2013

Wind Turbine Fault Detection and Isolation Using Support Vector Machine and a Residual-Based Method

Jianwu Zeng
University of Nebraska-Lincoln, jzeng@huskers.unl.edu

Dingguo Lu
University of Nebraska-Lincoln, stan1860@huskers.unl.edu

Yue Zhao
University of Nebraska-Lincoln

Zhe Zhang
University of Nebraska-Lincoln

Wei Qiao
University of Nebraska-Lincoln, wqiao@engr.unl.edu

See next page for additional authors

2013 American Control Conference (ACC)
Washington, DC, USA, June 17-19, 2013
Pages: 3661 - 3666, DOI: 10.1109/ACC.2013.6580398

Wind Turbine Fault Detection and Isolation Using Support Vector Machine and a Residual-Based Method

Jianwu Zeng, *Student Member*, *IEEE*, Dingguo Lu, *Student Member*, *IEEE*, Yue Zhao, *Student Member*, *IEEE*, Zhe Zhang, *Student Member*, *IEEE*, Wei Qiao*, Senior *Member*, *IEEE*, and Xiang Gong, *Student Member*, *IEEE*

***Abstract*—This paper proposes a novel scheme combining support vector machines (SVM) and a residual-based method for wind turbine fault detection and isolation (FDI). SVMs with radius basis function kernels are used for detecting and identifying sensor stuck and offset faults, where binary codes of fault types are used as the outputs of the SVMs to minimize the number of SVMs being used. The same output of a SVM may correspond to different types of faults and the final decision is made by all SVMs instead of one SVM. Moreover, a residual-based fault detection method using a time-variant threshold is developed to identify the abrupt change and scaling faults. Monte Carlo simulations are carried out in MATLAB to test the effectiveness and robustness of the proposed FDI methods using a wind turbine FDI benchmark model. Results show that the proposed methods can always detect the faults successfully within the required time limits.**

I. Introduction

As the number of wind turbines continuously grows, fault detection and isolation (FDI) has become a more and more important and urgent issue in modern wind turbine operation, where control systems play a vital role in satisfying power capture and load alleviation [1]. Sensors and actuators are key components in a wind turbine control system. A faulty sensor or actuator may cause process performance degradation, process shutdown, or even a fatal accident. Early FDI can provide necessary information for the control system and, therefore, helps reduce the cost of wind energy and increase penetration of wind power into electrical grids.

The purpose of fault detection is to generate symptoms, which indicate the difference between nominal and faulty conditions. Fault isolation is then performed to localize the fault and to identify the fault type based on the observed analytical and heuristic symptoms [2].

Existing FDI techniques can be broadly classified into two major categories, including model-based methods and signal processing-based methods [3]. For model-based FDI, the system model could be mathematical- or knowledge-based [4]. Faults are detected and isolated based on the residual generated by state variable or model parameter estimation [5]-[8]. For signal processing-based FDI, mathematical or statistical operations [9]-[11] are performed on the measurements, or artificial intelligence (AI) techniques [12], [13] are applied to the measurements to extract the information about the faults.

Among all the AI techniques, support vector machine (SVM) is a widely used method in binary classification application [14]. Recently, SVM has been utilized to detect and isolate sensor and actuator faults in wind turbine control system [12], [13]. However, in these papers, one SVM was used to detect each type of fault. Therefore, n SVMs were required in order to detect n types of faults. This requires significant computational load and increases the complexity of the problem.

This paper proposes a method of using SVMs for FDI of wind turbines. To reduce the number of SVMs, binary codes of fault types are generated as the output of the SVMs. The final decision is then made by all SVMs instead of one SVM for fault detection. Furthermore, a residual-based fault detection (RFD) method is proposed as well. Compared to other residual-based methods, the threshold in the proposed method is a time-variant variable rather than a constant. The detection law is also more general than those counter-based methods since it can be used for signals in different forms, e.g., different orders of derivatives.

II. Fault Description and Analysis

This paper investigates two categories of faults, i.e., sensor faults and actuator faults, addressed in the second challenge call on wind turbine fault detection and fault tolerant control [15]. Table I shows the 10 fault scenarios in [15].

A. Sensor Faults

Sensor faults (i.e., Faults 1-6) include the faults in sensor measurements that are stuck, scaled from the true values, or offset from the true values. The details are listed as follows:

Fault 1: the blade root bending moment sensor measurement at Blade 2 ($M_{B,2}$) is scaled by a factor of 0.95.

Fault 2: an offset of -0.5 m/s^2 on the tower top accelerometer in both the fore-aft ($\ddot{x}_x$) and side-to-side ($\ddot{x}_y$) directions.

Fault 3: the generator speed sensor (ω_g) is scaled by a factor of 0.95.

Fault 4: Blade 1 has a stuck pitch angle (β_1) sensor, which holds a constant value of one degree.

Fault 5: the generator power sensor (P_g) is scaled with a factor of 1.1.

The authors are with the Department of Electrical Engineering, University of Nebraska-Lincoln, Lincoln, NE 68588-0511 USA (e-mail: jzeng@huskers.unl.edu; wqiao@engr.unl.edu)

Fault 6: a bit error in the low speed shaft encoder (ϕ).

TABLE I: FAULT SCENARIOS.

No.	Time Range (s)	Fault Component	Fault Type
1	[20, 45]	Blade root bending moment sensor ($M_{B,2}$)	Scaling
2	[75, 100]	Tower top accelerometer ($\ddot{x}_t$)	Offset
3	[130, 155]	Generator speed sensor (ω_g)	Scaling
4	[185, 210]	Pitch angle sensor (β_1)	Stuck
5	[240, 265]	Generator power sensor (P_g)	Scaling
6	[295, 320]	Low speed shaft position encoder (ϕ)	Bit error
7	[350, 410]	Pitch actuator (β_1, β_2, β_3)	Slow change in dynamics
8	[440, 465]	Pitch actuator (β_1, β_2, β_3)	Abrupt change in dynamics
9	[495, 520]	Torque transducer	Offset
10	[550, 575]	Yaw drive (Ξ_e)	Stuck

B. Actuator Faults

Faults 7-10 belong to actuator faults, which include the faults of the blade pitch actuators, the generator torque converter, and the yaw drive.

Fault 7: slow change in Blade 2 pitch angle dynamics, which is introduced linearly from 350 s to 370 s, fully active from 370 s to 390 s, and linearly outfaced from 390 s to 410 s.

Fault 8: abrupt change in Blade 3 pitch actuator, which is active from 440 s to 465 s, and linearly introduced and outfaced within one second.

Fault 9: offset of 1000 Nm on the generated generator torque (T_g).

Fault 10: yaw actuator is stuck at zero rad/s.

C. Fault Analysis

Consider a sensor dataset X consisting of m variables. In this paper, $X^{(i)}$ is the i^{th} ($i = 1, \cdots, m$) column of the matrix X and represents a time series of the i^{th} variable. $X^{(i)}(t)$ represents the value of $X^{(i)}$ at time t. The features used for FDI are the q^{th} ($q = 0, \cdots, 3$) order derivatives of the sensor data calculated as follows:

$$X_q^{(i)}(t) = \frac{d^q}{dt^q} X^{(i)}(t) \tag{1}$$

where X_q represents the features used for FDI. Particularly, when $q = 0$, $X_q^{(i)} = X^{(i)}$.

Let $Y \in \{+1, -1\}$ be a matrix with n columns that represents the state of the system. The positive and negative values of Y indicate that the system is in fault or normal state, respectively. $Y^{(j)}$ is the j^{th} ($j = 1, \cdots, n$) column of the matrix Y and $Y^{(j)}(t)$ represents the system state of the j^{th} fault at time t. Then the FDI problem becomes how to construct the system fault matrix $\hat{Y}$ with the use of features X_q.

III. FAULT DETECTION AND ISOLATION

Two techniques are proposed for FDI in this paper, including a SVM method and an RFD method. The SVM is used for detecting the Faults 2, 4, 6 and 10, while the rest faults are detected by the RFD method.

A. Support Vector Machine

SVM has been successfully applied to classification problems, especially in binary classification applications. In this paper, SVM is used for FDI with the use of the measured signals. Since the ranges of different signals are different, they are normalized as follows:

$$\overline{X}_q^{(i)}(t) = \frac{X_q^{(i)}(t) - \mu_{X_q^{(i)}}}{\sigma_{X_q^{(i)}}} \tag{2}$$

where $\mu_{X_q^{(i)}}$ and $\sigma_{X_q^{(i)}}$ are the mean and standard deviation of $X_q^{(i)}$, respectively. Then a classifier is constructed as follows:

$$\hat{Y}^{(j)}(t) = \operatorname{sgn}\left(w^T \cdot \varphi(x_t) + b\right) \tag{3}$$

where $\hat{Y}^{(j)}(t)$ represents the detection result of the j^{th} fault at time t; $sgn(\cdot)$ is the sign function; w and b are the weights and bias of SVM, respectively; $\varphi(\cdot)$ is a nonlinear mapping function; $x_t = [\overline{X}_q^{(i)}(t), \overline{X}_q^{(i)}(t-1), \ldots, \overline{X}_q^{(i)}(t-d+1)]$ is the current and historical values of the time series $\overline{X}_q^{(i)}$, where d represents the dimension of the vector x_t.

The key issue is to find the optimal values of the SVM parameters w and b. This can be done by solving the following constrained optimization problem.

$$\begin{aligned} \min \quad & \frac{1}{2} w^T w + \gamma \sum_{t=1}^{N} \xi^2(t) \\ s.t \quad & \begin{cases} Y^{(j)}(t) \cdot sign\left(w^T \varphi(x_t) + b\right) = 1 - \xi(t) \\ \xi(t) \geq 0, \quad t = 1, 2, \ldots, N \end{cases} \end{aligned} \tag{4}$$

where $Y^{(j)}(t)$ is the observed value of $\hat{Y}^{(j)}(t)$; $\xi(t)$ is a slack variable; γ is a regularization parameter, which balances the fitting in the training stage and generalization in the implementation stage. Equation (4) can be solved by using Lagrange multipliers and the solution is expressed in its dual form. Then (3) can be rewritten as follows:

$$\hat{Y}^{(j)}(t) = \sum_{k=1}^{N} \alpha_i K\left(x_t, x_k\right) + b \tag{5}$$

where α_k ($k = 1, \cdots, N$) is the nonnegative Lagrange multiplier of (3); and $K(x_t, x_k) = \varphi(x_t)\varphi(x_k)$ is a positive-definite radial basis function (RBF) kernel function defined as follows:

$$K(x_t, x_k) = \exp\left(-\frac{\left\|x_t - x_k\right\|^2}{\sigma^2}\right) \tag{6}$$

where $\|\cdot\|$ represents the operation of the Frobenius norm and σ is the width of the RBF kernel.

In order to identify multiple faults, the SVM-based method is usually implemented by combining several SVMs with either a one-versus-all or one-versus-one method [1], [16]. Therefore, n SVMs are required for identification of n types of faults.

In this paper, Fault 10 is detected by using a SVM (SVM_{10}), Faults 2, 4, and 6 are detected using two SVMs (i.e., SVM_{24} and SVM_{26}). The outputs of the two SVMs are binary coded and listed in Table II. As shown in Table II, two SVMs are used to identify four different states of the system.

TABLE II: BINARY CODES OF THE SVMS' OUTPUTS.

SVM_{24}	SVM_{26}	Fault Type
1	1	2
1	-1	4
-1	1	6
-1	-1	No fault

B. Residual-Based Fault Detection

To identify the pitch actuator dynamics faults, i.e., Faults 7 and 8, the value of pitch angle is firstly estimated. Then the value estimated by the following model is compared to the observed value.

$$\hat{\beta}(t) = f(z_t) \tag{7}$$

where $\hat{\beta}(t)$ is the estimated value of pitch angle $\beta(t)$; $z_t = [\beta(t\text{-}1), \beta(t\text{-}2),\ldots, \beta(t\text{-}d), \beta_r(t), \beta_r(t\text{-}1),\ldots, \beta_r(t\text{-}d\text{+}1)]$ is the time series that consists of the historically observed values of β and β_r, where β_r is the reference pitch angle; $f(\cdot)$ is the estimation function, which is realized by a three-layer artificial neural network (ANN). The number of hidden neurons is 5 and the logistic function is chosen as the activation function for the ANN.

The ANN model is trained to estimate the value of the pitch angle as expressed by (7). Then the residual, i.e., error, can be calculated as follows:

$$r_q(t) = \beta(t) - \hat{\beta}(t) \tag{8}$$

which is used as the feature for fault detection. If $|r_q(t)|$ is larger than a threshold value, $\sigma_{T_q}(t)$, then it indicates that a fault occurs. The threshold is calculated as follows.

$$\sigma_{T_q}(t) = k \cdot \sigma_{r_q(t)} = k \cdot \sqrt{\frac{1}{t}\sum_{i=1}^{t}\left(r_q(i) - \mu_{r_q(t)}\right)^2} \tag{9}$$

where $\mu_{r_q(t)}$ and $\sigma_{r_q(t)}$ are the mean value and the standard deviation of $r_q(t)$; $k > 1$ is a parameter to specify the distance between the threshold value and the standard deviation. It should be noted that $r_q(t) = X_q^{(i)}(t)$ if there is no estimated value of $X_q^{(i)}(t)$.

However, for those abrupt change faults or sensor scaling faults, the residual value only exceeds the threshold value at the instant when the fault occurs and disappears. Fig. 1 shows the values of $r_1(t)$ and $r_2(t)$ when the j^{th} fault occurs.

As shown in Fig. 1, the fault occurs and disappears at t_1 and t_2, respectively. The values of $r_1(t)$ and $r_2(t)$ exceed their corresponding threshold values in a short period, e.g., T_s or $2T_s$, where T_s is the sampling period of $r_1(t)$ and $r_2(t)$. Taking $r_1(t)$ and $r_2(t)$ into account, a new fault detection law can be written as follows:

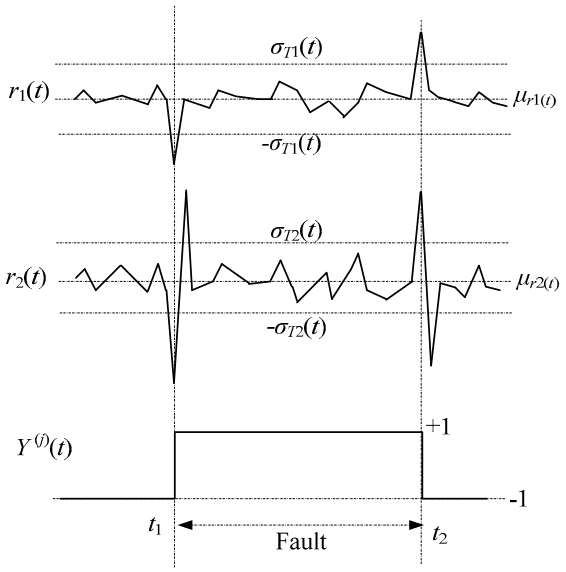


Fig. 1. Two residual signals when the j^{th} fault occurs.

$$Y^{(j)}(t) = Y^{(j)}(t-1) \cdot \mathrm{sgn}\left(r_q(t) - \mu_{r_q(t)} + \sigma_{Tq}(t)\right) \tag{10}$$

where $Y^{(j)}(t\text{-}1)$ is the latest value of the j^{th} fault; $sgn(\cdot)$ is the sign function. Obviously, (10) is suitable for fault detection with both $r_1(t)$ and $r_2(t)$ signals. Therefore, (10) can be used as a general law for fault detection.

Based on the fault analysis and the proposed two methods, the input feature(s) and the corresponding method used for detection of each fault are listed in Table III.

TABLE III: FEATURES AND METHODS FOR FDI.

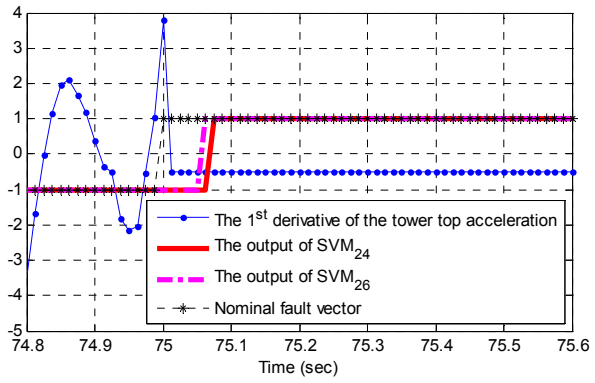
No.	Inputs	Dimension (d)	Method
2, 4	$d(\dot{x}_y)/dt$, $\dot{\beta}_i$	5	SVM_{24}
3	$\ddot{\omega}_g$	1	RFD
5	$\ddot{P}_g$	1	RFD
2, 6	$d(\dot{x}_y)/dt$, $\dot{\phi}$	1	SVM_{26}
7, 8	β_i, $\dot{\beta}_i$, $\ddot{\beta}_i$	1	RFD
	β_{ri} (i = 1,2,3)	1	
9	$\dot{P}_g$, $\ddot{P}_g$	1	RFD
10	Ξ_e	20	SVM_{10}
	$\beta' = \beta_1\cos(\phi) + \beta_2\cos(\phi+2\pi/3) + \beta_3\cos(\phi-2\pi/3)$	20	

IV. SIMULATION RESULTS

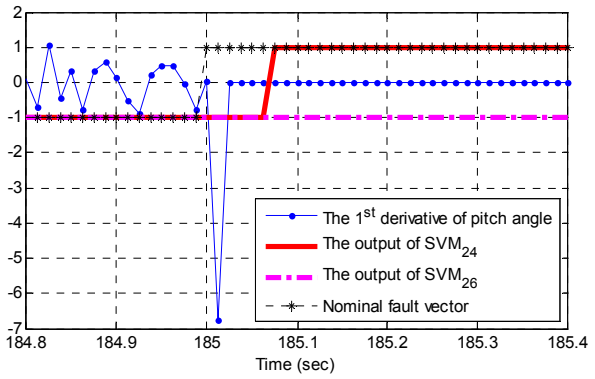
Simulation studies are carried out in MATLAB environment to obtain the FDI results. The simulation step is set as T_s, which equals the sampling time of the sensor measurements.

A. SVM-Based FDI Results

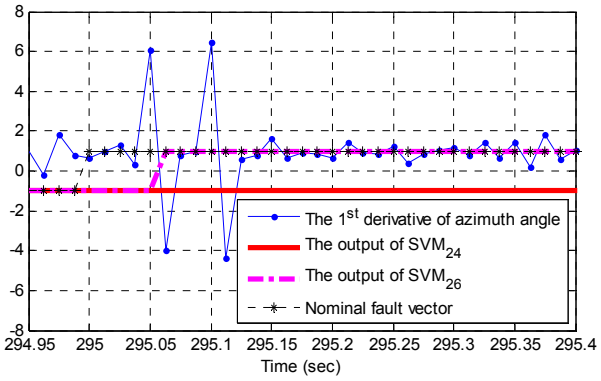
As listed in Table III, Faults 2, 4, 6, and 10 are detected by SVMs. For Faults 2, 4, and 6, the inputs of SVM_{24} and



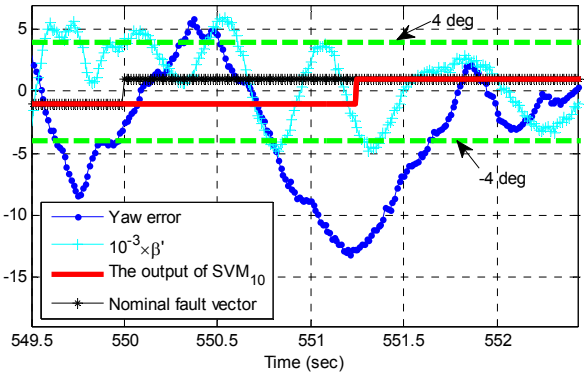
(a)



(b)



(c)



(d)

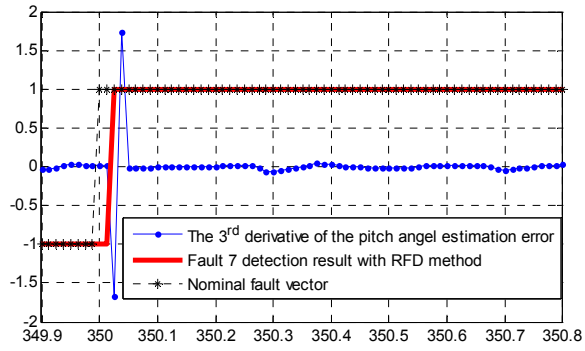
Fig. 2. The SVM-based FDI results: (a) Fault 2; (b) Fault 4; (c) Fault 6; (d) Fault 10.

SVM_{26} are the latest five (d = 5) values of $d(\ddot{x}_y)/dt$ and $\dot{\beta}_1$, and $d(\ddot{x}_y)/dt$ and $\dot{\phi}$, respectively. Fig. 2(a)-(d) shows the detection results when Faults 2, 4, 6, and 10 occur, respectively. The curves marked with dots and plus signs represent sensor signals. The curves marked with stars are the nominal fault vectors [15], which indicate the actual state (i.e., healthy or faulty state) of the system. The solid lines without any marks represent the state of the system identified by the proposed SVM-based fault detection method. As shown in Fig. 2(a)-(c), Faults 2, 4, and 6 occur at the 75th, 185th, and 295th second, respectively. All of them are successfully detected within $6T_s$. It should be noted that the detection time of these faults can be shortened if the value of d is set smaller. However, a smaller d may result in the increase of the possibility of false detection. In this study, d = 5 is the optimal value.

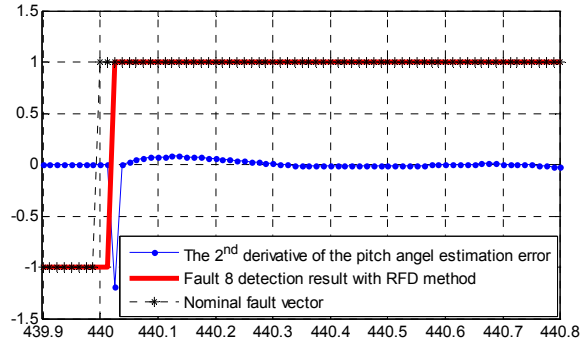
For Fault 10, the inputs of the SVM are the latest 20 samples of Ξ_e and β'. Fig. 2(d) shows the detection result of Fault 10. The yaw drive is stuck at the 550th second. However, since the yaw controller will not be activated to control the yaw drive to correct the yaw error as long as the yaw error is less than 4 degrees, according to wind turbine control system, the yaw stuck will not be detected before the yaw error exceeds 4 degrees. The detection time for Fault 10 is $36T_s$ after the yaw error exceeds the allowed range, which is less than the required detection time limit of $50T_s$.

B. *RFD-Based FDI Results*

Faults 3, 5, 7, 8, and 9 are detected by using the RFD method. For Faults 7 and 8, the ANN-based model is constructed for estimating the value of pitch angle, which is then used for residual generation with (8). The parameter k in (9) is set as 3 and 4 for Faults 7 and 8, respectively. Fig. 3 shows the detect results of Faults 7 and 8. The faults are



(a)

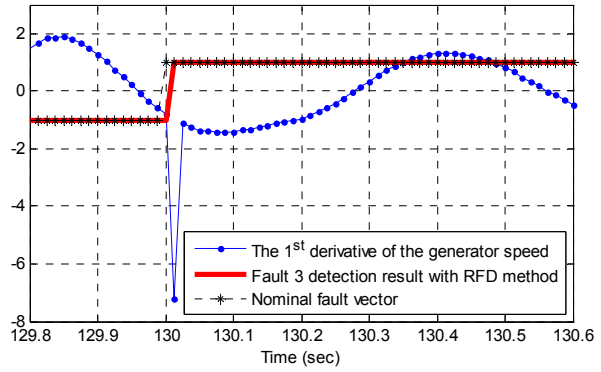


(b)

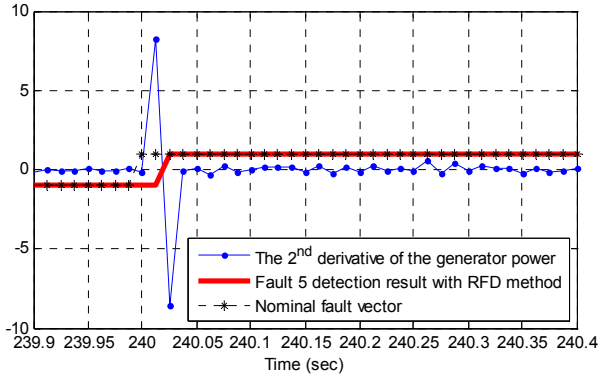
Fig. 3. RFD-based FDI results: (a) Fault 7; (b) Fault 8.

detected within $2T_s$. Such a quick detection is largely attributed to the ANN-based estimation model, which is sensitive to the change of the state value.

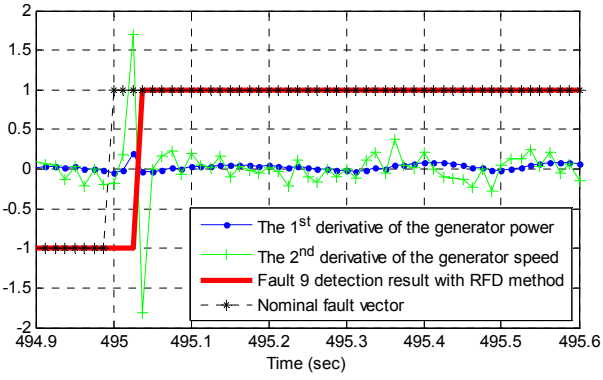
For Faults 3, 5, and 9, residuals are calculated from the original signals because there are no estimated values provided by the reference models. As listed in Table III, the 2nd order derivative of the generator speed ($\ddot{\omega}_g$) and generator power ($\ddot{P}_g$) are used for detecting Faults 3 and 5, respectively. For Fault 9, both the 1st and 2nd order derivatives of generator power are utilized for fault detection. The values of the parameter k are set to be 8, 20, and 5 for Faults 3, 5, and 9, respectively. Fig. 4 shows the detection results for these faults. As shown in Fig. 4 (a)-(b), Faults 3 and 5 are detected within T_s and $2T_s$, respectively. Fault 9 is detected within $3T_s$, as shown in Fig. 4(c).



(a)



(b)



(c)

Fig. 4. RFD-based FDI results: (a) Fault 3; (b) Fault 5; (c) Fault 9.

Table IV summarizes the detection time for each fault. It shows that the detection times of all nine faults are less than the corresponding required time limits.

TABLE IV: DETECTION TIME OF EACH FAULT.

Fault No.	Actual (Required) Detection Time	Fault No.	Actual (Required) Detection Time
2	$6T_s$ ($10T_s$)	7	$2T_s$ ($100T_s$)
3	T_s ($10T_s$)	8	$2T_s$ ($8T_s$)
4	$6T_s$ ($10T_s$)	9	$3T_s$ ($3T_s$)
5	$2T_s$ ($10T_s$)	10	$36T_s$ ($50T_s$)
6	$6T_s$ ($10T_s$)		

C. *Monte Carlo Simulation*

Monte Carlo simulation has been applied to test the robustness of the proposed FDI methods. In the Monte Carlo simulation, the system is simulated 100 times where different measurement noise is generated for different simulation runs. Fig. 5 shows a typical result in a Monte Carlo simulation for each fault. It shows that the fault indices (the solid lines without any marks) obtained by using the proposed methods follow closely their nominal fault vectors (the lines marked with stars). These results indicate that faults 2–10 have always been successfully detected.

V. CONCLUSION

This paper has presented a novel scheme combining SVM and RFD methods for wind turbine FDI. The binary codes of fault types have been used as the outputs of the SVMs. This has reduced the number of SVMs needed for fault detection. Moreover, a RFD method, in which the threshold value is a time-variant value instead of a constant, has been developed to identify the abrupt change and scaling faults. The detection time of most faults is less than that required. Simulation results on the benchmark model provided in [15] have validated the effectiveness of the proposed FDI methods.

REFERENCES

[1] Z. Hameed, Y.S. Hong, Y. Cho, S. Ahn, and C. Song, "Condition monitoring and fault detection of wind turbines and related algorithms: A review," *Renewable and Sustainable Energy Reviews*, vol. 13, pp. 1-39, 2009.

[2] V. Venkatasubramanian, R. Rengaswamy, K. Yin, and S.N. Kavuri, "A review of process fault detection and diagnosis part I: quantitative model-based methods," *Computer & Engineering*, vol. 27, no. 3, pp. 293-311, 2003.

[3] P. Frank, "Analytical and qualitative model-based fault diagnosis - a survey and some new results," *European Journal of Control*, vol. 2, no. 1, pp. 6-28, 1996.

[4] S. Ding, *Model-Based Fault Diagnosis Techniques: Design Schemes, Algorithms, and Tools*, Berlin: Springer, 2008.

[5] P. Odgaard and J. Stoustrup, "Unknown input observer based detection of sensor faults in a wind turbine", in *Proc. IEEE Multiconference on Systems and Control*, 2010.

[6] W. Chen, S. Ding, A. Sari, A. Naik, A.Q. Khan, and S. Yin, "Observer-based FDI schemes for wind turbine benchmark," in *Proc. IFAC World Congress*, Sept. 2011, pp. 7073–7078.

[7] X. Zhang, Q. Zhang, S. Zhao, R. M. Ferrari, M. Polycarpou, and T. Parisini, "Fault detection and isolation of the wind turbine benchmark: An estimation-based approach," in *Proc. IFAC World Congress*, Sept. 2011, pp. 8295–8300.

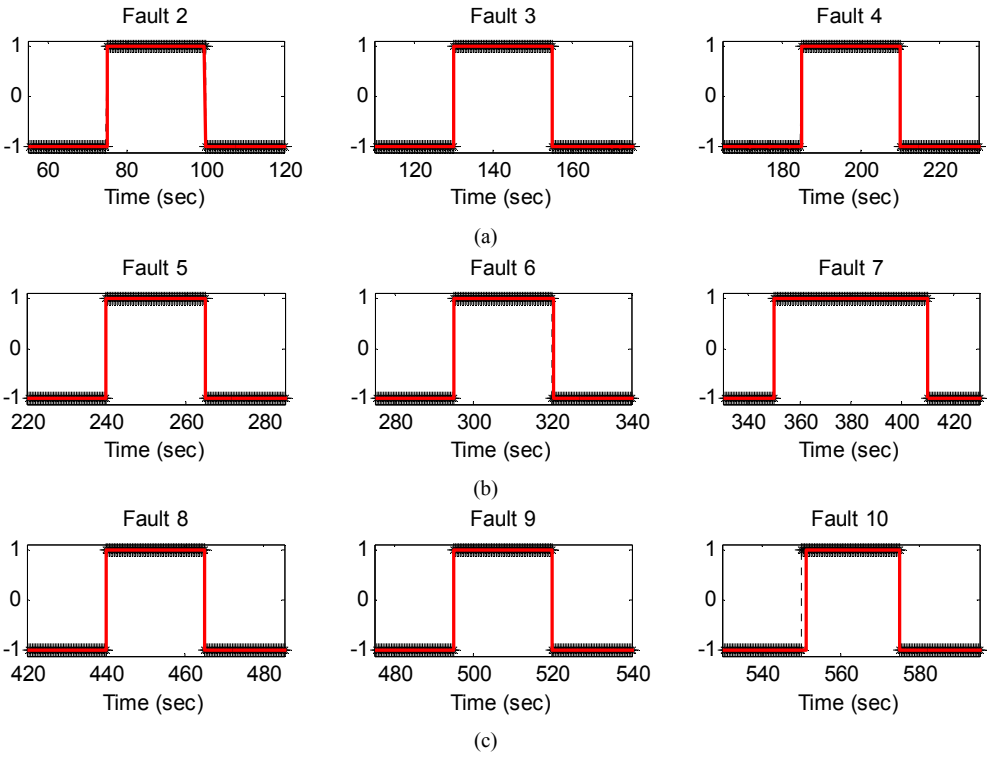


Fig. 5. Typical results in a Monte Carlo simulation (The curves marked with stars are the nominal fault vectors. The solid lines without any marks represent the state of the system identified by the proposed fault detection methods): (a) Faults 2-4; (b) Faults 5-7; (c) Faults 8-10.

[8] A. Ozdemir, P. Seiler, and G. Balas, "Wind turbine fault detection using counter-based residual thresholding," in *Proc. IFAC World Congress*, Sept. 2011, pp. 8295–8300.

[9] J. Dong and M. Verhaegen, "Data driven fault detection and isolation of a wind turbine benchmark," in *Proc. IFAC World Congress*, Sept. 2011, pp. 8295-8300.

[10] S. Simani, P. Castaldi, and M. Bonfe, "Hybrid model-based fault detection of wind turbine sensors," in *Proc. IFAC World Congress*, Sept. 2011, pp. 8295-8300.

[11] S. Simani, P. Castaldi, and A. Tilli, "Data-driven approach for wind turbine actuator and sensor fault detection and isolation," in *Proc. IFAC World Congress*, Sept. 2011, pp. 8295-8300.

[12] N. Laouti, N. Sheibat-Othman, and S. Othman, "Support vector machines for fault detection in wind turbines," in *Proc. IFAC World Congress*, Sept. 2011, pp. 8295-8300.

[13] F. Stoican, C.-F. Raduinea, and S. Olaru, "Adaptation of set theoretic methods to the fault detection of wind turbine benchmark," in *Proc. IFAC World Congress*, Sept. 2011, pp. 8295-8300.

[14] V. Vladimir, S. Golowich, and A. Smola, *Support Vector Method for Function Approximation, Regression Estimation, and Signal Processing*, MIT Press, 1997.

[15] P. Odgaard and K. Johnson, "Wind turbine fault detection and fault tolerant control-a second challenge." to appear in *Proc. 2013 American Control Conference*, Washington, D.C., June 2013.

[16] K. Duan and S. Sathiya Keerthi, "Which is the best multiclass SVM method? An empirical study," in *Proc. 6th International Workshop on Multiple Classifier Systems*, June 2005, pp. 278-285.