Research Article

Early FDI Based on Residuals Design According to the Analysis of Models of Faults: Application to DAMADICS

Yahia Kourd,[1] Dimitri Lefebvre,[2] and Noureddine Guersi[3]

[1] *Department of Control Engineering, University of Mohamed Khider, Biskra 07000, Algeria*

[2] *Electrical Engineering and Automatic Control Research Group (GREAH), University of Le Havre, 25 rue Philippe Lebon, 76058 Le Havre, France*

[3] *Department of Electronics, University of Badji Mokhtar, Annaba 23000, Algeria*

Correspondence should be addressed to Dimitri Lefebvre, dimitri.lefebvre@univ-lehavre.fr

The increased complexity of plants and the development of sophisticated control systems have encouraged the parallel development of efficient rapid fault detection and isolation (FDI) systems. FDI in industrial system has lately become of great significance. This paper proposes a new technique for short time fault detection and diagnosis in nonlinear dynamic systems with multi inputs and multi outputs. The main contribution of this paper is to develop a FDI schema according to reference models of fault-free and faulty behaviors designed with neural networks. Fault detection is obtained according to residuals that result from the comparison of measured signals with the outputs of the fault free reference model. Then, Euclidean distance from the outputs of models of faults to the measurements leads to fault isolation. The advantage of this method is to provide not only early detection but also early diagnosis thanks to the parallel computation of the models of faults and to the proposed decision algorithm. The effectiveness of this approach is illustrated with simulations on DAMADICS benchmark.

1. Introduction

Fault tolerant control and reliability issues for industrial systems and technological processes require the development of advanced fault detection and isolation (FDI) approaches. The main objective of fault detection and isolation is to provide early warnings to operators, such that appropriate actions can be taken to prevent the breakdowns of the system after the occurrence of faults. As a consequence, FDI methods help to avoid system breakdowns and material damages. During the last decades many investigations have been made using analytical approaches for FDI, based on mathematical models. Among the model-based methods, parameter estimation, parity relation, and observers design are the most often applied techniques [1–5]. Unfortunately these approaches require a mathematical description that is often not available in engineering practice. In order to solve this problem, system identification strategies can be applied [6, 7]. One of the most popular nonlinear systems identification approaches is based on the application of artificial neural networks (ANNs) [8–14]. ANNs are computational models with particular properties such as ability to learn, simplicity of implementation, generalization, and good approximation properties.

The aim of fault detection is to deliver alarms when faults occur. The aim of fault diagnosis is to determine the type, magnitude, and location of faults. Detection and diagnosis procedures are based on the observed system and knowledge about the process. So, the inputs of a knowledge-based fault diagnosis system are observed measurements and fault-relevant knowledge about the process [15, 16]. Then, diagnosis of actuators, sensors, and system components can be achieved either via a remote and supervisory diagnosis system or using local intelligence and self-validation methods [17–20]. Such self-validation techniques and condition monitoring are popular for numerous applications in various domains. The usual difficulty with FDI is to provide early diagnosis. Even if detection is generally obtained with short delays algorithms, diagnosis requires more time in order to collect data and process history [21–23]. So, diagnosis needs generally the observation of the fault consequence over a quite long time window and is achieved a posteriori.

This paper focuses on the problem of early fault detection and diagnosis using nonlinear models that are designed with neural networks. Fault detection is based on residuals design and analysis that also estimate the time of fault occurrence. Fault isolation results from the analysis of a bank of additional residuals. These residuals are obtained according to models of faults designed from the faults candidate database. Such models run simultaneously and are updated with the estimated time of occurrence of the faults. The idea to start several models of faults simultaneously once a fault is detected leads to parallel computation that accelerates the diagnosis. Each model behaves according to a single expected fault and is compared using the Euclidean distance with the collected data to provide a rapid diagnosis. The effectiveness of this approach is illustrated according to simulation results obtained with DAMADICS benchmark.

2. FDI Method

The potential of ANNs for FDI problems has been demonstrated in recent years. The neural network approach is applicable to systems for which mathematical models are difficult to obtain. Adaptive ANNs are used to differentiate various faults from the normal conditions, and from one another, according to different fault patterns. Such patterns are extracted from the measured input-output system data, either by offline training or by online learning. In addition ANNs are also helpful to model the nonlinear dynamics according to nonlinear autoregressive structures with exogenous inputs (NARX) [24]. In this section, a FDI method is proposed, that is based on the design of neural models that represent fault-free behaviors as well as faulty behaviors.

2.1. Fault-Free Model Design. In what follows we consider dynamic systems with q inputs $u_i(t)$ and n outputs $y_k(t)$, and we assume that the state variables are not measurable. Such systems often exhibit complex nonlinear dynamics. As a consequence, knowledge-based models are not easy to obtain. Another approach lies in the data-based models. Artificial neural networks are often used for that purpose [25, 26]. The goal is to design models for fault-free and faulty behaviors that will be used for the design of residuals (Figure 1).

In order to get the best ANN architecture, several configurations are tested according to a trial-error processing that uses pruning methods to eliminate useless nodes. The learning of the ANN is obtained according to Levenberg-Marquardt algorithm with early stopping. This algorithm is known for its rapid convergence. During learning stage, the ANN is trained with data collected during the normal functioning (fault-free model). Then the ANN reference models are validated with another set of data. Details about the sizing and training of ANN can be found in [25–27].

2.2. Fault Detection. During monitoring, the direct comparison of the system outputs $y_k(t)$ and fault-free model outputs $y_k'(t)$ leads to n residuals $r_{k0}(t)$:

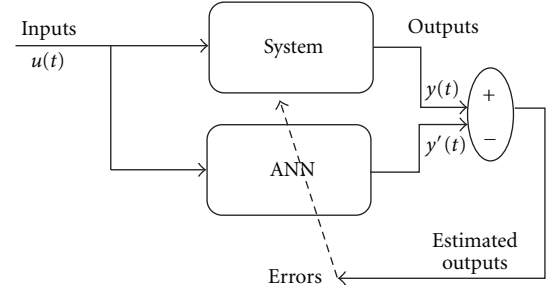$$r_{k0}(t) = y_k(t) - y_k'(t), \quad k = 1, \ldots, n. \tag{1}$$

FIGURE 1: Data-based model design.

The residuals $r_{k0}(t)$ provide a source of information about faults for further processing. Fault detection is based on the evaluation of residuals magnitude. It is assumed that each residual $r_{k0}(t)$, $k = 1, \ldots, n$ should normally be close to zero in the fault-free case, and it should be far from zero in the case of a fault. Thus, faults are detected by setting threshold S_{k0} on the residual signals (Figure 2, a single residual and a single fault are considered for simplicity). The analysis of residuals $r_{k0}(t)$ also provides an estimate τ_k of the time of occurrence t_f used for diagnosis issue. When several residuals are used, the estimate τ of the time of occurrence of faults is given by

$$\tau = \min\{\tau_k, \ k = 1, \ldots, n\}. \tag{2}$$

The faults are detected when the absolute value of one residual $|r_{k0}(t)|$ becomes larger than the threshold S_{k0}:

$$|r_{k0}(t)| \leq S_{k0} : \text{ no fault is detected at time } t,$$
$$|r_{k0}(t)| > S_{k0} : \text{ a fault is detected at time } t. \tag{3}$$

The main difficulty with this evaluation is that the measurement of the system outputs $y_k(t)$ is usually corrupted by disturbances (e. g., measurement noise). In practice, due to the modeling uncertainties and disturbances, it is necessary to assign large thresholds S_{k0} in order to avoid false alarms. Such thresholds usually imply a reduction of the fault detection sensitivity and can lead to nondetections.

2.3. Design of the Models of Faults. When multiple faults are considered, the isolation of the detected faults is no longer trivial, and early diagnosis becomes a difficult task. One can multiply the measurements and use some analysis tools (residuals analysis) in order to isolate the faults. But the number of sensors limits the use of such approach. Another approach is to use a history of collected data to improve the knowledge about the faulty behaviors and then to use this knowledge to design models of faults and additional residuals. We design and use such models to estimate each fault candidate and compare with measurements to provide the most probable fault according to the Euclidean distance between estimated and measured signals [28]. This approach is developed in the following: p faults candidates are considered. For each fault candidate, a model FM(j), $j = 1, \ldots, p$ based on ANNs is designed (Figure 3). These models are trained according to a procedure similar to
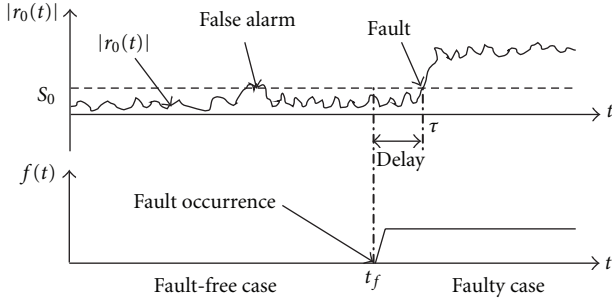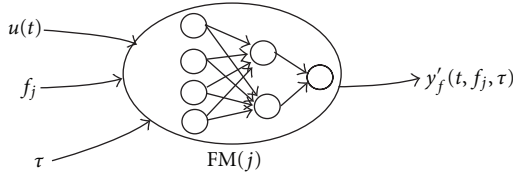
FIGURE 2: Residual-based fault detection.

FIGURE 3: Model of fault f_j.

the one used for fault-free reference models design. Each model of fault is trained with data that result from the observation of the faulty behavior corresponding to a single fault f_j. Such data can be founded in history collected for the considered systems or with specific study where faults are enforced in order to investigate the consequences on the system safety, and availability.

Once the model of fault is trained and validated, it produces estimates $y'_k(t, f_j, \tau)$ of the signals $y_k(t)$, $k = 1, \ldots, n$, under the assumption that the fault f_j occurs at time τ. The comparison of this estimates with the actual measurements $y_k(t)$ is used to isolate the fault.

The inputs of network FM(j) are the input signals $u_i(t)$, the fault candidate f_j, and the estimation τ of the time of occurrence for the fault provided by the detection stage. The outputs of network FM(j) are the estimated faulty outputs $y'_k(t, f_j, \tau)$ $k = 1, \ldots, n$, $j = 1, \ldots, p$ obtained assuming that fault f_j disturbs the system from time τ. The models design and learning are obtained with a method similar to the one used for fault-free model.

2.4. Early Fault Diagnosis. The models of faults run simultaneously once a fault is detected according to the estimate τ of the time of occurrence of the fault. Each model will behave according to a single fault candidate, and the resulting behaviors will be compared with the collected data to provide a rapid diagnosis. In case of numerous fault candidates f_j, $j = 1, \ldots, p$, the outputs $y'_k(t, f_j, \tau)$ of the model FM(j) are compared with the measurements $y_k(t)$ to compute additive residuals. The most probable fault candidate is determined according to the comparison of all residuals $r_{kj}(t, \tau)$, $k = 1, \ldots, n$, $j = 1, \ldots, p$ resulting from the n outputs and p models of faults:

$$r_{kj}(t, \tau) = y_k(t) - y'_{kj}(t, f_j, \tau), \qquad (4)$$

The proposed method uses a time window that can be sized according to the time requirement. Multistep diagnosis with a large window includes a diagnosis delay but will lead to a decision with a high confidence index. On the contrary single step diagnosis leads to immediate diagnosis but with a lower confidence index. To evaluate the probability of each fault candidate, let us define $R_{kj}(t, T, \tau)$ as the cumulative residuals over the sliding time interval $[\max(0, t - T), t]$ of maximal size T:

$$R_{kj}(t, T, \tau) = \sqrt{\int_{\max(0, t-T)}^{t} \left(r_{kj}(u, \tau) \right)^2 \cdot du}. \qquad (5)$$

Then, $D_j(t, T, \tau)$ is the Euclidean norm of the vector $R_j(t, T, \tau) = (R_{kj}(t, T, \tau))$ of dimension n:

$$D_j(t, T, \tau) = \sqrt{\sum_{k=1}^{k=n} \left(R_{kj}(t, T, \tau) \right)^2}. \qquad (6)$$

$D_j(t, T, \tau)$ is used to decide the most probable fault according to single or Multistep (i.e., immediate or delayed) diagnosis.

Multistep diagnosis at time t results from the a posteriori analysis of $D_j(t, t, \tau)$ computed for the time interval $[0, t]$ (i.e., $T = t$). The most probable fault at time t is given by

$$j^*(0, t) = \arg \min_j \left\{ D_j(t, t, \tau), j = 1, \ldots, p \right\}. \qquad (7)$$

A confidence factor CF$_j(t, t, \tau)$ that the current fault is f_j will be given by

$$\mathrm{CF}_j(t, t, \tau) = \frac{\sum_{k=1, k \neq j}^{k=p} (D_k(t, t, \tau))}{\sum_{k=1}^{k=p} (D_k(t, t, \tau))}. \qquad (8)$$

CF$_j(t, t, \tau)$ is near 1 when $D_j(t, t, \tau)$ is near 0 and CF$_j(t, t, \tau)$ is far from 1 when $D_j(t, t, \tau)$ is far from 0.

Immediate diagnosis results from the analysis of $D_j(t, 0, \tau)$ computed at time t according to

$$j^*(t, t) = \arg \min_j \left\{ D_j(t, 0, \tau), j = 1, \ldots, p \right\}. \qquad (9)$$

In order to attenuate the effects of noise and outlaw values, the most probable fault candidate is determined according to the comparison of the cumulative residuals over a sliding time interval $[\max(0, t - T), t]$ of maximal size T. Single-step diagnosis results as a consequence:

$$j^*(t - T, t) = \arg \min_j \left\{ D_j(t, T, \tau), j = 1, \ldots, p \right\} \qquad (10)$$

and the confidence factor CF$_j(t, T, \tau)$ that the current fault is f_j will be given by

$$\mathrm{CF}_j(t, T, \tau) = \frac{\sum_{k=1, k \neq j}^{k=p} (D_k(t, T, \tau))}{\sum_{k=1}^{k=p} (D_k(t, T, \tau))}. \qquad (11)$$

The window width T (i.e., number of steps) is selected in order to satisfy real time requirements for rapid diagnosis.

3. Application to the DAMADICS Benchmark

The proposed method is applied on signals obtained with the DAMADICS simulator. Such system has been used to validate and discuss several FDI systems [29–31]. In [30], binary-valued evaluation of the fault symptoms is explored, and the authors focus on the optimization of the neural network architecture according to Akaike Information Criteria and Final Prediction Error. Both criteria include the learning error and also a term that depends on the complexity (size of the network in number of nodes) and on the dimension of the learning set in order to optimize the ratio complexity/performance. The authors provide interesting performances with small networks for detection but some faults are not isolable. In comparison, our approach requires a larger number of networks, and the networks have more nodes but all faults are detected and isolated. In [31], multiple-valued evaluation of the fault symptoms is introduced to improve the distinguishability of faults. Such a method requires a heuristic knowledge about influence of faults on residuals. In comparison, our approach uses binary-valued evaluation of the residuals but needs analytical models of faults of the actuator including the faults candidates that are not used in [31].

3.1. DAMADICS Description. The DAMADICS benchmark is an engineering research case study that can be used to evaluate detection and isolation methods. The benchmark is an electropneumatic valve actuator in the Lublin sugar factory in Poland [32]. The actuator consists of a control valve, a pneumatic servomotor, and a positioner (Figure 4).

In Figure 4, D/A is the data acquisition unit, PC is the positioner processing unit, E/P is the electropneumatic transducer, V1, V2, V3 are bypass valves, DT stands for displacement, PT for pressure, FT for value flow transducer, and TT for temperature.

In the actuator, faults can appear in control valve, servomotor, electropneumatic transducer, piston rod travel transducer, pressure transmitter, or microprocessor control unit. 19 types of faults are considered ($p = 19$, Table 1). The faults are emulated under carefully monitored conditions, keeping the process operation within acceptable quality limits.

Five available measurements and 1 control value signal have been considered for benchmarking purposes: process control external signal CV, values of liquid pressure on the valve inlet $P1$ and outlet $P2$, liquid flow rate F, liquid temperature $T1$, and stem displacement X (Table 2) [32].

3.2. Residuals Design for Detection. The positioner and control valve are modeled with two multilayer ANNs: net$X$ and net$F$ (Figure 5) that represent the interaction between $q = 4$ inputs and the $n = 2$ outputs in fault-free case according to

$$X' = \text{net}X(CV, P1, P2, T),$$
$$F' = \text{net}F(X, P1, P2, T). \tag{12}$$

To select the structure of the neural networks net$X$ and net$F$, numerous tests have been carried out to obtain the

TABLE 1: Faults to be detected and isolated.

Fault	Description
f1	Valve clogging
f2	Valve or valve seat sedimentation
f3	Valve or valve seat erosion
f4	Increasing of valve or bushing friction
f5	External leakage
f6	Internal leakage (valve tightness)
f7	Medium evaporation or critical flow
f8	Twisted servomotor's piston rod
f9	Servomotor's housing or terminals tightness
f10	Servomotor's diaphragm perforation
f11	Servomotor's spring fault
f12	Electropneumatic transducer fault
f13	Rod displacement sensor fault
f14	Pressure sensor fault
f15	Positioner feedback fault
f16	Positioner supply pressure drop
f17	Unexpected pressure change across the valve
f18	Fully or partly opened bypass valves
f19	Flow rate sensor fault

TABLE 2: Input and output variables.

Input	Range	Unit	Description
CV	[0,1]	—	control signal from external PI controller
P1	[2000, 4e + 6]	Pa	Inlet liquid pressure
P2	[2000, 4e + 6]	Pa	Outlet liquid pressure
T1	[30, 110]	°C	Liquid temperature
Output	Range	Unit	Description
X	[0,1]	—	Position of the rod
F	[0,1]	—	Average flow

best architectures (i.e., number of hidden layers and number of neurons by layer) in order to model the operation of the actuator. The training and test data were generated by simulation using Matlab-Simulink models [27]. The selected structures are ANNs with 6 nodes in layer 1, 3 nodes in layer 2, and a single output neuron. The Table 3 sums some results obtained during the training stage in order to select the best architecture [25, 27]. For this purpose, the mean square error (MSE) is worked out over the set of training data and for a training of 1000 epochs.

When the training is over, net$F$ provides estimates of the outputs, and MSE over the training data is about $3.3 \cdot e - 4$ for net$X$ and about $2 \cdot e - 4$ for net$F$ according to a training of 1000 epochs.

Validation is done with the measured data provided by the Lublin Sugar Factory in 2001 [32]. Validation is illustrated in Figure 6.
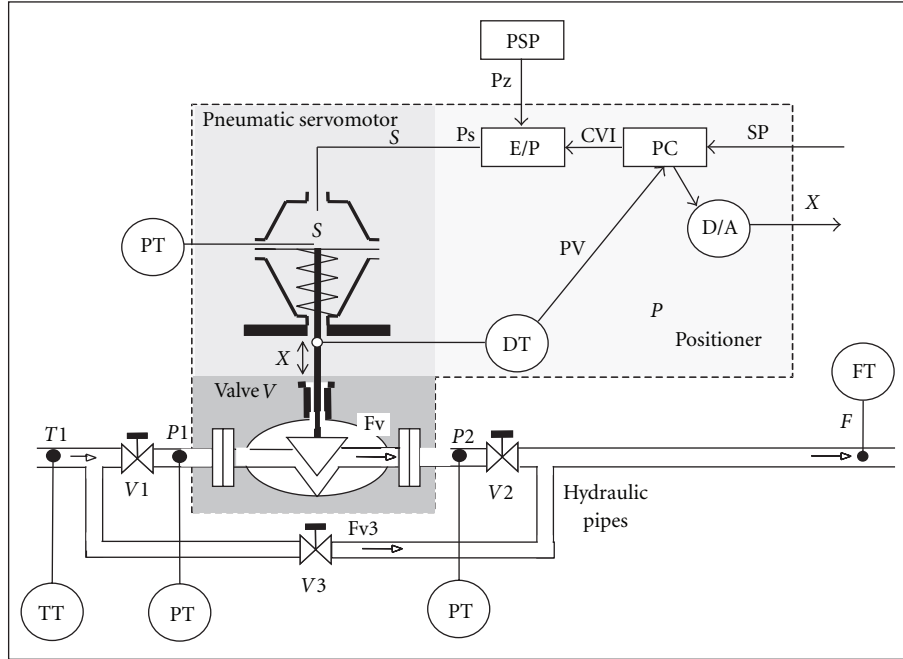
FIGURE 4: DAMADICS actuator.

TABLE 3: Design of the reference models net$X$ and net$F$.

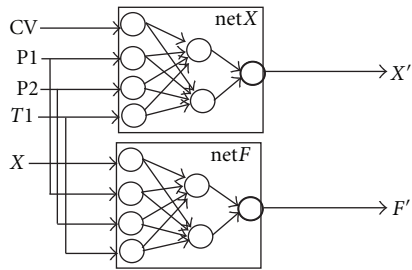| Net$X$	Layer 1	Layer 2	Output layer	MSE
Structure 1	6	3	1	0.00033
Structure 2	10	8	1	0.00149
Structure 3	21	12	1	0.00491
Net$F$	Layer 1	Layer 2	Output layer	MSE
Structure 1	6	3	1	0.000199
Structure 2	4	2	1	0.000849
Structure 3	8	4	1	0.00949

FIGURE 5: ANN fault-free model.

Two residuals are designed according to

$$r_{10}(t) = X(t) - X'(t),$$
$$r_{20}(t) = F(t) - F'(t). \qquad (13)$$

3.3. Fault Detection. Fault detection is obtained according to the comparison of the current residuals value with some thresholds. The thresholds are determined according to the standard deviation of the residual for fault-free case. For output X, $\sigma_1 = 0.0259$ and $S_{10} = 5\sigma_1 = 0.1295$. For output F, $\sigma_2 = 0.0040$ and $S_{20} = 5\sigma_2 = 0.0199$. During normal operation the residuals remain near zero. In Figure 7, the residuals r_{10} and r_{20} are depicted when the fault $f1$ is simulated during interval $[20\,s\ 80\,s]$ time units (time units are in seconds). A fault is detected by r_{10} and by r_{20} at time $t = 22\,s$ when the residuals reach the threshold. From this figure one can also evaluate the delay to detection that is about 2 s for fault f_1. Such information will be used further for diagnosis issue.

The residuals r_{10} and r_{20} are binary valued according to the detection threshold. The Table 4 sums up the residual signatures for the 19 types of faults ($p = 19$). r_{i0^+} means that residual r_{i0} is large positive (according to the previous thresholds) and r_{i0^-} means that residual r_{i0} is large negative. To conclude, all faults are detected according to residual r_{10} and r_{20}.

The residuals analysis is an essential step in FDI systems. The choice of constant or adaptive detection thresholds strongly influences the quality and performance of the FDI. The problem of the threshold selection is closely linked to the behavior of residuals and also to constraints that may be imposed such as security margins tolerance [29]. For this reason, we study the variation of the delay to detection $t_f - \tau$ according to the magnitude of detection thresholds S_{10} and S_{20}. The results are summarized in the Table 5.

According to Table 5 one notices that if the detection thresholds increase, the delay of detection also increases in exponential manner and may lead to nondetections (ND). On the contrary small thresholds may lead to false alarms (FA). So, the thresholds must be thoroughly selected. For the continuation of our work we select the thresholds $S_{10} = 5\sigma_1$ and $S_{20} = 5\sigma_2$.
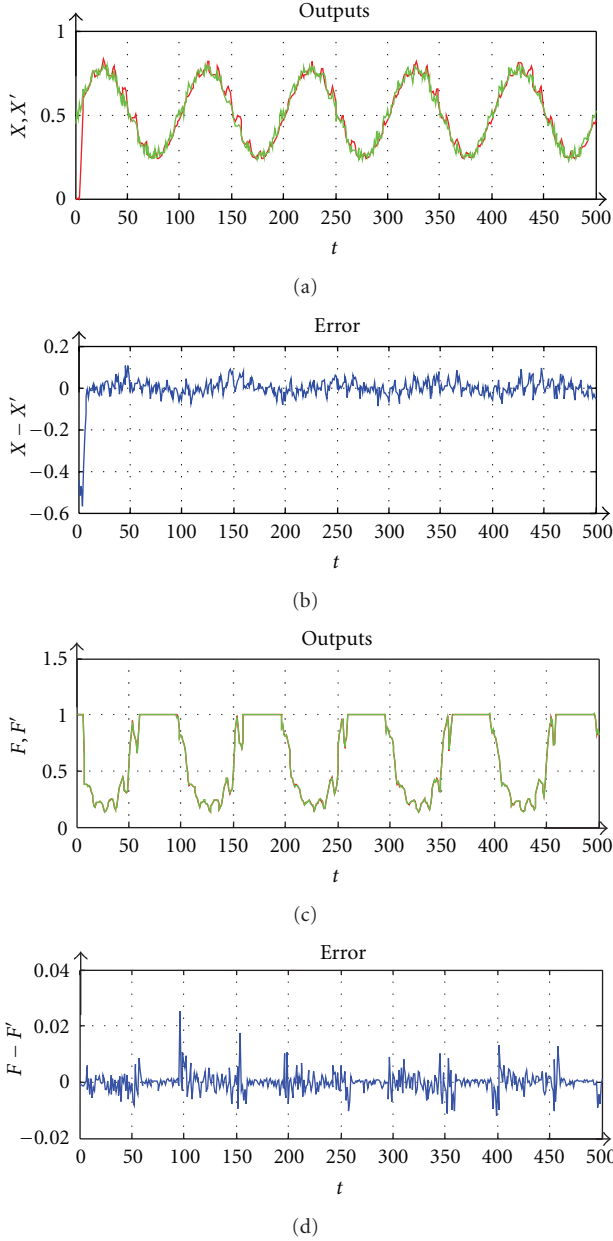
Figure 6: Actual output X and estimated output X' (a); residual r_{10} (b); actual output F, and estimated output F' (c); residual r_{20} (d).

Table 4: Fault detection and isolation with residuals r_{10} and r_{20}.

Faults	residuals			
	r_{10^+}	r_{10^-}	r_{20^+}	r_{20^-}
Fault free	0	0	0	0
$f1$	1	1	1	1
$f2$	0	0	1	0
$f3$	0	0	0	1
$f4$	0	0	1	1
$f5$	0	0	0	0
$f6$	0	0	0	1
$f7$	1	1	1	1
$f8$	0	0	0	0
$f9$	0	0	0	1
$f10$	1	1	1	1
$f11$	1	1	1	0
$f12$	0	0	0	1
$f13$	0	1	0	1
$f14$	0	0	0	0
$f15$	1	1	1	1
$f16$	1	0	0	1
$f17$	1	1	1	1
$f18$	0	0	0	1
$f19$	0	0	0	1

Table 5: Delays to detection.

	Res.	σ	2σ	3σ	4σ	5σ	6σ	7σ
$f12$	r_{10}	2 s	3 s	12 s	39 s	40 s	60 s	ND
$t_f = 500$ s	r_{20}	2 s	2 s	2 s	2 s	2 s	2 s	2 s
$f18$	r_{10}	FA	ND	ND	ND	ND	ND	ND
$t_f = 231$ s	r_{20}	1	2 s	2 s	2 s	2 s	2 s	2 s
$f11$	r_{10}	FA	2 s	2 s	2 s	2 s	3s	ND
$t_f = 485$ s	r_{20}	FA	2 s	2 s	2 s	2 s	2 s	2 s
$f15$	r_{10}	7 s	7 s	8 s	8 s	9 s	17 s	51 s
$t_f = 444$ s	r_{20}	FA	FA	7 s	7 s	7 s	8 s	8 s
$f3$	r_{10}	FA	ND	ND	ND	ND	ND	ND
$t_f = 405$ s	r_{20}	FA	FA	29 s	29 s	35 s	45 s	47 s

3.4. Fault Diagnosis. According to Table 4, three groups of faults with similar symptoms can be separated:

group 1 = {$f3$, $f6$, $f9$, $f12$, $f18$, $f19$},

group 2 = {$f1$, $f7$, $f10$, $f15$, $f17$},

group 3 = {Fault-free, $f5$, $f8$, $f14$}.

Within each group, faults are not isolable. For this reason we propose to use the method described in Sections 2.3 and 2.4 in order to improve the isolability of faults and to perform the complete early diagnosis. For this purpose, 19 models of faults FM(j) $j = 1, \ldots, 19$ are designed according to the history of data available with DAMADICS benchmark. Each

model FM(j) computes two estimated outputs $X'(t, fj, \tau)$, and $F'(t, f_j, \tau)$ and comparisons with measured data lead to the residuals $r_{1j}(t, f_j, \tau)$ and $r_{2j}(t, f_j, \tau)$:

$$r_{1j}\left(t, f_j, \tau\right) = X(t) - X'\left(t, f_j, \tau\right),$$
$$r_{2j}\left(t, f_j, \tau\right) = F(t) - F'\left(t, f_j, \tau\right). \tag{14}$$

The application of diagnosis stage leads to the results in Table 6 and Figures 9 to 11. Let us define the cumulative residuals $R_{1j}(t, T, \tau)$, $R_{2j}(t, T, \tau)$, and the distance $D_j(t, T, \tau)$, according to (5) and (6). Single- and Multistep diagnoses are obtained according to (7) and (10).

For example, $f15$ is simulated within time interval [444 s 1000 s] (Figure 8). The residuals r_{10} and r_{20} are also depicted as the detection thresholds $S_{10} = 5\sigma_1$ and $S_{20} = 5\sigma_2$. According to the detection stage, a fault is detected
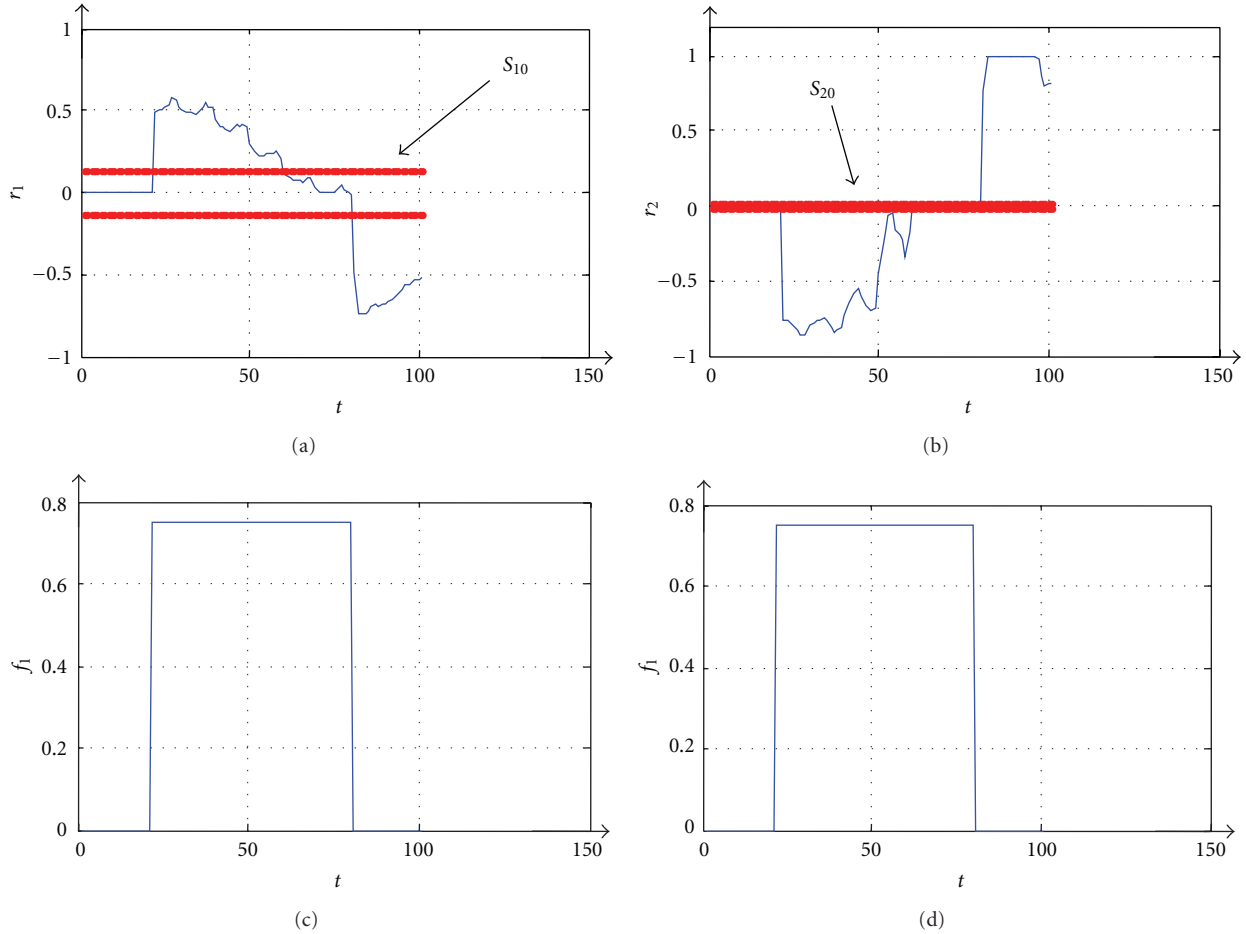
FIGURE 7: Residuals r_{10}, r_{20} when $f1$ is simulated within time interval [20 s 80 s].

at time $\tau = 451$ s with a delay of 7 s, and the group 2 $= \{f1, f7, f10, f15, f17\}$ is isolated (Figure 8). Multistep diagnosis is illustrated with a large time interval $T = t = 1000$ s (Figure 9).

For Multistep diagnosis, Figure 9 and Table 6 report the location of each model FM(j) in plan (R_1, R_2) and also the distance $D_j(t, t, 451)$ at time $t = 1000$ s. The model FM(15) corresponding to the fault candidate f_{15} provides the estimated outputs with the smallest Euclidean distance from the measured outputs. To conclude f_{15} is the most probable fault when residuals are analyzed within time interval [0, 1000 s]. Similar conclusions have been obtained for numerous other simulations.

Another example is provided when fault $f12$ is simulated within time interval [500 s, 1000 s]. According to the detection stage, a fault is detected at time $\tau = 502$ s, and the group $1 = \{f3, f6, f9, f12, f18, f19\}$ is isolated. Early diagnosis is illustrated with a small time interval $T = 50$ s. Figures 10 and 11 plot the location of each model FM(j) in plan (R_1, R_2) for $t \in [0, 700]$. For any $t \in [0, 700]$, the trajectory with minimal distance to the origin (i.e., minimal value of $D_j(t, 50, \tau)$) corresponds to the most probable fault. In Figures 10 and 11, the most probable fault f_{12} is highlighted. Figure 11 plots details about the trajectory for model FM(12).

One can notice that the trajectories start near the origin (i.e., the effects of the expected faults on residuals are weak) and then go far from origin (i.e., the effects of the expected faults increase). The trajectory corresponding to FM(12) (i.e., the expected fault is the actual one) remains near origin in comparison to the other trajectories. One can conclude that the fault candidate $f12$ is the most probable fault because the distance to the origin is the smallest one. One can also notice that cumulative residuals $R_{1j}(t, T, \tau)$ and $R_{2j}(t, T, \tau)$ cover the positive part of plan (R_1, R_2). The repartition of the cumulative residuals in plan (R_1, R_2) confirms the significance of both outputs $X(t)$ and $F(t)$ to design residuals. Thus, Figures 10 and 11 are also useful to check if the considered outputs are helpful for diagnosis issues. Similar conclusions have been obtained for numerous other simulations.

3.5. Discussion. Kościelny et al. [31] have introduced the distinguishability factor Γ that depends on the cardinal of the set of distinguishable faults. As long as all faults are isolable with our approach, we obtain $\Gamma = 1$. In comparison, $\Gamma = 0.54$ has been obtained in [31]. In [30], Patan et al. just consider 3 faults and mention that $f1$ is not isolable with his contribution, so that $\Gamma < 1$ for sure. In addition, the delay to
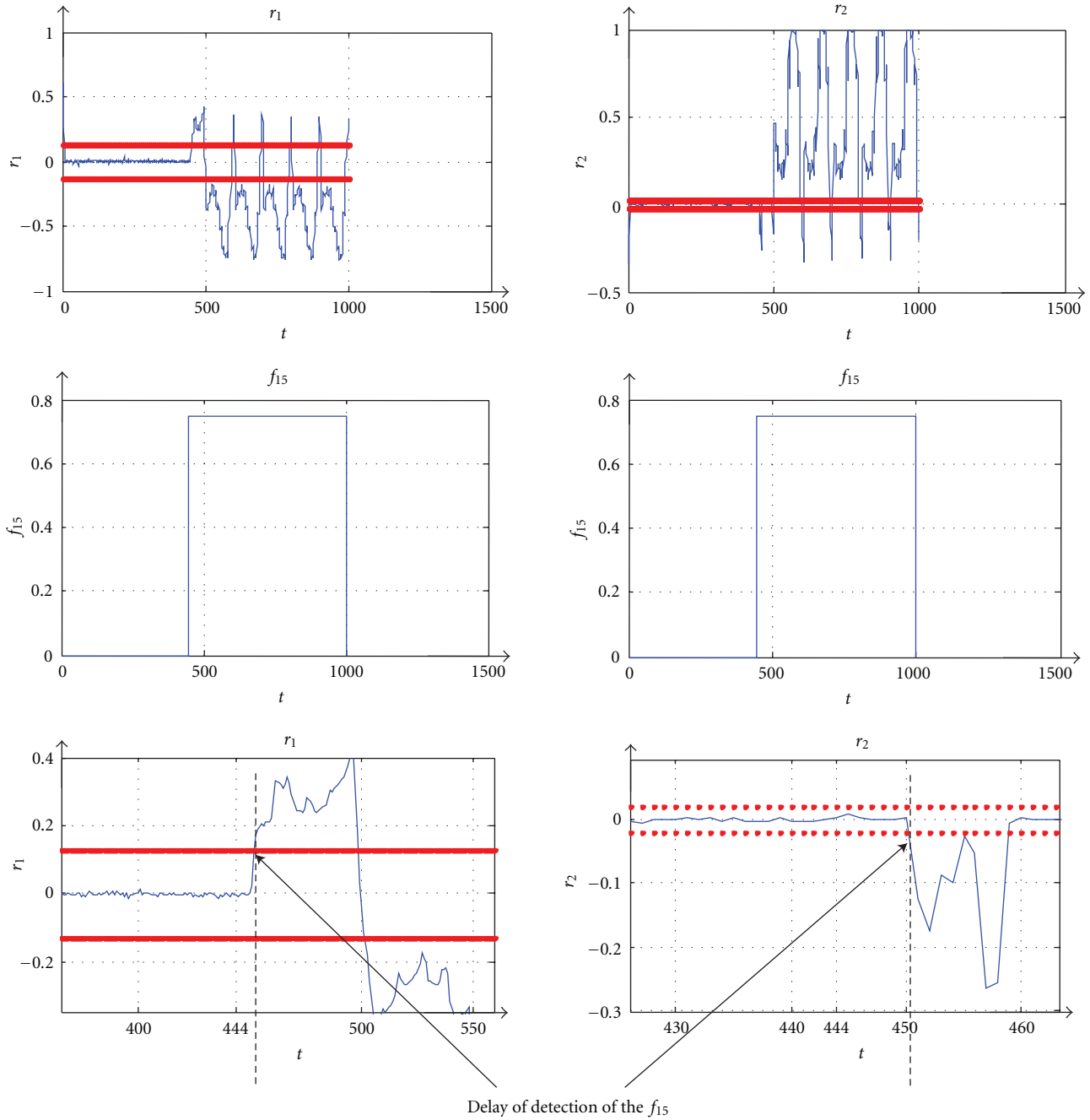
Figure 8: Residual r_{10}, r_{20} when f_{15} is simulated within time interval [444 s 1000 s].

detection is already not discussed in [30, 31]. From our point of view, delay to detection is quite important to consider as long as this delay will also influence the rapidity (and efficiency) of the diagnosis.

It is important to notice that the good performances of our approach are due to a large computation effort: 20 models with 2 outputs each of them are required to compute 40 residuals. In comparison [30] uses only 4 models with 2 outputs for each of them and [31] uses 10 residuals. The Akaike Information Criterion (AIC) and Final Prediction Error (FPE) [30] that measure the ratio complexity/performances remain quite good: FPE (net$F$) = 0.15

and IAC (net$F$) = -1.85, but these criteria do not include the complexity due to the number of networks working in parallel. Anyway, in our work, we do not consider the optimization of the ratio complexity/performance as introduced in [30], and we just use NNs of appropriate dimensions so that the training error will be small enough.

4. Conclusion

The method proposed in this work for early detection and diagnosis of faults combines the computing power and the robustness of neural networks with simple real time decision

TABLE 6: Multistep diagnosis according to the bank of models of faults.

| Fault Cand. f_j	Model of faults FM(j)		
	R_{1j} (1000, 1000, 451)	R_{1j} (1000, 1000, 451)	D_j (1000, 1000, 451)
$f1$	14.96	20.02	25.00
$f2$	9.94	11.23	15.00
$f3$	10.03	14.79	17.88
$f4$	9.99	14.29	17.44
$f5$	10.01	13.47	16.78
$f6$	10.01	18.13	20.71
$f7$	14.15	20.02	24.52
$f8$	10.01	13.57	16.86
$f9$	10.08	13.65	16.97
$f10$	11.33	16.65	20.14
$f11$	9.93	13.59	16.83
$f12$	8.12	11.01	13.68
$f13$	10.09	20.01	22.41
$f14$	10.01	13.57	16.86
$f15$	*0.53*	*0.24*	*0.58*
$f16$	11.30	16.42	19.93
$f17$	10.19	15.29	18.37
$f18$	10.01	20.02	22.38
$f19$	10.01	8.65	13.23
Fault free	10.01	13.57	16.86
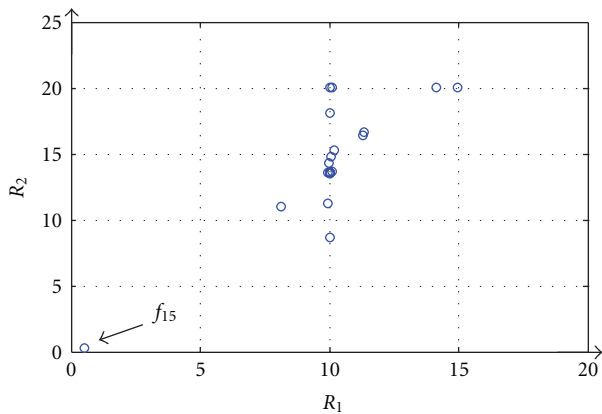
FIGURE 9: Multistep diagnosis: location of the models FM(j) in plan in plan (R_1, R_2) for fault candidates $f1$ to $f19$.

according to the Euclidean distance of cumulative residuals in residual space. The method leads to fault detection, time to failure estimation, and most probable fault evaluation. The results obtained with DAMADICS benchmark illustrate the performance of the method. But it is important to notice that the good performances of our approach are due to a large computation effort. Twenty models with two outputs each of them are required, and these networks work in parallel.

The hardest limitation of the proposed method is the necessity to design models of faults according to each fault
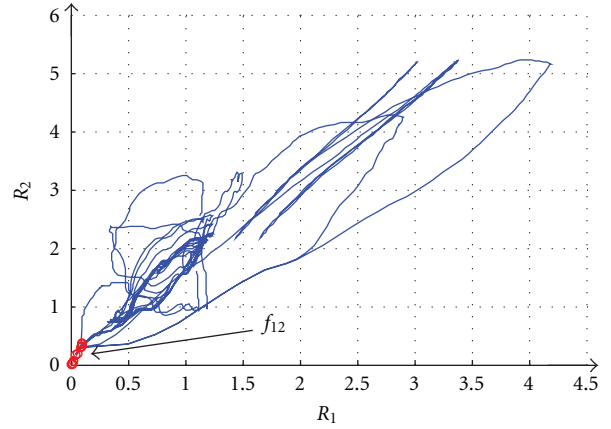

FIGURE 10: Early diagnosis: location of the models FM(j) in plan (R_1, R_2) for fault candidates $f1$ to $f19$.
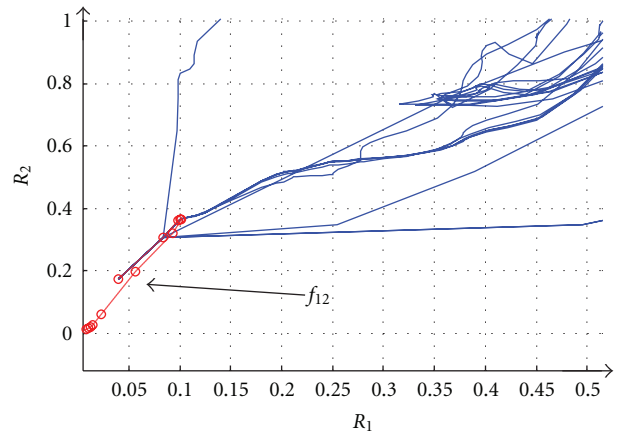

FIGURE 11: Details of the location of model FM(12) in plan (R_1, R_2).

candidate. Such design requires time, computational resources, and large history of data. We will consider systematic design of models of faulty behaviors in our future works. Another perspective is to take benefit from the correlation of diagnosis performances with the selection of estimated outputs for residuals design. The analysis of cumulative residuals in the residual space provides an interesting point of view to continue this investigation, and the covering of the residual space will be used to select appropriate residuals. Our next works will also include a deeper interpretation of the distance as a probability or likelihood]

Acknowledgment

References

[1] J. Korbicz, J. M. Koscielny, Z. Kowalczuk, and W. Cholewa, *Fault Diagnosis. Models, Artificial Intelligence, Applications*, Springer, Berlin, Germany, 2004.

[2] R. J. Patton and J. Chen, "On eigenstructure assignment for robust fault diagnosis," *International Journal of Robust and Nonlinear Control*, vol. 10, no. 14, pp. 1193–1208, 2000.

[3] H. R. Scola, R. Nikoukah, and F. Delebecque, "Test signal design for failure detection: a linear programming approach," *International Journal of Applied Mathematics and Computer Science*, vol. 13, no. 4, pp. 515–526, 2003.

[4] M. Witczak, *Modeling and Estimation Strategies for Fault Diagnosis of Non-Linear Systems. From Analytical to Soft Computing Approaches*, Springer, Berlin, Germany, 2007.

[5] E. Delaleau, J. P. Louis, and R. Ortega, "Modeling and control of induction motors," *International Journal of Applied Mathematics and Computer Science*, vol. 11, no. 1, pp. 105–129, 2001.

[6] T. Soderstrom et al., *System Identification*, Prentice-Hall International, Hemel Hempstead, Hertfordshire, UK, 1989.

[7] T. Bouthiba, "Fault location in ehv transmission lines using artificial neural networks," *International Journal of Applied Mathematics and Computer Science*, vol. 14, no. 1, pp. 69–78, 2004.

[8] M. Mrugalski, M. Witczak, and J. Korbicz, "Confidence estimation of the multi-layer perceptron and its application in fault detection systems," *Engineering Applications of Artificial Intelligence*, vol. 21, no. 6, pp. 895–906, 2008.

[9] M. M. Gupta, *Static and Dynamic Neural Networks*, John Wiley & Sons, Hoboken, NJ, USA, 2003.

[10] O. Nelles, *Non-Linear Systems Identification. From Classical Approaches to Neural Networks and Fuzzy Models*, Springer, Berlin, Germany, 2001.

[11] S. C. Tan, C. P. Lim, and M. V. C. Rao, "A hybrid neural network model for rule generation and its application to process fault detection and diagnosis," *Engineering Applications of Artificial Intelligence*, vol. 20, no. 2, pp. 203–213, 2007.

[12] V. Uraikul, C. W. Chan, and P. Tontiwachwuthikul, "Artificial intelligence for monitoring and supervisory control of process systems," *Engineering Applications of Artificial Intelligence*, vol. 20, no. 2, pp. 115–131, 2007.

[13] J. Vieira, F. M. Dias, and A. Mota, "Artificial neural networks and neuro-fuzzy systems for modelling and controlling real systems: a comparative study," *Engineering Applications of Artificial Intelligence*, vol. 17, no. 3, pp. 265–273, 2004.

[14] B. Michal, R. Patton, M. Syfert, S. de las Heras, and J. Quevedo, "Introduction to the DAMADICS actuator FDI benchmark study," *Control Engineering Practice*, vol. 14, no. 6, pp. 577–596, 2006.

[15] R. Isermann, "Model-based fault-detection and diagnosis—status and applications," *Annual Reviews in Control*, vol. 29, no. 1, pp. 71–85, 2005.

[16] R. Isermann, *Fault-Diagnosis Systems. An Introduction from Fault Detection to Fault Tolerance*, Springer, Berlin, Germany, 2006.

[17] J. C. Yang and D. W. Clarke, "The self-validating actuator," *Control Engineering Practice*, vol. 7, no. 2, pp. 249–260, 1999.

[18] M. Henry, "Plant asset management via intelligent sensors digital, distributed and for free," *Computing and Control Engineering Journal*, vol. 11, no. 5, pp. 211–213, 2000.

[19] M. Tombs, "Intelligent and self-validating sensors and actuators," *Computing and Control Engineering Journal*, vol. 13, no. 5, pp. 218–220, 2002.

[20] J. B. Gomm, D. L. Yu, and D. Williams, "Sensor fault diagnosis in a chemical process via RBF neural networks," *Control Engineering Practice*, vol. 7, no. 1, pp. 49–55, 1999.

[21] L. Yaguol, H. Zhengjia, and Z. Yahyang, "A new approach to intelligent fault diagnosis of rotating machinery," *Expert Systems with Applications*, vol. 35, no. 4, pp. 1593–1600, 2008.

[22] H. T. Mok and C. W. Chan, "Online fault detection and isolation of nonlinear systems based on neurofuzzy networks," *Engineering Applications of Artificial Intelligence*, vol. 21, no. 2, pp. 171–181, 2008.

[23] J. Zhao, J. Huang, and W. Sun, "On-line early fault detection and diagnosis of municipal solid waste incinerators," *Waste Management*, vol. 28, no. 11, pp. 2406–2414, 2008.

[24] H. Hjalmarsson, A. Juditsky, J. Sjöberg et al., "Nonlinear black-box modeling in system identification: a unified overview," *Automatica*, vol. 31, no. 12, pp. 1691–1724, 1995.

[25] Y. Kourd, N. Guersi, and D. Lefebvre, "A two stages diagnosis method with neuronal networks," in *Proceedings of the International Conference on Electrical Engineering Design and Technologies (ICEEDT '08)*, Hammamet, Tunisie, 2008.

[26] Y. Kourd, N. Guersi, and D. Lefebvre, "A two stages diagnosis method with Neuro-fuzzy approach," in *Proceedings of the 6th Conférence Internationale Francophone d'Automatique*, Nancy, France, 2010.

[27] Y. Kourd, N. Guersi, and D. Lefebvre, "Neuro-fuzzy approach for fault diagnosis: application to the DAMADICS," in *Proceedings of the 7th international Conference on Informatics in Control, Automation and Robotics (ICINCO '10)*, Funchal, Madeira, Portugal, 2010.

[28] M. Blanke, M. Kinnaert, J. Lunze, and M. Staroswiecki, *Diagnosis and Fault Tolerant Control*, Springer, New York, NY, USA, 2003.

[29] D. Lefebvre, H. Chafouk, and M. Lebbal, *Modélisation et Diagnostic des Systèmes. Une Approche Hybride*, Éditions universitaires européennes, 2010.

[30] K. Patan and T. Parisini, "Identification of neural dynamic models for fault detection and isolation: the case of a real sugar evaporation process," *Journal of Process Control*, vol. 15, no. 1, pp. 67–79, 2005.

[31] J. M. Kościelny, M. Bartyś, P. Rzepiejewski, and J. Sá da Costa, "Actuator fault distinguishability study for the DAMADICS benchmark problem," *Control Engineering Practice*, vol. 14, no. 6, pp. 645–652, 2006.

[32] DAMADICS, 2002, http://diag.mchtr.pw.edu.pl/damadics/.