Robust neural network fault estimation approach for nonlinear dynamic systems with applications to wind turbine systems

Reihane Rahimilarki, *Student Member, IEEE,* Zhiwei Gao, *Senior Member, IEEE,* Aihua Zhang, and Richard Binns

In this paper, a robust fault estimation approach is proposed for multi-input and multi-output nonlinear dynamic systems on the basis of back propagation neural networks. The augmented system approach, input-to-state stability theory, linear matrix inequality optimization, and neural network training/learning are integrated so that a robust simultaneous estimate of system states and actuator faults are achieved. The proposed approaches are finally applied to a 4.8 MW wind turbine benchmark system, and the effectiveness is well demonstrated.

Keywords—Fault estimation, robust, linear matrix inequalities, artificial neural networks, wind turbines, input-to-state stability.

I. Introduction

Over the past few years, wind energy has received a significant attention owing to the concerns on global warming, environmental problems, and fossil fuels reduction. During the past decade, numerous investments went to wind energy industries and the wind turbine installed capacity had a constant increase, and the overall capacity of all wind turbines installed worldwide reached 539GW by the end of 2017. However, similar to other industries, faults may occur in wind turbines due to age or unexpected events, which may cause breakdown and relatively high-cost maintenance. In addition, substandard reliability directly decreases the availability of wind power in the grid. Based on the mentioned issues, fault diagnosis plays an important role in increasing the reliability of wind turbines.

In recent years, fruitful results were reported in the field of fault diagnosis, e.g [1]–[3], which can be categorized into three main methods: model-based, signal-based, and knowledge-based approaches following the classical survey literature [4]. In ***model-based*** methods, models of systems to be monitored should be available to the designers [5], [6]. In [7], hidden Markov model based on fuzzy scalar quantisation was proposed to solve the problem of accuracy and sensitivity of fault diagnosis in wind turbines. Performance degradation was addressed in [8] and an approach based on classifier adapting and regression model was proposed to cope with this problem. In ***signal-based*** methods, the input-output model is not necessary to be available. However, the measured signals become essential and the decision on fault diagnosis is made based on these signals and their attributes [9], [10]. In [11], fault diagnosis in wind turbine planetary gearboxes was studied and an approach based on automatic sparse representation was proposed for detecting weak transients. [12] addressed multiscale filtering construction approach, to solve fault diagnosis problem under speed varying and noisy conditions in wind turbine gearboxes.

Alternatively, ***knowledge-based*** methods are particularly suitable for the cases with a large amount of historical data, and the explicit relationships of the system dynamics are challenging to derive. From this aspect, knowledge-based fault diagnosis is called *data-driven* approach [13]. Based on the fact that it is very difficult to model fault dynamics for a system like a wind turbine, it is very challenging to study fault diagnosis in this complex system. Therefore, data-driven methods can be beneficial in industrial area [14]. For instance, [15] proposed a fault diagnosis and isolation approach in order to handle uncertain models and noisy signals, using fuzzy method in wind turbine systems. [16] addressed a data-driven method in order to monitor nonlinear systems using available measurements. Recent results in key-performance-indicator oriented prognosis and diagnosis with a Matlab toolbox Db-kit were reported in [17].

Artificial Neural Networks (ANN) is widely applicable in the area of mapping nonlinear functions and complex systems. It is worth to mention that, multi-layer neural networks as one of the most effective computational intelligence (CI) approaches, has gotten exceptional attention due to its ability as universal approximator [18], in identification and modeling of industrial systems. In [19], power curve modeling was studied in a wind turbine benchmark and an ANN method was studied for parameters estimation. In [20], by having standard deviation, the previous output power, and the wind speed average, an ANN nonlinear model of wind turbine was developed in order to estimate the output power in future. [21] investigated using of the experimental results to train ANN in order to confront the problem of parameters finding of a counter-rotation wind turbine. A data-driven method fault diagnosis was proposed in [22] based on convolutional neural networks to cope with the difficulty of extracting features in new datasets of industrial systems. In [23], an ANN concept

Manuscript received May 2, 2018; revised August 18, 2018, 30 October, 2018, December 12, 2018; accepted December 22, 2018. This work was supported in part by the National Natural Science Foundation of China under Grant 61673074, in part by the Alexander von Humboldt Foundation under Grant GRO/1117303 STP, and in part by the E&E Faculty, University of Northumbria. (Corresponding author: Zhiwei Gao.)

R. Rahimilarki, Z. Gao and R. Binns are with the Faculty of Engineering and Environment, University of Northumbria, Newcastle upon Tyne, NE1 8ST, U.K. (e-mail: reihane.rahimilarki@northumbria.ac.uk; zhiwei.gao@northumbria.ac.uk; richard.binns@northumbria.ac.uk). A. Zhang is with the College of Engineering, Bohai University, Jinzhou 121000, China (e-mail: jsxinxi_zah@163.com).

was studied in order to estimate the imbalance faults in a wind turbine.

Based on the facts that there are always unexpected faults and disturbances in industrial systems, it is very challenging to design an observer to tackle with this problem and guarantee the robustness of the system in conjunction with fully estimating the occurred faults. In [24], a robust neural network was proposed to design an unknown input observer for the application of wind turbine to deal with unmodeled nonlinearities. This approach was considered in discrete-time systems. In [25], a robust H_∞ observer was designed to deal with the environmental disturbances based on ANN.

Augmented system methods achieve many advantages in estimating states and faults simultaneously, among distinct fault estimation approaches, and the pioneering works can be found in [26], [27]. Recently in [28], a discrete-time robust fault tolerant control approach is proposed based on linear matrix inequalities (LMI) techniques and augmented system approach in order to achieve input-to-states stability of the system subjected to input disturbances.

In this paper, the goal is to achieve a fault estimator, which is robust against unknown inputs. As it is obvious, unknown inputs such as modeling defects, perturbations, disturbances and parameters uncertainties can influence the stability of the system. To do so, an augmented robust LMI optimization is proposed based on back propagation neural network (BPNN). The stability of the system is guaranteed via Lyapunov and input-to-state stability criteria.

The rest of the paper is arranged as follows: In Section II, some preliminaries are brought including the basic notations and the introduction on ANN. In section III, the problem is stated and developed in ANN fault estimation based on robust performance index. In Section IV, the model of the wind turbine is investigated and the related simulation results are given.

II. Preliminaries

A. Notations and Definitions

In this paper, the notations are standard. A^{-1} is the inverse of matrix A. A^T is the transpose of matrix A. I and 0 denote an identity matrix and a zero matrix with appropriate dimension, respectively.

The vector $x \in R^n$ can have a norm defined as

$$||x|| = \sqrt{x^T x},$$

while $||d||_{Tf} = (\int_0^{Tf} d^T(\tau)d(\tau)d\tau)^{1/2}$. L_∞ space is space of all L_∞ bounded signals that can be defined as

$$x(t) \in L_\infty \quad \text{if} \quad ess\ sup_t|f(t)| < \infty,$$

σ is a bounded function that is considered as:

$$\sigma(x) = \frac{2}{1+e^{-2x}} - 1, \tag{1}$$

Derivative of sigmoid function is given as follows:

$$\frac{d\sigma(x)}{dx} = \frac{4e^{-2x}}{(1+e^{-2x})^2} = 1 - \sigma^2(x). \tag{2}$$

The function $f(x)$ is *Lipschitz*, if for all x and y, there is a constant C, which is independent of x and y, such that

$$|f(x) - f(y)| \leq C|x-y|. \tag{3}$$

This definition leads to the fact that any function with a bounded first derivative is Lipschitz [29].

The root mean square (RMS) value of estimation error (RMSE) is defined as follows:

$$RMSE = \sqrt{\frac{1}{T_f - T_s}\int_{T_f}^{T_s}(x_i - \hat{x}_i)^2}, \tag{4}$$

in which, T_s is the start time, T_f is the final time, x_i is the ith parameter, and $\hat{x}_i$ is the estimation of the ith parameter. In addition, the normalized RMSE (NRMSE) can be intoduced as belows:

$$NRMSE = \frac{RMSE}{max(x_i) - min(x_i)}, \tag{5}$$

in which, $max(x_i)$ is the maximum of x_i and $min(x_i)$ is the minimum of x_i.

B. Back Propagation Neural Networks

System identification in gray-box modeling is very influential in understanding the behavior of the system in tackling of the unpredicted faults. One of the capable tools in modeling and identification of the nonlinear functions is multi-layer perceptron (MLP) neural networks. The schematic of a fully connected MLP is presented in Figure 1. As it can be seen in this figure, a typical MLP contains of an input layer, which can be the states of the system ($z_i : i = 1, ..., n$), a hidden layer and an output layer ($t_i : i = 1, ..., p$), that designed to be an approximation of the system output. The $\hat{V}$ and $\hat{W}$ are weight matrices for the hidden and output layers. This network is called fully connected, since each neuron in hidden and output layers is connected to every neuron in the previous layer.

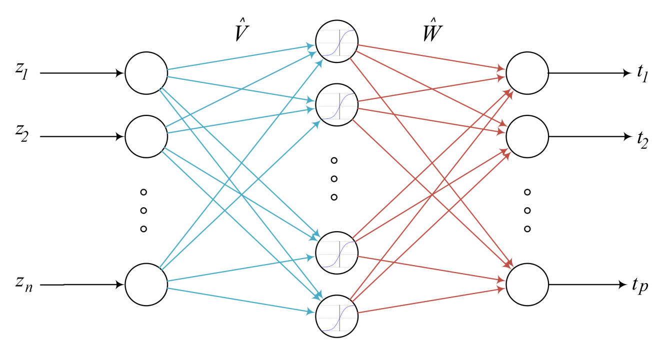

Fig. 1. A Three-layer Fully Connected Neural Network.

One of the most widespread approaches to update the weights in an MLP is *Back Propagation Neural Networks* (BPNN) algorithm. This method is composed of three main steps: initializing, feed-forward, and backward. The last two steps recurred in each iteration until the error is less than a predefined value.

In *initializing*, all the wights are initialized by unsupervised approach, e.g. random quantities. In *feed-forward*, input vector

$Z = [z_1, z_2, ..., z_n]^T$ goes through the network as feed-forward and with the previous weight matrices, the output vector $T = [t_1, t_2, ..., t_m]^T$ is achieved as in (6):

$$T = \hat{W}(\sigma(\hat{V}Z)), \tag{6}$$

In the last part of feed-forward step, the output of the network is compared to the output of the system and the error vector is obtained.

In *backward* step, the error vector is applied to train the weights $\hat{W}$ and $\hat{V}$ using an updating rule. The main task of each BPNN designing is to derive an equation to update the training weights in each iteration.

III. Neuro-Robust Fault Estimation

Consider a nonlinear multi-input an multi-output (MIMO) system, as follows:

$$\begin{aligned} \dot{x}(t) &= f(x(t), u(t)) + B_f f_a(t) + B_d d(t), \\ y(t) &= Cx(t), \end{aligned} \tag{7}$$

in which the $x \in R^n$ is the state vector of the system, $u \in R^m$ is the input signal, $y \in R^p$ is the output and $f(x(t), u(t)) \in R^n$ is an unknown nonlinear function. $f_a(t) \in R^m$ is the occurred actuator fault and B_f is the related fault matrix. $d(t)$ is continuously differentiable and bounded disturbance and B_d is the distribution matrix. In multi-layer ANN mapping, all states of the model should be available. Nevertheless, in some problems, like wind turbines, not all states are measurable. State estimation in a system plays an important role in detecting and diagnosing the faults and monitoring the process more vividly. Based on this fact, it would better to have an augmented system to not only estimate the states but also identify the unanticipated faults at the same time.

In order to obtain this aim, two assumptions are considered: first, the nonlinear model is observable. Second, the states are bounded in L_∞ [30].

Now, by adding and subtracting the term Ax, system (7) becomes

$$\begin{aligned} \dot{x}(t) &= Ax(t) + g(x(t), u(t)) + B_f f_a(t) + B_d d(t), \\ y(t) &= Cx(t), \end{aligned} \tag{8}$$

where $g(x(t), u(t)) = f(x(t), u(t)) - Ax(t)$, A is an arbitrary Hurwitz matrix, which has been chosen in the way that the pair of (C, A) is observable. The main reason to decouple the system into linear and nonlinear blocks is to be able to design a robust observer based on the LMI for linear part while the nonlinear block error is augmented into the disturbance vector. This vector plays the role of exogenous input in the process of formulating the LMI. By this approach the nonlinear observability criteria is relaxed into linear observability criteria. More explanation will be brought in Theorem 1 later on.

As it is obvious that the dynamics of the fault is unknown, there are some methods to deal with this problem. For instance, in [31], it is assumed that the second-order derivative of the occurred fault is zero. However, in this paper, this condition will be relaxed by considering the following equation that is correct in all situations.

$$\dot{f}_a = \dot{f}_a - f_a + f_a. \tag{9}$$

It is worth to mention that f_a and $\dot{f}_a$ should be continuously differentiable and bounded. By augmenting (9) and (8), the model of the system can be written as:

$$\begin{aligned} \dot{X}(t) &= \bar{A}X(t) + G + \bar{B}_d\bar{d}, \\ y &= \bar{C}X(t), \end{aligned} \tag{10}$$

in which, $\bar{A} = \begin{bmatrix} A & B_f \\ 0 & -I \end{bmatrix}$, $X(t) = [x(t) \quad f_a]^T$, $G = [g(x(t), u) \quad 0]^T$, $\bar{B}_d = \begin{bmatrix} B_d & 0 & 0 \\ 0 & I & I \end{bmatrix}$, $\bar{d} = [d \quad \dot{f}_a \quad f_a]^T$, and $\bar{C} = [C \quad 0]$. The main goal in this approach is to design a model identification system to minimize the augmented state error vector in (11):

$$\widetilde{X}(t) = X(t) - \hat{X}(t), \tag{11}$$

where $\hat{X}$ is the estimated state vector and $\widetilde{X}$ is the estimation error vector.

[30] proposed a model to design a neural network observer (NNO) by decoupling systems into linear and nonlinear blocks. By using this model and modifying it by adding faults and disturbances, the NNO model can be seen in Figure 2.

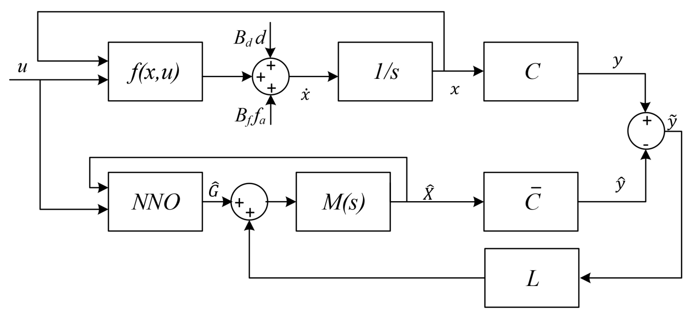

Fig. 2. The Scheme of ANN Based Observer.

In this model, NNO is designed to estimate the nonlinear block of the model, G, and a robust observer (described in the equation (12)) is designed to cope with the disturbance and unexpected fault, which are augmented in $\bar{d}$. Moreover, $M(s) = (sI - \bar{A})^{-1}$ and $\hat{y}$ is the estimation of y. The upper part is the main model with its inputs, while the lower part is the estimation system. The input and the output of the main model are applied to the estimation system. The whole system is not close loop since no signal from the estimation part is entering the main model via feedback.

The observer model of the system (8) can be defined as follows:

$$\begin{aligned} \dot{\hat{X}}(t) &= \bar{A}\hat{X}(t) + \hat{G} + L(y - \bar{C}\hat{X}(t)), \\ \hat{y}(t) &= \bar{C}\hat{X}(t), \end{aligned} \tag{12}$$

where $L \in R^{(n+m)\times p}$ is selected so that the augmented system becomes robust against the disturbance term of $\bar{d}$. Moreover, $\hat{G}$ is the output of the neural networks of NNO and the estimation of G. As it was discussed in (6), $\hat{G}$ can be written as:

$$\hat{G} = \hat{W}\sigma(\hat{V}\hat{\bar{X}}(t)), \tag{13}$$

in which, $\hat{\bar{X}}(t) = [\hat{X}(t) \quad u]^T$ is the input of NNO.

By substituting (13) into (12), the following observer equation can be obtained:

$$\begin{aligned}\dot{\hat{X}}(t) &= (\bar{A}-L\bar{C})\hat{X}(t)+\hat{W}\sigma(\hat{V}\hat{\hat{X}}(t))+L\bar{C}X(t),\\ \hat{y}(t) &= \bar{C}\hat{X}(t),\end{aligned} \tag{14}$$

Procedure 1. For achieving the goal, the following procedure is applied:

i By having the idea of BPNN that was introduced in section II, in feed-forward step after having the output of NNO, a robust observer gain is designed via LMI in order to reduce influence of the unmodeled dynamics and disturbances. In addition, the stability of the system is guaranteed through Lyapunov function (to be addressed in Theorem 1). The output error is calculated at the end of this step.

ii In the next step, the backward step of BPNN is applied and the updating rules for the weights of the NNO are obtained (to be presented in Theorem 2) via the predefined cost function and the output error, which is assessed in the earlier step.

By considering the error function of (11), the error dynamics can be written as follows:

$$\begin{aligned}\dot{\tilde{X}}(t) =&\dot{X}(t)-\dot{\hat{X}}(t)\\ =&\bar{A}X(t)-\bar{A}\hat{X}(t)+G-\hat{W}\sigma(\hat{V}\hat{\hat{X}}(t))\\ &-L(\bar{C}X(t)-\bar{C}\hat{X}(t))+\bar{B}_d\bar{d}.\end{aligned} \tag{15}$$

By substituting $\tilde{G}=G-\hat{W}\sigma(\hat{V}\hat{\hat{X}}(t))$, which is the error of nonlinear function estimation, equation (15) is given as below:

$$\dot{\tilde{X}}(t)=\tilde{A}\tilde{X}(t)+\tilde{G}+\bar{B}_d\bar{d}, \tag{16}$$

in which $\tilde{A}=\bar{A}-L\bar{C}$. (16) can be further simplified as:

$$\dot{\tilde{X}}(t)=\tilde{A}\tilde{X}(t)+\tilde{B}_1\tilde{F}, \tag{17}$$

in which $\tilde{F}=[\tilde{G}\quad \bar{d}]^T$. Now, the system has the state vector of $\tilde{X}(t)$ and the exogenous input of $\tilde{F}$.

Lemma 1 [29]: Consider $f(x,u)$ is continuously differential function and globally Lipschitz in (x,u). If $\dot{x}=f(x,0)$ has a globally exponentially stable equilibrium point at the origin, then the system $\dot{x}=f(x,u)$ is input-to-state stable.

Before presenting the main result of Theorem 1, we firstly give the definitions of the robust performance index and associated Hamiltonian function as follows [32]:

$$J_{11}=||\widetilde{X}(t)||_{Tf}-\gamma^2||\tilde{F}||_{Tf}<0. \tag{18}$$

The associated *Hamiltonian* function is defined as:

$$J_{12}=\int_0^{Tf}\left(\frac{dV(\widetilde{X}(t))}{dt}+\widetilde{X}^T(t)\widetilde{X}(t)-\gamma^2\widetilde{F}^T\widetilde{F}\right)dt \tag{19}$$

Theorem 1: There exists robust observer (14) for the augmented system of (10), so that: (i) the estimation error dynamics in (17) is input-to-state stable; (ii) the estimation error satisfy the robust performance index (18), if there are a positive definite matrix P and a matrix Q so that

$$\begin{bmatrix}P\bar{A}+\bar{A}^TP-Q\bar{C}-\bar{C}^TQ^T+I & P\tilde{B}_1\\ \tilde{B}_1^TP & -\gamma^2 I\end{bmatrix}<0 \tag{20}$$

in which $\tilde{B}_1=[I\quad \bar{B}_d]$. Then, the observer gain is calculated as $L=P^{-1}Q$.

Proof: The proof of this theorem is divided into two parts: (i) input-to-state stability, and (ii) robust performance index.

(i) **Proof of the input-to-state stability**. For any $\tilde{X}_1(t)$, $\tilde{X}_2(t)$, $\tilde{F}_1$, and $\tilde{F}_2$, we can have:

$$\begin{aligned}&|h(\tilde{X}_1(t),\tilde{F}_1)-h(\tilde{X}_2(t),\tilde{F}_2)|\\ &\qquad=|\tilde{A}(\tilde{X}_1-\tilde{X}_2)+\tilde{B}_1(\tilde{F}_1-\tilde{F}_2)|\\ &\qquad\le\alpha|\tilde{X}_1(t)-\tilde{X}_2(t)|+\beta|(\tilde{F}_1-\tilde{F}_2)|,\end{aligned} \tag{21}$$

where $\alpha=||\tilde{A}||$, and $\beta=||\tilde{B}_1||$. As a result, $h(\tilde{X}(t),\tilde{F})$ is globally Lipschitz in $(\tilde{X}(t),\tilde{F})$. It is evident that $h(\tilde{X}(t),\tilde{F})$ is continuously differentiable.

Since the matrix $\tilde{A}$ is Hurwitz, the unforced system $\dot{\tilde{X}}(t)=\tilde{A}\tilde{X}(t)=h(\tilde{X}(t),0)$ is globally exponentially stable at the origin. Therefore, by using Lemma 1, we can conclude that the estimation error dynamics $\dot{\tilde{X}}(t)=h(\tilde{X}(t),\tilde{F})$ is input-to-state stable.

(i) **Proof of robust performance index**. One can take the Lyapunov candidate of $V(\tilde{X})$ as follows:

$$V(\tilde{X}(t))=\tilde{X}(t)^TP\tilde{X}(t), \tag{22}$$

in which P is positive definite symmetric matrix. By having derivative of (22), one can have:

$$\begin{aligned}\dot{V}(\tilde{X}(t)) =&\tilde{X}(t)^TP\dot{\tilde{X}}(t)+\dot{\tilde{X}}(t)^TP\tilde{X}(t)\\ =&\tilde{X}(t)^TP(\tilde{A}\tilde{X}(t)+\tilde{B}_1\tilde{F})\\ &+(\tilde{A}\tilde{X}(t)+\tilde{B}_1\tilde{F})^TP\tilde{X}(t).\end{aligned} \tag{23}$$

Therefore, by substituting (23) into (19), J_{12} can be obtained as:

$$\begin{aligned}J_{12}=\int_0^{Tf}&(\tilde{X}(t)^TP(\tilde{A}\tilde{X}(t)+\tilde{B}_1\tilde{F})\\ &+(\tilde{A}\tilde{X}(t)+\tilde{B}_1\tilde{F})^TP\tilde{X}(t)\\ &+\widetilde{X}^T(t)\widetilde{X}(t)-\gamma^2\widetilde{F}^T\widetilde{F})\,dt.\end{aligned} \tag{24}$$

By extracting the vector block of $Z=[\tilde{X}(t)\quad \tilde{F}]^T$ and using *Schur Complement*, (24) can be rewritten as:

$$J_{12}=\int_0^{Tf}Z^TRZ\ dt, \tag{25}$$

in which,

$$R=\begin{bmatrix}P\tilde{A}+\tilde{A}^TP+I & P\tilde{B}_1\\ \tilde{B}_1^TP & -\gamma^2 I\end{bmatrix} \tag{26}$$

Consequently, for having $J_{12}<0$, R should be negative definite. By substituting $\tilde{A}=\bar{A}-L\bar{C}$, $R<0$ is equivalent to the following LMI:

$$\begin{bmatrix}P\bar{A}+\bar{A}^TP-Q\bar{C}-\bar{C}^TQ^T+I & P\tilde{B}_1\\ \tilde{B}_1^TP & -\gamma^2 I\end{bmatrix}<0$$

where $Q=PL$. As a result, the condition (20) implies $R<0$, then $J_{12}<0$. It is noticed that $V(\tilde{X}(t))\ge 0$, and from (19) and $J_{12}<0$, the robust performance index (18) can thus be obtained. Therefore, the gain matrix of $L=P^{-1}Q$ can be calculated $\therefore$

Now, by calculating the output error vector of $\widetilde{y} = y - \hat{y}$, the feed-forward step of designing is finished. The next step is to design a neural network and propose an updating rule for weight matrices by using the output error $\widetilde{y}$.

As it was discussed in [33], based on the *Universal Approximator* theorem, a multi-layer neural network (MLP) with three layers and updating rule of BPNN has the capability of identifying any nonlinear function. Therefore, as it was brought earlier, for estimating nonlinear function of G, equation (13) with the estimated weight matrices of $\hat{W}$ and $\hat{V}$ is considered. Moreover, the basic updating rules in BPNN are as follows:

$$\begin{aligned} \dot{\hat{W}} &= -\eta_1(\frac{\partial J_2}{\partial \hat{W}}) - \rho_1||\widetilde{y}||\hat{W}, \\ \dot{\hat{V}} &= -\eta_2(\frac{\partial J_2}{\partial \hat{V}}) - \rho_2||\widetilde{y}||\hat{V}, \end{aligned} \tag{27}$$

in which, J_2 is the cost function of the system that should be minimized. For finding an updating rule to minimize the cost function, the following theorem is discussed.

Theorem 2: Given the nonlinear model of (7) and the observer scheme of Figure 2 with observer equation of (14). If the ANN weights are trained as

$$\begin{aligned} \dot{\hat{W}} =& -\eta_1(\widetilde{y}^T C\tilde{A}^{-1})^T(\sigma(\hat{V}\hat{\hat{X}}))^T - \rho_1||\widetilde{y}||\hat{W}, \\ \dot{\hat{V}} =& -\eta_2(\widetilde{y}^T C\tilde{A}^{-1}\hat{W}(I-\Lambda(\hat{V}\hat{\hat{X}})))^T\hat{\hat{X}}^T \\ & - \rho_2||\widetilde{y}||\hat{V}, \end{aligned} \tag{28}$$

where $\Lambda(\hat{V}\hat{\hat{x}}) = diag\left\{\sigma_i^2(\hat{V}_i\hat{\hat{X}})\right\}$, and $i = 1, 2, ..., m$, then, $\widetilde{X}, \widetilde{W}, \widetilde{V}, \widetilde{y} \in L_\infty$ are the estimation error, the weights error, and the output error which are all bounded. η_1 and η_2 are positive learning rate and ρ_1 and ρ_2 are small positive numbers.

Proof: By defining cost function $J_2 = \frac{1}{2}(\widetilde{y}^T\widetilde{y})$ and using the basic updating rule for BPNN that is introduced in (27), it is obvious that the only terms that should be changed into simpler terms are $\frac{\partial J_2}{\partial \hat{W}}$ and $\frac{\partial J_2}{\partial \hat{V}}$. In order to solve this issue, two terms are introduced as below:

$$\begin{aligned} net_{\hat{W}} &= \hat{W}\sigma(\hat{V}\hat{\hat{X}}(t)), \\ net_{\hat{V}} &= \hat{V}\hat{\hat{X}}(t). \end{aligned} \tag{29}$$

Now, $\frac{\partial J_2}{\partial \hat{W}}$ and $\frac{\partial J_2}{\partial \hat{V}}$ can be decoupled into four partial derivatives of:

$$\begin{aligned} \frac{\partial J_2}{\partial \hat{W}} &= \frac{\partial J_2}{\partial \widetilde{y}} \times \frac{\partial \widetilde{y}}{\partial \hat{X}} \times \frac{\partial \hat{X}}{\partial net_{\hat{W}}} \times \frac{\partial net_{\hat{W}}}{\partial \hat{W}} \\ \frac{\partial J_2}{\partial \hat{V}} &= \frac{\partial J_2}{\partial \widetilde{y}} \times \frac{\partial \widetilde{y}}{\partial \hat{X}} \times \frac{\partial \hat{X}}{\partial net_{\hat{V}}} \times \frac{\partial net_{\hat{V}}}{\partial \hat{V}} \end{aligned} \tag{30}$$

By using the cost function equation, one can get:

$$\frac{\partial J_2}{\partial \widetilde{y}} = \widetilde{y}^T. \tag{31}$$

By having $\widetilde{y} = \bar{C}(X(t) - \hat{X}(t))$, it can be obtained that:

$$\frac{\partial \widetilde{y}}{\partial \hat{X}} = -\bar{C}. \tag{32}$$

For the third term of each equation in (30), by considering (14) and (29), following equations can be achieved:

$$\begin{aligned} \frac{\partial \dot{\hat{X}}}{\partial net_{\hat{W}}} &= (\bar{A} - L\bar{C})\frac{\partial \hat{X}}{\partial net_{\hat{W}}} + I, \\ \frac{\partial \dot{\hat{X}}}{\partial net_{\hat{V}}} &= (\bar{A} - L\bar{C})\frac{\partial \hat{X}}{\partial net_{\hat{V}}} + \hat{W}^T\frac{\partial(\sigma(\hat{V}\hat{\hat{X}}(t)))}{\partial net_{\hat{V}}}. \end{aligned} \tag{33}$$

By having partial derivatives of a vector on a vector, and by considering (2), one can have:

$$\begin{aligned} \frac{\partial(\sigma(\hat{V}\hat{\hat{X}}(t)))}{\partial net_{\hat{V}}} &= \frac{\partial(\sigma(\hat{V}\hat{\hat{X}}(t)))}{\partial(\hat{V}\hat{\hat{X}}(t))} \\ &= \begin{bmatrix} 1-\sigma^2(\hat{V}_1\hat{\hat{X}}(t) & & 0 \\ & \ddots & \\ 0 & & 1-\sigma^2(\hat{V}_n\hat{\hat{X}}(t) \end{bmatrix} \end{aligned}$$

in which, $\hat{V}_i \quad i = 1, ..., n$ is the ith row of the weight matrix $\hat{V}$. Therefore, the above equation can be written as belows:

$$\frac{\partial(\sigma(\hat{V}\hat{\hat{X}}(t)))}{\partial net_{\hat{V}}} = 1 - \Lambda(\hat{V}\hat{\hat{X}}(t)), \tag{34}$$

in which, $\Lambda(\hat{V}\hat{\hat{X}}(t)) = diag[\sigma^2(\hat{V}_i\hat{\hat{X}}(t)] \quad i = 1, ..., n$.

In (33), static approximation of the gradient can be assumed due to the fact that the network converges relatively fast [30]. Therefore, $\frac{\partial \dot{\hat{X}}}{\partial net_{\hat{W}}} = 0$ and $\frac{\partial \dot{\hat{X}}}{\partial net_{\hat{V}}} = 0$. Based on this assumption, (33) can be written as:

$$\begin{aligned} 0 &= (\bar{A} - L\bar{C})\frac{\partial \hat{X}}{\partial net_{\hat{W}}} + I, \\ 0 &= (\bar{A} - L\bar{C})\frac{\partial \hat{X}}{\partial net_{\hat{V}}} + \hat{W}^T\frac{\partial(\sigma(\hat{V}\hat{\hat{X}}(t)))}{\partial net_{\hat{V}}}. \end{aligned}$$

Consequently, by using above equation and (34), following equations can be obtained:

$$\begin{aligned} \frac{\partial \hat{X}}{\partial net_{\hat{W}}} &= -(\bar{A} - L\bar{C})^{-1}, \\ \frac{\partial \hat{X}}{\partial net_{\hat{V}}} &= -(\bar{A} - L\bar{C})^{-1}\hat{W}^T(1 - \Lambda(\hat{V}\hat{\hat{X}}(t))), \end{aligned} \tag{35}$$

By considering the definition of (29), the forth term of equations of (30) are achieved:

$$\begin{aligned} \frac{\partial net_{\hat{W}}}{\partial \hat{W}} &= \sigma(\hat{V}\hat{\hat{X}}(t)), \\ \frac{\partial net_{\hat{V}}}{\partial \hat{V}} &= \hat{\hat{X}}(t). \end{aligned} \tag{36}$$

Now, by substituting (31), (32), (35) and (36) in (30), the following equation is obtained:

$$\begin{aligned} \frac{\partial J_2}{\partial \hat{W}} &= (\widetilde{y}^T C\tilde{A}^{-1})^T(\sigma(\hat{V}\hat{\hat{X}}))^T \\ \frac{\partial J_2}{\partial \hat{V}} &= (\widetilde{y}^T C\tilde{A}^{-1}\hat{W}(I - \Lambda(\hat{V}\hat{\hat{X}})))^T\hat{\hat{X}}^T \end{aligned} \tag{37}$$

By replacing (37) into the updating rule of (27), the equations of (28) can be obtained and based on the BPNN approach and universal approximator theorem, the neural network whose weight matrices are updated based on (28) is stable ∴.

Procedure 2. For designing the robust neural network fault estimator, the following procedure is noted:

i Select the matrix A so that pair of (C, A) is observable.
ii Construct the augmented system in the form of (17).
iii Solve the LMI (20) to achieve the matrices P and Q in order to have $L = P^{-1}Q$.
iv Consider a three-layer back propagation neural network with initial random weights.
v Update the weight matrices W and V using (28).
vi Obtain the augmented state of $\hat{X}$ and compare it to the real value of X.

The flowchart of the algorithm is depicted in Figure 3. In this flowchart, T_s is the initial time of simulation and T_{Final} is the end of it.

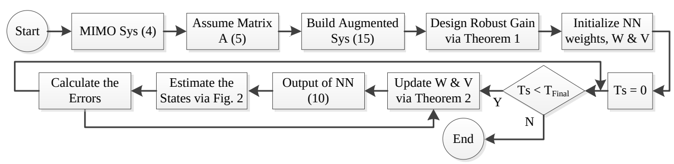

Fig. 3. Flowchart of the Combined Algorithm.

IV. Fault Estimation for 4.8 MW Wind Turbine Benchmark

In this section, the proposed robust fault estimation based on ANN in section III is simulated for wind turbine model. First, the nonlinear model of the system is investigated and then the method is applied to the benchmark.

A. Model Dynamics

For the detailed nonlinear model of the system and internal relations, one can see [34]. In addition, the state-space benchmark can be modeled as follows [31]:

$$\begin{aligned} \dot{x} &= A(x)x + Bu, \\ y &= Cx, \end{aligned} \tag{38}$$

where $x = [\omega_r \quad \omega_g \quad \theta_\delta \quad \dot{\beta} \quad \beta \quad \tau_g]^T$ is the state vector and $u = [\tau_{g,r} \quad \beta_r]^T$ is the control input from the internal controller. The related matrices of the system are given in (39) and (40):

$$A = \begin{bmatrix} A_{11} & A_{12} & A_{13} & 0 & 0 & 0 \\ A_{21} & A_{22} & A_{23} & 0 & 0 & -\dfrac{1}{J_g} \\ 1 & -\dfrac{1}{N_g} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -2\zeta\omega n & -\omega_n^2 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -\alpha_{gc} \end{bmatrix}, \tag{39}$$

$$B = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & \alpha_{gc} \\ 0 & 0 & 0 & \omega_n & 0 & 0 \end{bmatrix}^T, \quad C = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}, \tag{40}$$

where $A11 = -\dfrac{B_{dt} + B_r}{J_r} + \dfrac{1}{2J_r\lambda^2}\rho\pi R^5 C_q(\lambda, \beta)\omega_r$, $A_{12} = \dfrac{B_{dt}}{N_g J_r}$, $A_{13} = -\dfrac{K_{dt}}{J_r}$, $A_{21} = \dfrac{n_{dt} B_{dt}}{N_g J_g}$, $A_{22} = \dfrac{-\dfrac{\eta_{dt} B_{dt}}{N_g^2} - B_g}{J_g}$, and $A_{23} = \dfrac{\eta_{dt} K_{dt}}{N_g J_g}$. The physical meaning of the parameters can be found in [31]. In addition, the numerical quantity of these parameters can be seen in [35].

B. Validating on Wind Turbine

B1. Luenbeger NN observer for WT

Before validating the proposed algorithm in wind turbine benchmark, we test the system by an approach based on Luenberger observer and ANN without considering fault estimation capability [34]. For having such observer, the model of (41) is considered. Detailed information on the steps of designing neural network Luenberger observer can be found in [34].

$$\begin{aligned} \dot{x}(t) &= Ax + g(x, u), \\ y(t) &= Cx(t), \end{aligned} \tag{41}$$

However, as the scenario in this paper is faulty system, the input actuator faults are considered to be 20% effectiveness loss on $\tau_{g,r}$ occurred in $t = 2500s - 3500s$ and 20% effectiveness loss on β_r occurred in $t = 3000s - 4000s$. No disturbances are considered for this problem. The expectation is that, two faulty states of τ_g and β are estimated accurately. The results of this observer can be seen in Figures 4, and 5 from which we can see the system states cannot be estimated well. As a result, the algorithm in [34] can track the healthy system states rather than faulty system states, while having nothing capability to track the faulty signals.

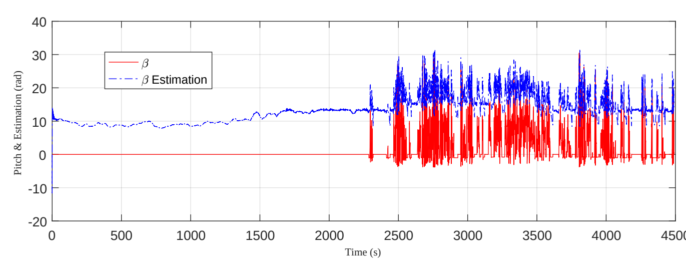

Fig. 4. Pitch Angle Signal and its Estimation Using Luenbeger Observer.

B2. NN fault estimator for WT

First step to validate robust neural network fault estimator is to obtain the robust LMI gain in theorem 1. In order to get this gain, A is assumed as:

$$A = \begin{bmatrix} -20 & 3 & 4 & 2 & 3 & 0 \\ 5 & -30 & 4 & 3 & 6 & 1 \\ 10 & 2 & -20 & 3 & 4 & 5 \\ 3 & 17 & 2 & -21 & 11 & 9 \\ 9 & 12 & 2 & 0 & -25 & 4 \\ 6 & 20 & 8 & 1 & 0 & -35 \end{bmatrix},$$

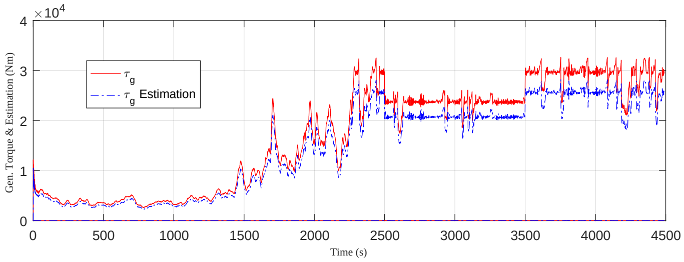

Fig. 5. Generator Torque Signal and its Estimation Using Luenbeger Observer.

The important criteria in choosing A is that it should be Hurwitz and the pair of (C, A) is observable. B_f is chosen equal to B in equation (40) to fully cover the actuators of the system. Therefore, $\bar{A}$ in (10) can be written as:

$$\bar{A} = \begin{bmatrix} -20 & 3 & 4 & 2 & 3 & 0 & 0 & 0 \\ 5 & -30 & 4 & 3 & 6 & 1 & 0 & 0 \\ 10 & 2 & -20 & 3 & 4 & 5 & 0 & 0 \\ 3 & 17 & 2 & -21 & 11 & 9 & 0 & \omega_n \\ 9 & 12 & 2 & 0 & -25 & 4 & 0 & 0 \\ 6 & 20 & 8 & 1 & 0 & -35 & \alpha_{gc} & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix},$$

By considering $B_d = [1 \quad 1 \quad 1 \quad 1 \quad 1 \quad 1]^T$, one can write:

$$\bar{B}_d = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}^T,$$

Now, by considering $\gamma = 0.1$, LMI of (20) can be obtained through LMI solver in MATLAB. Then, L can be achieved via theorem 1 and be shown as:

$$L = 10^3 \times \begin{bmatrix} 51.91 & -0.67 & -0.67 & -0.67 \\ -2.51 & 43.75 & -0.37 & -0.36 \\ 370.24 & 397.04 & 367.59 & 456.04 \\ 137.05 & 163.01 & 118.87 & 133.62 \\ -2.52 & -0.38 & 43.75 & -0.39 \\ -2.54 & -0.40 & -0.41 & 43.70 \\ 304.20 & 318.73 & 318.71 & 573.21 \\ -0.02 & -0.02 & -0.02 & -0.02 \end{bmatrix}$$

As in Luenberger observer problem, the input actuator faults are 20% reduction on $\tau_{g,r}$ occurred in $t = 2500s - 3500s$ and 20% reduction on β_r occurred in $t = 3000s - 4000s$.

For the next step, ANN should be set with updating rule of (28). The ANN training data comes from the benchmark introduced in section IV-Part A. The input of model, which is $u = [\tau_{g,r} \quad \beta_r]^T$ goes directly to ANN model, However, only the error vector of the output of the system is applied to ANN for learning process. By choosing 20 neurons in hidden layers, $\eta_1 = 500$ and $\eta_2 = 500$, the results in Figures 6, 7, 8, and 9 can be compared. In these four diagrams, four measurable states of wind turbine model, e.g. ω_r, ω_g, β and τ_g are depicted (the red solid line) in comparison to the related output of robust neural network state estimator (the blue dash line). As it can be vividly seen, the estimations getting from the robust ANN algorithm can follow the outputs of the main system accurately. The estimation errors converge to zero in all outputs. In addition, in Figure 10 and 11, the two unmeasured states, e.g. θ_δ, and $\dot{\beta}$, are exhibited. The comparison of the red solid line (which is the output of the system) and the blue dash line (which is the estimation) can be vividly explained the effectiveness of the robust neural network algorithm in estimating the unmeasured states.

Now, by considering faults as described earlier on two inputs of the main system, $\tau_{g,r}$, and β_r, the influence of fault on β_r can not be easily seen in the state β (Figure 8). The healthy signal, which is green dash line is not so different with the red solid line, which is faulty signal. However, by comparing the healthy signal and faulty one in Figure 9, the effect of fault on $\tau_{g,r}$ is completely recognizable on the state τ_g. By the way, without considering that it is recognizable in the output or not, the robust neural network algorithm can precisely estimate the occurred faults. The results are also well-illustrated in Figures 12 and 13. In addition, one can see the RMS value of the estimation errors (RMSE) and normalized estimation error (NRMSE) in Table I, which is calculated based on (4) and (5). As one can see, the RMSE and NRMSE for each states and faults are very small, relatively.

TABLE I
RMSE VALUE OF EACH STATES AND FAULTS.

States	**Range**	**RMSE**	**NRMSE (%)**
ω_r	0 : 1.82 rad/s	0.0176	0.98
ω_g	0 : 180 rad/s	1.1782	0.65
β	-3.85 : 30.50 rad	0.1982	0.57
τ_g	0 : 32600 Nm	1.71	0.0052
θ_δ	0 : 0.0017	0.000041	2.14
$\dot{\beta}$	-183.7 : 132.5 rad/s	1.4	0.44
Faults on $\tau_{g,r}$	-6526 : 0 Nm	42.87	0.65
Faults on β_r	-10.60 : 0.2 Nm	0.067	0.62

B3. Some discussions on the proposed algorithm

It is well-illustrated in the literature that neural networks are powerful in estimating complex nonlinear models. However, there are some difficulties related to the simulation. The very challenging point is that due to the big value of the signals, great matrices, and computational cost, the training process is quite time-consuming. The other important issue is the solver steps in Matlab. By increasing the step size of the solver, one can get faster training results. However, it influences directly on the accuracy of the estimation performance. Having a trade-off between these items, an acceptable accuracy with satisfactory speed can be achieved.

V. CONCLUSION

In this paper, a robust fault estimation approach has been proposed based on artificial neural networks. The first dif-

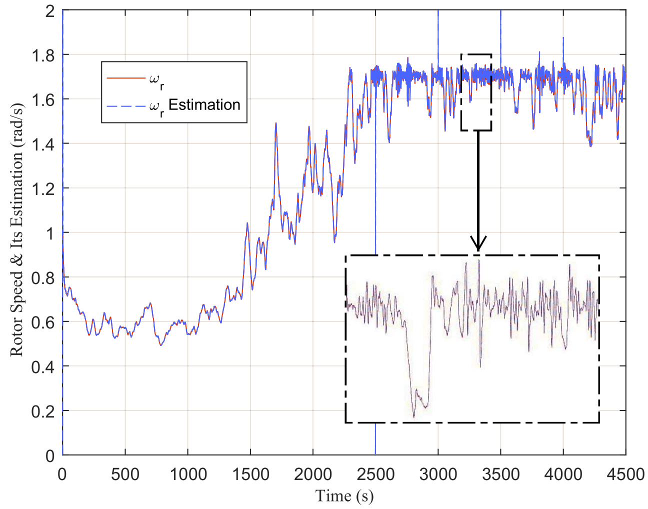

Fig. 6. Rotor Speed Signal and its Estimation.

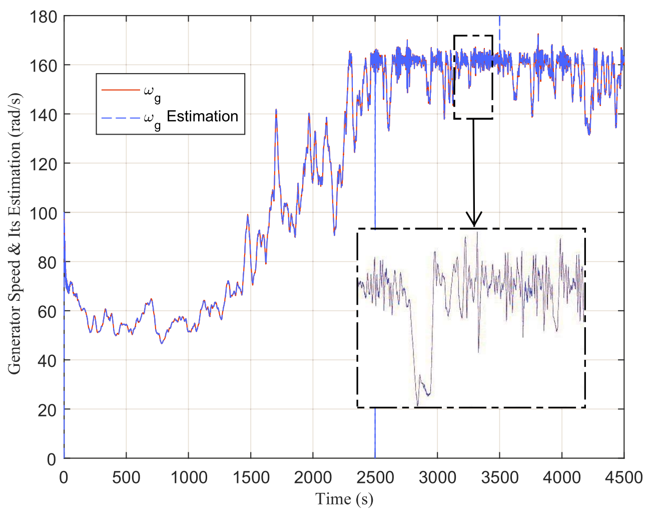

Fig. 7. Generator Speed Signal and its Estimation.

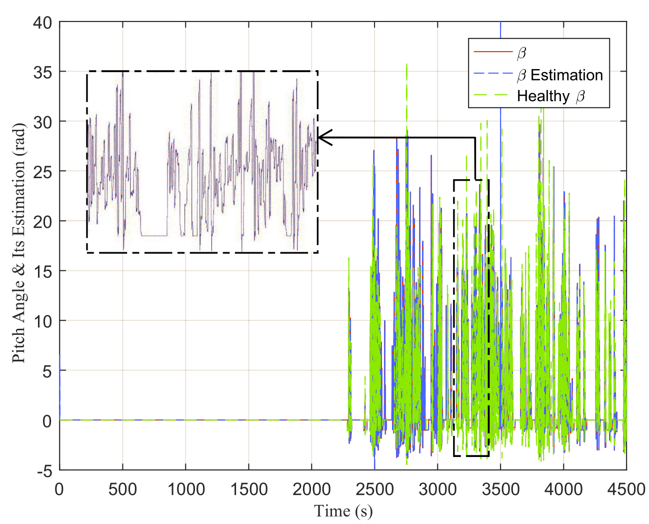

Fig. 8. Pitch Angle Signal and its Estimation.

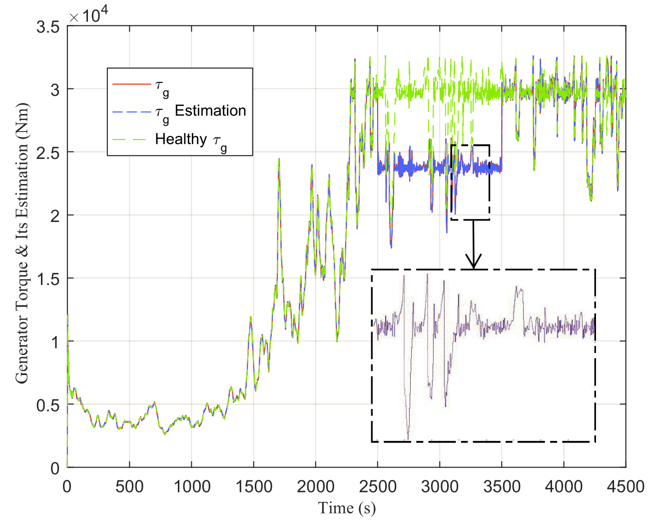

Fig. 9. Generator Torque Signal and its Estimation.

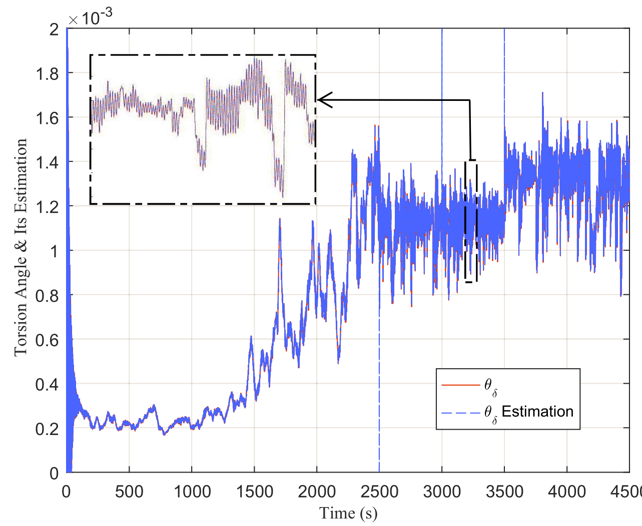

Fig. 10. Torsion Angle and its Estimation.

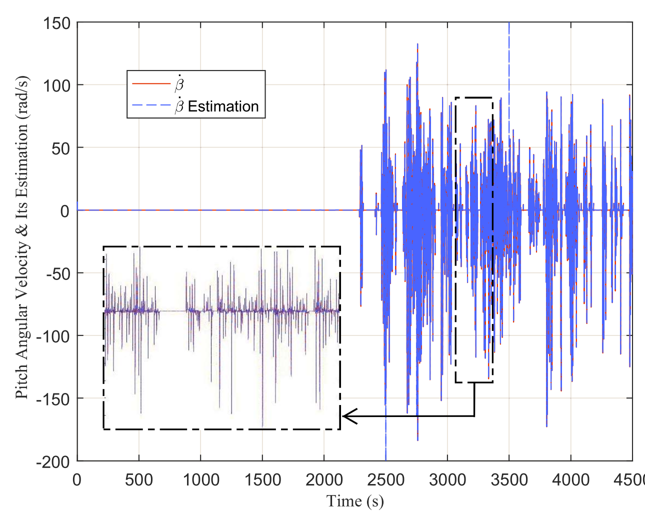

Fig. 11. Pitch Angular Velocity and its Estimation.

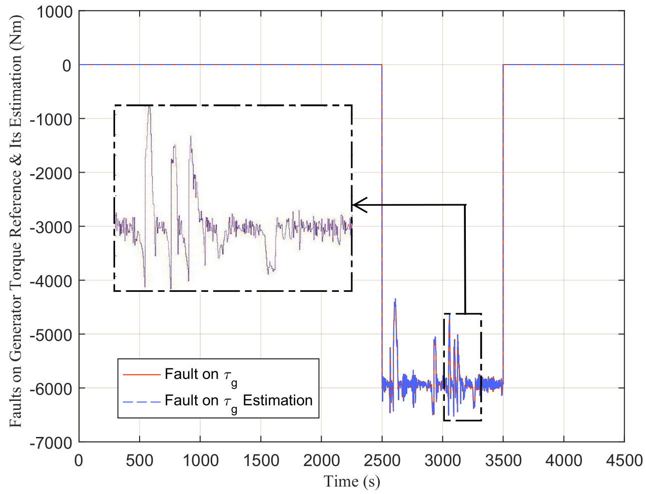

Fig. 12. 20% Faults on Reference of Generator Torque Actuator and its Estimation.

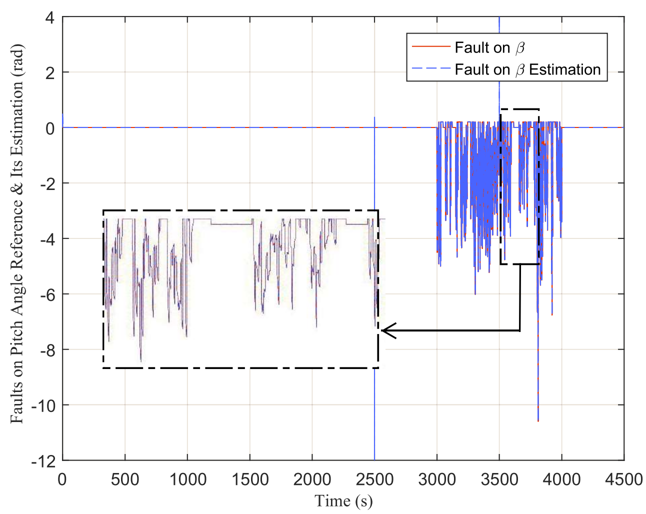

Fig. 13. 20% Faults on Reference of Pitch Angle Actuator and its Estimation.

ficulty to confront in this paper is the unmeasurable states in MIMO systems such as wind turbines. To cope with this problem, the dynamical model is decoupled into linear and nonlinear block. For the nonlinear one, a fully connected ANN is developed to identify the nonlinearities. For relaxing the conditions on fault modeling, a model is proposed and a robust LMI is studied to deal with unmodeled faults and disturbances using input-to-state stability lemma. The approach is validated on a 4.8 MW wind turbine benchmark. A case study is investigated for 20% loss of actuators on each actuator. The results validate the effectiveness of the proposed algorithm. The faults are estimated successfully and the outputs of the observer converge the real output, simultaneously.

In the future, it would be encouraging to develop neural network based fault tolerant control algorithms for nonlinear dynamic systems. Moreover, another interesting research topic is to develop prognostics algorithms for wind turbine systems by using neural network techniques.

REFERENCES

[1] Y. Wang, E. W. Ma, T. W. Chow, and K.-L. Tsui, "A two-step parametric method for failure prediction in hard disk drives," *IEEE Transactions on industrial informatics*, vol. 10, no. 1, pp. 419–430, Feb. 2014.

[2] Y. Jiang and S. Yin, "Recursive total principle component regression based fault detection and its application to vehicular cyber-physical systems," *IEEE Transactions on Industrial Informatics*, vol. 14, no. 4, pp. 1415–1423, Apr. 2018.

[3] M. Volk, S. Junges, and J.-P. Katoen, "Fast dynamic fault tree analysis by model checking techniques," *IEEE Transactions on Industrial Informatics*, vol. 14, no. 1, pp. 370–379, Jan. 2018.

[4] Z. Gao, C. Cecati, and S. X. Ding, "A survey of fault diagnosis and fault-tolerant techniques part i: Fault diagnosis with model-based and signal based approaches," *IEEE Transactions on Industrial Electronics*, vol. 62, no. 6, p. 3757–3767, Jun. 2015.

[5] L. Xu and H. E. Tseng, "Robust model-based fault detection for a roll stability control system," *IEEE Transactions on Control Systems Technology*, vol. 15, no. 3, pp. 519–528, May 2007.

[6] J. Su and W.-H. Chen, "Model-based fault diagnosis system verification using reachability analysis," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, no. 99, pp. 1–10, Jul. 2017.

[7] Y. Zhao, Y. Liu, and R. Wang, "Fuzzy scalar quantisation based on hidden markov model and application in fault diagnosis of wind turbine," *The Journal of Engineering*, vol. 2017, no. 14, pp. 2685–2689, May 2017.

[8] W. Lu, B. Liang, Y. Cheng, D. Meng, J. Yang, and T. Zhang, "Deep model based domain adaptation for fault diagnosis," *IEEE Transactions on Industrial Electronics*, vol. 64, no. 3, pp. 2296–2305, Mar. 2017.

[9] H. Chen and S. Lu, "Fault diagnosis digital method for power transistors in power converters of switched reluctance motors," *IEEE Transactions on Industrial Electronics*, vol. 60, no. 2, pp. 749–763, Feb. 2013.

[10] X. Gong and W. Qiao, "Bearing fault diagnosis for direct-drive wind turbines via current-demodulated signals," *IEEE Transactions on Industrial Electronics*, vol. 60, no. 8, pp. 3419–3428, Jan. 2013.

[11] Y. Qin, J. Zou, and F. Cao, "Adaptively detecting the transient feature of faulty wind turbine planetary gearboxes by the improved kurtosis and iterative thresholding algorithm," *IEEE Access*, vol. 6, pp. 14 602–14 612, Mar. 2018.

[12] J. Wang, F. Cheng, W. Qiao, and L. Qu, "Multiscale filtering reconstruction for wind turbine gearbox fault diagnosis under varying-speed and noisy conditions," *IEEE Transactions on Industrial Electronics*, vol. 65, no. 5, pp. 4268–4278, May 2018.

[13] Z. Gao, C. Cecati, and S. X. Ding, "A survey of fault diagnosis and fault-tolerant techniques part ii: Fault diagnosis with knowledge-based and hybrid/active approaches," *IEEE Transactions on Industrial Electronics*, vol. 62, no. 6, pp. 3768–3774, Jun. 2015.

[14] Y. Jiang, S. Yin, and O. Kaynak, "Data-driven monitoring and safety control of industrial cyber-physical systems: Basics and beyond," *IEEE Access*, vol. 6, pp. 47 374–47 384, Aug. 2018.

[15] S. Simani, S. Farsoni, and P. Castaldi, "Fault diagnosis of a wind turbine benchmark via identified fuzzy models," *IEEE Transactions on Industrial Electronics*, vol. 62, no. 6, pp. 3775–3782, Jun. 2015.

[16] S. Yin, C. Yang, J. Zhang, and Y. Jiang, "A data-driven learning approach for nonlinear process monitoring based on available sensing measurements," *IEEE Transactions on Industrial Electronics*, vol. 64, no. 1, pp. 643–653, Jan. 2017.

[17] Y. Jiang and S. Yin, "Recent advances in key-performance-indicator oriented prognosis and diagnosis with a matlab toolbox: Db-kit," *IEEE Transactions on Industrial Informatics*, Oct. 2018.

[18] K. S. Narendra and K. Parthasarathy, "Identification and control of dynamical systems using neural networks," *IEEE Transactions on neural networks*, vol. 1, no. 1, pp. 4–27, Mar. 1990.

[19] F. Pelletier, C. Masson, and A. Tahan, "Wind turbine power curve modelling using artificial neural network," *Renewable Energy*, vol. 89, pp. 207–214, Apr. 2016.

[20] S. Kelouwani and K. Agbossou, "Nonlinear model identification of wind turbine with a neural network," *IEEE Transaction on Energy Conversion*, vol. 19, no. 3, pp. 607–612, Sep. 2004.

[21] L. Romanski, J. Bieniek, P. Komarnicki, M. Debowski, and J. Detyna, "Estimation of operational parameters of the counter-rotating wind turbine with artificial neural networks," *Archives of Civil and Mechanical Engineering*, vol. 17, no. 4, pp. 1019–1028, Sep. 2017.

[22] L. Wen, X. Li, L. Gao, and Y. Zhang, "A new convolutional neural network-based data-driven fault diagnosis method," *IEEE Transactions on Industrial Electronics*, vol. 65, no. 7, pp. 5990–5998, Jul. 2018.

[23] H. Malik and S. Mishra, "Artificial neural network and empirical mode decomposition based imbalance fault diagnosis of wind turbine using turbsim, fast and simulink," *IET Renewable Power Generation*, vol. 11, no. 6, pp. 889–902, May 2017.

[24] P. Witczak, K. Patan, M. Witczak, V. Puig, and J. Korbicz, "A neural network-based robust unknown input observer design: Application to wind turbine," *IFAC-PapersOnLine*, vol. 48, no. 21, pp. 263–270, Oct. 2015.

[25] B. L. Boada, M. J. L. Boada, L. Vargas-Melendez, and V. Diaz, "A robust observer based on hinf filtering with parameter uncertainties combined with neural networks for estimation of vehicle roll angle," *Mechanical Systems and Signal Processing*, vol. 99, pp. 611–623, Jan. 2018.

[26] Z. Gao and H. Wang, "Descriptor observer approaches for multivariable systems with measurement noises and application in fault detection and diagnosis," *Systems & Control Letters*, vol. 55, no. 4, pp. 304–313, Apr. 2006.

[27] Z. Gao, S. X. Ding, and Y. Ma, "Robust fault estimation approach and its application in vehicle lateral dynamic systems," *Optimal Control Applications and Methods*, vol. 28, no. 3, pp. 143–156, May 2007.

[28] X. Liu, Z. Gao, and A. Zhang, "Robust fault tolerant control for discrete-time dynamic systems with applications to aero engineering systems," *IEEE Access*, vol. 6, pp. 18 832–18 847, Mar. 2018.

[29] H. K. Khalil, *Nonlinear control*. Pearson New York, 2015.

[30] F. Abdollahi, H. A. Talebi, and R. V. Patel, "A stable neural network-based observer with application to flexible-joint manipulators." *IEEE Transactions on Neural Networks*, vol. 17, no. 1, pp. 118–129, Jan. 2006.

[31] X. Liu, Z. Gao, and M. Z. Chen, "Takagi–sugeno fuzzy model based fault estimation and signal compensation with application to wind turbines," *IEEE Transactions on Industrial Electronics*, vol. 64, no. 7, pp. 5678–5689, Mar. 2017.

[32] A. J. van der Schaft, "L2-gain analysis of nonlinear systems and nonlinear state-feedback h-inf control," *IEEE Transactions on Automatic Control*, vol. 37, no. 6, pp. 770–784, Jun. 1992.

[33] G. Cybenko, "Approximation by superpositions of a sigmoidal function," *Mathematics of Control, Signals and Systems*, vol. 2, no. 4, pp. 303–314, Dec. 1989.

[34] R. Rahimilarki and Z. Gao, "Grey-box model identification and fault detection of wind turbines using artificial neural networks," in *IEEE International Conference on Industrial Informatics (INDIN), Porto*, Jul. 2018, pp. 647–652.

[35] P. Odgaard, J. Stoustrup, , and M. Kinnaert, "Fault tolerant control of wind turbines: a benchmark model," *IEEE transaction on control systems technology*, vol. 21, no. 4, pp. 1168–1182, May, 2013.

Reihane Rahimilarki received the B.S. and M.S. degrees from the Department of Electrical Engineering, Amirkabir University of technology, Tehran, Iran, in 2010 and 2013, respectively. She is currently working toward the Ph.D. degree in the Department of Mathematics, Physics and Electrical Engineering, University of Northumbria, Newcastle upon Tyne, U.K.

Her research interests include robust observers, data-driven fault diagnosis and prognosis, artificial neural networks, deep learning and wind turbine systems.

PLACE PHOTO HERE

Zhiwei Gao received the B.Eng. degree in electrical engineering and automation and the M.Eng. and Ph.D. degrees in systems engineering from Tianjin University, Tianjin, China, in 1987, 1993, and 1996, respectively.

He is currently a Reader with the Faculty of Engineering and Environment, University of Northumbria, Newcastle upon Tyne, U.K. His research interests include fault diagnosis, fault- tolerant control, intelligent optimization, power electronics, and wind turbine energy systems.

Dr. Gao is currently an Associate Editor of the IEEE TRANSACTIONS ON AUTOMATIC CONTROL, IEEE TRANSACTIONS ON INDUSTRIAL INFORMATICS, and IEEE TRANSACTIONS ON INDUSTRIAL ELECTRONICS.