ScienceDirect

IFAC PapersOnLine CONFERENCE PAPER ARCHIVE

IFAC PapersOnLine 51-24 (2018) 381–388

Fault Diagnosis of a Wind Turbine Simulated Model via Neural Networks ⋆

Silvio Simani * **Cihan Turhan** **

* *Department of Engineering, University of Ferrara, Ferrara (FE) 44122 Italy (e–mail: silvio.simani@unife.it).*

** *Mechanical Engineering, Izmir Institute of Technology, 35430 Izmir, Turkey (e–mail: cihanturhan@iyte.edu.tr)*

Abstract: The fault diagnosis of wind turbine systems has been proven to be a challenging task and motivates the research activities carried out through this work. Therefore, this paper deals with the fault diagnosis of wind turbines, and it proposes viable solutions to the problem of earlier fault detection and isolation. The design of the fault indicator involves a data–driven approach, as it represents an effective tool for coping with a poor analytical knowledge of the system dynamics, together with noise and disturbances. In particular, the data–driven proposed solution relies on neural networks that are used to describe the strongly nonlinear relationships between measurement and faults. The chosen network architecture belongs to the nonlinear autoregressive with exogenous input topology, as it can represent a dynamic evolution of the system along time. The developed fault diagnosis scheme is tested by means of a high–fidelity benchmark model, that simulates the normal and the faulty behaviour of a wind turbine. The achieved performances are compared with those of other control strategies, coming from the related literature. Moreover, a Monte Carlo analysis validates the robustness of the proposed solutions against the typical parameter uncertainties and disturbances.

Keywords: Fault diagnosis, wind turbine benchmark, neural networks, fault estimation, robustness and reliability.

1. INTRODUCTION

The worldwide increased level of wind generated energy in power grids induces further requirements in terms of reliability of wind turbines. Wind turbines should have the capability to generated the desired value of electrical power continuously, depending on the current wind speed level and on the grid demand.

As a consequence, the possible faults affecting the system have to be properly identified and treated, before they endanger the correct functioning of the turbines or become critical faults. Wind turbines in the megawatt size are extremely expensive systems, therefore their availability and reliability must be high, in order to assure the maximisation of the generated power while minimising the Operation and Maintenance (O & M) services. Alongside the fixed costs of the produced energy, mainly due to the installation and the foundation of the wind turbine, the O & M costs could increase the total energy cost up to about the 30%, particularly considering the offshore installation (Simani et al. (2017)).

These considerations motivate the introduction of fault diagnosis system coupled with fault tolerant controllers. Currently, most of the turbines feature a simply conservative approach against faults that consists in the shutdown of the system to wait for maintenance service. Hence, effective strategies coping with faults have to be studied and developed, for improving the turbine performance, particularly in faulty working conditions. Their benefits would concern the prevention of critical failures that jeopardise wind turbine components, thus avoiding unplanned replacement of functional parts, as well as the reduction of the O & M costs and the increment of the energy production. The advent of computerised control, communication networks and information techniques brings interesting challenges concerning the development of novel real–time monitoring and fault tolerant control design strategies for industrial processes.

Indeed, in the recent years, many contributions have been proposed related to the topics of fault diagnosis of wind turbines, see *e.g.* (Simani and Farsoni (2018)). Some of them highlight the difficulties to achieve the diagnosis of particular faults, *e.g.* those affecting the drive–train, at wind turbine level. However these fault are better dealt with at wind farm level, when the wind turbine is considered in comparison to other wind turbine of the wind farm (Odgaard and Stoustrup (2015)). Moreover, fault tolerant control of wind turbines has been investigated *e.g.* in (Odgaard and Stoustrup (2015)) and international competitions on these issues arose (Odgaard and Stoustrup (2012)).

Hence, the fault diagnosis of wind turbine systems has been proven to be a challenging task and motivates the research activities carried out through this work (Simani and Castaldi (2017)). In particular, this paper deals with

⋆ Invited paper for the special session on "Industrial Fault Diagnosis and Fault–tolerant Control" organised by Christophe Aubrun and Vicenc Puig.

Peer review under responsibility of International Federation of Automatic Control.
10.1016/j.ifacol.2018.09.605

the fault diagnosis of wind turbines, and it proposes viable solutions to the problem of earlier fault detection and isolation (Simani and Farsoni (2018)). The design of the fault indicator involves a data–driven approach (Simani et al. (2018)), as it represents an effective tool for coping with a poor analytical knowledge of the system dynamics, together with noise and disturbances. The data–driven proposed solution relies on neural networks that are used to describe the strongly nonlinear relationships between measurement and faults (Simani and Castaldi (2018)). The chosen network architecture belongs to the nonlinear autoregressive with exogenous input topology, as it can represent a dynamic evolution of the system along time. The training of the neural network fault estimators exploits the back–propagation Levenberg–Marquardt algorithm, that processes a set of acquired target data.

The developed fault diagnosis scheme is tested by means of a high–fidelity benchmark model, that simulates the normal and the faulty behaviour of a wind turbine. The achieved performances are compared with those of other control strategies, coming from the related literature. Moreover, a Monte Carlo analysis validates the robustness of the proposed solutions against the typical parameter uncertainties and disturbances.

The work is organised as follows. Section 2 recalls the wind turbine benchmark simulator. Section 3 describes the fault diagnosis scheme relying on neural network structures. The achieved results are reported in Section 4. Comparisons with different Fault Detecton and Isolation (FDI) strategies are also reported. Finally, Section 5 concludes the paper by summarising the main achievements of the work, and providing some suggestions for further research topics.

2. WIND TURBINE SIMULATOR

The single wind turbine benchmark model considered in this study is described in detail in (Odgaaard and Shafiei (2015)). It has been implemented in Matlab/Simulink environment and proposes a realistic simulator for a wind turbine system, as well as some common fault scenarios. It presents a specific three–blade horizontal–axis variable–speed pitch–controlled wind turbine with a full converter generator. The considered components have been reduced in the benchmark model considered in this study. The tower supports the nacelle, that contains the control systems and the equipment to convert energy from the rotating blades fixed in the hub to electrical energy for the grid. On the nacelle an anemometer provides the measurements of the current wind speed at the hub height. The internal components of the nacelle consist of a gear box that connects the rotor main shaft to the generator, adapting its torque and speed values; the generator, that converts the energy into the electrical form; a converter and a transformer that connect the turbine to the grid; finally the controller adjusts the pitch angle and the generator torque in order to follow the power reference.

The block diagram of Figure 1 shows how the main components are connected each other and the input/output variables which indicate the relationship among the blocks.

The wind turbine model of the benchmark considered in this work consists of four submodels: the wind model,

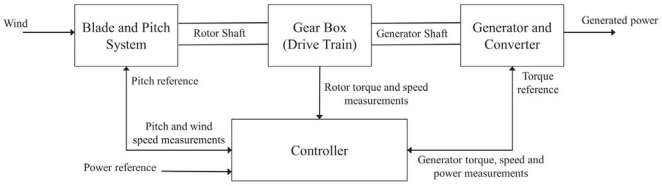

Fig. 1. Wind turbine simulator blocks.

the blade and pitch model, the drive–train model and the generator model. The wind is considered a stochastic process, with the additive contributions of the effects of wind shear and tower shadows. A complete description of the wind model and the wind turbine simulator is provided in (Odgaaard and Shafiei (2015)).

The measurements available to the monitoring system come directly from several sensors or, in one case, they are obtained via estimation. In particular, for each of the three blades, a redundant couple of sensors measures the current pitch angle. Then, a couple of sensors measures the speed of the rotor and another one the speed of the generator, while a single sensor is available for the wind speed at hub height and another one for the generator torque. The wind torque measurements are estimated exploiting the hub anemometer. Table 1 reports a summary of the measured variables. The model of the measurements consists in the sum of the actual value with a white Gaussian noise.

Table 1. Wind turbine simulator measurements.

Wind Turbine Variable	Measurement Description
$\beta_{1,m1}$	Blade 1 pitch angle from sensor #1
$\beta_{1,m2}$	Blade 1 pitch angle from sensor #2
$\beta_{1,m1}$	Blade 2 pitch angle from sensor #1
$\beta_{1,m2}$	Blade 2 pitch angle from sensor #2
$\beta_{1,m1}$	Blade 3 pitch angle from sensor #1
$\beta_{1,m2}$	Blade 3 pitch angle from sensor #2
$\omega_{r,m1}$	Rotor shaft speed from encoder #1
$\omega_{r,m2}$	Rotor shaft speed from encoder #2
$\omega_{g,m1}$	Generator shaft speed from encoder #1
$\omega_{g,m2}$	Generator shaft speed from encoder #2
$\tau_{g,m}$	Generator shaft torque measurement
$P_{g,m}$	Generated power measurement
$v_{w,m}$	Wind Speed measurement at hub height
$\tau_{r,m}$	Aerodynamic rotor torque

In the benchmark model three kinds of actual faults can be simulated: namely sensor, actuator and system faults. They are modelled as additive or multiplicative faults and they involve different degrees of severity, so that they can yield to the turbine shutdown in case of serious fault, or they can be accommodate by the controller if the risk for the system safety is low.

Regarding the considered sensor faults, they affect the measurements of the pitch angles and the measurements of the rotor speed, in form of a fixed value or a scaling error. They represent a common fault scenario of wind turbines, but their severity is low and they should be easy to identify and accommodate. In particular, an electrical or mechanical faults in the pitch sensors, if not handled, results in the generation of a wrong pitch reference system by the controller with the consequence of a loss in the generated power. The speed of the rotor is measured by means of two redundant encoders, an offset faulty signal can affect these measurements when the encoder does not

detect the updated marker, while a gain factor faulty signal represents the reading of excessive markers each loop, due to dirt on the rotating part.

The considered actuator faults are modelled either as a fixed value or a changed dynamics of the transfer function. They affect the converter torque actuator as well as the pitch actuator. In the former case, the fault is located in the electronics of the converter, while in the latter case the fault is on the hydraulic system: it models the pressure drop in the hydraulic supply system (*e.g.* due to a leakage in hose or a blocked pump) or the excessive air content in the oil that causes the variation of the compressibility factor. The severity of these fault is of medium/high level.

Finally, the considered system fault concerns the drive–train, in form of a slow variation of the friction coefficient in time due to wear and tear (months or year, but for benchmarking reason in the model it has been accelerated up to some seconds). It results in a combined faulty signal affecting the rotor speed and the generator speed. It can be listed as an high severe fault, as it can yield to the breakdown of the drive–train, but in a long time. Tables 2 and 3 describe the considered faults and their types, corresponding to the affected measurements.

Table 2. Fault case description.

Fault nr.	Fault Description
1	Fixed value of the blade 1 pitch sensor 1
2	Scaling error of the blade 2 pitch sensor 2
3	Fixed value of the blade 1 pitch sensor 1
4	Fixed value of the rotor speed sensor 1
5	Combined scaling error of the rotor speed sensor 2 and the generator speed sensor 2
6	Pitch system changed response for the pitch actuator of the blade 2 due to air content in oil
7	Pitch system changed response for the pitch actuator 3 due to low pressure
8	Fixed value of the converter torque control signal
9	Changed dynamics of the drive–train

Table 3. Fault typology.

Fault nr.	Fault Type
1	Sensor fault
2	Sensor fault
3	Sensor fault
4	Sensor fault
5	Sensor fault
6	Actuator fault
7	Actuator fault
8	Actuator fault
9	System fault

2.1 Wind Turbine Overall Model

With these assumptions, the complete model of the system under analysis (Odgaard and Stoustrup (2015)) can be represented by means of a non–linear continuous-time function $\mathbf{f}_{wt}$, that describes the evolution of the turbine state vector $\mathbf{x}_{wt}$ excited by the input vector $\mathbf{u}$:

$$\{\ \dot{x}_{wt}(t) = \mathbf{f}_{wt}(\mathbf{x}_{wt}, \mathbf{u}(t))\mathbf{y}(t) = \mathbf{x}_{wt}(t) \qquad (1)$$

where the state of the system is considered equal to the monitored system output *i.e.* the rotor speed, the generator speed and the generated power: $\mathbf{x}_{wt}(t) = \mathbf{y}(t) = [\omega_{g,m1},\ \omega_{g,m2},\ \omega_{r,m1},\ \omega_{r,m2},\ P_{g,m}]$. On the other hand, the input vector $\mathbf{u}(t) = [\beta_{1,m1},\ \beta_{1,m2},\ \beta_{2,m1},\ \beta_{2,m2},\ \beta_{3,m1},\ \beta_{3,m2},\ \tau_{g,m}]$ contains the measurements of the pitch angles from the three sensor couples as well as the measured torque. These vectors are sampled for obtaining a number of N input–output data $\mathbf{u}[k]$, $\mathbf{y}[k]$ with $k = 1, \ldots, N$, in order to implement the data–driven estimators at the sampling time T.

3. FAULT DIAGNOSIS VIA NEURAL NETWORKS

This work proposes a data–driven approach that is based on neural networks and is used to implement the fault diagnosis block. In this section, after a brief introduction on the general structure and the architecture of a Nonlinear AutoRegressive with eXogenous input (NARX) network is reported, as it represents, in combination with the back–propagation Levenberg–Marquardt training algorithm, the exploited solution for the implementation of the neural network fault estimators.

3.1 Neural Network Structure

In this work, a set of neural network estimators is designed and trained in order to reproduce the behaviour of the systems under investigation, thus accomplishing the modelling and identification task. The structure of the i–th single neuron is also called *perceptron*. It features a MISO system where the output y_i is computed as a function f of the weighted sum v_i of all the n_i neuron inputs $u_{i,1}, \ldots, u_{i,n_i}$, with the associated weights $w_{i,1}, \ldots, w_{i,n_i}$. The function f, denominated *activation function*, represents the engine of the neuron.

A structural categorisation of neural networks concerns the way in which their elements are connected each others. In a *feed-forward network*, also called *multilayer perceptron*, neurons are grouped into unidirectional layers. The first of them, namely the *input* layer, is fed directly by the network inputs, then each successive *hidden* layer takes the inputs from the neurons of the previous layer and transmits the output to the neurons of the next layer, up to the last *output* layer, in which the final network outputs are produced. Therefore, neurons are connected from one layer to the next, but not within the same layer. The only constraint is the number of neurons in the output layer, that has to be equal to the number of actual network outputs. On the other hand, *recurrent networks* are multilayer networks in which the output of some neurons is fed back to neurons belonging to previous layers, thus the information flow in forward as well as in backward directions allowing a dynamic memory inside the network.

A noteworthy intermediate solution is provided by the multilayer perceptron with a tapped delay line, which is a feed–forward network whose inputs come from a delay line. This kind of network represents a suitable tool to model, or predict, the evolution of a dynamic system. In particular the open loop Nonlinear AutoRegressive with eXogenous inputs (NARX) network belongs to this latter category as its inputs are delayed samples of the system inputs and outputs. Indeed, if properly trained, a NARX network can estimate the current (or the next) system output on the basis of the acquired past measurements of system inputs and outputs.

In general, by considering a MIMO system, the elaborations of the open–loop NARX network follow the law:

$$\hat{\mathbf{f}}(k) = f_{net}\big(\mathbf{u}(k), \ldots, \mathbf{u}(k-d_u), \mathbf{y}(k-1), \ldots, \mathbf{y}(k-d_y)\big) \quad (2)$$

where $\hat{\mathbf{f}}(k)$ is the estimation of the process faults, $\mathbf{u}$ and $\mathbf{y}$ are the measured system inputs and outputs, k is the time step, d_u and d_y are the number of delay of inputs and outputs, respectively. $f_{net}(\cdot)$ is the function realized by the network, that depends on the layer architecture, the number of neurons, their weights and their activation functions. Therefore, as highlighted by Eq. (2), the neural network is used as fault estimator.

The parameters on which the designer can act concerns the overall architecture (number of neurons, connections between layers), while the value of the weights inside each neuron are derived from the network training.

3.2 Fault Diagnosis System

In the following the monitored systems, *i.e.* the wind turbine under diagnosis, is assumed to be affected by additive faults on the input (actuator) and output (sensor) measurements, as represented in Fig. 2, in the form of Eqs. 3:

$$\begin{cases} \mathbf{u}(k) = \mathbf{u}^*(k) + \mathbf{f}_u(k) \\ \mathbf{y}(k) = \mathbf{y}^*(k) + \mathbf{f}_y(k) \end{cases} \quad (3)$$

where $\mathbf{u}^*(k), \mathbf{y}^*(k)$ are the actual unmeasurable variables, $\mathbf{u}(k)$ and $\mathbf{y}(k)$ represent the sensor acquisitions, affected by both the measurement noise and the faults. $\mathbf{f}_u(k), \mathbf{f}_y(k)$ are additive signals, that assume values different from zero only in presence of faults.

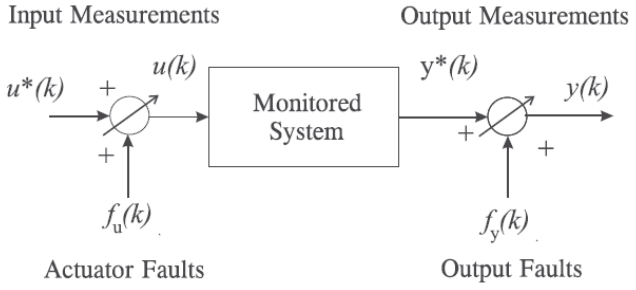

Fig. 2. Fault model.

Figure 2 shows the general scheme with the faults affecting the system under diagnosis, *i.e.* the wind turbine, as additive signals on the input (actuator) and output measurements.

Among the different approaches to generate the residual signals, summarised in Section 1 the solution adopted in this work exploits neural network models, which provide an on–line estimation of the faulty signals. Hence, as shown in Fig. 3 residuals $\mathbf{r}$ are generated by means of the neural networks used as fault estimators:

$$\mathbf{r}(k) = \hat{\mathbf{f}}(k) \quad (4)$$

Figure 3 highlights the residual generation scheme, achieved by using the neural network structures, which are thus

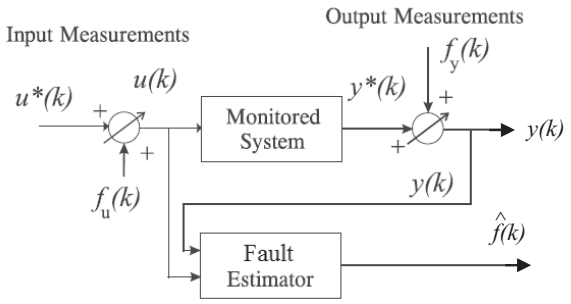

Fig. 3. Residual generation scheme based on the fault estimator.

exploited as fault estimators. The fault diagnosis process involves, as first step, the fault detection task. It is performed here by using a proper thresholding logic operating on the residuals after their elaboration into a proper evaluation function:

$$\mathbf{r}_e(k) = F(\mathbf{r}(k)) \quad (5)$$

where the proposed function F can be the moving average or variance, in case of neural networks, as explained in Section 4. Then, the occurrence of the i–th fault can be detected according to:

$$\begin{cases} \bar{r}_{e_i} - \delta\sigma_{r_i} \le r_{e_i} \le \bar{r}_{e_i} + \delta\sigma_{r_i} & \text{fault–free} \\ r_{e_i} < \bar{r}_{e_i} - \delta\sigma_{r_i} \text{ or } r_{e_i} > \bar{r}_{e_i} + \delta\sigma_{r_i} & \text{faulty} \end{cases} \quad (6)$$

where the i–th item r_{e_i} of the residual vector $\mathbf{r}_e$ is considered a random variable, whose unknown mean $\bar{r}_{e_i}$ and variance $\sigma^2_{r_i}$ can be estimated in fault–free condition, after the acquisition of N samples, as follows:

$$\begin{cases} \bar{r}_{e_i} = \dfrac{1}{N}\displaystyle\sum_{k=1}^{N} r_{e_i}(k) \\ \sigma^2_{r_i} = \dfrac{1}{N}\displaystyle\sum_{k=1}^{N} (r_{e_i}(k) - \bar{r}_{e_i})^2 \end{cases} \quad (7)$$

The tolerance parameter $\delta \ge 2$ has to be properly tuned in order to separate the fault-free from the faulty condition. The δ value determines the trade–off between the false alarm rate and the fault detection probability. A common choice of δ relies on the three–sigma rule, otherwise extensive simulations can be performed to optimise the δ value.

Consequently to the fault detection, the fault isolation task is achieved by means of two observer schemes. Faults are here subdivided into two main groups: the faults concerning the inputs $\mathbf{f}_u$ and the faults concerning the output $\mathbf{f}_y$. Following the generalised observer scheme of Fig. 4, in order to uniquely isolate one of the input faults, under the assumption that no output faults occur in the meanwhile, a bank of MISO estimators is used, whose number is equal to the number of input faults to be isolate. Then, the i-th fault estimator is driven by all but the i–th input, so that, when the i–th fault occurs, its residual signal is the only one that does not detect the fault. In particular, when the i–th neural network estimator, insensitive to the i–th fault has to be designed, all but the i-th input, in addition to the output, are considered in the training of the neural networks.

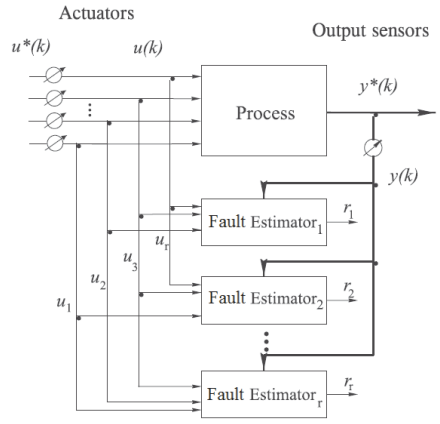

Fig. 4. General observer scheme for FDI.

Figure 4 shows the general observer scheme where the fault estimators are driven by all but one input, so that the relative residual is insensitive only to the fault affecting that input. It is worth noting that multiple faults occurring at the same time cannot be correctly isolated, using this configuration.

On the other hand, in order to isolate one or multiple output faults, under the assumption of no input faults occurring, another bank of estimator is used. In this case, the i–th estimator provides directly the i–th residual as it is driven by all the system inputs and only the i–th output. This configuration is better known as dedicated observer scheme.

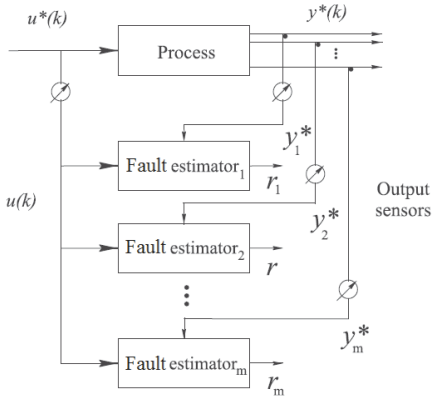

Fig. 5. Dedicated observer scheme for FDI.

Figure 5 shows the dedicated observer scheme where all the fault estimators are driven by all inputs and each of them generates the residual relative to only one output faulty signal.

The isolation capabilities of the adopted observer banks can be summarised by means of the so–called *fault signature*, depicted in Table 4, where each entry that is characterised by a value equal to 1 means that the considered residual is sensitive to the fault (zero otherwise), under the hypothesis above mentioned.

A fault sensitivity analysis (Simani and Farsoni (2018)), has to be executed before the design of the observers. This procedure leads to the selection of the inputs–output con-

Table 4. Fault signatures.

	f_{u_1}	f_{u_2}	...	f_{u_r}	f_{y_1}	f_{y_2}	...	f_{y_m}
r_{u_1}	0	1	...	1	0	0	...	0
r_{u_2}	1	0	...	1	0	0	...	0
⋮			⋱				...	⋮
r_{u_r}	1	1	...	0	0	0	...	0
r_{y_1}	0	0	...	0	1	0	...	0
r_{y_2}	0	0	...	0	0	1	...	0
⋮			...				⋱	⋮
r_{y_m}	0	0	...	0	0	0	...	1

figuration for the fault estimator blocks. Then, the design of the neural networks model can be performed. Finally, the threshold test logic of Eq. 6 allows the achievement of the fault diagnosis task.

4. SIMULATION RESULTS

This section summarises the simulations related to the considered benchmark system, for which the proposed solution for the fault diagnosis solutions are implemented. The focus is placed on the single wind turbine benchmark, where the neural network fault estimators are analysed and validated by means of a Monte Carlo analysis. Then, their performances are compared to those of other fault diagnosis methods, commonly adopted in the related literature.

With reference to the wind turbine benchmark model of Section 2, all the simulations are driven by the same wind sequence. It represents a good coverage of typical operating condition, as it ranges from 5 to 20 m/s, with a few spikes at 25 m/s. The other wind speed components are represented by uniform random variables. The simulations last for 4400 s, during which only one fault may occur. The discrete–time benchmark model runs at a sampling frequency of 100 Hz (*i.e.* $T = 0.01$), so that $N = 440000$ samples per simulation are acquired. With reference to the different scenarios described in Section 2, Table 5 reports the shape and the time of the fault signals affecting the system. They are modelled as input (actuator) or output (sensor) additive fault, as described in (Odgaard and Stoustrup (2015)).

Table 5. Fault characteristics.

Fault nr.	Fault Shape	Time (s)
1	step	2000 – 2100
2	step	2300 – 2400
3	step	2600 – 2700
4	step	1500 – 1600
5	step	1000 – 1100
6	step	2900 – 3000
7	trapezoidal	3500 – 3600
8	step	3800 – 3900
9	step	4100 – 4300

In order to highlight how faults affect the system, the comparison between the faulty and the fault free signal is represented in Fig. 6, regarding the most affected signals of the fault sensitivity test. This sensitivity analysis aimed at estimating the most sensitive measurements with respect to the simulated fault conditions.

In practice, the monitored fault signals have been injected into the benchmark simulators, assuming that only a single fault may occur in the considered plant. Then, the Relative

Mean Square Errors (RMSE) between the fault-free and faulty measured signals are computed, so that, for each fault, the most sensitive signal can be selected. The results of this fault sensitivity analysis are shown in Table 6 for the wind turbine benchmark.

Table 6. The most sensitive measurements with respect to the faults.

Fault	1	2	3	4	5
Measurement	$\beta_{1,m1}$	$\beta_{2,m2}$	$\beta_{3,m1}$	$\omega_{r,m1}$	$\omega_{r,m1}$
RMSE	11,29	0,98	2,48	1,44	1,45
Fault	6	7	8	9	
Measurement	$\beta_{2,m1}$	$\beta_{3,m2}$	$\tau_{g,m}$	$\omega_{g,m1}$	
RMSE	0,80	0,73	0,84	0,77	

As an example, the cases of faults 1, 2, 3, and 8 are considered in Fig. 6.

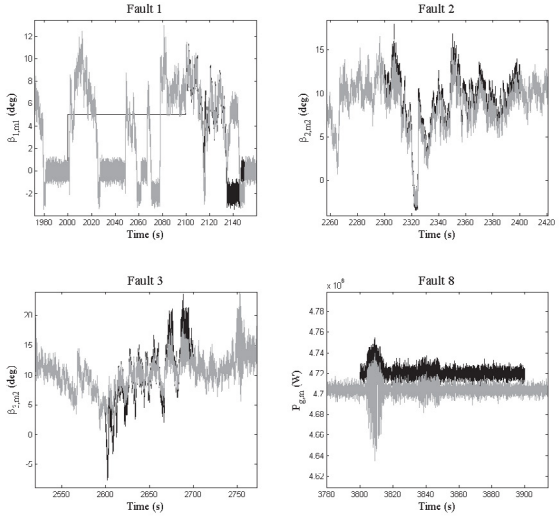

Fig. 6. The faulty signals (black line) compared with the fault-free signals (grey line).

4.1 Fault Diagnosis

Nine open–loop NARX neural network described in Section 3.1 have been designed to estimate the nonlinear behaviour between the acquired measurements and the proposed faults. The selected architecture of the networks involves two layers, namely the hidden layer and the output layer. The number of neurons in the hidden layer has been fixed to $n_h = 16$. Finally, a number of $d_u = d_y = 4$ has been chosen for the input–output delays. The neural networks modelling capabilities have been tested in terms of Root Mean Square Error (RMSE) and the results are reported in Table 7 in fault–free conditions.

The fault detection task is achieved by comparing the residual with a fixed optimised threshold. However, the residuals are filtered by an evaluation function ahead of the threshold comparison. This evaluation function can be either a Mobile Average (MA) or a Mobile Variance (MV), with a properly tuned window size, as reported in Table 8.

Figure 7 shows some meaningful residual signal for actuator faults, together with the relative thresholds, while Fig. 8 shows the residual regarding the sensor faults. Further details on validation and comparative results are described in the following.

Table 7. Neural network performance in terms of RMSE.

Fault Estimator nr.	**RMSE**
1	0.009
2	0.009
3	0.009
4	0.012
5	0.011
6	0.011
7	0.009
8	0.009
9	0.014

Table 8. The residual filter functions for each fault estimator.

Residual nr.	Evaluation function	Window samples
1	MV	20
2	MA	60
3	MV	20
4	MA/MV	45/55
5	MA	50
6	MV	60
7	MV	70
8	MA	50
9	MA	50

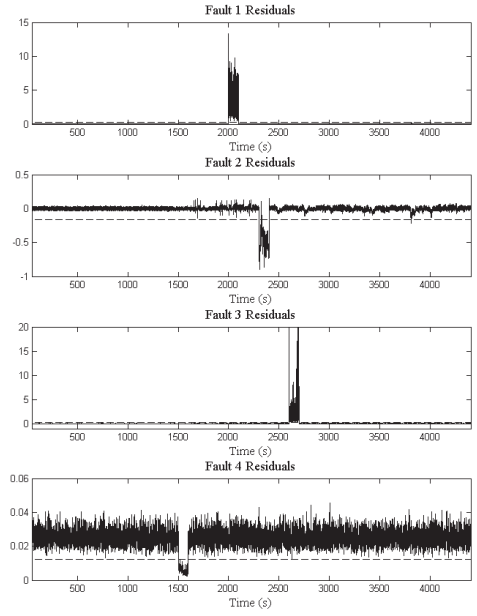

Fig. 7. Faulty residuals (continuous line) and fixed thresholds (dashed line) for the actuator faults 1 - 4.

In particular, Figure 7 shows the residuals generated in faulty conditions by neural network estimators (continuous line) compared the fixed thresholds (dashed line). The considered residuals concern the actuator faults 1 - 4.

On the other hand, Figure 8 shows the residuals generated in faulty conditions by neural network estimators (continuous line) compared the fixed thresholds (dashed line). The considered residuals concerns the sensor fault cases 6 - 8.

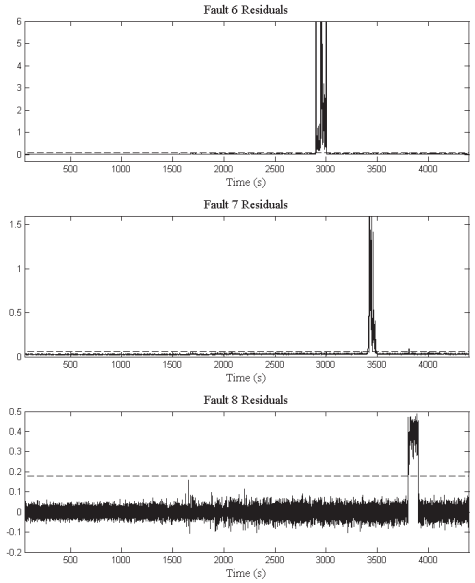

Fig. 8. Faulty residuals (continuous line) and fixed thresholds (dashed line) for the sensor faults 6 - 8.

4.2 Validation and Comparative Analysis

The evaluation of the performances of the considered fault diagnosis strategies is based on the computation of the following indices:

- **False Alarm Rate** (FAR): the ratio between the number of wrongly detected faults and the number of simulated faults;
- **Missed Fault Rate** (MFR): the ratio between the total number of missed faults and the number of simulated faults;
- **True FDI Rate** (TFR): the ratio between the number of correctly detected faults and the number of simulated faults (complementary to MFR);
- **Mean FDI Delay** (MFD): the delay time between the fault occurrence and the fault detection.

A proper Monte Carlo analysis has been performed in order to compute these indices and to test the robustness of the considered FDI scheme. Indeed, the Monte Carlo tool is useful at these stage, as the efficacy of the diagnosis depends on both the model approximation capabilities and the measurements errors. In particular, a set of 1000 Monte Carlo runs has been executed, during which realistic wind turbine uncertainties have been considered by modelling some meaningful variables as Gaussian stochastic processes around the nominal values and with standard deviations corresponding to the realistic minimal and maximal error values of Table 9.

Table 9. Monte–Carlo analysis parameter variations.

Parameter	Nominal Value	Min. Error	Max. Error
ρ	$1.225 Kg/m^3$	$\pm 0.1\%$	$\pm 5\%$
J	$7.794 \times 10^6 Kg/m^3$	$\pm 0.1\%$	$\pm 25\%$
C_p	C_{p0}	$\pm 0.1\%$	$\pm 35\%$

In addition to the proposed neural network fault estimators, the performance indices of other fault diagnosis schemes are analysed.

The first alternative approach considered here uses a Support Vector machine based on a Gaussian Kernel (GKSV) (Odgaard and Stoustrup (2015)). The scheme defines a vector of features for each fault, which contains relevant signals obtained directly from measurements, filtered measurements or their combinations. These vectors are subsequently projected onto the kernel of the Support Vector Machine (SVM), which provides suitable residuals for all of the defined faults. Data with and without faults were used for learning the model for the FDI of the specific faults.

The second scheme consists in an Estimation-Based (EB) solution (Odgaard and Stoustrup (2015)). In particular, a fault detection estimator is designed to detect a fault, and an additional bank of estimators is derived to isolate them. The method was designed on the basis of a system linear model and used fixed thresholds. Each estimator for fault isolation was computed on the basis of the particular fault scenario under consideration.

The third method relies on Up-Down Counters (UDC) (Odgaard and Stoustrup (2015)). These tools, are commonly used in the aerospace framework, and they provide a different approach to the decision logic applied to the FDI residuals. Indeed, the decision to declare the fault occurrence involves discrete-time dynamics and is not simply a function of the current residual value.

The fourth approach Combines Observer and Kalman filter (COK) methods (Odgaard and Stoustrup (2015)). It relies on an observer used as a residual generator for diagnosing the faults of the drivetrain, in which the wind speed is considered a disturbance. This diagnosis observer was designed to decouple the disturbance and simultaneously achieve optimal residual generation in a statistical sense. For the other two subsystems of the wind turbine, a Kalman filter–based approach was applied. The residual evaluation task used a generalised likelihood ratio test, and cumulative variance indices were applied. For fault isolation purpose, a bank of residual generators was exploited. Sensor and system faults were thus isolated via a decision table.

Finally, the fifth method is a General Fault Model (GFM) scheme, which is a method of automatic design (Odgaard and Stoustrup (2015)). The FDI strategy consists of three main steps. In the first step, a large set of potential residual generators was designed. In the second step, the most suitable residual generators to be included in the final FDI system were selected. In the third step, tests for the selected set of residual generators were performed, which were based on comparisons of the estimated probability distributions of the residuals, evaluated with fault-free and faulty data.

The comparative analysis results are reported in Table 10. In particular, different approaches to the fault diagnosis of the wind turbine benchmark model, *i.e.* the neural network estimators, are shown.

Table 10. Comparison of the FDI results.

Fault	Index	GKSV	EB	UDC	COK	GFM	Neural
1	FAR	0.001	0.001	0.001	0.001	0.001	0.001
	MFR	0.002	0.003	0.002	0.003	0.002	0.001
	TFR	0.978	0.977	0.987	0.977	0.982	0.999
	MFD (s)	0.03	0.03	0.04	10.32	0.05	0.01
2	FAR	0.234	0.224	0.123	0.003	0.235	0.228
	MFR	0.343	0.333	0.232	0.029	0.532	0.001
	TFR	0.657	0.667	0.768	0.971	0.468	0.999
	MFD (s)	47.24	44.65	69.03	19.32	13.74	0.08
3	FAR	0.004	0.141	0.123	0.056	0.135	0.001
	MFR	0.006	0.132	0.241	0.128	0.232	0.001
	TFR	0.974	0.868	0.769	0.872	0.768	0.999
	MFD (s)	0.05	0.54	0.05	19.32	0.74	0.01
4	FAR	0.006	0.005	0.123	0.056	0.236	0.001
	MFR	0.005	0.006	0.113	0.128	0.333	0.001
	TFR	0.975	0.994	0.887	0.872	0.667	0.999
	MFD (s)	0.15	0.33	0.04	19.32	17.64	0.69
5	FAR	0.178	0.004	0.234	0.256	0.236	-
	MFR	0.223	0.005	0.254	0.329	0.242	-
	TFR	0.777	0.995	0.746	0.671	0.758	-
	MFD (s)	25.95	0.07	0.04	31.32	9.49	-
6	FAR	0.897	0.173	0.334	0.156	0.096	0.001
	MFR	0.987	0.234	0.257	0.129	0.042	0.001
	TFR	0.013	0.766	0.743	0.871	0.958	0.999
	MFD (s)	95.95	11.37	12.94	34.02	9.49	0.01
7	FAR	0.899	0.044	0.134	0.134	0.123	0.676
	MFR	0.899	0.035	0.121	0.101	0.098	0.001
	TFR	0.101	0.965	0.879	0.899	0.902	0.999
	MFD (s)	99.95	26.17	13.93	35.01	29.79	6.87
8	FAR	0.004	0.045	0.144	0.109	0.099	0.466
	MFR	0.007	0.011	0.101	0.032	0.124	0.001
	TFR	0.993	0.989	0.899	0.968	0.876	0.999
	MFD (s)	0.07	0.08	0.09	0.06	8.94	0.20
9	FAR	-	-	-	-	-	-
	MFR	-	-	-	-	-	-
	TFR	-	-	-	-	-	-
	MFD (s)	-	-	-	-	-	-

The results show the efficacy of the proposed FDI solutions. In more detail, the neural network estimators seem to work better than other approaches, and they have a noteworthy performance level considering the mean delay time, which is significantly lower than 10 s for all the fault cases. Also false alarm and missed fault rate are often lower than those of other approaches, particularly neural networks features an almost null missed fault rate for all the considered faults. However, in the case of the neural networks FDI design, optimisation stages are required, for example for the selection of the optimal thresholds. Furthermore, GKSV involves delays bigger than 25 s, with false alarms and missed fault rate up to 35 %. EB has comparable performance with respect to GKSV in terms of false alarm, true detection and missed fault rate, but with shorter detection delay. UDC often involves high false alarm rates, bigger than 12% for all the detectable faults. COK and GFM have similar performances, with delay time higher than 10 s, false alarm and missed fault bigger than 10 %. Fault 9 concerns the drive–train. This fault is difficult to detect at wind turbine level, therefore it is investigated also in the context of the wind farm benchmark.

5. CONCLUSION

The paper proposed a solution to the problem of earlier fault detection and isolation. The design of the fault indicator involved a data–driven approach, as it represented an effective tool for coping with a poor analytical knowledge of the system dynamics, together with noise and disturbances. In particular, the data–driven proposed solution was based on neural networks used to describe the strongly nonlinear relationships between measurement and faults. The chosen network architecture belongs to the nonlinear autoregressive with exogenous input topology, as it can represent a dynamic evolution of the system along time. The developed fault diagnosis scheme was tested by means of a high–fidelity benchmark model, that simulated the normal and the faulty behaviour of a wind farm. The achieved performances were compared with those of other control strategies, coming from the related literature. Moreover, a Monte Carlo analysis served to analyse the robustness of the proposed solutions against the typical parameter uncertainties and disturbances. Further studies will consider the verification and the validation of the proposed solutions by means of data acquired from real installations, and for fault tolerant control applications.

REFERENCES

Odgaaard, P.F. and Shafiei, S.E. (2015). Evaluation of wind farm controller based fault detection and isolation. In Elsevier (ed.), *Proceedings of the IFAC SAFEPROCESS Symposium 2015*, volume 48, 1084–1089. IFAC, Elsevier, Paris, France. DOI: 10.1016/j.ifacol.2015.09.671.

Odgaard, P.F. and Stoustrup, J. (2012). Results of a Wind Turbine FDI Competition. In C. Verde, C.M. Astorga Zaragoza, and A. Molina (eds.), *Proceedings of the 8th IFAC Symposium on Fault Detection, Supervision and Safety of Technical Processes – SAFEPROCESS 2012*, volume 8, 102–107. National Autonomous University of Mexico, Mexico City, Mexico. DOI: 10.3182/20120829-3-MX-2028.00015.

Odgaard, P.F. and Stoustrup, J. (2015). A benchmark evaluation of fault tolerant wind turbine control concepts. *IEEE Transactions on Control Systems Technology*, 23(3), 1221–1228.

Simani, S. and Castaldi, P. (2017). Robust Control Examples Applied to a Wind Turbine Simulated Model. *Applied Sciences*, 8(29), 1–28. DOI: 10.3390/app8010029. Invited paper for the special issue "Renewable Energy 2018".

Simani, S. and Castaldi, P. (2018). Adaptive Robust Control and its Applications. In A.T. Le (ed.), *Robust Control Applications to a Wind Turbine Simulated System*, chapter 11, 217–233. InTech, Rijeka, Croatia. ISBN: 978-953-51-5729-8. DOI: 10.5772/intechopen.71526.

Simani, S., Castaldi, P., and Farsoni, S. (2017). Data-driven fault diagnosis of a wind farm benchmark model. *Energies*, 10(7), 1–26. Invited paper for the special issue "Wind Turbine 2017". ISSN: 1996–1073. DOI: 10.3390/en10070866.

Simani, S. and Farsoni, S. (2018). *Fault Diagnosis and Sustainable Control of Wind Turbines: Robust data–driven and model–based strategies*. Mechanical Engineering. Butterworth–Heinemann – Elsevier, Oxford (UK), 1st edition. ISBN: 9780128129845.

Simani, S., Farsoni, S., and Castaldi, P. (2018). Data–Driven Techniques for the Fault Diagnosis of a Wind Turbine Benchmark. *International Journal of Applied Mathematics and Computer Science – AMCS*, 28(2), 1–13.