International Journal of Information, Control and Computer Sciences
ISSN: 2517-9942
Vol:9, No:2, 2015

Fault Diagnosis of Nonlinear Systems Using Dynamic Neural Networks

E. Sobhani-Tehrani, K. Khorasani, N. Meskin

***Abstract*—**This paper presents a novel integrated *hybrid* approach for fault diagnosis (FD) of nonlinear systems. Unlike most FD techniques, the proposed solution *simultaneously* accomplishes fault detection, isolation, and identification (FDII) within a unified diagnostic module. At the core of this solution is a bank of adaptive neural parameter estimators (NPE) associated with a set of single-parameter fault models. The NPEs continuously estimate *unknown* fault parameters (FP) that are indicators of faults in the system. Two NPE structures including series-parallel and parallel are developed with their exclusive set of desirable attributes. The *parallel* scheme is extremely robust to measurement noise and possesses a simpler, yet more solid, fault isolation logic. On the contrary, the *series-parallel* scheme displays short FD delays and is robust to closed-loop system transients due to changes in control commands. Finally, a fault tolerant observer (FTO) is designed to extend the capability of the NPEs to systems with partial-state measurement.

***Keywords*—**Hybrid fault diagnosis, Dynamic neural networks, Nonlinear systems.

I. Introduction

INCREASING demand on reliable operation of safety-critical control systems such as intelligent vehicles and future planned autonomous spacecraft/probes has made fault detection, isolation, and identification an essential component of an autonomous system. There is a high demand for developing intelligent systems that are able to autonomously detect and isolate the location of faults occurring in different components of complex systems. Furthermore, accurate estimation of fault severities is essential for development of reliable autonomous recovery procedures as well as component health monitoring and condition-based maintenance, where accurate estimation of a component's health state and consequently prediction of its remaining useful life is of utmost importance.

During the past two decades, a number of approaches have been developed for fault detection and isolation (FDI) of both deterministic and stochastic nonlinear systems. Many techniques utilize either analytical model-based [1]-[3] or learning-based methodologies [3]-[5] using qualitative or quantitative modeling. The problem of FDI of nonlinear stochastic systems often represented by general hidden Markov models have been solved using adaptive change detection [6], likelihood ratio approach with adaptive Monte Carlo as well as particle filters [7], [8], and entropy optimization filtering [9]. However, only few works have been reported in the literature that exploits both mathematical models of a system and adaptive nature of intelligent techniques such as neural networks [10]-[12]. Furthermore, the importance of utilizing an integrated framework to simultaneously achieve FDI and fault severity estimation has not been fully addressed.

The fault diagnosis technique proposed in this work is essentially a hybrid approach due to the use of neural networks in conjunction with a mathematical model of the system as a basis for fault modeling. Fault modeling can be accomplished in a variety of ways and perspectives. For example, [12], [9] have modeled a fault as an unknown nonlinear function of the system states and inputs. On the other hand, neural networks are used in [5] to identify the full system dynamics including nominal and faulty dynamics, under different fault scenarios. The fault modeling approach adopted in this paper is based on the notion of fault parameters (FP) as defined in [11] to parameterize a known mathematical model of the system with unknown parameters that reflect the occurrence of faults.

In this paper, a new integrated solution to the problems of nonlinear FDI and fault severity estimation is proposed. By integrating fault detection, isolation, and severity estimation into a single module, our proposed solution reduces the structural complexity of conventional fault diagnosis schemes. The idea of an integrated framework for simultaneous FDI and fault severity estimation has been previously proposed by, for example [12] for a special class of faults where the state equation is an *affine* function of the FPs. Moreover, Xia [12] assumes full-state measurement (as can be seen from equation (1) of their paper). Also, conventional methods consist of a combination of independent or interconnected subsystems such as residual post-processing subsystem and parameter estimation subsystem that perform one of the above three tasks.

The idea of using a bank of estimators/observers/models for fault detection and isolation has been previously pursued in the literature by many researchers (see for example, [11], [14]-[18]). However, *none* of them have addressed the problem of fault severity identification. In this paper, we have developed a bank of estimators that allow accurate identification of fault severity while the fault is being detected and isolated. More

E. S. Tehrani was with the Department of Electrical and Computer Engineering, Concordia University, Montreal, Quebec, Canada. He is now with Globvision, Inc.

K. Khorasani is with the Department of Electrical and Computer Engineering, Concordia University, Montreal, Quebec, Canada (e-mail: kash@ece.concordia.ca).

N. Meskin is with the Department of Electrical Engineering, Qatar University, Doha, Qatar.

This publication was made possible by NPRP grant No. 5-045-2-017 from the Qatar National Research Fund (a member of Qatar Foundation). The statements made herein are solely the responsibility of the authors.

specifically, once a bank of parameterized fault models is constructed, a corresponding bank of neural parameter estimators (NPE) is designed to estimate fault parameters and thus accomplish fault identification. Therefore, even in terms of methodology, the proposed *hybrid* fault diagnosis approach can be viewed as an *integration* of two well-known fault diagnosis methodologies including multiple-model (MM) approach and parameter estimation method.

Furthermore, two NPE structures, namely *series-parallel* and *parallel*, are proposed with their respective fault isolation policies, where each structure possesses an exclusive set of desirable properties. For example, the proposed *parallel* scheme is extremely robust to measurement noise, hence making it suitable for low SNR applications. On the other hand, the *series-parallel* scheme displays very fast convergence rates desirable for systems requiring short delay in fault diagnosis. Thus, the choice of the appropriate FDI structure really depends on the specifications and requirements of the specific problem at hand.

The robust *parallel* FDII scheme proposed in this work is an entirely *novel* development in the literature. One of the main contributions of this work is the integration of fault-tolerant observers with the fault parameter estimators in order to accomplish FDII under partial-state measurements. To the best of our knowledge, this has not been pursued by any previous work in the literature. On the contrary, [11] and [10] have previously developed FDI techniques similar to the *series-parallel* scheme. However, the *series-parallel* scheme proposed in this paper possesses the following *three novelties*: (i) more solid fault isolation results due to the first-time use of a bank of *single-parameter* fault models (4) extracted from the *multi-parameter* fault model (3) employed by [11], (ii) remarkably simpler neural network architecture and adaptation laws than those employed by [11] and [10], which makes the proposed methodology more suitable for real-time implementation, and (iii) added fault identification capability – the simulation results presented in [11] do not demonstrate such capability.

II. Problem Definition and Fault Modeling

In this section, the problem of fault diagnosis in components of a general nonlinear system is formally stated. Consider a nonlinear system that is described by the following discrete-time state space representation:

$$\begin{cases} x_{k+1} = f(x_k, u_k) + \Gamma(x_k) w_k \\ y_k = H x_k + \nu_k \end{cases} \quad (1)$$

where $x_k \in \Re^n$ is the system state vector, $f : \Re^n \times \Re^r \to \Re^n$, is a smooth nonlinear vector-valued function (or vector field) on its domain, $u_k \in R^r$ is the control input vector, $y_k \in R^m$ is the system output vector, and w_k and ν_k represent system disturbances and measurement noise, respectively. The vector field f represents the nonlinear dynamics and the matrix H represents the linear output equation of the nominal model of the system. The state-dependent function $\Gamma(.)$ essentially represents the channel over which the external disturbances are applied to the system. It is assumed that the disturbances are bounded signals, that is:

$$\|w_k\| \le D_{\max} \ , \ \|\nu_k\| \le N_{\max} \quad \forall k \in N \quad (2)$$

Under the full-state measurement assumption the output equation in (1) can be redefined as $y_k = x_k + \nu_k$, whereas under the partial state measurement the general output equation (1) is considered.

Our objective is to design and develop a fault diagnosis scheme that is capable of autonomously detecting the presence, isolating the location, and identifying the severity of faults in the system within a unified framework. We make the following assumptions regarding the occurrence of faults in the system, which comprise the basis for fault diagnosis design, development, and verification.

Assumption (i). The control signals and the state vector remain bounded prior to and after the occurrence of a fault.

Assumption (ii). The faults do not *occur* precisely at the same time; i.e., at each instant of time only one fault may occur in the system. Note that this does *not* exclude *existence* of concurrent faults in the system.

Assumption (iii). The rate of variation of fault severities is "slow" compared to the dynamics of the system states.

Assumption (iv). The input and output signals in model (1) satisfy the so-called persistent excitation condition to ensure that the parameter estimators that are designed subsequently guarantee convergence to their true values.

It should be noted that this is a reasonable assumption for most engineering systems. Because for *abrupt* faults, once they occur it is not likely that their severity changes over time and for *incipient* faults, since they occur due to wear and tear of system components, the fault growth rates are often much slower than system dynamics.

Generally speaking, different models of a faulty system may be constructed. In this paper, following the work of [11]-[20], we have assumed that the system component faults are reflected in the physical system parameters. Thus, the presence of faults in the system can be represented by changes in the system parameters. The faulty system is described by the following parameterized nonlinear model, called *multi-parameter* fault model:

$$\Omega : \begin{cases} x_{k+1} = f(x_k, u_k, \alpha_k) + \Gamma(x_k) w_k \\ y_k = H x_k + \nu_k \end{cases} \quad (3)$$

where $\alpha_k \in \Re^L$ is the fault parameter vector containing L fault elements. Furthermore, $\alpha_k = \alpha_H$ entails the absence of faults in the system, i.e., healthy mode of operation. The value of α_H

depends on the way the FP vector affects the system model parameters in (3); that is being either additive or multiplicative. The representation adopted in this paper is the additive form, hence making $\alpha_H = [0]_{L\times 1}$.

The fault model given by (3) enables one to state the problem of nonlinear fault diagnosis in the form of an on-line nonlinear parameter estimation problem, where the unknown fault parameters are being estimated by using system inputs and measurements.

Within our proposed fault diagnosis framework, fault detection is accomplished by first estimating the fault parameter vector using system input-output measurements and then comparing the estimated parameters against α_H. For fault isolation and fault severity estimation purposes, however, we propose a bank of parameter estimators where each estimator in the bank is designed based on a *single-parameter* fault model as described below.

Consider the *multi-parameter* fault model given by (3) with L fault parameters. One can extract L *single-parameter* models, $\Omega_i, i = 1,..., L$, from model (3) as follows:

$$\Omega_i : \begin{cases} x_{k+1} = f(x_k, u_k, \alpha_k^i) + \Gamma(x_k) w_k \\ y_k = Hx_k + v_k \end{cases}; \; i = 1,..., L \tag{4}$$

A bank of L parameter estimators may then be designed based on each *single-parameter* fault model in (3), where the i^{th} parameter estimator will essentially estimate the i^{th} fault parameter, namely α_k^i. While [8] has introduced subset parameter estimation for detection and isolation of faults, in our opinion, the extraction of *single-parameter* fault models from the *multi-parameter* fault model to enable fault severity estimation (in addition to FDI) is introduced here for the first time in the literature.

To overcome the shortcomings in estimating the parameters of a nonlinear system with arbitrary disturbance/process noise distributions, we integrate multi-layer feed-forward (static) neural networks with the true nonlinear dynamical model of the system in our proposed nonlinear parameter estimation scheme. The estimation of parameters is then accomplished based on an on-line minimization of instantaneous output estimation error as described in the next sections.

III. Proposed FDII Using Series-Parallel Architecture of Neural Parameter Estimators and Full State Measurement

Fig. 1 depicts the structure of a bank of series-parallel NPEs designed and developed to simultaneously achieve the three objectives of fault detection, isolation, and fault severity estimation. The residual signals $r_k^i, i = 1,..., L$ and the FP estimates $\hat{\alpha}_k^i, i = 1,..., L$ comprise the outputs of the series-parallel scheme and the three tasks of FDII are accomplished by examining these quantities. The fault detection and isolation (FDI) decision logic of this scheme is presented in Section III *B*.

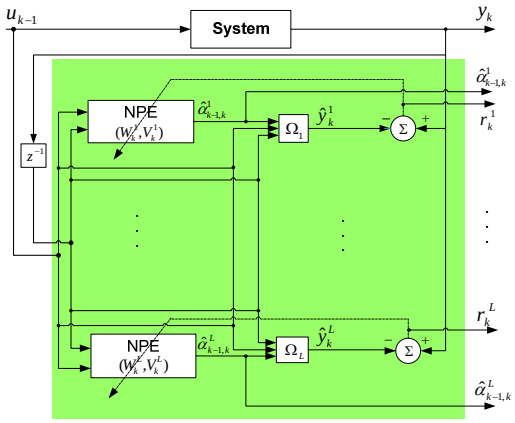

Fig. 1 *Series-Parallel* scheme of the proposed hybrid FDII approach

The series-parallel structure is composed of two main subsystems: (1) the feed-forward (static) neural networks (FFNN) (i.e., the NPEs) utilized to adaptively approximate the nonlinear FP estimation functions and (2) the nonlinear *single-parameter* fault models given by (4) utilized for state/output estimation (or prediction) based on the estimated FPs. Accordingly, at each time-step k, the following two sets of calculations are performed for each NPE:

1) Calculation of FP Estimates

$$\hat{\alpha}_{k-1,k}^i = g(\bar{y}_k, W_k^i, V_k^i) \; ; \; i = 1,..., L \tag{5}$$

$$\bar{y}_k = [y_{k-1} \quad u_{k-1}]^T \tag{6}$$

where $\hat{\alpha}_{k-1,k}^i$ is the estimate of the i^{th} fault parameter at time $k-1$ calculated at time k, W_k^i, V_k^i are respectively the output and the hidden layer weight matrices of the i^{th} NPE, $\bar{y}_k$ is the input vector of the NPEs, and g is the nonlinear mapping implemented by a single hidden layer FFNN with linear activation functions for the neurons at the output layer and nonlinear activation functions for the neurons at the hidden-layer. Thus,

$$g(\bar{y}_k, W_k^i, V_k^i) = W_k^i \sigma(V_k^i \bar{y}_k) \tag{7}$$

where $\sigma(.)$ is the activation function of the hidden-layer neurons that is usually set to be a *sigmoidal* function:

$$\sigma_j(V_k^i \bar{y}_k) = \frac{2}{1 + \exp(-2V_{k_j}^i \bar{y}_k)} - 1 \tag{8}$$

where $V_{k_j}^i$ is the j^{th} row of V_k^i and $\sigma_j(V_k^i \bar{y}_k)$ is the j^{th} element of $\sigma(V_k^i \bar{y}_k)$.

2) State/Output Estimation (or Prediction) Based On FP Estimates

In this step, the states and consequently the outputs of the system are estimated (or predicted) by the deterministic part of the *single-parameter* fault models in (3) (i.e., without *unknown* external disturbances w_k and measurement noise v_k) and using the FP vector estimate from step 1, namely $\hat{\alpha}_{k-1,k}$. Hence,

$$\begin{cases} \hat{x}_k^i = f\left(x_{k-1}, u_{k-1}, \hat{\alpha}_{k-1,k}^i\right) \\ \hat{y}_k^i = \hat{x}_k^i \end{cases} ; i = 1,...,L \tag{9}$$

where $x_{k-1} = y_{k-1}$ are the measured states of the system.

A. Weight Update Laws of the Series-Parallel Scheme

The weights of the NPEs are updated with the objective of minimizing the weighted L_2 norm of the instantaneous output estimation error vector defined as:

$$\tilde{y}_k^i = y_k - \hat{y}_k^i \quad ; i = 1,...,L \tag{10}$$

Thus, the objective function, at time-step k, of the i^{th} NPE is the instantaneous output error:

$$J_k^i = \frac{1}{2}\left\|\tilde{y}_k^i\right\|_Q^2 = \frac{1}{2}\tilde{y}_k^{i\,\mathrm{T}} Q \tilde{y}_k^i \tag{11}$$

where $Q \in \Re^{n \times n}$ is the estimation error weight matrix. The weights of the NPEs are updated by using the well-known gradient descent (GD) algorithm:

$$\begin{aligned} W_{k+1}^i &= W_k^i - \eta_w^i \left(\frac{\partial J_k^i}{\partial W_k^i} \right) \\ V_{k+1}^i &= V_k^i - \eta_v^i \left(\frac{\partial J_k^i}{\partial V_k^i} \right) \end{aligned} \quad ; i = 1,..., L \tag{12}$$

where $\eta_w^i, \eta_v^i > 0; i = 1,..., L$ are the learning rates.

In order to derive the weight update laws, let us define for $i = 1,..., L$; $net_{v_k}^i = V_k^i \bar{y}_k$, and $net_{w_k}^i = W_k^i \sigma\left(V_k^i \bar{y}_k\right)$. Thus, the partial derivatives $\partial J_k^i / \partial W_k^i$, $\partial J_k^i / \partial V_k^i$ can be computed according to:

$$\frac{\partial J_k^i}{\partial W_k^i} = \frac{\partial J_k^i}{\partial net_{w_k}^i} \frac{\partial net_{w_k}^i}{\partial W_k^i} \tag{13}$$

$$\frac{\partial J_k^i}{\partial V_k^i} = \frac{\partial J_k^i}{\partial net_{v_k}^i} \frac{\partial net_{v_k}^i}{\partial V_k^i} \tag{14}$$

where

$$\frac{\partial J_k^i}{\partial net_{w_k}^i} = \frac{\partial J_k^i}{\partial \tilde{y}_k^i} \frac{\partial \tilde{y}_k^i}{\partial \hat{y}_k^i} \frac{\partial \hat{y}_k^i}{\partial \hat{x}_k^i} \frac{\partial \hat{x}_k^i}{\partial \hat{\alpha}_{k-1,k}^i} \frac{\partial \hat{\alpha}_{k-1,k}^i}{\partial net_{w_k}^i} = -\tilde{y}_k^{i\,\mathrm{T}} Q \frac{\partial \hat{x}_k^i}{\partial \hat{\alpha}_{k-1,k}^i} \frac{\partial \hat{\alpha}_{k-1,k}^i}{\partial net_{w_k}^i}$$

$$\frac{\partial J_k^i}{\partial net_{v_k}^i} = \frac{\partial J_k^i}{\partial \tilde{y}_k^i} \frac{\partial \tilde{y}_k^i}{\partial \hat{y}_k^i} \frac{\partial \hat{y}_k^i}{\partial \hat{x}_k^i} \frac{\partial \hat{x}_k^i}{\partial \hat{\alpha}_{k-1,k}^i} \frac{\partial \hat{\alpha}_{k-1,k}^i}{\partial net_{v_k}^i} = -\tilde{y}_k^{i\,\mathrm{T}} Q \frac{\partial \hat{x}_k^i}{\partial \hat{\alpha}_{k-1,k}^i} \frac{\partial \hat{\alpha}_{k-1,k}^i}{\partial net_{v_k}^i}$$

$$\frac{\partial net_{w_k}^i}{\partial W_k^i} = \sigma\left(V_k^i \bar{y}_k\right), \quad \frac{\partial net_{v_k}^i}{\partial V_k^i} = \bar{y}_k$$

The partial derivative $\partial \hat{x}_k^i / \partial \hat{\alpha}_{k-1,k}^i$; $i = 1,..., L$ is calculated by using the i^{th} state estimation equation (9) as:

$$\frac{\partial \hat{x}_k^i}{\partial \hat{\alpha}_{k-1,k}^i} = \frac{\partial f\left(x_{k-1}, u_{k-1}, \hat{\alpha}_{k-1,k}^i\right)}{\partial \hat{\alpha}_{k-1,k}^i} \tag{15}$$

which is essentially the Jacobian of the vector-valued function f with respect to the *scalar* parameter $\hat{\alpha}_{k-1,k}^i$. However, it should be noted that we do not need to calculate the Jacobian matrix of the system with respect to its states, which is an advantage from implementation point of view.

Finally, the well-known standard back-propagation (BP) algorithm is used to calculate the partial derivatives $\partial \hat{\alpha}_{k-1,k}^i / \partial net_{w_k}^i$ and $\partial \hat{\alpha}_{k-1,k}^i / \partial net_{v_k}^i$ for $i = 1,..., L$. Due to the *linearity* of the output layer of the NPEs, we simply have $\frac{\partial \hat{\alpha}_{k-1,k}^i}{\partial net_{w_k}^i} = 1$, and taking into account the sigmoidal activation functions of the hidden layer of the NPEs, we have

$$\frac{\partial \hat{\alpha}_{k-1,k}^i}{\partial net_{v_k}^i} = W_k^i \left(I - \Lambda\left(V_k^i \bar{y}_k\right) \right)$$

where $\Lambda\left(V_k^i \bar{y}_k\right) = \mathrm{diag}\left[\sigma_j^2\left(V_{k_j}^i \bar{y}_k\right)\right]$, $j = 1,..., S^i$; and S^i is the number of neurons in the hidden-layer of the i^{th} NPE and $V_{k_j}^i$ is, once again, the j^{th} row of V_k^i.

B. FDI Decision Logic of the Series-Parallel Scheme and the Threshold Selection Criteria

To formulate the FDI decision logic, we need to define a set of residual vectors as follows (note that a total of L residual vectors can be defined one per state estimator in the bank):

$$r_k^i = y_k - \hat{y}_k^i \quad ; i = 1,..., L \tag{16}$$

Given *Assumption (ii)*, the FDI decision logic for the series-parallel scheme is quite straight-forward and can be stated as:

$$\left(C_k^F, T_F^C\right) = \left\{ (i,k) \,\middle|\, \left|r_k^{i,j}\right| \le \delta^j \wedge \left|\hat{\alpha}_k^i - \alpha_H^i\right| > \varepsilon^i \right\} \quad l = 1,..., L;\ l \ne i;\ j = 1,..., n \tag{17}$$

where $\left|r_k^{i,j}\right|$ is the absolute value of the j^{th} element of the residual vector corresponding to the i^{th} NPE in the bank; $\delta^j; j = 1,..., n$ denotes the thresholds associated with the output residuals of the NPEs; $\varepsilon^i; i = 1,..., L$ denotes the thresholds corresponding to the FP estimate of the i^{th} NPE in the bank; C_k^F specifies (the index of) the faulty component(s) within the system at each instant of time (i.e., the *health state* of the system); α_H^i is the value of the i^{th} FP under nominal, healthy

conditions (which is "zero" for additive FPs and "1" for multiplicative FPs); and finally, T_F^C represents the detection and isolation time (or time-step) of the fault(s). Under healthy conditions, C_k^F should ideally (i.e., under *perfect detection*) be an *empty* set (i.e., $C_k^F = \varnothing$). On the other hand, in presence of *only* one faulty component in the system, C_k^F should ideally (i.e., under perfect *isolation*) belong to the set $\{1,...,L\}$. However, in case of imperfect isolation, C_k^F could be a subset of the set $\{1,...,L\}$, consisting of more than one elements.

It should be noted that as opposed to the thresholds $\varepsilon^i; i = 1,...,L$, the thresholds $\delta^j; j = 1,...,n$ are common (or equal) across all NPEs in the bank of filters. As can also be seen from the FDI decision logic in (17), both the *residual* signals and the FP estimates $\hat{\alpha}_k^i; i = 1,...,L$ are examined in the series-parallel scheme to detect the presence and isolate the location of faults in the monitored system. Once a fault is detected and the faulty component is isolated, the severity of the fault is essentially the value of the corresponding FP estimate, namely $\hat{\alpha}_{k-1,k}^{C_k^F}$.

In the series-parallel scheme fault detection can be ensured if the well-known worst-case noise/disturbance analysis is employed for assigning the thresholds $\delta^j; j = 1,...,n$ in (17). However, this does not guarantee that fault isolation will be *perfectly* achieved. More precisely, the i^{th} FP estimate $\hat{\alpha}_{k-1,k}^i$ is *not* perfectly decoupled from *all fault sources but the* i^{th} *one* (i.e., the fault sources $j = 1,..., i-1, i+1,..., L$). In fact, there is always a *weak* impact from the fault sources $j = 1,..., i-1, i+1,..., L$ on the i^{th} FP estimate $\hat{\alpha}_{k-1,k}^i$, as will be demonstrated using simulations in Section VIII. However, this *weak* impact can be resolved by properly setting the thresholds $\varepsilon^i; i = 1,...,L$. A good rule of thumb that augments the reliability of FDII and ensures the safety of the system is to select the thresholds $\varepsilon^i; i = 1,...,L$ in a way that the occurrence of the i^{th} fault with a severity level below its respective threshold ε^i does not significantly deteriorate the system performance.

IV. Robust FDII Using Parallel Architecture of NPEs

The series-parallel scheme developed in the previous section possesses several advantages including simple FDI decision logic (as discussed earlier) and fast convergence (which will be demonstrated in Section VIII). However, it may incorrectly isolate faults specially when there is a strong coupling between two fault sources. Furthermore, as illustrated in the simulation results of Section VIII, the series-parallel scheme suffers from lack of robustness to measurement noise. In particular, measurement noise significantly deteriorates the fault isolation and identification performance of the series-parallel scheme. This is due to the fact that measurement noise directly propagates through the network, thus affecting the FP estimates.

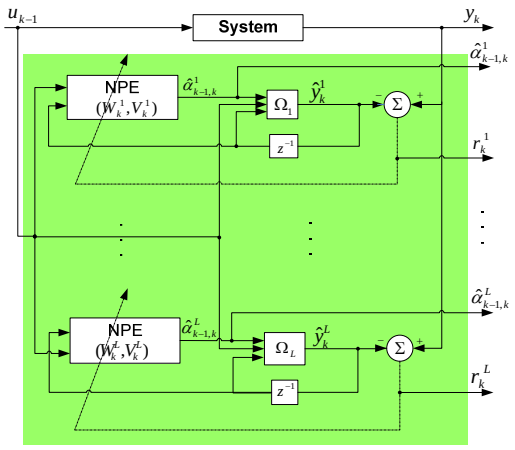

Fig. 2 *Parallel* scheme of the proposed hybrid FDII approach

The sensitivity of the series-parallel scheme to measurement noise makes it impractical and unreliable for fault diagnosis in low SNR applications. The parallel scheme developed in this section intelligently resolves this issue by feeding back the estimated rather than the measured outputs to the NPE input. This slight restructuring of the series-parallel scheme makes the measurement noises to be filtered out in the NPE weight adaptation process of the parallel FDII scheme, hence making it extremely robust to measurement noise. The strong insensitivity of the FDII performance of the parallel scheme to measurement noise will be demonstrated in Section VIII. The schematic of the *robust* parallel structure of the proposed *hybrid* FDII methodology is shown in Fig. 2.

Furthermore, by using a special formulation of the FDI decision logic, the parallel scheme allows fault isolation to be *perfectly* achieved in contrast to the series-parallel scheme. The reason for *perfect* isolation in the parallel scheme is that the only signal common among the inputs of all state estimators (or predictors) and the NPEs in the bank is the control input signal. More specifically, each NPE and state estimator in the bank utilizes its own state estimate (or prediction), which automatically enforces a structural decoupling between the units. Clearly, this restructuring also has a disadvantage of slower convergence rate for the state estimators and the NPEs of the parallel scheme as compared to the series-parallel approach. This slower convergence rate causes longer fault diagnosis delays and makes the parallel scheme sensitive to transients of the closed-loop system (due to changes in the control command). More specifically, while the state estimates from the series-parallel scheme very quickly converge to the measured states (thus extremely robust to closed-loop system transients), the parallel scheme generates false alarms during the transients until the steady state of the closed-loop system is reached.

The NPE weight adaptation laws of the parallel structure remain essentially similar to those of the series-parallel scheme with only slight modifications; however, the FDI decision logics of the two are a bit different as described below.

Since instead of the actual measurements, the output estimates (or predictions) are fed back to the NPEs' and state estimators, y_{k-1} in (6) should be replaced by $\hat{y}^i_{k-1}$, and x_{k-1} (equal to y_{k-1} under full-state measurement assumption) in (9) must be replaced by $\hat{x}^i_{k-1}$ for $i=1,...,L$. Hence, for the robust parallel FDII scheme we have:

$$\hat{\alpha}^i_{k-1,k} = g\left(\bar{y}^i_k, W^i_k, V^i_k\right) \; ; i=1,...,L \tag{18}$$

$$\bar{y}^i_k = \begin{bmatrix} \hat{x}^i_{k-1} & u_{k-1} \end{bmatrix}^{\mathrm{T}} ; i=1,...,L \tag{19}$$

Moreover,

$$\begin{cases} \hat{x}^i_k = f\left(\hat{x}^i_{k-1}, u_{k-1}, \hat{\alpha}^i_{k-1,k}\right) \\ \hat{y}^i_k = \hat{x}^i_k \end{cases} ; i=1,...,L \tag{20}$$

A. Weight Update Laws of the Robust Parallel Scheme

Once the above adjustments are made to (6) and (9), the weight update laws remain practically intact, since they are written in terms of $\bar{y}_k$ in (6), which represents the input vector of the NPEs. The only required adjustment to the weight update laws of the series-parallel scheme that may need to be explicitly re-emphasized is in (15). For the robust parallel structure, this equation should be reinstated for $i=1,...,L$ as:

$$\frac{\partial \hat{x}^i_k}{\partial \hat{\alpha}^i_{k-1,k}} = \frac{\partial f\left(\hat{x}^i_{k-1}, u_{k-1}, \hat{\alpha}^i_{k-1,k}\right)}{\partial \hat{\alpha}^i_{k-1,k}} \tag{21}$$

B. Fault Isolation Policy of the Parallel Scheme

Once again, we need to define a set of L residual vectors – one per state estimator in the bank – as:

$$r^i_k = y_k - \hat{y}^i_k ; \;\; i=1,...,L \tag{22}$$

In the sequel, the FDI decision strategy is defined as:

$$\left(C^F_k, T^C_F\right) = \left\{(i,k) \,\middle|\, \left|r^{i,j}_k\right| \le \delta^j \wedge \left|r^{l,j}_k\right| > \delta^j ; \; l=1,...,L; \; l \ne i; \; j=1,...,n \right\} \tag{23}$$

where $r^{i,j}_k$ denotes the j^{th} element of residual vector r^i_k and $\delta^j ; j=1,...,n$ are the thresholds corresponding to the state residuals of the NPEs. It should be noted that the thresholds $\delta^j ; j=1,...,n$ are common (or equal) across all NPEs in the bank. The above fault isolation policy states that the fault model with residuals within the threshold bounds is actually the current active mode of the system. In the parallel scheme, threshold values are determined by using the worst-case disturbance/noise analysis. Once the fault source is isolated, the severity of the fault is essentially the value of the corresponding FP estimate. It should be noted that the FDI decision logic of the robust parallel scheme is simpler to design than that of the series-parallel scheme. This can be simply observed by comparing (23) with (17). Therefore, the FDI logic of the parallel scheme has only n parameters to be specified corresponding to the residual thresholds $\delta^j ; j=1,...,n$, with n being the order of the monitored system. On the other hand, in the series-parallel scheme $n+L$ parameters need to be specified, where the thresholds associated with FPs $\varepsilon^i ; i=1,...,L$ have to be determined as well.

Remark: The reason for having different FDI decision logics is indeed due to the difference between the architecture of the series-parallel and the parallel schemes. As opposed to the series-parallel scheme, the only common input signal of the state estimators and the NPEs of the parallel FDII strategy is the control input (represented by u_k). More specifically, each NPE and state estimator in the bank of parallel scheme utilizes its own state estimate as opposed to the series-parallel scheme that uses a common output measurement vector (in addition to the common control input). This automatically enforces a structural decoupling between the units in the bank of the parallel scheme. Hence, the fault isolation logic of the parallel scheme states that the fault model with residuals within the threshold bounds is actually the current active mode of the system. However, in case of the series-parallel scheme this is not necessarily the case because the output measurements of the faulty system that is fed back to the state estimators of the series-parallel scheme can maintain more than one residual within the threshold bounds with slight changes in parameter values. Hence, there is a need to check the estimated parameter values in order to distinguish the actual faulty mode of the system.

V. Proposed FDII under Partial State Measurements

Fig. 3 depicts a block diagram representing the extension of the series-parallel FDII scheme to partial-state measurement conditions. As shown in Fig. 3 this extension is based upon integration of the *hybrid* NPEs of the series-parallel FDII scheme with a fault tolerant observer.

The states are decomposed into measured and unmeasured as:

$$x = \begin{bmatrix} x^m & x^{unm} \end{bmatrix} = \begin{bmatrix} y & x^{unm} \end{bmatrix} \tag{24}$$

where x^m denotes the subset of system states directly measured by sensors (i.e., system outputs) and x^{unm} represents the subset of *unmeasured* states of the system.

As can be seen from Fig. 3, the measured states $x^m_k = y_k$ are fed directly to the series-parallel FDII scheme, while the unmeasured states are first estimated by the FTO using system inputs and output measurements and then these estimates $\hat{x}^{unm}_k$ are fed as inputs to the bank of NPEs of the FDII module. Accordingly, the NPE update laws and the FDI decision logic of the series-parallel FDII scheme have to be slightly modified as follows:

(i) In (6), y_{k-1} is replaced by $\begin{bmatrix} x^m_{k-1} & \hat{x}^{unm}_{k-1} \end{bmatrix}$; that is,

$$\bar{y}_k = \begin{bmatrix} \begin{bmatrix} x^m_{k-1} & \hat{x}^{unm}_{k-1} \end{bmatrix} & u_{k-1} \end{bmatrix}^{\mathrm{T}} \tag{25}$$

(ii) In (9), x_{k-1} is replaced by $\left[x_{k-1}^{m} \quad \hat{x}_{k-1}^{unm} \right]$; that is,

$$\begin{cases} \hat{x}_k^i = f\left(\left[x_{k-1}^{m} \quad \hat{x}_{k-1}^{unm} \right], u_{k-1}, \hat{\alpha}_{k-1,k}^i\right) \\ \hat{y}_k^i = \hat{x}_k^i \end{cases} ; i = 1,...,L \tag{26}$$

(iii) The instantaneous output estimation error of the NPEs in (10) is redefined as:

$$\tilde{y}_k^i = \left[x_{k-1}^{m} \quad \hat{x}_{k-1}^{unm} \right] - \hat{y}_k^i \quad ; i = 1,...,L \tag{27}$$

(iv) The residuals corresponding to the L NPEs in the bank given by (10) is redefined as:

$$r_k^i = \left[x_{k-1}^{m} \quad \hat{x}_{k-1}^{unm} \right] - \hat{y}_k^i \quad ; i = 1,...,L \tag{28}$$

The rest of the equations for the weight update laws and the FDI decision logic essentially remain the same as those given for the series-parallel FDII scheme.

The FDII using the parallel scheme is accomplished under partial-state measurement using exactly the same principle as the one described for the series-parallel scheme. More specifically, an FTO is integrated with the parallel NPEs as depicted in Fig. 4.

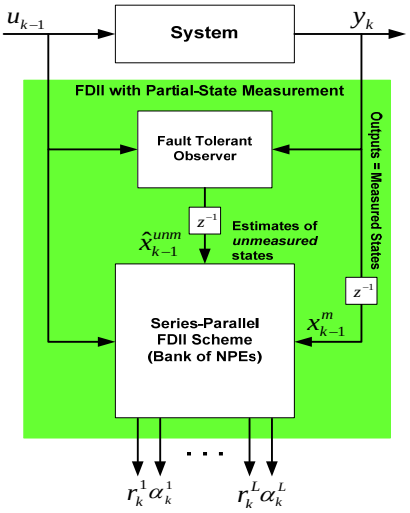

Fig. 3 The series-parallel FDII scheme under partial-state measurements using the integration of the *hybrid* NPEs and an FTO

It is important to note that Fig. 4 looks exactly the same as Fig. 3 except for the internal structure of the two FDII schemes. More specifically, the difference between the two figures is internal to the FDII blocks and is in the way the vector $\left[x_{k-1}^{m} \quad \hat{x}_{k-1}^{unm} \right]$ is being used. Equations (25)-(28) essentially describe how the vector $\left[x_{k-1}^{m} \quad \hat{x}_{k-1}^{unm} \right]$ affects the equations governing the series-parallel scheme. For the robust parallel scheme, however, the changes (i) and (ii) above are *not* required and (18)-(20) still remain valid. Indeed, for the robust parallel scheme *only* the instantaneous output estimation error and the residual vector have to be redefined as in (27) and (28), respectively.

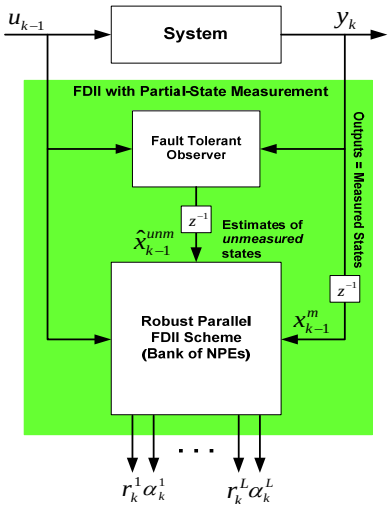

Fig. 4 The robust parallel FDII scheme under partial-state measurements using the integration of the *hybrid* NPEs and an FTO

The FDII schemes under partial state measurements (depicted in Figs. 3 and 4) consist of two main modules, namely the *hybrid* NPEs and an FTO. The design and development of hybrid NPEs was the subject of Sections III and IV and both series-parallel and parallel NPE schemes were thoroughly treated. Hence, the focus of the next section is on the design and development of an FTO, which enables FDII under partial state measurements.

VI. Kalman Filter Structure Preserving Neural State Estimator (NSE)

To solve the filtering problem for nonlinear dynamical systems, [24] used the so-called concept of "linear-structure preserving principle" (LISP), which is designed to imitate the structure of an optimal linear recursive least squares (RLS) [22] or similarly the standard Kalman filter [23]. In other words, the linear state prediction is replaced by a nonlinear one, using the *exact* nonlinear dynamics of the system. Furthermore, the filter gain matrix is replaced by a parameterized nonlinear function that is a function of the prediction error. For the parameterized nonlinear function, we use a multi-layer perceptron (MLP) neural network with neural weights as the parameters that are continuously adapted; hence the name Kalman filter structure-preserving neural state estimator (NSE). To summarize, the recursive state equations of the NSE are as follows:

$$\begin{cases} \text{Prediction Step:} \quad \hat{x}_k^- = f\left(\hat{x}_{k-1}, u_{k-1}\right) \\ \text{Correction Step:} \quad \hat{x}_k = \hat{x}_k^- + g\left(e_k^-, W_k^{obs}, V_k^{obs}\right) \end{cases} \tag{29}$$

with the output equation defined as:

$$\hat{y}_k = H\hat{x}_k \tag{30}$$

where $e_k^- = y_k - \hat{y}_k^- = y_k - H\hat{x}_k^-$ is the prediction error, $g\left(e_k^-, W_k^{obs}, V_k^{obs}\right)$ is a multilayer feed-forward neural network with prediction error e_k^- as the input and with sigmoidal activation functions for the hidden-layer neurons and linear neurons in the output layer. The parameters W_k^{obs} and V_k^{obs} denote the weights of the output layer and the hidden layer of the network, respectively.

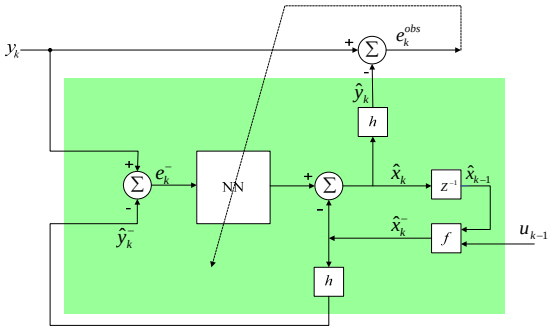

Fig. 5 The Kalman filter structure preserving neural state estimator (NSE) redrawn with modifications from [24]

The recursive equation of the NSE, given by (29), can also be written as:

$$\hat{x}_k = f\left(\hat{x}_{k-1}, u_{k-1}\right) + g\left(e_k^-, W_k^{obs}, V_k^{obs}\right) \tag{31}$$

The block diagram representation of the Kalman filter structure preserving NSE is shown in Fig. 5. The only assumption made in this scheme is that the process and measurement noise are zero mean, mutually independent and identically distributed (i.i.d.). Parisini and Zoppoli [24] applied this neural filter to a subclass of *target motion analysis* problems. Simulation results presented therein revealed that this neural filter outperforms the EKF algorithm especially in presence of model uncertainties or model parameter variations. The results showed significant performance gains over the EKF filter, especially in situations where the EKF diverges due to numerical instability of the covariance matrix. The other advantage of this recursive scheme is that it does not have the computational complexity issues of the Lo's approach [25] when the observation period is too large or has essentially no *a priori* bound as in on-line health monitoring and fault diagnosis applications. It is important to note that the structure/architecture of the developed NSE is *not* a novelty of this work and has been taken from [24]. However, the development of *new* weight update laws for the NSE comprises another contribution of this work, which is the subject of the next section.

A. Update Laws for the NSE: Recursive On-line Backpropagation

Parisini and Zoppoli [24] update the neural filter weights by using the standard back-propagation algorithm. Their neural weight adaptation was performed at the time-step k+1 through a nonlinear optimization on the set of weights $W_{k+1}^{obs}, V_{k+1}^{obs}$, while freezing the set of k previously computed weights $\{W_i^{obs}, V_i^{obs}\}_{i=1}^{k}$. However, this procedure may result in suboptimal performance or even filter divergence due to the presence of feedback in the NSE architecture (as seen from Fig. 5, where the neural network output is fed back to its input after passing through the system dynamics).

In order to adapt the parameters of the *closed-loop* system, a partial derivative of the associated dynamical system must be calculated. Due to the presence of feedback, the calculation of this derivative can be quite complex. However, [26] correctly argues that the *ordered partial derivative*, which is a partial derivative whose constant and varying terms are defined by using ordered set of equations, provides a mathematical tool for computing derivatives of dynamical systems.

As shown by [26], two classes of steepest descent adaptation (or training) algorithms based on ordered partial derivatives can be derived for a general closed-loop nonlinear system. These include: (i) *epochwise training algorithms* and (ii) *on-line training algorithms*. An *epochwise training algorithm* is any algorithm in which the adaptation takes place after each epoch or after a number of epochs, where an *epoch* is an iteration to iteration cycling of a discrete-time system from initial to final iteration (i.e., $k = k_f$).

Due to the real-time limitations of the epochwise training algorithms, we adopt *on-line training algorithms.* The error is defined at each iteration as the instantaneous difference between the desired response and the output of the system:

$$e_k = \frac{1}{2}\left(d_k - y_k\right)^{\mathrm{T}}\left(d_k - y_k\right) \tag{32}$$

where k is the current time step (or iteration). However, calculation of the exact *ordered partial derivative* of the error with respect to the weight vector (i.e., error gradient) is not possible. Instead, an approximation of the error gradient must be used to update the weights. Therefore, the on-line update rule at step k is expressed as:

$$W_{k+1} = W_k - \eta \frac{\partial^+ \widehat{E_k}}{\partial W_k} \tag{33}$$

where $\partial^+ \widehat{E_k} / \partial W_k$ is the approximate error gradient which is calculated as [26]:

$$\frac{\partial^+ \widehat{E_k}}{\partial W_k} = -\left(d_k - y_k\right)^{\mathrm{T}} \frac{\partial^+ \widehat{y_k}}{\partial W_k} \tag{34}$$

where the *approximate* output derivative $\partial^+ \widehat{y_k} / \partial W_k$ is *recursively* obtained from (with $0 < \beta < 1$):

$$\frac{\partial^+ \widehat{y_k}}{\partial W_k} = \frac{\partial y_k}{\partial W_k} + \sum_{m=1}^{M} \beta^m \frac{\partial y_k}{\partial y_{k-m}} \frac{\partial^+ \widehat{y_{k-m}}}{\partial W_{k-m}} \tag{35}$$

To obtain the weight update laws of the Kalman filter structure preserving NSE, let us define the observer error as:

$$e_k^{obs} = y_k - \hat{y}_k \tag{36}$$

where y_k denotes the outputs (i.e., the measured states) of the system and $\hat{y}_k$ is the output estimates from the FTO (i.e., the NSE). By using the observer error in (36) and (30), the cost function of the NSE is defined as:

$$J_k^{obs} = \frac{1}{2}\left\|e_k^{obs}\right\|^2 = \frac{1}{2}\left\|y_k - \hat{y}_k\right\|^2 = \frac{1}{2}\left\|y_k - h(\hat{x}_k)\right\|^2 \tag{37}$$

By utilizing the on-line scheme given in (32), the weights of the NSE must are updated as:

$$W_{k+1}^{obs} = W_k^{obs} - \eta_w^{obs}\left(\frac{\partial^+ J_k^{obs}}{\partial W_k^{obs}}\right) \tag{38}$$

$$V_{k+1}^{obs} = V_k^{obs} - \eta_v^{obs}\left(\frac{\partial^+ J_k^{obs}}{\partial V_k^{obs}}\right) \tag{39}$$

where W_k^{obs} and V_k^{obs} are the output layer and hidden layer weights of the NSE, respectively, and η_w^{obs} and η_v^{obs} are the corresponding learning rates.

Using (38)-(39), the *approximate* gradient of the cost function with respect to the output layer weights W_k^{obs} is given by:

$$\begin{aligned}\frac{\partial^+ \widehat{J_k^{obs}}}{\partial W_k^{obs}} &= -e_k^{obs}\frac{\partial h(\hat{x}_k)}{\partial \hat{x}_k}\frac{\partial^+ \widehat{\hat{x}_k}}{\partial W_k^{obs}} \\ &= -e_k^{obs} H \frac{\partial^+ \widehat{\hat{x}_k}}{\partial W_k^{obs}} \\ &= -e_k^{obs}\sum_{j=1}^{n} h_j \frac{\partial^+ \widehat{\hat{x}_k^j}}{\partial W_k^{obs}}\end{aligned} \tag{40}$$

Similarly, the *approximate* gradient of the cost function with respect to the hidden layer weights V_k^{obs} is as:

$$\begin{aligned}\frac{\partial^+ \widehat{J_k^{obs}}}{\partial V_k^{obs}} &= -e_k^{obs}\frac{\partial h(\hat{x}_k)}{\partial \hat{x}_k}\frac{\partial^+ \widehat{\hat{x}_k}}{\partial V_k^{obs}} \\ &= -e_k^{obs} H \frac{\partial^+ \widehat{\hat{x}_k}}{\partial V_k^{obs}} \\ &= -e_k^{obs}\sum_{j=1}^{n} h_j \frac{\partial^+ \widehat{\hat{x}_k^j}}{\partial V_k^{obs}}\end{aligned} \tag{41}$$

Now, by invoking the on-line *recursive* algorithm in (35) we have:

$$\begin{cases}\dfrac{\partial^+ \widehat{\hat{x}_k^j}}{\partial W_k^{obs}} = \dfrac{\partial \hat{x}_k^j}{\partial W_k^{obs}} + \beta\dfrac{\partial \hat{x}_k^j}{\partial \hat{x}_{k-1}}\dfrac{\partial^+ \widehat{\hat{x}_{k-1}}}{\partial W_{k-1}^{obs}} \\ \dfrac{\partial^+ \widehat{\hat{x}_k^j}}{\partial V_k^{obs}} = \dfrac{\partial \hat{x}_k^j}{\partial V_k^{obs}} + \beta\dfrac{\partial \hat{x}_k^j}{\partial \hat{x}_{k-1}}\dfrac{\partial^+ \widehat{\hat{x}_{k-1}}}{\partial V_{k-1}^{obs}}\end{cases};\ j = 1,\ldots,n \tag{42}$$

where $\hat{x}_k \in \Re^n$ is the estimate of the system state vector at time step k and $\hat{x}_k^j$ is the estimate of the jth state of the system. It is important to note that the parameter M in (35) is equal to 1, as can be seen from (42) of the Kalman filter structure preserving NSE. This is due to the fact that in the NSE architecture only the last state estimate generated at the output of the NSE, namely $\hat{x}_{k-1}$, is fed back to the NSE input, as can also be seen from Fig. 5. Furthermore, it is important to note that (42) is a *recursive* equation that requires calculating the following terms in each iteration:

$$\begin{cases}\dfrac{\partial \hat{x}_k^j}{\partial W_k^{obs}} = \dfrac{\partial g_j\left(e_k^-, W_k^{obs}, V_k^{obs}\right)}{\partial W_k^{obs}} \\ \dfrac{\partial \hat{x}_k^j}{\partial V_k^{obs}} = \dfrac{\partial g_j\left(e_k^-, W_k^{obs}, V_k^{obs}\right)}{\partial V_k^{obs}}\end{cases};\ j = 1,\ldots,n \tag{43}$$

The above terms can be easily calculated using the standard back-propagation (BP) algorithm as:

$$\begin{cases}\dfrac{\partial g_j\left(e_k^-, W_k^{obs}, V_k^{obs}\right)}{\partial W_k^{obs}} = \sigma\left(V_k^{obs} e_k^-\right) \\ \dfrac{\partial g_j\left(e_k^-, W_k^{obs}, V_k^{obs}\right)}{\partial V_k^{obs}} = \left(W_{k_j}^{obs}\left(I - \Lambda\left(V_k^{obs} e_k^-\right)\right)\right)^T e_k^-\end{cases};\ j = 1,\ldots,n \tag{44}$$

Furthermore, the term $\partial \hat{x}_k^j / \partial \hat{x}_{k-1}$ in (42) is defined as:

$$\frac{\partial \hat{x}_k^j}{\partial \hat{x}_{k-1}} \triangleq \left[\frac{\partial \hat{x}_k^j}{\partial \hat{x}_{k-1}^1}, \ldots, \frac{\partial \hat{x}_k^j}{\partial \hat{x}_{k-1}^i}, \ldots, \frac{\partial \hat{x}_k^j}{\partial \hat{x}_{k-1}^n}\right];\ j = 1,\ldots,n \tag{45}$$

where $\partial \hat{x}_k^j / \partial \hat{x}_{k-1}^i;\ j = 1,\ldots,n;\ i = 1,\ldots,n$ is the $(j,i)^{\text{th}}$ element of the above matrix. Consider the Jacobian matrix of the nonlinear system defined as:

$$F_{k-1} = \frac{\partial f(\hat{x}_k, u_k)}{\partial \hat{x}_{k-1}} = \begin{bmatrix}\dfrac{\partial f_1(\hat{x}_k, u_k)}{\partial \hat{x}_{k-1}^1} & \cdots & \dfrac{\partial f_1(\hat{x}_k, u_k)}{\partial \hat{x}_{k-1}^n} \\ \vdots & & \vdots \\ \dfrac{\partial f_n(\hat{x}_k, u_k)}{\partial \hat{x}_{k-1}^1} & \cdots & \dfrac{\partial f_n(\hat{x}_k, u_k)}{\partial \hat{x}_{k-1}^n}\end{bmatrix} \tag{46}$$

with the $(j,i)^{\text{th}}$ element defined as follows:

$$F_{k-1}^{ji} = \frac{\partial f_j(\hat{x}_k, u_k)}{\partial \hat{x}_{k-1}^i};\ j = 1,\ldots,n;\ i = 1,\ldots,n \tag{47}$$

The $(j,i)^{\text{th}}$ element of the matrix in (45) – and correspondingly in (42) – can be calculated as follows (in conjunction with (31)):

$$\frac{\partial \hat{x}_k^j}{\partial \hat{x}_{k-1}^i} = F_{k-1}^{ji} + W_{k_j}^{obs}\left(I - \Lambda\left(V_k^{obs} e_k^-\right)\right)\frac{\partial e_k^-}{\partial \hat{x}_{k-1}^i} \tag{48}$$

where the term $W_{k_j}^{obs}\left(I-\Lambda\left(V_k^{obs}e_k^-\right)\right)$ is the partial derivative of the neural network output with respect to its input, which is obtained using the standard BP algorithm.

Finally, the partial derivative $\partial e_k^-/\partial\hat{x}_{k-1}^i$ can simply be calculated as:

$$\frac{\partial e_k^-}{\partial\hat{x}_{k-1}^i}=\frac{\partial\left(y_k-h\left(\hat{x}_k^-\right)\right)}{\partial\hat{x}_{k-1}^i}=-\frac{\partial h\left(\hat{x}_k^-\right)}{\hat{x}_k^-}\cdot\frac{\partial\hat{x}_k^-}{\partial\hat{x}_{k-1}^i} \tag{49}$$

where $\partial h\left(\hat{x}_k^-\right)/\hat{x}_k^-=H$, when the system output equation (30) is linear.

VII. Conclusion

In this paper a novel integrated *hybrid* solution to the problem of fault diagnosis of components of nonlinear systems has been presented. This work can be considered as an extension of our earlier work in [27]-[30]. Unlike most existing fault diagnosis techniques, the proposed solution is able to *simultaneously* detect, isolate, and identify the severity of faults in system components within a single unified diagnostic module.

The core of the proposed *hybrid* nonlinear fault detection, isolation, and identification (FDII) scheme is a bank of adaptive neural parameter estimators (NPE), where each NPE in the bank was designed based on a separate single-parameter fault model. At each instant of time, the NPEs provide estimates of the *unknown* fault parameters (FP), which in conjunction with the output residuals determine the health state of the system being monitored.

Two NPE structures, namely series-parallel and parallel, have been proposed and their respective FDI decision logics and weight update laws are presented. The notion of a fault tolerant observer (FTO) was introduced, which enables the estimation of *unmeasured* states of the system even in presence of faults in the system. A Kalman structure preserving neural state estimator (NSE) was designed and developed that adaptively estimates system states by constantly minimizing a performance index comprising of the instantaneous observation error. The adaptive capability of neural networks has been exploited in the proposed NSE in order to achieve robustness with respect to faults. Due to presence of output feedback in the architecture of the NSE and the on-line nature of the proposed FTO, new update laws are derived using the on-line recursive back-propagation algorithm.

Applications of the proposed solutions are currently under investigation to fault diagnosis of unmanned underwater vehicle (UUV) systems.

References

[1] S. Simani, C. Fantuzzi, R. J. Patton, Model-based Fault Diagnosis in Dynamic Systems Using Identification Techniques, Springer 2003.
[2] R. Isermann, Model-based Fault Detection and Diagnosis - Status and Applications, Annual Reviews in Control 29 (2005) 71-85.
[3] R. Rengaswamy, D. Mylaraswamy, V. Venkatasubramanian, K. E. Arzen, A Comparison of Model-based and Neural Network-based Diagnostic Methods, Engineering Applications of Artificial Intelligence 14 (2001) 808-818.
[4] P. Frank, Fault Diagnosis in Dynamic Systems using Analytical and Knowledge-based Redundancy: A Survey and Some New Results, Automatica 26 (1990) 459-474.
[5] R. J. Patton, C.J. Lopez-Toribio, F. J. Uppal, Artificial Intelligence Approaches to Fault Diagnosis," in: IEE Colloquium on Condition Monitoring: Machinery, External Structures and Health (Ref. No. 1999/034), April 1999, pp. 5/I-5/18.
[6] N. Vaswani, Adaptive Change Detection in Nonlinear Systems with Unknown Change Parameters, IEEE Trans. on Signal Processing 55 (3) (2007) 859-872.
[7] P. Li, V. Kadirkamanathan, Particle Filtering based Likelihood Ratio Approach to Fault Diagnosis in Nonlinear Stochastic Systems, IEEE Trans. on Sys, Man Cybernetics 31 (3) (2001) 337-343.
[8] P. Li, V. Kadirkamanathan, Fault Detection and Isolation in Non-linear Stochastic Systems - A Combined Adaptive Monte Carlo Filtering and Likelihood Ratio Approach, Int. J. of Control 77 (12) (2004) 1101-1114.
[9] L. Guo, L. Yin, H. Wang, T Chai, Entropy Optimization Filtering for Fault Isolation of Nonlinear Non-Gaussian Stochastic Systems, IEEE Trans. on Automatic Control 54 (4) (2009) 804-810.
[10] E. Sobhani-Tehrani, K. Khorasani, S. Tafazoli, Dynamic Neural Network-based Estimator for Fault Diagnosis in Reaction Wheel Actuator of Satellite Attitude Control System, in: Proceedings of the IEEE Int. Joint Conf. on Neural Networks (IJCNN), Montreal, Canada, 2005, pp. 2347-2352.
[11] A. Alessandri, Fault Diagnosis for Nonlinear Systems using a Bank of Neural Estimators, Computers in industry 52 (2003) 271-289.
[12] Z. Xiaodong, M. M. Polycarpou, T. Parisini, A Robust Detection and Isolation Scheme for Abrupt and Incipient Faults in Nonlinear Systems, IEEE Trans. on Auto. Control 47 (4) (2002) 576 – 593.
[13] M. M. Polycarpou, A. J. Helmicki, Automated Fault Detection and Accommodation: A Learning Systems Approach, IEEE Trans. on Sys. Man Cybernetics 25 (11) (1995) 1447 – 1458.
[14] R. K. Mehra, C. Rago, S. Seereeram, Failure Detection and Identification using a Nonlinear Interactive Multiple Model (IMM) Filtering Approach with Aerospace Applications, in: Proceedings of the 11th IFAC Symposium on System Identification, Fukuoka, Japan, July 1997.
[15] Y. Zhang, Xiao, Detection and Diagnosis of Sensor and Actuator Failures using Interacting Multiple-Model Estimator, in: Proceedings of the 36th IEEE Conference on Decision and Control, San Diego, CA, Dec. 1997, pp. 4475-4480.
[16] T. Jiang, K. Khorasani, S. Tafazoli, Parameter Estimation-Based Fault Detection, Isolation and Recovery for Nonlinear Satellite Models, IEEE Transactions on Control Systems Technology 16 (4) (2008) 799-808.
[17] N. Tudoroiu, K. Khorasani, Fault Detection and Diagnosis for Satellite's Attitude Control System using An Interactive Multiple Model (IMM) Approach, in: Proceedings of the 2005 Conference on Control Applications, Toronto, Canada, August 2005, pp. 1287-1292.
[18] N. Tudoroiu, E. Sobhani-Tehrani, K. Khorasani, Interactive Bank of Unscented Kalman Filters for Fault Detection and Isolation in Reaction Wheel Actuators of Satellite Attitude Control System, in: Proceedings of the 32nd IEEE Conference on Industrial Electronics, Paris, France, 2006, pp. 264-269.
[19] C. Rago, R. Pransanth., R. K. Mehra, R. Fortenbaugh, Failure Detection and Identification and Fault Tolerant Control using the IMM-KF with Applications to the Eagle-Eye UAV, in: 37th IEEE Conf. on Decision and Control, Tampa, FL, December 1998, vol. 4, pp. 4208-4213.
[20] A. Medvedev, State Estimation and Fault Detection by a Bank of Continuous Finite-Memory Filters, Int. J. of Control 69 (4) (1998) 499-517.
[21] R. Isermann, Fault Diagnosis of Machines via Parameter Estimation and Knowledge Processing - A Tutorial Paper, Automatica 29 (4) (1994) 815-835.
[22] A. Houacine, Regularized Fast Recursive-Least Squares Algorithms for Finite Memory Filtering, IEEE Transactions on Signal Processing 40 (1992) 758-769.
[23] S. Haykin, Kalman filters, in: S. Haykin (Ed.), Kalman Filtering and Neural Networks, Wiley/Interscience, 2001, pp. 1-22.
[24] T. Parisini, R. Zoppoli, Neural Networks for Nonlinear State Estimation, International Journal of Robust Control 4 (1994) 231-248.
[25] J. T. Lo, Synthetic Approach to Optimal Filtering, IEEE Transactions on Neural Networks 5 (5) (1994) 803-811.

[26] S. W. Piche, Steepest Descent Algorithms for Neural Network Controller and Filters, IEEE Transactions on Neural Networks 5 (2) (1994) 198-212.
[27] N. Tudoroiu, K. Khorasani, Fault Detection and Diagnosis for Reaction Wheels of Satellite's Attitude Control System using a Bank of Kalman Filters, in: Int. Symp. on Signals, Circuits and Systems, Iasi, Romania, vol. 1, July 2005, pp. 199-202.
[28] H. A. Talebi, R.V. Patel, An Intelligent Fault Detection and Recovery Scheme for Reaction Wheel Actuator of Satellite Attitude Control Systems, in: IEEE Conf. on Control Applications, Munich, Germany, October 2006, pp. 3282 – 3287.
[29] H. A. Talebi, K. Khorasani, S. Tafazoli, A Recurrent Neural Network-based Sensor and Actuator Fault Detection, Isolation for Nonlinear Systems with Application to a Satellite's Attitude Control Subsystem, IEEE Transactions on Neural Networks 20 (1) (2009) 45-60.
[30] A. Valdes, K. Khorasani, A Pulsed Plasma Thruster Fault Detection and Isolation Strategy for Formation Flying Satellites, Applied Soft Computing 10 (3) (2010) 746-758.