These contact points are called the support vectors. In order to allow classifying nonlinearly separable sets, a nonlinear kernel function can be used. The main differences between SVM and many other statistical methods are therefore: First, the structural risk minimization (training by traditional classifiers usually minimizes only the empirical risk) that improves the ability of generalization even with a reduced number of samples and avoids over-fitting in view of good parameter tuning. Second, SVMs use nonlinear kernels which allow separation of nonlinearly separable data. SVMs have been extensively used to solve classification problems in many domains ranging from face, object and text detection and categorization, information and image retrieval and so on. Their use for fault detection started in 1999 and was found to improve the detection accuracy. Reference [37] presents a review about the use of SVMs for fault detection. They reported 37 papers in academic journals on this subject. Nowadays, the number of journal papers using SVMs for fault detection has importantly increased. The concerned domains are in majority restricted to mechanical machinery with slight extension to electro-mechanical machinery, semi-conductors and chemical processes<sup>[38–40]</sup>.

Consider the problem of separating the set  $D$  composed of  $N$  training vectors belonging to two classes  $(\Omega_1, \Omega_2)$  (for

more details see [34–36])

$$D = \{(x_1, z_1), (x_2, z_2), \dots, (x_N, z_N)\}, \Omega_1 = \{x_i | z_i = +1\}$$

$$\Omega_2 = \{x_i | z_i = -1\}, \quad i = 1, \dots, N$$

where  $x_i \in \mathbf{R}^p$  denote the input vectors, each vector being characterized by a set of  $p$  descriptive variables  $x_i \in \{x_{i1}, x_{i2}, \dots, x_{ip}\}$ , and  $z_i \in \{-1, +1\}$  defines the class label of a given vector  $x_i$ . The purpose of SVM is to find an optimal separating hyperplane  $f(x)$  that maximizes the margin ( $\frac{1}{\|w\|}$ ) between the hyperplane and the data points from each side such that all points of the same class are on the same side of the hyperplane (Fig. 1). The support vectors correspond to points located exactly at a distance equal to the margin. The weight  $w$  is a  $p$ -dimensional vector orthogonal to the hyperplane. Since it is not always possible to perfectly separate the data (for instance, due to measurement noise/errors), a slack variable  $\zeta_i$  is introduced to relax the margin constraints and allow misclassification.  $\zeta_i$  measures the degree of misclassified vectors (lying on the wrong side of the hyperplane or inside the margin). In this case, the optimisation problem for soft margin classification can be written as follows (linearly separable data with error tolerance):

$$\min_{w, b, \zeta} \left( \frac{1}{2} \|w\|^2 + C \sum_{i=1}^N \zeta_i \right) \quad (1)$$

subject to  $\begin{cases} z_i f(x_i) \geq 1 - \zeta_i \\ \zeta_i \geq 0 \end{cases}$ , where  $f(x_i)$  is the predicted output and  $C \geq 0$  is a regularization parameter that governs the tolerance to misclassification. Increasing the value of  $C$  will increase the cost of misclassifying points but reduce the importance of minimizing the model complexity (minimizing  $\|w\|^2$ ). It can be tuned by optimisation and cross validation. If criterion  $\|w\|^2$  is convex and all constraints are linear, this problem can be solved by constructing a Lagrange function. Solving the dual optimization problem gives the Lagrangian multipliers ( $\alpha_i$ ), the support vectors and  $b$  (the bias). According to the Karush-Kun-Tucker complementary condition, the solution must satisfy:  $\alpha_i [z_i f(x) - 1 + \zeta_i] = 0$  which means either  $\alpha_i = 0$  or  $z_i f(x) - 1 + \zeta_i = 0$ . The latter condition corresponds to the support vectors (inputs lying on the margin), where  $\alpha_i \neq 0$ .

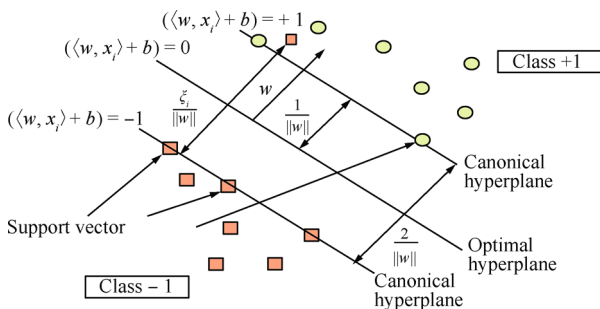


Fig. 1 SVM classification of two linearly separable classes (here  $x_i \in \mathbf{R}^2$ )

For nonlinearly separable data, which is the case of many real problems, the data can be mapped by some nonlinear function  $\phi(x)$  into a high-dimensional feature space where linear classification becomes possible. Rather than fitting nonlinear curves to the data, SVMs handle this by using a kernel function  $K(x_i, x) \leq \phi(x_i), \phi(x) >$  to map the data into a different space, where a hyperplane can be used to do the separation. The obtained decision function is

$$f(x) = \langle w, \phi(x) \rangle + b = \sum_{i=1}^N \alpha_i z_i K(x_{\text{sup } i}, x) + b \quad (2)$$

with the properties

$$w = \sum_{i=1}^N \alpha_i z_i \phi(x_{\text{sup } i})$$

where  $b$  is the bias term (a scalar). It is clear that this decision function is only influenced by the non-zero  $\alpha_i$  (support vectors). This gives two features to the SVM algorithm: ability of adjusting the error with a reduced training set, and fast computation in decision-making (allowing online implementation). Therefore,  $N$  can be replaced by  $n_{sv}$  (size of support vectors  $x_{\text{sup}}$ ) in (2). The kernel function can be any function that satisfies Mercer's theorem, namely any continuous positive definite function can be considered as a kernel function that represents an inner product function in some space. The Gaussian kernel (which is a radial basis function) is the most widely used:

$$K(x_i, x) = e^{-\frac{\|x_i - x\|^2}{2\sigma^2}}$$

where  $\sigma$  is the variance. A small  $\sigma$  is known to perfectly fit the training data but to be unable to evaluate the fault for new data (overfitting, reduced ability of generalization). It can also be tuned by optimisation and cross validation.

## 2.2 Kalman-like observer

The Kalman-like observer was developed by [38] for a class of nonlinear systems, where the state matrix may depend on the inputs; outputs or on time and all the inputs are regularly persistent. The observer equations are based on the minimization of a quadratic convex criterion. For such a class of nonlinear systems, the Kalman-like observer is easier to implement than the Kalman filter since the gain matrix relies on a unique tuning parameter, which justifies the choice of this observer for this FDI in WT. Also, no need for a change of variables is required since this system is under a canonical form of observability.

Consider the following nonlinear system:

$$\begin{cases} \dot{x} = A(u, y)x + G(u) \\ y = Cx \end{cases} \quad (3)$$

with a single output,  $y \in \mathbf{R}$ . The state matrix  $A$  might depend on the inputs and the outputs. Let us call  $\phi_u(s, t_0)$  the unique solution of

$$\frac{d\phi_u(S, t_0)}{ds} = A(u(s))\phi_u(S, t_0)$$

with  $\phi_u(t_0, t_0) = I$  the identity matrix and  $\phi_u(t_0, s) = \phi_u^{-1}(s, t_0)$ . We denote  $\phi_u(s, t) = \phi_u(s, t_0)\phi_u(t_0, t)$  and

$G(u, t_0, t_0 + t_1)$  is the Gramian of observability related to the input  $u$  on the interval  $[t_0, T]$ :

$$G(u, t_0, t_0 + t_1) = \int_{t_0}^{t_0+t_1} \phi_u^T(t, t_0) C^T C \phi_u(t, t_0) dt$$

where  $T$  stands for the transposed matrix or vector.

**Definition 1.** An input  $u \in \mathbf{R}^m$  is regularly persistent for system 3 if  $\exists t_1 > 0$ ,  $\exists \alpha_1 > 0$ ,  $\exists \alpha_2 > 0$  and  $\exists t_0 \geq 0$  such that  $\forall t \geq t_0$ :

$$\begin{aligned} \lambda_{\min}(G(u, t_0, t_0 + t_1)) &\geq \alpha_1 \\ \lambda_{\max}(G(u, t_0, t_0 + t_1)) &\leq \alpha_2 \end{aligned}$$

where  $\lambda_{\min}$  and  $\lambda_{\max}$  stand for the less and largest eigenvalues of  $G$ , respectively.

**Theorem 1**<sup>[41]</sup>. If  $u$  is regularly persistent, then

$$\begin{cases} \dot{\hat{x}} = A(u)\hat{x} + G(u) - R^{-1}C^T(C\hat{x} - y) \\ \dot{R} = -\theta R - A^T(u)R - RA(u) + C^T C \end{cases} \quad (4)$$

is an observer for system 3 with  $\theta > 0$ ,  $\hat{x}(n) \in \mathbf{R}^n$ . Moreover, the norm of the estimation error goes exponentially to zero. The tuning parameter of the Kalman-like observer is  $\theta$ , which must be superior than zero. The convergence of the observer is guaranteed if matrix  $R$  is a symmetric positive definite matrix.

### 3 Wind turbine description

A horizontal axis variable speed turbine composed of three blades is considered in this work<sup>[12]</sup>. The system has a full converter coupled to a generator that allows converting the mechanical energy to electrical energy. A drive train is used to increase the rotational speed from the rotor (the three blades) to the generator.

**Sensor faults:** The system is equipped with duplicated sensors (Fig. 2) measuring:

- 1) the three pitch positions ( $\beta_{k,mi}$ ,  $k = 1, 2, 3$ ,  $i = 1, 2$ ),
- 2) and the generator and rotor speeds ( $\omega_{g,mi}$ ,  $\omega_{r,mi}$ ,  $i = 1, 2$ ),

where  $i$  indicates the sensor number and  $k$  the pitch number (each pitch has 2 sensors measuring its position). This gives a total of ten sensors all subject to two kinds of faults: stuck or scaled measurements that are to be detected within 10 sampling periods (desired number of samples for detection  $n_s^{\text{des}} < 10$ ), where the sampling time is  $T_s = 0.01$  s (Table

- 1). The process has other sensors, measuring for instance the wind speed and generator power that are not supervised in this work.

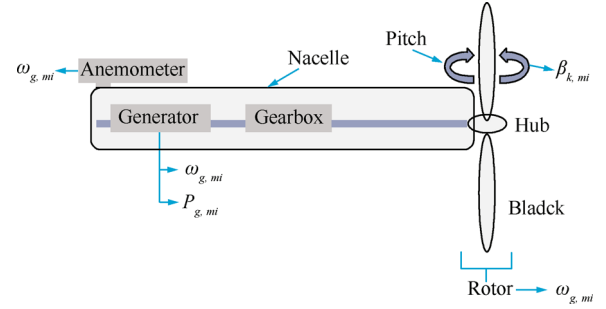


Fig. 2 Measurements of the WT

**Actuator faults:** As a function of the wind speed, a control system allows controlling the aerodynamics of the turbine to get the optimal power. The benchmark allows simulating the wind turbine control under normal operation: Zone II: power optimization, and Zone III: constant power production. Zones I and IV correspond to the start and stop operations. The actuators manipulate the three pitch systems and the converter torque. They allow respectively pitching the blades and setting the generator torque to control the generator and rotor rotation speeds. These actuators are also subject to fault. The converter system that sets the generator torque might have an offset that should be detected rapidly ( $n_d^{\text{des}} < 5$ ). The three pitching systems might also have a change in the dynamics that can be due to abrupt change in the hydraulic system or to high air content in the oil at a slower rate.

**System faults:** In the used benchmark, system faults might also occur for instance in the driving train due to friction changes with time that might break down the train, but this is not considered in this work. In real life, other kinds of faults might also occur, like data transmission, or raw position that are not considered either.

**Hints on the model:** Fault detection will be studied based on closed-loop simulations in Zones II and III with a real measured sequence of wind of 4400 s. The detailed model of the turbine is given in [12]. Some hints are given below. Note that the system contains nonlinear parts, the measurements are noisy and the control system is switched between both zones, which all add difficulties for FDI.

Table 1 Fault locations and fault detection results based on few scenarios ( $n_s^{\text{des}}$  and  $n_s$  are the desired and real numbers of sampling periods for detection, the sampling period being  $T_s = 0.01$  s)

	Fault $n^\circ$	Fault location	Fault type	$n_s^*$	$n_s^{\text{des}*}$
Sensor faults	1a	Pitch angle sensor	Stuck ( $\beta = \text{constant}$ )	2	<10
	1b		Scaling factor $\times \beta$	10 if $\Delta\beta > 1.5$	
	2a	Rotor speed sensor	Stuck ( $\omega_r = \text{constant}$ )	2	
	2b		Scaling factor $\times \omega_r$	47 if $\Delta\omega_r \geq 0.3$	
	3a	Generator speed sensor	Stuck ( $\omega_g = \text{constant}$ )	2	
Actuator faults	3b		Scaling factor $\times \omega_g$	2 if $\Delta\omega_g \geq 4$	< 5
	4	Generator torque	Offset $\tau_g = \tau_g + \Delta\tau_g$	2	
	5	Error in parameters $w_n$ and $\zeta$	Abrupt or slow drift	400	
					8–600 (8 if sever)



Let us recall the pitch system and converter models that will explicitly be referred to in the fault scenarios. The pitch system is hydraulic and the relation between the measured and desired pitch angle (the reference obtained by the controller) can be modelled by a second-order transfer function:

$$\frac{\beta_k^m(s)}{\beta^d(s)} = \frac{w_n^2}{s^2 + 2\xi w_n s + w_n^2} \quad (5)$$

where  $\beta_k^m(s)$  and  $\beta^d(s)$  are the measured and desired positions of pitches  $k = 1, 2, 3$ , and  $[w_n, \xi] = [11.11 \text{ rad/s}, 0.6]$  are respectively the natural frequency and a damping factor. The pitch rate ( $\dot{\beta}$ ) may take values between  $-8$  and  $8 \text{ deg/s}$  and the pitch angle ( $\beta$ ) between  $-2$  and  $-90 \text{ deg}$ . As shown below,  $\beta$  remains lower than  $20^\circ$  with the used wind sequence in this benchmark.

Similarly, the converter dynamics, represented by the measured to the desired generator torque, can be modelled by a first-order transfer function:

$$\frac{\tau_g^m(s)}{\tau_g^d(s)} = \frac{1}{\tau s + 1} \quad (6)$$

where  $\tau_g^m$  and  $\tau_g^d$  are the real and desired generator torques, and  $\tau = 0.02 \text{ s}$  is a combined convertor and generator model parameter. The real generator torque, being non measured, is calculated from the measured generator speed  $\omega_{g,mi}$  and the power produced by the generator  $P_g$ , which are related by the following equation:

$$P_g(t) = \eta_g \omega_g(t) \tau_g(t) \quad (7)$$

where  $\eta_g = 0.98$  is the generator efficiency.

## 4 FDI implementation

First of all, SVM is applied alone to FDI of all faults. For actuator faults, both SVM and a Kalman-like observer are compared.

### 4.1 Support vector machines

Fault detection and isolation by SVM is developed in two parts: training of models for FDI and validation of the obtained models. The main steps in the development are detailed below.

1) Data generation: First of all, a set of measured data  $x$  (inputs, references and outputs) without fault or with different fault amplitudes is generated to train models for detection of each fault separately (to ensure isolation). This set was generated using a real wind sequence as input to the benchmark. For each fault, about six scenarios were considered, with different fault amplitudes. Each sample is attributed  $z = +1$  (with/without fault) for the considered fault. Note that when a particular fault is considered, normal data might contain faults in the other sensors or actuators.

2) Data pre-treatment: Data filtering was found primordial before model development in order to reduce the sensitivity to process disturbances or measurement noise. A first order filter with a time constant  $\tau$  was used

(filtered data is noted with a hat). Data were not normalized.

3) Features selection: The key step in training SVM models is features selection. The input vector  $x_i$  used for classification should contain the most pertinent information related to the considered fault. But, all the data cannot be used since the important information might be affected by high variation amplitudes of useless variables. This vector may include inputs, outputs, set-points, combination of both or derivation of the measurements with time. It is important to mention that the selection of  $x_i$  should insure both fault detection and isolation. In this work,  $x_i$  is selected based on observing the process outputs for each fault. Note that for each fault a different vector was proposed. Using some statistical analysis such as principal component analysis or partial least square can be useful for pre-treatment, but was not found interesting in the present study since some information was lost during treatment.

4) Parameter tuning: The kernel used for learning all the faults is the Gaussian kernel. Initial values of the kernel variance  $\sigma$  and generalization parameter  $C$  are obtained based on the correlations proposed by [42]. These values were then refined based on a few simulations. Cross validation may also be used, but this would require reduction of the data size before optimising the parameters for each fault. Indeed, the wind sequence duration is  $4400 \text{ s}$  with  $T_s = 0.01 \text{ s}$ , which gives  $4 \times 10^5$  samples. For parameter tuning, it is well known that high  $\sigma$  values lead to improved generalisation, but very high  $\sigma$  might not fit the data at all. Small  $\sigma$  on the contrary might perfectly fit the learning data (overfitting), but might be unable to evaluate the fault for new data (reduced ability of generalization). For the regularization parameter  $C$ , higher values allow more misclassification and minimizing the function complexity.

5) Model development: The SVM learning algorithm then uses the inputs  $x$  and their corresponding outputs  $z$  to identify  $\alpha_i$  and the support vectors ( $x_{\text{sup},i}$ ) to be used in (2) for decision making. Note that the same “model” ( $x_i$  and  $\alpha_i$ ) is used for faults of the same type (e.g., pitch position fault 1a  $\beta_{k,mi}$ ,  $\forall k = 1, 2, 3, \forall i = 1, 2$ , so one model for 6 sensors). Eight different models were therefore developed.

6) Validation: the obtained SVM models are evaluated in new fault scenarios. Parameter adjustment is done based on the number of false alarms and misdetections.

The features used for learning SVM models are detailed below for each fault.

#### 4.1.1 Stuck measurements

Data exchange with the system might be interrupted for few sampling periods, especially in wireless systems, which is frequent in WT. The measurement is therefore stuck at the last exchange with the system. In order to detect such a fault, the use of the derivative of the measurement can be very useful. A wise filter is however necessary in order to overcome difficulties due to measurement noise. The sensors concerned by this type of fault are the pitch position sensor and the rotor and generator speed sensors.

**Pitch position sensor.** For stuck pitch position sensor fault (fault 1a), the following vector is used for detection

and isolation:

$$x = \begin{bmatrix} \hat{\beta}_k^{m1}(t_j) - \hat{\beta}_k^{m2}(t_j) \\ \beta_1^{m1}(t_j) - \beta_1^{m1}(t_{j-1}) \\ \beta_1^{m2}(t_j) - \beta_1^{m2}(t_{j-1}) \end{bmatrix} \quad (8)$$

where  $t_j$  and  $t_{j-1}$  are the time instances  $j$  and  $j-1$ , respectively,  $\beta_k^{mi}$  is the measured pitch position, and  $\hat{\beta}$  is the measured pitch position filtered using  $\tau = 6 \times T_s$ . The first line allows fault detection. The second and third lines allow fault isolation (sensor number 1 or 2). They give a kind of derivative of the sensor measurement (without the division by  $T_s$ , and using the absolute value).

For practical applications, when

$$|\beta_k^{mi}(t_j) - \beta_k^{mi}(t_{j-1})| = 0$$

this term is replaced by a large constant value (5000) in order to enhance distinguishability between the fixed value fault and normal case (no fault) as these values oscillate between  $1 \times 10^{-2}$  and 2. For all sensors measuring the pitch positions ( $\beta_k^{mi}$ ,  $k = 1, 2, 3$ ,  $i = 1, 2$ ), the same model is used with the variance  $\sigma$  tuned at 10.

**Generator and rotor speed sensors.** For stuck rotor (fault 2a) and generator (fault 3a) speed sensor faults ( $\omega_{g,mi}$ ,  $\omega_{r,mi}$ ,  $i = 1, 2$ ), the following vector is used for detection and isolation:

$$x = \begin{bmatrix} |\hat{\omega}_p^{m1}(t_j) - \hat{\omega}_p^{m2}(t_j)| \\ |\omega_p^{m1}(t_j) - \omega_p^{m1}(t_{j-1})| \\ |\omega_p^{m2}(t_j) - \omega_p^{m2}(t_{j-1})| \end{bmatrix}, \quad p = g, r \quad (9)$$

where  $\hat{\omega}_g$  is obtained using a filter with  $\tau = 2 \times T_s$  and  $\hat{\omega}_r$  using  $\tau = 60 \times T_s$ . The Gaussian variance is tuned at  $\sigma = 15$ .

#### 4.1.2 Scaled measurements

Scaled faults might occur in the sensors, for instance, due to calibration errors, or drifts in some components of the sensor with time. Therefore, these faults might appear progressively. In order to detect such faults, the difference (in absolute value) between the “desired” and measured values is considered after filtering. It is important to note that in the benchmark these faults are simulated as a multiplicative gain ( $\beta_k^{mi} = k \times \beta_k$ ). Therefore, once the drift attains a sufficiently detectable limit, it can be detected and isolated. When the measurement equals zero, the faulty measurement coincides with the real one, even though the sensor might be defected. Therefore, in the considered scenarios, the faults are introduced at instances where  $\beta_k^{mi} \neq 0$ , otherwise it is not detected.

**Pitch position sensors.** Drift in the pitch position sensor fault (fault 1b) is detected and isolated in two steps: First of all, the fault is detected using the following measurement vector

$$x = \begin{bmatrix} |\hat{\beta}_k^{m1}(t_j) - \hat{\beta}_k^{m2}(t_j)| \\ |\beta_k^{m1}(t_j) - \beta_k^{m1}(t_{j-1})| \\ |\beta_k^{m2}(t_j) - \beta_k^{m2}(t_{j-1})| \end{bmatrix}. \quad (10)$$

The second and third lines in 8 are important in order to exclude faults of type stuck sensor (fault 1a). In a second

step, if a fault of type b is detected, for isolation between sensors 1 and 2, the following vector is used:

$$x = \begin{bmatrix} |\hat{\beta}^d(t_j) - \beta_k^{m1}(t_j)| \\ |\hat{\beta}^d(t_j) - \beta_k^{m2}(t_j)| \end{bmatrix} \quad (11)$$

where  $\beta^d$  is the desired value of the pitch angle and  $\hat{\beta}$  is obtained by filtering  $\beta_k^{mi}$  using a first order filter with  $\tau = 0.08$  s.

**Generator and rotor speed sensors.** For the detection of generator and rotor speed sensor faults of type b (scaled measurement), excluding faults of type a (stuck measurement), the following vector is first applied:

$$x = \begin{bmatrix} |\hat{\omega}_p^{m1}(t_j) - \hat{\omega}_p^{m2}(t_j)| \\ |\hat{\omega}_p^{m1}(t_j) - \hat{\omega}_p^{m1}(t_{j-1})| \\ |\hat{\omega}_p^{m2}(t_j) - \hat{\omega}_p^{m2}(t_{j-1})| \end{bmatrix}, \quad p = g, r. \quad (12)$$

In a second step, isolation between sensors 1 and 2 is done using the following vector

$$x = \begin{bmatrix} \frac{\hat{\tau}_g^m \times \hat{\omega}_{p,m1}(t_j)}{\hat{P}_g^m} \\ \frac{\hat{\tau}_g^m \times \hat{\omega}_{p,m2}(t_j)}{\hat{P}_g^m} \end{bmatrix}, \quad p = g, r \quad (13)$$

where  $P_g^m$  is the measured power of the generator. The measurements are filtered with  $\tau = 6 \times T_s$  for the estimation of faults of  $\hat{\omega}_s$  and using  $\tau = 60 \times T_s$  for  $\hat{\omega}_r$ .

#### 4.1.3 Actuators

**Offset in the generator torque actuator.** The generator torque actuator fault (fault 4) is assumed to be an offset in the benchmark, which is comparable to scaled measurements. Therefore, comparison to the desired value is considered:

$$x = \begin{bmatrix} |\tau_g^d - \hat{\tau}_g^m| \\ \lambda_2 \times \left| \hat{\omega}_g^d - \frac{\omega_g^{m1} + \omega_g^{m2}}{2} \right| \end{bmatrix}. \quad (14)$$

In the first line, the measured generator torque ( $\tau_g^m$ ) is compared to the desired one ( $\tau_g^d$ ), and in the second line, the measured generator speed is compared to the mean measured values from the two sensors  $\omega_{g,mi}$ . The desired generator speed is calculated from the desired generator torque  $\tau_g^d$  (7) which gives

$$\omega_g^d = \frac{P_g^d}{\tau_g^d \eta_g}$$

with  $P_g^d$  the desired generator power.  $\tau_g^d$  is filtered using a first order filter,  $\tau = 2 \times T_s$  is a time constant. The objective of this filter is to take into account the dynamics of the control system (time necessary for  $\tau_g^m$  to attain  $\tau_g^d$ , (6). The factor  $\lambda_2 = 10^{-10} \times \nu_{\text{wind}}^6$  in the 2nd component of  $x$  is used to take into account the wind speed with a kind of normalization with respect to the first term. The kernel variance is tuned at  $\sigma = 10$ .

**Scaled pitch position actuator.** Pitch position actuator faults (fault 5) might be due to abrupt change in the hydraulic system or to high air content in the oil which appears at a slower rate. Both types of faults are modelled by

varying  $w_n$  and  $\zeta$  in (1) either abruptly or more smoothly over 30 s.

As can be seen, (5) is a second order linear relation between  $\beta$  and  $\beta^d$ . In the case of alteration in parameters  $w_n$  and  $\zeta$ , the stationary state does not change, but the transient dynamics changes. In order to estimate the dynamics, the transient behavior resulting from different operating conditions should be included in the data for SVM training, which might increase importantly the data volume. Based on a few simulations, as for scaled sensors faults, the vector proposed for the detection of the pitch position actuator fault is thought to take into account the difference between the real and desired pitch positions. Note that due to measurement noise and control dynamics, this difference will be subject to oscillations even under normal situation. The difference between both sensors is also considered in order to distinguish this fault from sensor faults. Finally, the generator speed measurement is considered since a fault in the pitch position will affect the rotation of the rotor and so the generator at a secondary level.

$$x = \begin{bmatrix} \omega_g^{m1} - \omega_g^{m2} \\ \beta_k^{m1} - \beta_k^{m2} \\ \hat{\beta}^d - \beta_k^{m1} \\ \hat{\beta}^d - \beta_k^{m2} \end{bmatrix} \quad (15)$$

where  $\beta_k^{mi}$  is the response of the second order system given by (5) to the input  $\beta^d$ . A Gaussian variance  $\sigma=10$  is used.

Since it is difficult to obtain a comprehensive training data including transitory states, and since the mechanical model is known, it is obvious to develop a model-based method for fault detection of the pitch position actuators. Therefore, before showing the simulation results of SVM, a Kalman-like observer will be developed for fault detection and isolation of this fault.

## 4.2 Kalman-like observer for pitch position actuator

As mentioned previously, the pitch position actuator fault (fault 5) is simulated by deviating the natural frequency and the dumping factor  $w_n$  and  $\zeta$  from their nominal values. Therefore, for fault detection, an observer is developed to estimate these parameters and evaluate the drift from nominal values. The difference between the estimated values of any of these parameters with respect to its nominal value can be used as a residual with some threshold, that is to be defined as a function of the noise.

Considering (5) and by applying the following change of coordinates  $x_1 = \beta_k, x_2 = \beta_k, u = \beta^d, k = 1, 2, 3$ , one gets the following state equations:

$$\begin{cases} \dot{x}_1 = x_2 \\ \dot{x}_2 = -w_n^2 x_1 - 2\zeta w_n x_2 + w_n^2 u \\ y = x_1. \end{cases}$$

To estimate parameters  $w_n$  and  $\zeta$ , we construct the follow-

ing augmented system, with  $x_3 = 2\zeta w_n$  and  $x_4 = w_n^2$ :

$$\begin{cases} \dot{x}_1 = x_2 \\ \dot{x}_2 = -x_4 x_1 + x_3 x_2 + x_4 u \\ \dot{x}_3 = 0 \\ \dot{x}_4 = 0 \\ y = x_1. \end{cases} \quad (16)$$

Necessary conditions for the observability of  $x_3$  and  $x_4$  in system 16 are:

- 1)  $\dot{x}_1 \neq 0$  and  $\dot{x}_2 \neq 0$  (non null dynamics  $\beta_k \neq 0$  and  $\dot{\beta}_k \neq 0$ , which requires that  $\beta_r \neq 0$ );
- 2) Regularly persistent inputs are required ( $\beta^d$ );
- 3) And  $(\dot{x}_1 + \dot{u})x_2 \neq (x_1 + u)\dot{x}_2$ .

Under these conditions, a Kalman-like observer<sup>[43]</sup> can be developed for system (16) as follows:

$$\begin{cases} \dot{\hat{x}} = A(u)\hat{x} + G(u) - R^{-1}C^T(C\hat{x} - y) \\ \dot{R} = -\theta R - A^T(u)R - RA(u) + C^T C \end{cases} \quad (17)$$

with  $C = [1 \ 0 \ 0 \ 0]$

$$A(u, y) = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & -s & u - x_1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}, \quad \hat{X} = \begin{bmatrix} \hat{x}_1 \\ \hat{x}_2 \\ \hat{x}_3 \\ \hat{x}_4 \end{bmatrix},$$

$$G(u) = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix},$$

$$R = \begin{bmatrix} r_{11} & r_{12} & r_{13} & r_{14} \\ r_{21} & r_{22} & r_{23} & r_{24} \\ r_{31} & r_{32} & r_{33} & r_{34} \\ r_{41} & r_{42} & r_{43} & r_{44} \end{bmatrix}.$$

Matrix  $R$  is initialized at identity. The observer is tuned using  $\theta=3$ .  $s = \dot{x}_1$  is the filtered output derivative signal which should be bounded and non null. A first order filter with  $\tau = T_s s$  is employed on  $\beta_k$ . Note however that no filtering was considered neither for the output nor its set-point  $\beta_k^{mi}$  and  $\beta^d$ . Indeed, filtering these entities differently would create a static error (between the output and the set-point), which would affect the observability of the unknown parameters.

Also note that since the system is not observable if  $\beta^d = 0$ , the observer gain is set to zero in Zone II and is activated only in Zone III. Moreover, when  $\beta^d = 0$ , the estimated values are reinitialized at their nominal values.

## 5 Results and discussion

### 5.1 SVM results

SVM models were applied to fault detection and isolation of all the discussed sensors and actuators. All faults of type stuck measurement could be detected within 2 sampling periods while introduced at different instances, and therefore under different dynamics (e.g., control phases). The detection time (and robustness) for scaled measurement faults depended on the scaling factor and on the system behaviour at the fault time. Based on a few simulations, an average value  $n_s$  was calculated, as reported in Table 1.

In the case of occurrence of one fault at a time, fault detection and isolation is insured for all the considered faults. For simultaneous faults, fault isolation is insured if the estimation is based on a different vector  $x$ . For instance, one of the scenarios will show the efficient isolation of simultaneous scaled faults in  $\omega_r$  and  $\omega_g$ . Another scenario will show the efficiency of isolation of simultaneous faults of different types stuck/scaled in two different sensors measuring the pitch position.

#### 5.1.1 Pitch position sensor faults

In a first scenario, two simultaneous faults were considered in the pitch positions  $\beta_1^{m1}$  and  $\beta_2^{m2}$  of types stuck and scaled measurements, respectively. Moreover, two scaling factors were compared: 1.2 and 1.8. The faults could be detected and isolated in both cases. Fig. 3 shows the detection results of the fault of type stuck measurement in  $\beta_1^{m1}$ . Figs. 4 and 5 show the effect of the scaling factor on the FDI results of  $\beta_2^{m2}$ . The scaled measurement fault occurs in  $\beta_2^{m2}$  at 2800 s, where  $\beta_2^{m2}$  holds a high value (around 10 deg), and finishes at 2900 s, where it decreases almost to zero. It can be seen that the detection results depend on the measured value of  $\beta$ , since the offset is calculated as a scaling factor of this measurement. Therefore, the smaller the value of  $\beta$ , the smaller the offset, and at  $\beta=0$ , it can be said that the fault is eliminated. For this reason, it can also be seen that the fault is detected only by intermittence if the scaling factor equals 1.2 (Fig. 5), while it is well detected with a scaling factor of 1.8 (Fig. 4).

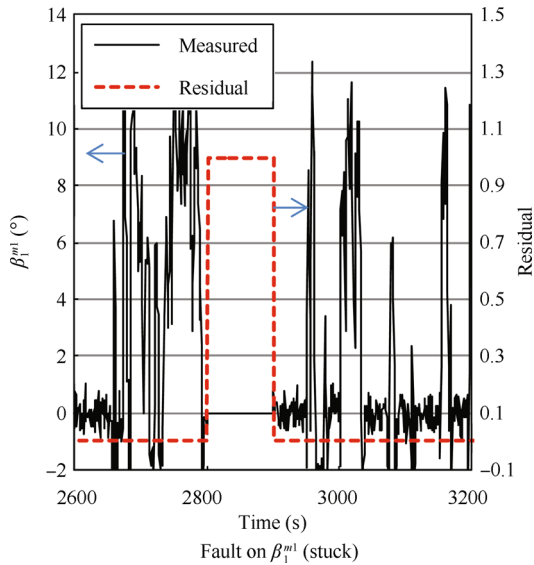


Fig. 3 Fault detection and isolation of stuck sensor  $\beta_1^{m1}$  (fault 1a). Fault duration: 2800–2900 s. The scale of the measured value is shown on the left and the residual's scale is given on the right (as indicated by the arrows)

A threshold is to be attributed to each fault, to be compared to the residual and to announce fault occurrence. The threshold should be  $\geq 0$  and its amplitude depends on the absolute value of used  $x$ . For instance, in Figs. 3–5, the threshold can reasonably be set to 0.5, which means that the system is considered non faulty if the residual is less than 0.5. (The color figures in this paper can be found in the electric version.)

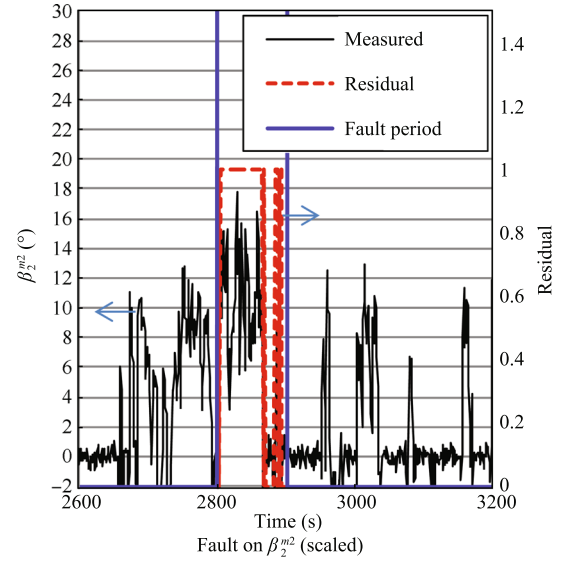


Fig. 4 Fault detection and isolation of pitch position  $\beta_2^{m2}$  between 2800–2900 s (fault 1b), scaling factor = 1.8

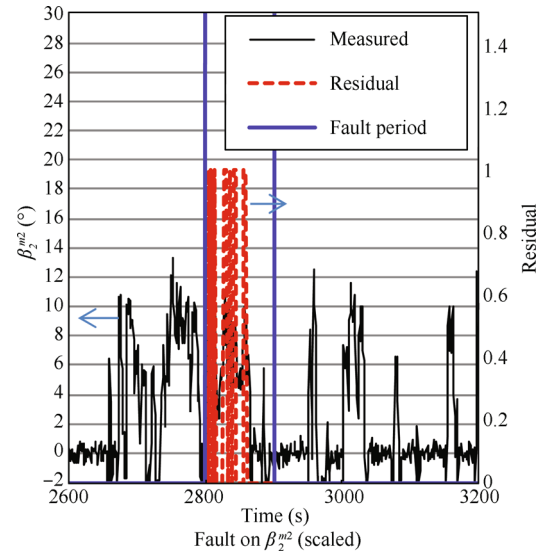


Fig. 5 Fault detection and isolation of pitch position  $\beta_2^{m2}$  between 2800–2900 s (fault 1b), scaling factor = 1.2

It can be concluded that fault isolation is robust to simultaneous faults in this case. Fault detection is robust for faults of type stuck measurements, but depends on the dynamics and the offset in scaled measurements.

#### 5.1.2 Generator and rotor speed sensor faults

As for the pitch position case, stuck measurement could be detected in 2 sampling periods while introduced in  $\omega_r$  or  $\omega_g$  sensor at different instances under different dynamics. Fig. 6 shows the detection results for the rotor speed sensor  $\omega_r^{m1}$ , where the measurement was stuck at 1.2 rad/s.

Concerning scaled measurement faults, fault detection results were more robust than those of the pitch position, which is due to the lower noise level with respect to the absolute values of  $\omega_g$  and  $\omega_r$ . An example of simultaneous faults occurring in both  $\omega_r^{m2}$  and  $\omega_g^{m1}$  with a scaling factor of 1.2 for both sensors is shown in Figs. 7 and 8. This leads



to  $\Delta\omega_r \approx 0.35$  rad/s and  $\Delta\omega_g \approx 30$  rad/s. Based on a number of simulations, it can be concluded that fault isolation of these sensors is efficient for both fault types (stuck/scaled). Again, the threshold can be set to 0.5 for both types of faults related to these sensors.

In Fig. 9, the fault was introduced in  $\omega_r^{m2}$  at 100 s, where the absolute value of  $\omega_r$  was around 0.6 rad/s (while  $\omega_r \approx 1.7$  at 3560 s, Fig. 7). Therefore, with a comparable scaling factor of 0.8, the fault leads to a lower  $\Delta\omega_r$  ( $\Delta\omega_r \approx 0.1$  rad/s). The fault could however be detected but the detection results were intermittent.

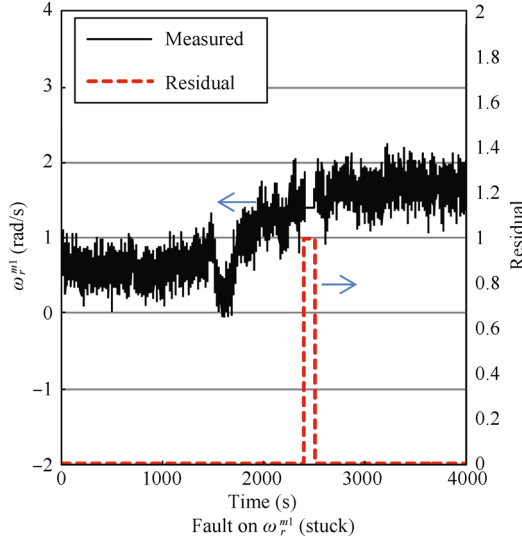


Fig. 6 Fault detection and isolation of the rotor speed (fault 2a), fault duration: 2400–2500 s

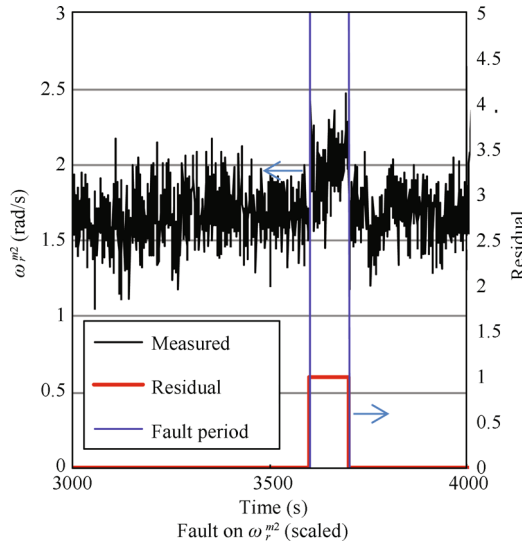


Fig. 7 Simultaneous faults in  $\omega_r^{m2}$  and  $\omega_g^{m1}$ , both of them scaled by a factor of 1.2 (faults 2b and 3b), fault duration: 3600–3700 s

### 5.1.3 Torque actuator faults

As in scaled faults, the estimation of the rotator torque actuator faults depends on the offset. Fig. 10 shows that the detection is very fast and free of oscillations or false alarms with an offset of 200 N·m (i.e., 1.3 % of  $\tau_g$  at the time of fault occurrence). This fault could be detected in both controller zones (I and II). When considering faults with 100 N·m off-

set, the residual had some oscillations (not shown in the figure), which reveals the detectable limit (about 0.07 %).

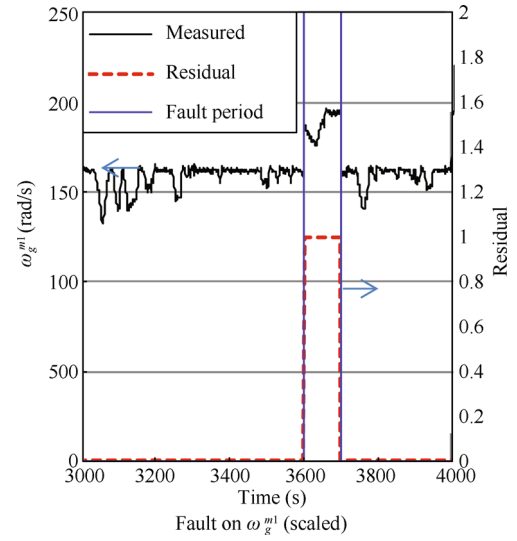


Fig. 8 Simultaneous faults in  $\omega_r^{m2}$  and  $\omega_g^{m1}$ , both of them scaled by a factor of 1.2 (faults 2b and 3b), fault duration: 3600–3700 s

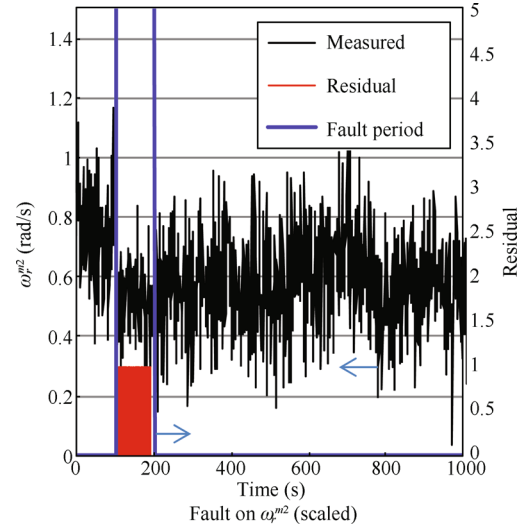


Fig. 9 Fault in the generator speed  $\omega_r^{m2}$ , when scaled by a factor of 0.8 (faults 2b), fault duration: 100–200 s

## 5.2 Pitch position actuators (by SVM and observer)

In this section, FDI results of the pitch position actuators are discussed by two approaches: SVM and Kalman-like observer. The necessity of using an observer to estimate this fault is due to the fact that only the transitory measurements are affected by drifts in  $\zeta$  and  $w_n$ , which seems to be fast. Training of transitions by SVM would require a number of simulations under different conditions. As specified in the observer development section, this fault cannot be detected by the observer in Zone II (while  $\beta^d = 0$ ). Therefore, the fault scenarios are realised mainly in Zone III, but not exclusively. The observer gain is set at zero in interval II, by making a test on the value of  $\beta^d$  and the estimated states are reinitialised at their nominal values when  $\beta^d = 0$ .

No test about the zones is done for SVM. Fig. 11 shows the desired pitch position  $\beta^d$  which presents high oscillations. Note that these oscillations that are due to measurements noise and control dynamics are expected to create oscillations in the estimates of the observer. The estimates by the observer are therefore filtered using a first order filter with  $\tau = 20T_s$  s.

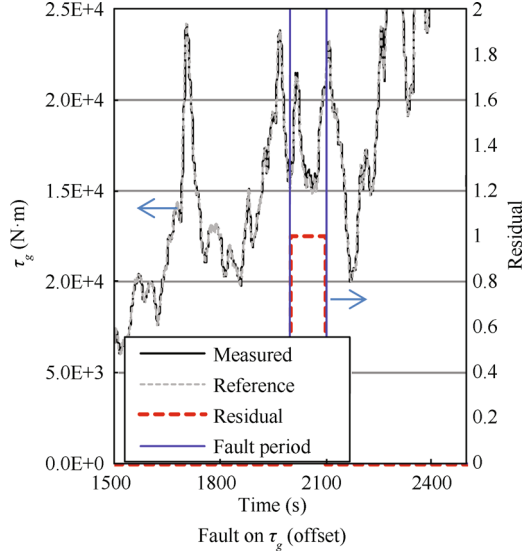


Fig. 10 Fault in the converter torque actuator (fault 4a), with an offset = 200 N·m, fault duration: 2000–2100 s

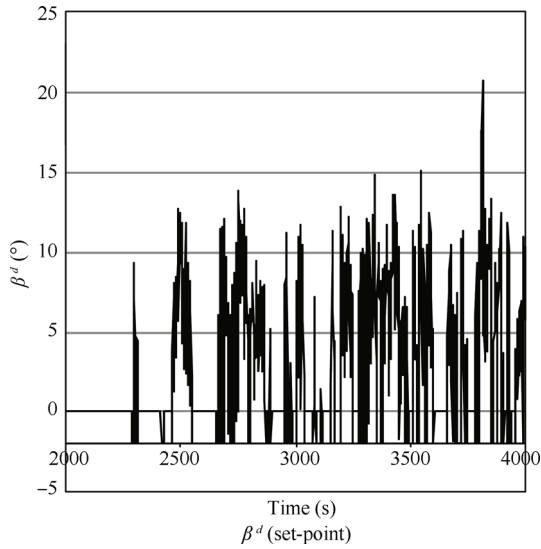


Fig. 11 Desired pitch position  $\beta^d$

Figs. 12–15 show a first scenario where an abrupt drift occurs in  $w_n$  and  $\zeta$  from their nominal values [11.11 rad/s, 0.6] to [6, 0.3]. The fault occurs at 2460 s for 100 s. This fault is assumed to be due to a severe failure in the hydraulic system. Fig. 12 shows that the estimation of  $w_n$  by the Kalman-like observer is clearly affected by the fault but it is noisy, similarly for  $\zeta$  (Fig. 13). Note that the observer gain is set to zero when  $\beta^d = 0$ , as for instance between 2000–2200 s.

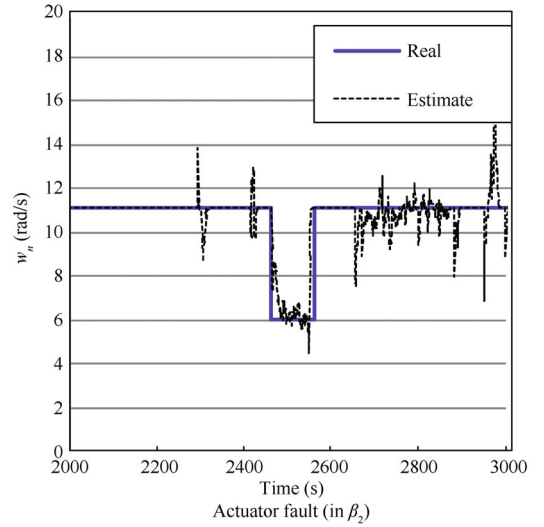


Fig. 12 Kalman-like observer results. Estimation of  $w_n$  when a fault occurs in the pitch actuator  $\beta_2$  at 2460 s (fault 5), where  $[w_n, \zeta]$  drift suddenly to [6, 0.3] over 100 s

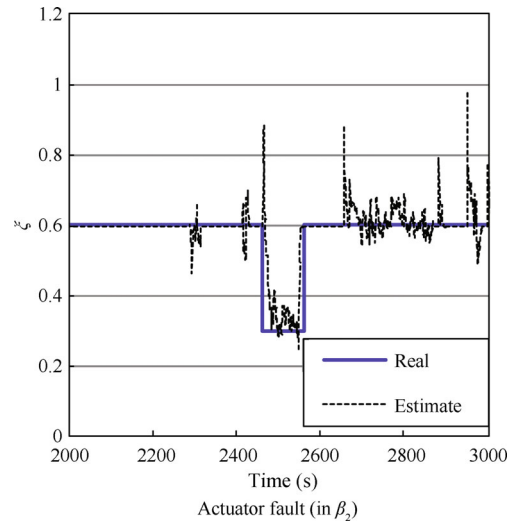


Fig. 13 Kalman-like observer results. Estimation of  $\zeta$  when a fault occurs in the pitch actuator  $\beta_2$  at 2460 s (fault 5), where  $[w_n, \zeta]$  drift suddenly to [6, 0.3] over 100 s

Based on the estimates of  $w_n$  and  $\zeta$ , a residual can be calculated for the pitch actuator fault, using the difference between  $w_n$  or  $\zeta$  and their nominal values. In Fig. 14, the residual is chosen to be equal to one if both of these conditions are verified:  $|\Delta\zeta| > 0.2$  and  $|\Delta w_n| > 2$ . Based on this assumption, the detection time by the observer is approximately 4 s. Fig. 15 shows the residual obtained by SVM, where the detection time is estimated to be 3.94 s. However, the SVM residual is oscillating and soon after a first detection it goes back to zero and continues oscillating, but without false alarms. Note that the presence of false alarms using the observer cannot completely be avoided, as the employment of important filtering will delay the detection.

It can be concluded in this example that the observer FDI results are more precise than those of SVM. Moreover, the estimates of  $w_n$  or  $\zeta$  can be useful for fault reconfiguration.

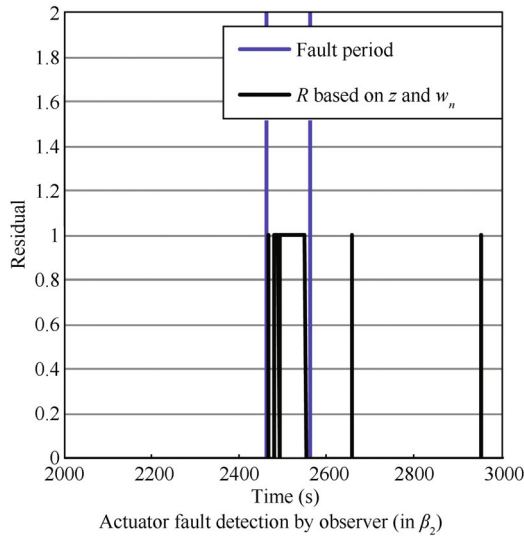


Fig. 14 FDI results using the Kalman-like observer. Detection of actuator fault at 2460 s in  $\beta_2$  (fault 5) where  $[w_n, \zeta]$  drift suddenly to  $[6, 0.3]$  over 100 s

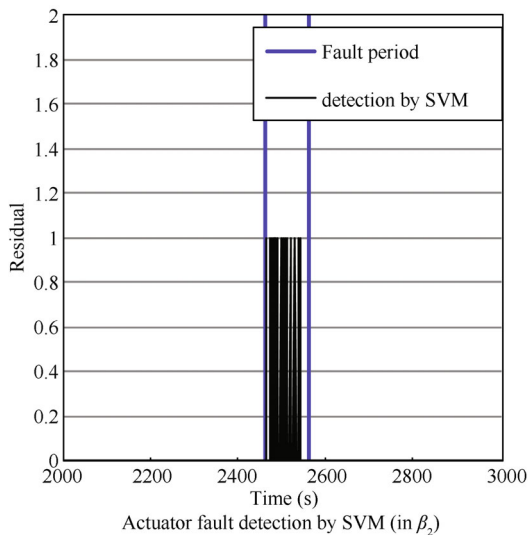


Fig. 15 FDI results using SVM. Detection of actuator fault at 2460 s in  $\beta_2$  (fault 5) where  $[w_n, \zeta]$  drift suddenly to  $[6, 0.3]$  over 100 s

In a second scenario, the values of  $w_n$  and  $\zeta$  are supposed to drift smoothly (over 30 s) from their nominal values  $[11.11, 0.6]$  to  $[7, 0.4]$ , which is assimilated to the presence of high air content in the oil of the hydraulic system (Figs. 16 and 18). This fault is assumed less severe than the first scenario. The estimates of  $w_n$  and  $\zeta$  are shown in Figs. 16 and 17, respectively. Due to oscillations and the fact that the fault (air content in the oil) appears relatively quickly (over 30 s), it cannot be distinguished from a sudden fault (Figs. 12–15). The final values of  $w_n$  and  $\zeta$  seem however to be well estimated in both fault types, which can be useful for fault reconfiguration.

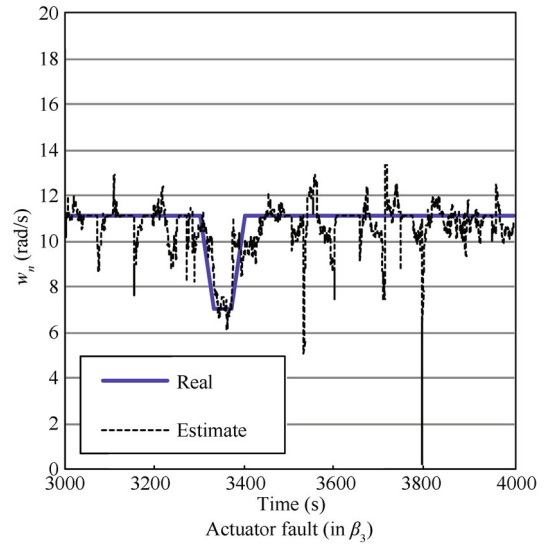


Fig. 16 Kalman-like observer results. Estimation of  $w_n$  when a fault occurs in the pitch actuator  $\beta_3$  at 3300 s (fault 5), where  $[w_n, \zeta]$  drift suddenly to  $[7, 0.4]$  over 100 s

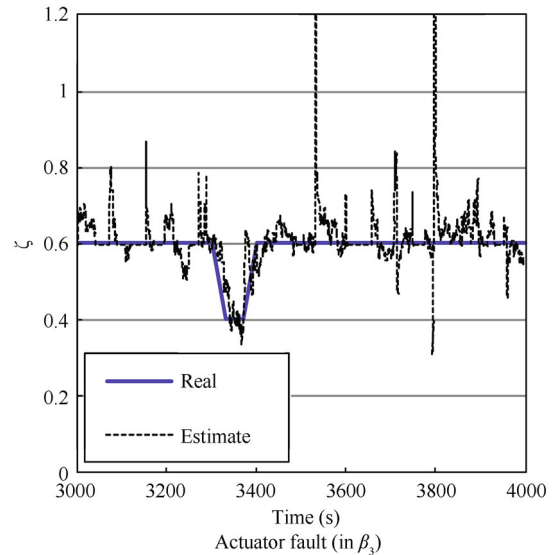


Fig. 17 Kalman-like observer results. Estimation of  $\zeta$  when a fault occurs in the pitch actuator  $\beta_3$  at 3300 s (fault 5), where  $[w_n, \zeta]$  drift suddenly to  $[7, 0.4]$  over 100 s

Fig. 18 shows the residual as obtained by the observer (based on the conditions  $|\Delta\zeta| > 0.2$  and  $|\Delta w_n| > 2$ ) and Fig. 19 the residual obtained from SVM in this scenario. Both residuals are oscillating, with some false alarms using the observer. This slow drift fault is therefore slightly more difficult to estimate, which indicates that small offset fault (resulting at the beginning of the drift) cannot be detected.

## 6 Conclusions

The wind energy is profitable if the technology of the turbines is optimized and online supervised. In view of the large number of components in the system, large number of frequent but noisy measurements besides the system disturbances, a good method of supervision should be used

for fault detection and isolation. In this work, sensor faults were treated by SVM which was found to be a good method for pattern recognition and to be adapted to online implementation. For detection of sensors stuck at some measurement, the derivative of the measurement is used in the training data. For detection of scaled measurements, the training data contain the difference between the measurement and the set-point. Offset in the actuator torque was also learned by SVM, based on the difference between the desired and real values.

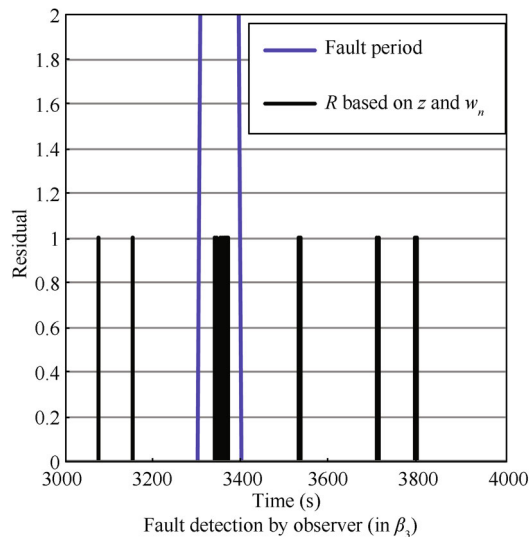


Fig. 18 FDI results using the Kalman-like observer. Detection of actuator fault at 3300 s in  $\beta_3$  (fault 5) where  $[w_n, \zeta]$  drift suddenly to  $[7, 0.4]$  over 100 s

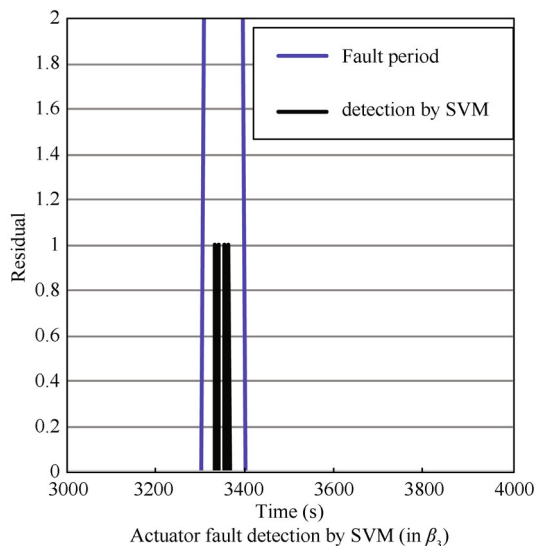


Fig. 19 FDI results using SVMs. Detection of actuator fault at 3300s in  $\beta_3$  (fault 5) where  $[w_n, \zeta]$  drift suddenly to  $[7, 0.4]$  over 100 s

For the pitch position actuator, two methods were used: SVM and Kalman-like observer. Both methods give comparable results, with the observer having a higher sensitivity to the fault, but more false alarms. The observer interest

is however incontestable for fault configuration.

## References

- [1] Y. Amirat, M. E. H. Benbouzid, E. Al-Ahmar, B. Bensaker, S. Turri. A brief status on condition monitoring and fault diagnosis in wind energy conversion systems. *Renewable and Sustainable Energy Reviews*, vol. 13, no. 9, pp. 2629–2639, 2009.
- [2] Z. Hameed, Y. S. Hong, Y. M. Cho, S. H. Ahn, C. K. Song. Condition monitoring and fault detection of wind turbines and related algorithms: A review. *Renewable and Sustainable Energy Reviews*, vol. 13, no. 1, pp. 1–39, 2009.
- [3] B. Lu, Y. Y. Li, X. Wu, Z. Zang. A review of recent advances in wind turbine condition monitoring and fault diagnosis. In *Proceedings of the Power Electronics and Machines in Wind Applications*, IEEE, Lincoln, NE, USA, pp. 1–7, 2009.
- [4] P. F. Odgaard, J. Stoustrup, R. Nielsen, C. Damgaard. Observer based detection of sensor faults in wind turbines. In *Proceedings of European Wind Energy Conference*, Marseille, France, 2009.
- [5] K. Rothenhagen, F. W. Fuchs. Current sensor fault detection and reconfiguration for a doubly fed induction generator. In *Proceedings of IEEE Power Electronics Specialists Conference (PESC)*, IEEE, Orlando, FL, USA, pp. 2732–2738, 2007.
- [6] P. Poure, P. Weber, D. Theilliol, S. Saadate. Fault-tolerant power electronic converters: Reliability analysis of active power filter. In *Proceedings of IEEE International Symposium on Industrial Electronics ISIE*, IEEE, Vigo, Spain, pp. 3174–3179, 2007.
- [7] S. H. Li, D. C. Wunsch, E. A. O'Hair, M. G. Giesselmann. Using neural networks to estimate wind turbine power generation. *IEEE Transactions on Energy Conversion*, vol. 16, no. 3, pp. 276–282, 2001.
- [8] M. Schlechtingen, I. F. Santos. Comparative analysis of neural network and regression based condition monitoring approaches for wind turbine fault detection. *Mechanical Systems and Signal Processing*, vol. 25, no. 5, pp. 1849–1875, 2011.
- [9] P. F. Odgaard, J. Stoustrup. Estimation of uncertainty bounds for the future performance of a power plant. *IEEE Transactions on Control Systems Technology*, vol. 17, no. 1, pp. 199–206, 2009.
- [10] A. Kusiak, W. Y. Li. The prediction and diagnosis of wind turbine faults. *Renewable Energy*, vol. 36, no. 1, pp. 16–23, 2011.
- [11] T. Barszcz, R. B. Randall. Application of spectral kurtosis for detection of a tooth crack in the planetary gear of a wind turbine. *Mechanical Systems and Signal Processing*, vol. 23, no. 4, pp. 1352–1365, 2009.
- [12] P. F. Odgaard, J. Stoustrup, M. Kinnaert. Fault tolerant control of wind turbines: A benchmark model. In *Proceedings of the 7th IFAC Symposium on Fault Detection, Supervision and Safety of Technical Processes*, IFAC, Barcelona, Spain, pp. 155–160, 2009.

- [13] N. Laouti, N. Sheibat-Othman, S. Othman. Support vector machines for fault detection in wind turbines. In *Proceedings of the 18th IFAC World Congress*, IFAC, Università Cattolica del Sacro Cuore, Milan, Italy, pp. 7067–707, 2011.
- [14] X. Zhang, Q. Zhang, S. Zhao, R. M. G. Ferrari, M. M. Polycarpou, T. Parisini. Fault detection and isolation of the wind turbine benchmark: An estimation-based approach. In *Proceedings of the 18th IFAC World Congress*, IFAC, Università Cattolica del Sacro Cuore, Milan, Italy, pp. 8295–8300, 2011.
- [15] A. A. Ozdemir, P. Seiler, G. J. Balas. Wind turbine fault detection using counter-based residual thresholding. In *Proceedings of the 18th IFAC World Congress*, IFAC, Università Cattolica del Sacro Cuore, Milan, Italy, pp. 8289–8294, 2011.
- [16] P. L. Negre, V. Vicenc Puig, I. Pineda. Fault detection and isolation of a real wind turbine using LPV observers. In *Proceedings of the 18th IFAC World Congress*, IFAC, Universitat Politècnica de Catalunya, Milan, Italy, pp. 12372–12379, 2011.
- [17] S. Simani, P. Castaldi, M. Bonf. Hybrid model-based fault detection of wind turbine sensors. In *Proceedings of the 18th IFAC World Congress*, IFAC, Università Cattolica del Sacro Cuore, Milan, Italy, pp. 7061–7066, 2011.
- [18] B. Ayalew, P. Pisu. Robust fault diagnosis for a horizontal axis wind turbine. In *Proceedings of the 18th IFAC World Congress*, IFAC, Università Cattolica del Sacro Cuore, Milan, Italy, pp. 7055–7060, 2011.
- [19] J. F. Dong, M. Verhaegen. Data driven fault detection and isolation of a wind turbine benchmark. In *Proceedings of the 18th IFAC World Congress*, IFAC, Università Cattolica del Sacro Cuore, Milan, Italy, pp. 7086–7091, 2011.
- [20] W. Chen, S. X. Ding, A. H. A. Sari, A. Naik, A. Q. Khan, S. Yin. Observer-based FDI schemes for wind turbine benchmark. In *Proceedings of 18th IFAC World Congress*, IFAC, Università Cattolica del Sacro Cuore, Milan, Italy, pp. 7073–7078, 2011.
- [21] X. Sun, H. J. Marquez, T. Chen, M. Riaz. An improved PCA method with application to boiler leak detection. *ISA Transactions*, vol. 44, no. 3, pp. 379–397, 2005.
- [22] S. X. Ding. *Model-based Fault Diagnosis Techniques*, Berlin: Springer, 2008.
- [23] S. Simani, C. Fantuzzi, R. J. Patton. *Model-based Fault Diagnosis in Dynamic Systems Using Identification Techniques*, XIV, 298. Series: Advances in Industrial Control. London: Springer-Verlag, 2002.
- [24] R. Isermann. *Fault-diagnosis Systems from Fault Detection to Fault Tolerance*, New York: Springer Verlag, 2006.
- [25] O. A. Z. Sotomayor, D. Odloak. Observer-based fault diagnosis in chemical plants. *Chemical Engineering Journal*, vol. 112, no. 1–3, pp. 93–108, 2005.
- [26] R. J. Patton, J. Chen. On eigenstructure assignment for robust fault diagnosis. *International Journal of Robust Non-linear Control*, vol. 10, no. 14, pp. 1193–1208, 2000.
- [27] C. Edwards, S. K. Spurgeon, R. J. Patton. Sliding mode observers for fault detection and isolation. *Automatica*, vol. 36, no. 4, pp. 541–553, 2000.
- [28] Y. J. Huang, G. V. Reklaitis, V. A. Venkatasubramanian. Heuristic extended Kalman filter based estimator for fault identification in a fluid catalytic cracking unit. *Industrial & Engineering Chemistry Research*, vol. 42, no. 14, pp. 3361–3371, 2003.
- [29] P. M. Frank. Fault diagnosis in dynamic systems using analytical and knowledge-based redundancy – A survey and some new results. *Automatica*, vol. 26, no. 3, pp. 459–474, 1990.
- [30] E. A. Garcia, P. M. Frank. Deterministic nonlinear observer-based approaches to fault diagnosis: A survey. *Control Engineering Practice*, vol. 5, no. 5, pp. 663–670, 1997.
- [31] H. Hammouri, P. Kaboré, S. Othman, J. Biston. Failure diagnosis and nonlinear observers: Application to a hydraulic process. *Journal of the Franklin Institute*, vol. 339, no. 4–5, pp. 455–478, 2002.
- [32] P. Kaboré, S. Othman, T. F. McKenna, H. Hammouri. Observer based fault diagnosis for a class of non-linear systems: Application to a free radical copolymerisation reaction. *International Journal of Control*, vol. 73, no. 9, pp. 787–803, 2000.
- [33] C. De Persis, A. Isidori. A geometric approach to nonlinear fault detection and isolation. *IEEE Transactions of Automatic Control*, vol. 46, no. 6, pp. 853–865, 2001.
- [34] V. N. Vapnik, A. Chervonenkis. A note on one class of perceptrons. *Automation & Remote Control*, vol. 25, pp. 821–837, 1964.
- [35] B. E. Boser, I. M. Guyon, V. N. Vapnik. A training algorithm for optimal margin classifiers. In *Proceedings of the 5th Workshop on Computational Learning Theory*, ACM Press, Pittsburgh, PA, USA, pp. 144–152, 1992.
- [36] V. N. Vapnik. *The Nature of Statistical Learning Theory*, New York: Springer-Verlag, 1995.
- [37] A. Widodo, B. S. Yang. Support vector machine in machine condition monitoring and fault diagnosis. *Mechanical Systems and Signal Processing*, vol. 21, no. 6, pp. 2560–2574, 2007.
- [38] L. Meng, Q. H. Wu. Fast training of support vector machines using error-center-based optimization. *International Journal of Automation and Computing*, vol. 2, no. 1, pp. 6–12, 2005.
- [39] X. Chen, T. Limchimchol. Monitoring grinding wheel redress-life using support vector machines. *International Journal of Automation and Computing*, vol. 3, no. 1, pp. 56–62, 2006.
- [40] X. H. Huang, X. J. Zeng, M. Wang. SVM-based identification and un-calibrated visual servoing for micro-manipulation. *International Journal of Automation and Computing*, vol. 7, no. 1, pp. 47–54, 2010.
- [41] H. Hammouri, G. Bornard. A high gain observer for a class of uniformly observable systems. In *Proceedings of the 30th IEEE Conference on Decision and Control*, IEEE, Brighton UK, pp. 1494–1496, 1991.



- [42] V. Cherkassky, Y. Ma. Practical selection of SVM parameters and noise estimation for SVM regression. *Neural Networks*, vol. 17, no. 1, pp. 113–126, 2004.
- [43] H. Hammouri, J. De Leon Morales. Observer synthesis for state-affine systems. In *Proceedings of the 29th IEEE Conference on Decision and Control*, IEEE, Honolulu, Hawaii, USA, pp. 784–785, 1990.



**Nassim Laouti** received his M.Sc. degree in electronics engineering from University of Science and Technology Houari Boumediene, Algiers in 2006, and in automatic control from University of Lyon, France in 2008. He received his Ph.D. degree in automatic control from University of Lyon, France in 2012. Since 2013 he has been an associate professor at the University of Science and Technology

Houari Boumediene.

His research interests include artificial intelligence, machine learning, support vector machines, statistical signal processing, nonlinear system identification and control, wind turbines, wastewater treatment plant and data mining.

E-mail: nassimlat@yahoo.fr



**Sami Othman** received his M.Sc. degree in automatic control from University of Lyon, France in 1988. He received his Ph.D. degree from University of Lyon in 1992, where he proposed the high gain observer for the first time. Since 1993 he has been an associate professor at LAGEP in the University of Lyon in the domain of automatic control.

His research interests include process monitoring using observers and linear and nonlinear control with

applications to chemical plants as well as wind turbines.

E-mail: Othman@lagep.univ-lyon1.fr



**Mazen Alamir** received his M.Sc. degree in mechanics from University of Grenoble, France in 1990, and in aeronautics from University of Toulouse, France in 1992. He received his Ph.D. degree in nonlinear model predictive control in 1995. Since 1996, he has been a CNRS research associate in the Control Systems Department of Gipsa-lab, Grenoble. He is member of the IFAC Technical Committee on Nonlinear Systems and serves as head of the “Nonlinear Systems and Complexity” Research Group of the Control Systems Department of Gipsa-lab, Grenoble.

His research interests include degree model predictive control, receding horizon observers, nonlinear hybrid systems, signature based diagnosis, optimal cancer treatment as well as industrial applications.

E-mail: mazen.alamir@gipsa-lab.grenoble-inp.fr



**Nida Sheibat-Othman** received her M.Sc. degree in automatic control from University of Lyon, France in 1997. She received her Ph.D. degree in chemical engineering from University of Lyon in 2000. Since 2002, she has been a CNRS researcher at LAGEP in the University of Lyon in the domain of modeling, monitoring and control of chemical plants.

Her research interests include multiscale modeling, population balance modeling, online sensors, monitoring using observers and data-based support vector machines, and nonlinear control.

E-mail: nida.othman@lagep.univ-lyon1.fr (Corresponding author)